



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
Facultad de Ingeniería en Electricidad y
Computación

“Implementación De Un Sistema De Despacho
Aduanero Utilizando “MERODE” Como Método De
Análisis”

TESIS DE GRADO

Previo a la obtención del Título de:

INGENIERO EN COMPUTACIÓN
ESPECIALIZACIÓN SISTEMAS DE INFORMACIÓN

INGENIERO EN COMPUTACIÓN
ESPECIALIZACIÓN SISTEMAS TECNOLÓGICOS

Presentada por:

Ricardo Sebastián Auhing Román

Roger Ernesto Salinas Robalino

GUAYAQUIL - ECUADOR

2006

AGRADECIMIENTO

A Dios, a nuestros padres, y a todas las personas que de una u otra manera colaboraron con nosotros en la realización de este proyecto de tesis; especialmente, al Ing. Xavier Riera, gerente de EIKON S.A., y a todo el personal de la misma empresa.

DEDICATORIA

A Dios, a nuestra familia y a nuestros compañeros.

TRIBUNAL DE GRADUACIÓN

Ing. Hólger Cevallos
SUBDECANO DE LA FIEC
PRESIDENTE

Ing. María V. Macías
DIRECTOR DE TESIS

Ing. Ana Tapia
VOCAL

Ing. Mónica Villavicencio
VOCAL

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de esta Tesis de Grado, me corresponde exclusivamente; y el patrimonio intelectual de la misma a la **Escuela Superior Politécnica del Litoral**”

Ricardo Auhing Román

Roger Salinas Robalino

RESUMEN

Este proyecto surgió ante la necesidad de saber en cuanto puede afectar una metodología de análisis nueva, como la es Model Driven Existence Dependency Relation Object Oriented (MERODE), en la calidad de un software resultante.

Para aplicar esta metodología se escogió como proyecto a desarrollar un Sistema de Despacho Aduanero. En la actualidad existen sistemas que ayudan al despachador aduanero con el manejo de los trámites respectivos pero que no satisfacen las necesidades de los clientes. La complejidad principal del sistema de despacho aduanero es que este debe ser amigable para que pueda ser usado por diferentes usuarios y además que este sea consecuente con las leyes de la aduana que cambian con mucha frecuencia.

Se aplica esta metodología dado que ha tenido mucho éxito con respecto a lo que a flexibilidad se refiere en la industria del software europea.

Los lenguajes de modelamiento, como UML ofrecen un grupo de modelos básicos para describir un sistema de software de diversas maneras de visualizar y en diversos niveles de la abstracción. Las herramientas que apoyan un uso sin restricción de estos modelos de UML no pueden garantizar la consistencia entre modelos de vistas múltiples, debido a la carencia de una definición formal de la semántica de los diagramas de UML.

MERODE se basa en el concepto de un modelo sencillo, para el cual se construyen diversas vistas, y de forma automática o semiautomática que comprueba la consistencia entre estas vistas.

Tres enfoques básicos a la comprobación de la consistencia son consistencia por análisis, consistencia por monitoreo y consistencia por construcción. La consistencia por construcción se basa en el modelamiento del dominio conceptual usando MERODE y su herramienta case asociada MERMAID.

En el Capítulo 1 de la Tesis se describen los objetivos del sistema, y su justificación.

En el Capítulo 2 se explican los fundamentos teóricos en los que se basa el proyecto, tales como que es MERODE, y los diferentes gráficos que se usan.

En el Capítulo 3 se presenta el análisis de los distintos componentes del sistema, en términos de requerimientos funcionales, viabilidad e interacción Hombre-Máquina.

En el Capítulo 4 se explica cómo están diseñadas la arquitectura y la interacción del sistema, los diferentes módulos que lo componen, y la comunicación entre estos.

En el Capítulo 5 se describen la fase de implementación y las pruebas que se realizaron en el sistema, para verificar la obtención de los resultados esperados.

ÍNDICE GENERAL

	Pág.
RESUMEN.....	I
ÍNDICE GENERAL.....	IV
ABREVIATURAS	VIII
ÍNDICE DE FIGURAS	IX
ÍNDICE DE TABLAS	XI
CAPÍTULO 1	
1. INTRODUCCIÓN Y JUSTIFICACIÓN.....	1
1.1.1. <i>Situación actual del uso de metodologías</i>	<i>1</i>
1.1.2. <i>Situación actual de los Sistemas de Despacho Aduanero.....</i>	<i>2</i>
1.3.1. <i>Objetos e Interacción de objetos</i>	<i>5</i>
1.3.2. <i>Orientación a Objetos en el Ciclo de Vida de los Sistemas.....</i>	<i>13</i>
1.4.1. <i>Arquitectura de desarrollo de sistemas de capas</i>	<i>15</i>
1.4.2. <i>Comparación con el Método de Jacobson</i>	<i>21</i>
1.4.3. <i>Interacción de Objetos en MERODE</i>	<i>22</i>

CAPÍTULO 2

2. FUNDAMENTOS TEÓRICOS26

2.1.1.	<i>Tipos de objetos de la empresa y tipos de eventos del negocio</i>	30
2.1.2.	<i>Especificaciones de las restricciones de secuencia</i>	35
2.1.3.	<i>Especificación de vectores de estado y métodos del objeto-evento</i>	37
2.2.1.	<i>El modelo de funcionalidad</i>	39
2.2.2.	<i>Descripción de la vista estática, dinámica y de la interacción</i>	42
2.2.3.	<i>Consistencia por Construcción</i>	44
<u>2.2.3.1.</u>	<u><i>Regla del Alfabeto</i></u>	<u>44</u>
<u>2.2.3.2.</u>	<u><i>Regla de Propagación y tipo de regla de implicación</i></u>	<u>45</u>
<u>2.2.3.3.</u>	<u><i>Detección de posibles trayectorias redundantes</i></u>	<u>47</u>
<u>2.2.3.4.</u>	<u><i>Regla del ciclo de vida por defecto</i></u>	<u>49</u>
2.3.1.	<i>El gráfico de la dependencia de existencia</i>	54
2.3.2.	<i>La tabla de objeto-evento</i>	58
2.3.3.	<i>Máquinas de estado finito</i>	60

CAPÍTULO 3

3. ANÁLISIS DEL SISTEMA64

3.4.1.	<i>Modelo MERODE</i>	75
3.4.2.	<i>Casos de uso y escenarios</i>	98

CAPÍTULO 4

4. DISEÑO DEL SISTEMA 115

4.1.1.	<i>Diseño de la Arquitectura del Sistema</i>	115
--------	--	-----

4.1.2.	<i>Modelo Lógico de la Base de Datos</i>	118
<u>4.1.2.1.</u>	<u><i>Tablas Maestras</i></u>	<u>118</u>
<u>4.1.2.2.</u>	<u><i>Tablas Transaccionales</i></u>	<u>119</u>
<u>4.1.2.3.</u>	<u><i>Tablas Híbridas</i></u>	<u>120</u>
<u>4.1.2.4.</u>	<u><i>Diseño de las Relaciones</i></u>	<u>121</u>
4.2.1.	<i>Mandatorios</i>	130
4.2.2.	<i>Condicionales</i>	130
4.2.3.	<i>No Aplicables</i>	131
4.3.1.	<i>Trámite</i>	132
4.3.2.	<i>Componentes de Entrada</i>	135
4.3.3.	<i>Componentes Generados o de Salida</i>	139
4.3.4.	<i>Componentes de Entrada/Salida</i>	140
4.3.5.	<i>Componente de Verificación</i>	141
4.3.6.	<i>Componente de Grabación</i>	144
4.4.1.	<i>Interacción con la Base de Datos</i>	153

CAPÍTULO 5

5.	IMPLEMENTACION Y PRUEBAS	159
5.1.1.	<i>Estructura del DATASET</i>	159
5.1.2.	<i>DATA BINDING</i>	163
5.2.1.	<i>IMAGE-DOCUMENTS</i>	171
5.2.2.	<i>Reportes Generados por el Sistema</i>	174
5.3.1.	<i>Creación de un Trámite Nuevo</i>	177

5.3.2.	<i>Apertura de un Trámite</i>	179
5.3.3.	<i>Verificación y Grabado de Datos</i>	181
CONCLUSIONES Y RECOMENDACIONES		193
APÉNDICE A		197
	DICCIONARIO DE DATOS	197
	RELACIONES	250
APÉNDICE B		267
	CONTROLES DESARROLLADOS PARA EL SISTEMA.....	267
APÉNDICE C		279
	CONCEPTOS GENERALES EN UN PROCESO ADUANERO	279

BIBLIOGRAFÍA

ABREVIATURAS

FIEC	Facultad de Ingeniería en Electricidad y Computación
FSM	Finite State Machine
JSD	Jacobson State Chart Diagram
EDG	Gráfico de la Dependencia de Existencia
MERODE	Model-Driven, Existence-dependency Relationship, Object-oriented Development
OET	Tabla de Objeto-Eventos
OOA	Análisis Orientado a Objetos
OOD	Diseño Orientado a Objetos
OOP	Programación Orientada a Objetos

ÍNDICE DE FIGURAS

Figura 1.1 Un sistema orientado a objeto	7
Figura 2.1 Comprobación de consistencia entre los 3 esquemas de modelamiento de la empresa	39
Figura 2.2 Vista y consistencia que comprueba reglas en MERODE	44
Figura 2.3 EDG extendido para la administración de proyecto	47
Figura 2.4 Liberación de Impuestos.....	55
Figura 2.5. Gráfico de la dependencia de la existencia por el ejemplo de la liberación de impuesto.....	58
Figura 2.6 FSM del Código Liberatorio	63
Figura 3.1 Modelo de Objetos del Sistema.....	75
Figura 3.2 FSM del Cliente	84
Figura 3.3 FSM del Objeto Pais.....	85
Figura 3.4 FSM del Convenio	85
Figura 3.5 FSM del Objeto factura.....	86
Figura 3.6 FSM del objeto Régimen	87
Figura 3.7 FSM del objeto Reg.Export.....	87
Figura 3.8 FSM del objeto Reg.Import.....	88
Figura 3.9 FSM del objeto Asoc.Export-Reg	88
Figura 3.10 FSM del objeto Asoc.Import-Reg.....	89

Figura 3.11 FSM del objeto Producto	90
Figura 3.12 FSM del objeto Partida	91
Figura 3.13 FSM del objeto Código	92
Figura 3.14 FSM del objeto Asoc.Doc-Proceso	92
Figura 3.15 FSM del objeto Documento	93
Figura 3.16 FSM del objeto Importación	96
Figura 3.17 FSM del objeto Exportación	97
Figura 4.1. Arquitectura Cliente – Servidor de 3 capas	116
Figura 4.2 Relación de la Tabla Process	122
Figura 4.3 Relaciones de las Tablas DUI,DAV,DAU	122
Figura 4.4 Relación de la Tabla Cliente	123
Figura 4.5 Relación de la Tabla Bank	124
Figura 4.6 Relaciones de la Tabla Transport	124
Figura 4.7 Relaciones de la Tabla Invoice	125
Figura 4.8 Relaciones de la Tabla Vehicle	126
Figura 4.9 Relaciones de la Tabla Document	127
Figura 4.10 Relaciones de la Tabla Warranty	128
Figura 4.13 Formularios que Intervienen en un trámite	135
Figura 4.14: Relaciones principales entre tablas	146
Figura 4.15: Opciones del Sistema	148
Figura 4.16: Familiaridad del Sistema	149
Figura 4.17: Consistencia entre formularios	151
Figura 4.18: Selector de Trámites	152
Figura 5.5: Instancias involucradas en el acceso a los formularios de entrada	171
Figura 5.6: Image-Document	172
Figura 5.8: Apertura de un trámite	181

ÍNDICE DE TABLAS

Tabla 2.1 Tabla del Objeto-Evento del sistema.....	62
Tabla 4.1: Tablas maestras de la base de datos del Sistema.....	119
Tabla 4.2: Tablas transaccionales de la base de datos del Sistema.....	120
Tabla 4.3: Tablas híbridas.....	121
Tabla 5.1: Controles derivados de System.Windows.Forms.Control	166
Tabla 5.3: Clases heredadas de DataRow para el sistema.....	175
Tabla 5.4: Clases heredadas de DataTable para el sistema.....	176

CAPÍTULO 1

1. INTRODUCCIÓN Y JUSTIFICACIÓN

1.1. Antecedentes del uso de metodologías de análisis para el desarrollo de software

1.1.1. Situación actual del uso de metodologías

Uno de los problemas fundamentales de la ingeniería de software es que cerca del 80% de los recursos disponibles deben ser usados para el mantenimiento del software existente. Parte de ese problema de mantenimiento es que de hecho, los métodos de desarrollo de software han adoptado un manejo de procesos o un manejo de datos. Durante los últimos 10 años se han desarrollado un número considerable de métodos de diseño y análisis orientado a objetos. Se creía

que aplicando la orientación a objetos a través del proceso de desarrollo de software, el re-uso de componentes sería mejorado, reduciendo de esta forma el costo de desarrollo de software.

Sin embargo, los métodos actuales de análisis y diseños orientados a objetos sufren de algunas desventajas severas.

El primer problema es que la mayoría de métodos usan una mezcla de lenguaje natural y notaciones gráficas semi-formales para la especificación del propósito.

El segundo problema es que la orientación a objetos primero apareció en los lenguajes de programación. Los mismos conceptos han sido aplicados a las etapas de diseño y análisis, sin repensar estos conceptos fundamentalmente. Esto dio como resultado, que muchos conceptos y métodos de análisis tengan una tendencia al uso de la tecnología orientada a objetos para la implementación de sistemas. Esta es una contradicción con la esencia del análisis.

1.1.2. Situación actual de los Sistemas de Despacho Aduanero

Es de conocimiento general que los trámites aduaneros constan de muchos detalles que deben ser manejados por los diferentes agentes que participan en los procesos de importación y exportación de las mercaderías. Uno de estos

agentes es el despachador aduanero, quien se encarga de hacer los trámites necesarios para "retirar" la carga importada o para "enviar" carga al exterior. Ya sea para importar o exportar mercadería, el despachador debe llevar un control del estado de los trámites y por supuesto garantizar al cliente la correcta finalización del proceso.

Existen sistemas que ayudan al despachador con el manejo de trámites aduaneros pero no satisfacen las necesidades de los clientes, por lo que se propone desarrollar un sistema generalizado y flexible para que pueda ser usado por diferentes usuarios, y además que éste sea consecuente con las disposiciones de la ley orgánica de aduanas ecuatoriana, satisfaciendo las necesidades de los clientes.

La complejidad principal del sistema de despacho aduanero es que este debe ser muy flexible, porque las leyes de la aduana cambian con una frecuencia muy alta. Como resultado de esto, el diseño del sistema debe ser construido para garantizar esta propiedad.

El modelo del sistema de despacho aduanero se lo realizará con un método nuevo de nombre MERODE (Model Driven

Existence Dependency Relation Object Oriented Development) desarrollado en la Universidad Católica de Lovaina de Bélgica por el Professor Ph.D. Guido Dedene y Professor Ph.D. Monique Snoeck. Este método ha sido usado con mucho éxito en Bélgica y en Holanda; y ha demostrado con casos de estudio que una de sus principales ventajas es brindar flexibilidad a los sistemas desarrollados; por lo que en teoría sería ideal para el desarrollo del sistema de despacho aduanero.

1.2. Objetivos

El objetivo general de este proyecto es:

- Aplicar el método de análisis orientado a objetos "Model Driven Existence Dependency Relation Object Oriented Development" (MERODE), en el desarrollo de un sistema de despacho aduanero y analizar en que forma dicha metodología afecta a la calidad del software resultante.

Los objetivos específicos de este proyecto son:

- Desarrollar un sistema de propósito general para los trámites que llevan a cabo los agentes aduaneros tanto para transporte aéreo como marítimo.

- Desarrollar un sistema más amigable que los existentes para hacer trámites aduaneros.

1.3. Conceptos Básicos de la Orientación a Objetos

1.3.1. Objetos e Interacción de objetos¹

Un sistema es un conjunto de elementos interconectados. En los sistemas orientados a objetos, estos elementos son llamados "objetos". Un objeto se lo puede definir como un elemento del sistema que:

(1) Tiene un estado. El estado de un objeto se puede describir por un grupo de atributos. Además, los atributos forman el vector del estado del objeto.

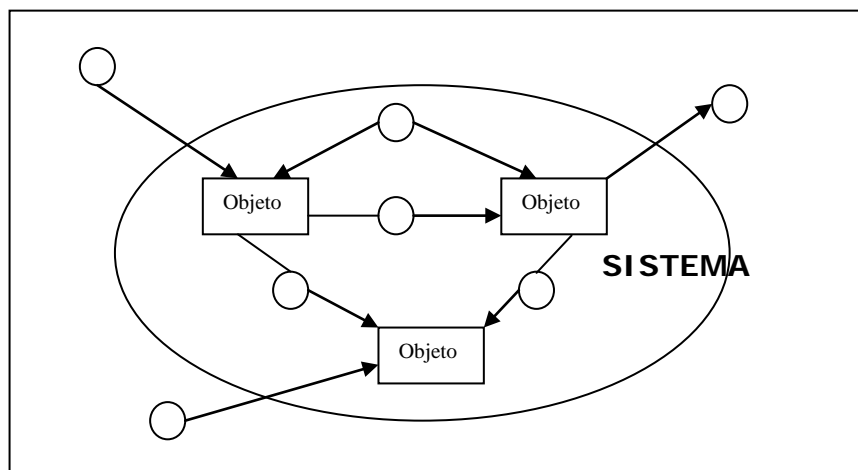
(2) Tiene un cierto comportamiento. El comportamiento de un objeto es disparado por los mensajes enviados por otros objetos o por el ambiente: un mensaje inicia la ejecución de un método (a veces también llamado 'rutina' o 'función'), tal como cada método ejecuta una clase particular de servicio.

¹ Use Cases: the Pros and cons, 2004, The Dangers of Misusing Use Cases, <http://www.ksc.com/article7.htm>

(3) Tiene una identidad. Cada objeto posee una ' identidad ' que no es un atributo. Esta identidad es definida por el sistema y en general no es visible para el usuario del sistema.

El concepto de "identidad" no es por lo tanto igual al concepto de "clave" que se refiere a la posibilidad de requerir atributos para tener valores únicos para identificar objetos por medio de estos valores.

Por ejemplo, en un ambiente de actividades bancarias, el número de cuenta permite identificar la cuenta. Pero puesto que puede ser requerido para permitir cambios a los valores de tales atributos que la identificaban, estos atributos no se pueden utilizar para establecer una identidad única para los objetos. Los ejemplos de objetos son un cliente y una reservación. Ambos se pueden describir por medio de atributos. Por ejemplo, un cliente puede ser descrito por medio de su nombre y dirección.

AMBIENTE**Figura 1.1 Un sistema orientado a objeto**

Una reservación se puede describir por la fecha en la que se hizo la reservación. Ambos objetos también demuestran un cierto comportamiento. Un cliente puede cambiar su dirección, llegar a ser inactivo, o ser registrado como moroso. Una reservación puede ser cancelada.

En el paradigma orientado a objetos, la noción de subordinación está totalmente ausente, en el sentido que todos los objetos son iguales: Los objetos no pueden ser dependientes de otros objetos; los objetos no pueden controlar otros objetos.

Clasificación

La clasificación es algo muy natural. La estamos haciendo a toda hora y sobre todo inconscientemente. Cuando hablamos de una silla, todos sabemos que es una cierta clase de mueble en la cual una persona se puede sentar, con cuatro piernas, un asiento y una parte posterior. Estamos haciendo la abstracción de las características de una silla en particular.

La misma cosa sucede al modelar los sistemas de información. Cuando un objeto cliente es modelado, sólo el comportamiento común que le interesa al futuro sistema de información será modelado.

Las características del cliente no se consideran. Esto significa que un modelo de análisis define solamente los tipos de objeto, es decir, plantillas para las ocurrencias individuales del objeto.

En el modelo de la empresa, el tipo del objeto del cliente describirá las cualidades que son de interés para la empresa (por ejemplo, el nombre y la dirección pero no la longitud de la nariz) y el comportamiento que es de interés para la empresa (por ejemplo, creación de un cliente, registrado como moroso, pero no de cambio del estado civil).

Objeto, Tipo de Objeto, Clase de Objeto, Definición de Clase

Un objeto es una ocurrencia distinta de un tipo de objeto. Como ejemplo, se puede hablar del objeto ' cliente 123 ' y sobre el tipo 'de objeto cliente'. Un tipo de objeto define una plantilla para todos los objetos individuales de ese tipo. La descripción de la plantilla, el tipo de objeto, se llama "un tipo de dato abstracto".

Un tipo de datos abstractos consiste en un número de partes, de las cuales las más esenciales son el grupo de atributos y el grupo de métodos por los cuales el objeto ejecuta sus servicios.

El concepto de clase del objeto denota un conjunto de objetos, que son del mismo tipo. Como ejemplo, en una compañía se puede decir que la clase de objeto CLIENTE contiene 507 objetos de tipo "cliente". En la literatura orientada a objetos, el término clase de objeto es también utilizado como tipo de objeto.

Encapsulamiento y ocultamiento de información

Uno de los defectos principales de los métodos clásicos de desarrollo ha sido la separación entre los datos y el proceso.

Los objetos encapsulan datos y comportamiento.

Esto significa que cada definición de tipo de objeto tiene los aspectos estructurales (los atributos y las relaciones con otros objetos) y aspectos del comportamiento (los métodos). Para alcanzar un acople entre los objetos, los objetos no tienen permitido cambiar directamente los valores de los atributos de otros objetos. Para obtener la información de realizar cambios en otro objeto, el objeto que hace la petición debe enviar un mensaje al otro objeto. Los atributos son de esta manera encapsulados en el objeto.

Para pasar el mensaje, los objetos deben hacer su definición de interfaz visible para otros objetos. La implementación detalla cómo la respuesta a un mensaje es escondida (ocultamiento de información). Consecuentemente, los objetos demuestran la forma posible más débil de acoplarse.

Herencia, Polimorfismo, Sustitubilidad y Dinamic Binding

El principio de herencia dice que una clase particular de objeto B está definida como heredera de las características y los métodos de otra clase A.

Además de las características heredadas, se permite definir atributos y métodos adicionales del vector que se aplican solo a la clase B. El uso de la herencia nos lleva a la definición de una jerarquía de generalización/especialización: Las clases de la especialización heredan las características de las clases de la generalización. La clase generalización y la clase especialización están relacionadas por la relación "esta" o "un tipo de". Además de la posibilidad de ampliar la definición heredada de la clase con las nuevas características, la mayoría de las metodologías y de los lenguajes también permiten eliminar o deshabilitar características heredadas.

En caso de herencia múltiple, una clase de la especialización hereda las características de dos o más clases de generalización distintas. Entonces la jerarquía de la generalización/especialización no es más un árbol, sino un enrejado. La sustitución, el enlazamiento dinámico y el

polimorfismo son conceptos que están relacionados con la herencia y la eliminación de características. El término de polimorfismo significa literalmente ' tomar varias formas'.

Cuando las clases eliminan características heredadas, muchas clases tendrán características con nombres e interfaces idénticas, pero con un contenido diferente. Obviamente, la herencia y los conceptos derivados proporcionan muchas características interesantes para la reusabilidad de especificaciones y de código. La herencia también evita redundancia en especificaciones, y si está bien aplicada, mejora la adaptabilidad en el sentido que cuando los atributos y métodos heredados cambien, el cambio tiene que ser hecho solamente en un lugar, en la clase general del objeto. El uso de la herencia en el nivel del modelo conceptual puede ser totalmente diferente del uso del mismo concepto en un lenguaje de programación. Sin embargo, cuando se pone en práctica, la herencia se utiliza a menudo con el fin de la reutilización del código.

Condiciones Previas y Poscondiciones

Como los objetos prestan servicios a otros objetos, es útil establecer un "contrato" que define estos servicios

precisamente. Tal contrato consiste en tres porciones para cada método:

La interfaz del método, la condición previa y la poscondición. La poscondición define qué resultado puede ser garantizado, la condición previa define bajo qué condiciones el método está garantizado para ejecutarse con éxito y la definición de la interfaz especifica el formato del mensaje que dispara el servicio.

1.3.2. Orientación a Objetos en el Ciclo de Vida de los Sistemas

Cada método de desarrollo del sistema divide el ciclo de vida de un sistema en un número de fases. Generalmente hablando, la mayoría de los métodos hacen una distinción entre el análisis, el diseño y la implementación.

Cada uno de estas fases se puede realizar en una manera orientada a objetos o no. Respecto a eso, la literatura se distingue generalmente en:

- Análisis orientado a objetos (OOA);
- Diseño orientado a objetos (OOD);
- Programación orientada a objetos (OOP).

Es posible adherir los principios de la orientación a objetos en algunas de estas fases y en otras no. Uno podría, por ejemplo, analizar de una manera orientada a objetos, pero realizar el diseño y la programación de otra manera. Sin embargo, aunque puede tener sentido hacer un análisis orientado a objetos pero de realizar el diseño y la implementación del sistema en otra tecnología, el análisis y el diseño orientados a objetos son un pre-requisito para la puesta en práctica orientada a objetos. De hecho, no tiene ningún sentido programar en una manera orientada a objetos y no usar orientación a objetos en las fases anteriores:

Los resultados del análisis de las técnicas estructuradas por ejemplo, son inflexibles, casi imposibles de traducir a un diseño orientado a objetos y de ponerlo en práctica.

El análisis orientado a objetos es un área joven de la investigación y todavía hay algunos problemas importantes que no se han tratado. Por ejemplo, aunque la orientación de objetos proporciona una manera natural de definir los módulos libremente juntados, el solo uso de la orientación no es suficiente para solucionar los dramáticos problemas de mantenimiento de software a los que los desarrolladores le

hacen frente hoy en día. Para mejorar más la capacidad de mantenimiento y la adaptabilidad de los sistemas de software, se requiere enfoque basado en el modelo en el desarrollo del software.

1.4. Comparación de MERODE con otras metodologías

1.4.1. Arquitectura de desarrollo de sistemas de capas²

El desarrollo basado en el modelo sigue la idea de que la complejidad de los sistemas de información es producto de la complejidad del mundo real.

Dando atención especial al modelar el mundo real antes que la especificación de la funcionalidad deseada, los desarrolladores de los sistemas de información pueden manejar mejor la complejidad de los sistemas desarrollados.

El enfoque basado en el modelo es también la respuesta a la carencia de modularidad en la mayoría de los métodos de análisis y de diseño orientado a objetos. Aunque los conceptos básicos de la orientación a objetos mejoran la capacidad de mantenimiento de los sistemas significativamente, hay lugar para una mejora importante organizando objetos en capas y particiones.

² Snoeck Monique, 1995, On a process Algebra Approach to the construction and análisis of MERODE based conceptual models.

Esto puede ser alcanzado adoptando el enfoque de desarrollo basado en el modelo.

Muy parecido a los métodos clásicos de desarrollo, el modelo de análisis que se construye en la primera etapa de una metodología orientada a objetos es una abstracción de lo que el sistema deseado debe hacer, no cómo lo hace. El modelo del objeto desarrollado en la fase de análisis define un conjunto de clases de objetos, de sus características, de los métodos e interacciones.

En general, las metodologías de análisis orientados a objetos no particionan este conjunto de clases en diversas categorías. Algunas metodologías proporcionan técnicas tales como áreas de temas definidos arbitrariamente para reducir la complejidad.

Para alcanzar un reparto tan sistemático, deben ser consideradas la naturaleza particular y las características de las especificaciones. Por ejemplo, hay diversas clases de especificaciones, y cada uno tiene su propia tasa de cambio: El índice del cambio de los requisitos de la interfaz del usuario es mucho más alto que el índice del cambio de un modelo de la

empresa. Cada unidad de la especificación dará lugar a un número de definiciones del tipo de objetos. Estos tipos de objetos tendrán la misma tasa de cambio de la cual se originaron las especificaciones. Se permite solamente utilizar los servicios de clases de la misma capa o de capas inferiores pero no los servicios de objetos en capas superiores. Esto asegura que las modificaciones en objetos en una capa externa tengan poco efecto en objetos de una capa interna. Poniendo clases estables de objetos en la capa internas, éstas pueden seguir sin cambiar cuando las clases en capas externas se modifican. Los sistemas que se pasaron a MERODE tienen una estructura natural de capas por especificaciones que se agrupaban según el aspecto del que provienen, algunas especificaciones pueden provenir de requisitos esenciales del negocio (incluyendo objetos del negocio, así como los limitantes del negocio). Este tipo de requisitos es también válido si no hay un sistema de información. Otras especificaciones se relacionan típicamente con la presencia de un sistema de información y se originan de un tipo de funcionalidad mas administrativo, como los servicios de ingreso al sistema, generación de informes, formato de EDI, etcétera.

Las especificaciones del primer tipo constituyen un modelo de la empresa (o el modelo del dominio del negocio) que contiene el conocimiento relevante del dominio para que funcione un negocio.

Alrededor de este modelo de la empresa, un modelo de la funcionalidad (o modelo de servicios) se construye como conjunto de servicios de entrada y de salida, ofreciendo la funcionalidad deseada de la información a los usuarios del sistema de información.

Los servicios de salida permiten que los usuarios extraigan la información del modelo del dominio del negocio, y lo presentan en el correcto formato en papel, en una estación de trabajo o en un formato electrónico. Los servicios de entrada permiten que los usuarios registren nueva información o modifiquen información que es relevante para el negocio. En MERODE, tal agrupación no se permite en la etapa de la especificación. Puede ocurrir en la etapa de la puesta en práctica.

La separación entre un modelo de la empresa y un modelo de la funcionalidad conduce a una partición natural de sistemas

en tres subsistemas débilmente acoplados, cada uno corresponde a un solo aspecto del problema, y se desarrolla independientemente del resto de los aspectos. Los tres aspectos independientes son:

- La lógica del negocio (el modelo de la empresa);
- La organización del trabajo (los servicios de entrada);
- Las necesidades de información (los servicios de salida).

Cuando los usuarios formulan requisitos del sistema, tienden a mezclar especificaciones del conocimiento del dominio con requisitos de la funcionalidad.

Por la metodología JSD³, modelar el mundo verdadero o el "dominio" permite construir una base sólida para un sistema de software. La complejidad de la mayoría de los sistemas es causada por la complejidad de la realidad con la que ellos tienen que tratar.

Además, las declaraciones sobre la funcionalidad requerida tienen siempre algunas asunciones subyacentes sobre el mundo verdadero.

³ J.A. Zachman, 1999, A framework for information systems architecture

Tiene así sentido construir un modelo del mundo real antes del desarrollo de un sistema de información. Por otra parte, no tiene ningún sentido construir un modelo del mundo real sin tener cierta funcionalidad en mente.

Aquí, ambos modelos son una parte integral de un modelo del análisis con una misma importancia.

La arquitectura de MERODE conduce a un incremento significativo de la productividad durante el mantenimiento.

Porque las funciones son módulos independientes, pueden ser agregadas y eliminadas fácilmente y las clases de objetos de la empresa se pueden agregar al modelo de la empresa sin efectos de propagación sobre funciones existentes y sobre otras clases de objetos de la empresa.

Además, debido a que las funciones se basan siempre en una parte específica del modelo de la empresa, el grupo de funciones que es afectado por una modificación de las clases de objetos de la empresa es fácil de determinar.

1.4.2. Comparación con el Método de Jacobson

A primera vista el enfoque de MERODE es parecido al enfoque de casos de uso de Ivar Jacobson. En el método de Jacobson, los requisitos funcionales son la base para el desarrollo del modelo de los requisitos. Los requisitos funcionales son especificados por los casos de uso.

El modelo de Casos de uso puede ser expresado en términos de un modelo de objetos del dominio, que es una clase de modelo de la empresa. En cambio, el modelo de análisis utiliza tres clases de objetos: Los objetos entidad, los objetos del interfaz y objetos de control. En general, los objetos del dominio serán transformados en objetos de la entidad y la funcionalidad de un caso del uso será repartida y asignada a los objetos de los tres tipos.

Al comparar el método de Jacobson con MERODE, uno podría decir que los tipos de objetos del dominio de MERODE son tipos del objeto de la entidad de Jacobson y que las funciones de MERODE son equivalentes a los casos de uso.

Sin embargo, MERODE da mucho más atención al dominio que modela que el método de Jacobson. Además, se requiere que las funciones sean independientes una de otra: La ejecución de una función nunca debería de depender de la ejecución o disponibilidad de otras funciones. Toda comunicación debe pasar a través del modelo de la empresa. Esto no es requerido para los casos de uso. Otra diferencia importante es que un modelo de la empresa de MERODE incluye la definición de eventos, atributos y operación del objeto; el enfoque basado en casos de uso pospone la definición de operaciones hasta la fase de diseño del sistema.

1.4.3. Interacción de Objetos en MERODE

El paradigma orientado a objetos proviene de lenguajes de programación. La orientación a objeto también se ha aplicado a las etapas de diseño y de análisis de los ciclos de vida del desarrollo del software.

En estas etapas, la mayoría de los métodos utilizan conceptos básicos de la orientación a objetos sin la adaptación substancial. Consecuentemente, casi todos los métodos de análisis y de diseño orientados a objetos actuales utilizan el paso de mensajes como paradigma de comunicación. Con este

formalismo, los objetos pueden invocar directamente operaciones de otros objetos. Esto seguramente es un mecanismo de comunicación eficiente y permite que los modeladores controlen fuertemente la secuencia y la sincronización. Sin embargo, usar el paso de mensajes para los propósitos de la especificación implica el uso de paso de mensajes para el diseño y la implementación. Entonces es posible utilizar tecnología no orientada a objetos para la implementación, pero esto requerirá un esfuerzo de traducción substancial. Además, debido al uso de diversos paradigmas para la especificación y la puesta en práctica, la trazabilidad de requisitos no puede ser asegurada. De hecho, la situación actual viola el acuerdo general que un modelo conceptual (en este caso un modelo de análisis orientado a objetos) debería ser un modelo del dominio del problema y no ser influenciado solamente por el dominio de la solución. Igualmente, las técnicas del análisis no se deben influenciar por opciones posibles de la implementación. Este tipo de interacción de objetos se puede modelar por medio del álgebra de procesos. Las ventajas de usar la comunicación por medio de eventos comunes son numerosas:

En el contexto del modelamiento del dominio o de la empresa, conduce a la identificación explícita de los eventos del negocio;

puede ser representado por una técnica de modelamiento sencilla: la tabla del objeto-evento.

La noción de los eventos del negocio se integra con la noción de la tarea.

La noción del evento combina bien con las máquinas de estado, donde los eventos son los disparadores para las transiciones de un estado a otro.

El comportamiento y la comunicación entre objetos son fáciles de concluir.

Permite una implementación manejada por eventos, que es fácil de realizar con herramientas de implementación no orientadas a objetos.

No imposibilita una puesta en práctica orientada a objetos con el paso de mensajes como mecanismo de la interacción.

1.5. Justificación de la tesis

Este proyecto de tesis provee una alternativa para solucionar los problemas planteados en el análisis de la situación actual y se enfoca principalmente en los siguientes puntos:

- El sistema con una interfaz muy amigable para los empleados de las despachadoras aduaneras.

- El sistema ahorrará tiempo en el movimiento de los papeles para hacer algún trámite aduanero.
- La implementación de este sistema resulta escalable ya que se adapta a los recursos tecnológicos y a los diferentes entornos de trabajo de las organizaciones.

1.6. Posibles usuarios

Los posibles usuarios para este sistema son aquellas empresas que de una manera u otra, prestan servicios como despachadores aduaneros.

CAPÍTULO 2

2. FUNDAMENTOS TEÓRICOS

2.1. Concepto de MERODE

Aunque los conceptos básicos de la orientación a objetos (como encapsulación, polimorfismo, herencia) incrementan significativamente el mantenimiento del sistema, el efecto de modificaciones dentro del diseño no debe ser desestimado. Desde que el número de objetos en un sistema aumenta, la maniobrabilidad se dificulta rápidamente. Esto también atenta contra un mantenimiento fácil. Como resultado, un sistema orientado a objetos que sea grande es muy difícil de mantener. El principal objetivo en el desarrollo de MERODE⁴ es dar una solución para los

⁴ MERODE es Model-Driven, Existence-dependency Relationship, Object-oriented Development

problemas anteriormente mencionados: La falta de facilidades para garantizar la calidad de especificaciones, la implementación con tendencia a muchas técnicas de análisis orientado a objetos, y la falta de un mayor nivel de modularidad en el desarrollo orientado a objetos. Esto ha resultado en tres conceptos que son las tres características sobresalientes del MERODE:

- Un desarrollo orientado al modelo.
- Un concepto de interacción de objetos más abstracto
- Una definición formal de todas las técnicas que nos aseguren un nivel más alto en el control de calidad.

MERODE fue diseñado por el departamento de ciencias económicas aplicadas de la Universidad Católica de Lovaina, Bélgica. La metodología ha sido probada en un gran número de compañías en Bélgica y Holanda teniendo mucho éxito. La característica más sobresaliente de MERODE es que usa el principio de dependencia de existencia. Esta es la piedra angular. Contribuye a la integridad, flexibilidad y al fácil modelamiento, permitiendo al mismo tiempo el uso de métodos formales para mejorar la calidad del sistema resultante. Segundo, la metodología hace una diferenciación clara entre la especificación y la implementación. Un sistema específico en MERODE puede ser implementado de diferentes maneras,

usando cualquier clase de ambiente de software disponible. Además, la implementación es hecha por la transformación de la especificación, no por la elaboración. Finalmente, MERODE es diferente de la mayoría de otras metodologías orientadas a objetos ya que garantiza un gran grado de independencia entre el modelo del negocio y el modelo de la funcionalidad y también entre los componentes que forman el modelo de negocio y los componentes que forman el modelo de funcionalidad. Esta característica mejora la flexibilidad y extensibilidad del sistema resultante.

Las fases más importantes del proceso de desarrollo de MERODE son: La fase de especificación y la fase de implementación. La especificación del sistema está dividida de acuerdo a la estructura de los requerimientos del usuario.

Dentro del modelo de negocio y el modelo de funcionalidad, tenemos los siguientes pasos:

1. Fase del modelamiento de la empresa.
 - a. Modelamiento de Tipo de objeto de la empresa y tipo de eventos del negocio.
 - i. Identificación del tipo de objeto de la empresa.
 - ii. Identificación del tipo de eventos del negocio.
 - iii. Construcción del gráfico de dependencia de existencia.

- iv. Construcción de la tabla objeto-eventos.
 - b. Especificación de las restricciones.
 - c. Especificación de los vectores de estado y métodos objeto-evento.
 - i. Definición de los vectores de estado.
 - ii. Definición de los métodos objetos-eventos.
 - iii. Definición de restricciones de datos.
2. Fase del modelamiento de funcionalidad:
- a. Identificación de todos los servicios requeridos.
 - b. Por cada servicio:
 - i. Identificación de tipos de objetos de información.
 - ii. Para servicios de entrada: identificación de eventos generados del negocio.
 - iii. Construcción del diagrama de la especificación de servicio.
 - iv. Definición del vector de estado.
 - v. Definición de los métodos.

El proceso entero es dirigido principalmente por un sistema de reglas rigurosas de verificación que determinan la calidad de las especificaciones.

2.1.1. Tipos de objetos de la empresa y tipos de eventos del negocio

Debido al concepto de la comunicación por medio de eventos comunes, un modelo de la empresa no está solo integrado por objetos de la empresa, sino también por eventos del negocio.

El primer paso en el modelamiento de la empresa es la identificación de los tipos de objetos de la empresa y los tipos de eventos del negocio que ocurren en la realidad descrita y que, al mismo tiempo, son relevantes para que el sistema sea construido.

Para ser incluido en la lista de elementos relevantes, los tipos de objetos tanto como los tipos de eventos deberán satisfacer algunos requisitos importantes.

Para ser un candidato válido de tipo de objeto, un elemento debe satisfacer las características básicas: Un objeto tiene una identidad, tiene un estado descrito por medio de atributos y demuestra un cierto comportamiento. En el contexto de modelamiento de la empresa, el aspecto del comportamiento se puede formular en términos de la participación del evento. Además, los objetos de la empresa

tienen que corresponder a los conceptos del mundo real, que conduce a los criterios adicionales. Consecuentemente, los tipos del objeto de la empresa deberían satisfacer el siguiente grupo de requisitos:

- Un objeto se puede describir por un número de características. Por ejemplo, un cliente puede ser descrito por medio de un nombre, por la fecha de nacimiento, una dirección. Una fecha no es un objeto sino un elemento que describe por ejemplo, fecha de nacimiento describe al cliente.
- Un objeto tiene una identidad, no necesariamente explícita y visible al usuario. (En otras palabras, el concepto de identidad no es el mismo que el concepto de clave (primaria) en el modelo relacional).
- Un objeto corresponde a un concepto del mundo real: Tiene contrapartes del mundo real que pueden o no ser físicas. Por ejemplo, un "cliente" es un objeto en el mundo real. Un "préstamo" en una biblioteca no es un objeto físico, pero es un objeto del mundo real también. Cuando se pide prestado un libro, no se crea ningún objeto físico, pero sin embargo el préstamo existe. Este requisito es típico para el modelamiento de la empresa. Los objetos en el modelo de la

puesta en práctica no necesitan tener contrapartes en el mundo real.

- Un objeto existe por cierto período de tiempo. Por ejemplo, un préstamo tiene cierta duración: Comienza cuando se pide prestado el libro y termina cuando se devuelve el libro.
- Un objeto está siempre implicado en por lo menos dos eventos: Su creación y su finalización. El CLIENTE es un tipo de objeto; la "fecha" y la "edad" no son, porque no hay tipos de eventos tales como `create_date`, `end_of_date`, `create_age` o `end_of_age`.

Además, el tipo de evento indica un cambio de estado de un objeto. Un cliente tiene un estado y puede tener un cambio de estado. Pero un razonamiento similar no se aplica a la fecha ni a la edad.

El estado de un objeto no debería ser confundido con su "estado de ciclo de vida". El estado de un objeto se refiere al conjunto de los valores de los atributos de los objetos, mientras que el estado del ciclo vital se refiere explícitamente a los estados por los que pasan todos los objetos de un tipo de clase determinado durante su tiempo de vida, y estos son los que aparecen en las máquinas de estado finito y los diagramas de estado de Harel. Los tipos

de objeto son señalados por un sustantivo. Para el negocio, los requisitos para los tipos de eventos son:

-Un evento del negocio de cierto tipo ocurre o es reconocido en un punto en el tiempo; su duración posible puede ser abstraída. Un buen ejemplo es el acontecimiento "paga". Cuando incluimos "paga" en la lista de los tipos relevantes de eventos, se refiere al momento en el cual la actividad de pagar termina. Por ejemplo, consideremos un pago en efectivo. Tal actividad durará cierto rato: El cliente tiene que tomar su cartera, dinero de la cuenta, pero la duración real de todo esto no tiene ningún interés para el modelo.

En una perspectiva de modelamiento de empresa, el acontecimiento de la "paga" ocurre en el momento que la actividad de pagar termina. Por otra parte, la actividad de "poseer" una característica no es un acontecimiento: No se puede hacer la abstracción de la duración de la propiedad.

Un acontecimiento del negocio representa algo que sucede en el mundo real dentro del universo discutido; no es un acontecimiento del sistema de información.

Un evento del negocio es atómico, no puede ser separado en eventos secundarios. Consideremos por ejemplo una asociación donde los miembros pueden registrarse y pagar su inscripción en calidad de miembro más adelante.

Algunos miembros sin embargo se registran como miembro directamente pagando su inscripción en calidad de miembro. En ese caso, 'convertirse en miembro' es un tipo de evento pero 'convertirse en miembro pagando la inscripción en calidad de miembro' no es. Es una combinación de los tipos del acontecimiento previamente mencionados.

Los tipos de evento se deben señalar preferiblemente por un verbo en infinitivo antes que por un tiempo conjugado: pagar pero no pagos. Uno de los propósitos de modelar el alcance es determinar el área relevante para el modelamiento de la empresa. Los elementos fuera de este alcance no se conservan.

Otra consideración es que solamente los elementos que son necesarios para entregar la funcionalidad que tenemos en mente deben ser conservados. El modelo de la empresa determina totalmente qué funcionalidad puede ser entregada. En este sentido es importante incluir bastantes tipos de objetos y tipos de eventos. Pero la tenencia de muchos elementos aumentará el costo del desarrollo y por lo tanto no es deseable.

En el modelo de la empresa, los tipos de objetos de la empresa y los tipos de eventos del negocio se relacionan el

uno con el otro y se definen más detalladamente. El modelo de la empresa de MERODE consiste en tres sub-modelos:

Un esquema del gráfico de la dependencia de existencia (EDG) o de la relación entre objetos, una tabla del objeto-evento (OET) y un modelo del comportamiento.

En MERODE se requiere que todas las relaciones expresen dependencia de la existencia. Un objeto (llamado 'objeto subordinado o esclavo') es dependiente de la existencia en otro objeto (llamado el 'objeto principal o maestro') si la vida del primer objeto siempre se contiene completamente en la vida del último.

2.1.2. Especificaciones de las restricciones de secuencia

Los eventos no pueden ocurrir al azar durante el ciclo vital de un objeto. Están conforme a las restricciones de secuencia.

La tabla del objeto-evento especifica algunas de estas restricciones de secuencia. De hecho, por cada columna que representa un objeto debe existir la creación, modificación y terminación de los mismos. Luego del evento de finalización del objeto no puede existir ninguna otra clase de evento para ese objeto.

Un objeto tiene que ser creado antes de que pueda ser modificado y/o eliminado. Luego de que el ciclo vital de un objeto llegue a su finalización no debería existir ningún otro evento para ese objeto.

El ciclo vital de los objetos se puede representar por medio de las máquinas de estado finito. En una máquina de estado finito, los estados se representan como círculos y las transiciones como flechas etiquetadas con los nombres de los tipos de eventos. El estado inicial es indicado por un círculo que lleva una flecha entrante y el estado final como círculo con la línea doble.

Como la tabla del objeto-evento también contiene la indicación del tipo de implicación (creación, modificación o finalización), los diagramas JSD o las máquinas de estado finito serán solamente necesarios para la especificación de las restricciones adicionales que aún no son especificadas por la tabla objeto-evento.

De hecho, el tipo de implicación define un ciclo vital por defecto que es igual para cada tipo de objeto: un objeto puede comenzar su vida de unas o más maneras (una

selección de las C-entradas), él puede entonces experimentar un número arbitrario de modificaciones y estos eventos se permiten suceder en cualquier orden (iteración de una selección de las M-entradas) y finalmente la vida de un objeto se termina en una forma u otra (una selección de E-entradas).

Esta clase de ciclo vital por defecto se representa como diagrama JSD o como máquina de estado finito.

Puesto que todas las restricciones se aplican simultáneamente, no hay necesidad de duplicar tales restricciones de la secuencia en diagramas JSD múltiples. Esto simplifica el proceso de modelamiento substancialmente. Si las restricciones cambian, necesitamos solamente rediseñar un diagrama únicamente identificable.

2.1.3. Especificación de vectores de estado y métodos del objeto-evento

Una vez que los aspectos estáticos y dinámicos de los tipos de objeto se definan, el modelo de la empresa es extendido agregando cualidades a los tipos de objeto y a los tipos de

evento especificando qué le sucede a un tipo de objeto cuando un evento en particular se ejecuta.

La definición de cualidades se requiere para hacer objetos visibles. Asimismo, la definición de un método hace un evento visible. Una vez que se hayan definido las cualidades y los métodos es posible hacer un prototipo del modelo de la empresa. El resultado es la definición de una clase para cada tipo de objeto, que contiene un vector (conjunto de atributos) para ese tipo de objeto y un método para cada tipo de evento que estén vinculado con el tipo de objeto.

Una vez que los atributos se hayan definido, el modelo de la empresa se puede extender con reglas adicionales del negocio. Estas reglas pueden ser restricciones (limitantes) de los atributos, condiciones previas para los tipos de evento (con excepción del estado actual del ciclo de vida).

2.2. Comprobación de consistencia entre los 3 esquemas de modelamiento de la empresa⁵

⁵ Monique Snoeck, Guido Dedene, Object- Oriented Enterprise Modelling with MERODE

El gráfico de dependencia de la existencia es una de las características que hace único a MERODE comparado a otros métodos de análisis orientado a objetos.

Este gráfico es la parte vital porque el control de consistencia entre los tres esquemas se logra mediante el aseguramiento de la consistencia entre el gráfico de la dependencia de la existencia y cada uno de los otros sub-esquemas.

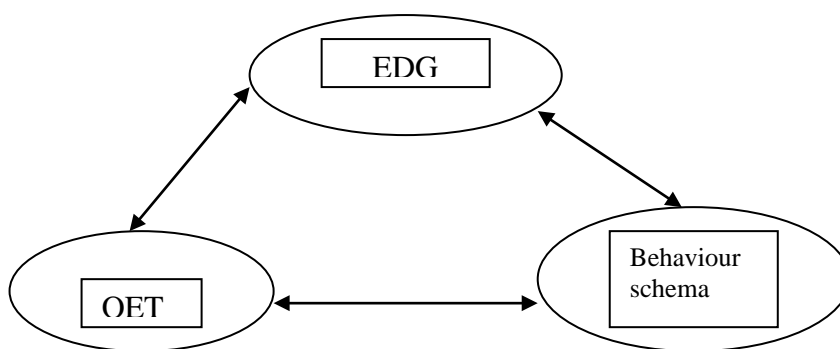


Figura 2.1 Comprobación de consistencia entre los 3 esquemas de modelamiento de la empresa

Usando el gráfico de dependencia de la existencia como la clave para el control de la consistencia, es posible lograr una mejor calidad en esquemas de empresa que cuando solamente se controla la concordancia sintáctica entre los sub-esquemas.

2.2.1. El modelo de funcionalidad

El modelo de la funcionalidad define como las funciones del sistema de información le dan al usuario final del sistema los

datos que se deben ingresar y la forma como se mostraran los resultados. Las funciones se diseñan como capas sobre el modelo de la empresa.

Se basan en dos tipos de herramientas:

Inspección del estado: Las funciones pueden examinar el valor de los atributos del objeto para darle al usuario los resultados en la forma de pantallas predefinidas o de informes impresos.

Generación de eventos del negocio: Las funciones pueden proporcionar la forma como se deben ingresar los datos a los usuarios finales generando eventos de las transacciones que se transmiten al modelo de la empresa.

Las funciones de entrada serán, a menudo, de una naturaleza interactiva: para aumentar la amigabilidad de la interfaz, las funciones pueden ser provistas para permitir a los usuarios buscar objetos existentes. Por ejemplo, una función de entrada que permita el cambio de dirección de un cliente que exista. Una función puede ser implementada con una lista que permita a los usuarios buscar a los clientes que se les vaya a cambiar la dirección. Aunque el registro de cambio de dirección es una función de entrada, la lista para escoger el cliente es de hecho una función de salida. Por lo

tanto las funciones interactivas son, a menudo, una combinación de los servicios de salida y de entrada.

Otra manera de separar funciones es mediante el iniciador: las funciones se pueden iniciar por algún agente externo (usualmente el interfaz de usuario o un contador de tiempo) o son iniciadas por el modelo de la empresa. La iniciación de una función depende del estado interno de los objetos del modelo de la empresa.

Las funciones pueden examinar el estado de objetos para recuperar la información. Esto se representa por medio de un diamante (rombo) conectado con la función y al tipo de objeto. Los eventos del negocio generados son representados mediante círculos.

Mientras que la mayoría de los métodos de análisis orientados a objeto solo discuten cómo ir desde el análisis orientado a objeto al diseño orientado a objeto, es posible implementar especificaciones orientadas a objeto en un ambiente convencional.

En la transformación a los ambientes de la implementación, se toma la tabla del objeto-evento como un punto de partida, para la implementación en ambientes convencionales y en tecnología orientada a objetos.

Si los cortes horizontales se toman en la tabla del objeto-evento, obtenemos un procedimiento por evento que se puede implementar en tecnología convencional, almacenando el estado del objeto en una base de datos que sea accesada por estos procedimientos.

La implementación en tecnología orientada a objetos es alcanzada tomando un corte vertical dentro de la tabla del objeto-evento: cada columna tiene la definición de la clase de un objeto de la empresa.

2.2.2. Descripción de la vista estática, dinámica y de la interacción

MERODE está hecho para el desarrollo orientado a Objetos. Es una metodología para modelar pero orientada a objetos de la empresa, que ha crecido fuera de la investigación sobre enfoques que modelaban semánticamente, el desarrollo de los sistemas de Jackson y el análisis orientado a objetos.

Las características que distinguen lo mejor de esta metodología son su orientación específica al dominio que modela, el uso de la dependencia de la existencia al modelar los aspectos estáticos del modelo del dominio, y el enfoque orientado a eventos para modelar el comportamiento. Por

medio de un ejemplo ilustraremos cómo una especificación puede terminar hasta cierto punto y cómo esta consistencia automatizada por la construcción contribuye a la validez y a tener especificaciones completas. Un modelo de MERODE consiste de 3 sub-vistas:

- Un gráfico de la dependencia de la existencia (EDG) que organiza el objeto de la empresa según dependencia y herencia de la existencia,
- Una tabla del objeto-evento (OET), que identifica los tipos de eventos del negocio y relaciona éstos con los tipos de objeto de la empresa,
- Un modelo del comportamiento, que consiste en una máquina de estado finito (FSM) por objeto de la empresa.

La semántica del EDG, de OET y del FSM y la consistencia de la visión se han definido por medio de álgebra de procesos. Existe también un sistema de reglas de comprobación de consistencia que prevé un cierto chequeo básico de la integridad. La Fig.2.2 da una descripción de las opiniones y de las reglas.

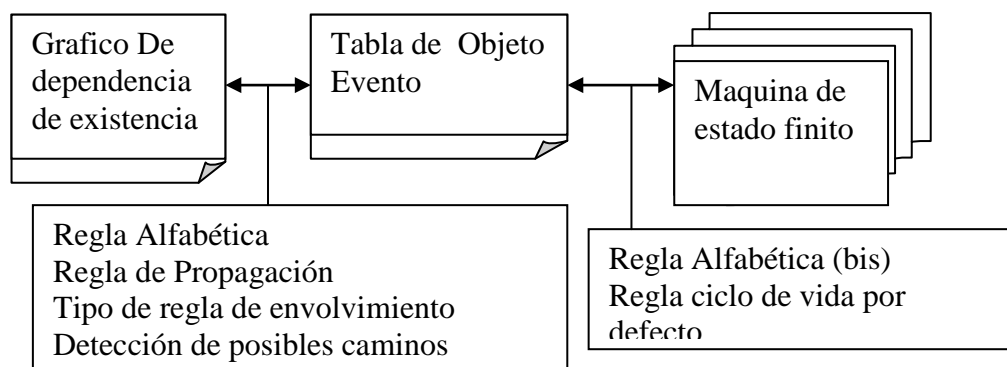


Figura 2.2 Vista de reglas que comprueban consistencia en MERODE

2.2.3. Consistencia por Construcción

2.2.3.1. Regla del Alfabeto

El alfabeto de una clase de objeto se va a definir mientras que todos los tipos de eventos estén marcados para un objeto dentro del OET. La regla del alfabeto indica que cada evento puede tener solamente un efecto en los objetos de una clase: el evento crea, modifica o suprime objetos. Además, la regla indica que cada clase de objeto necesita por lo menos un evento de creación y un evento de finalización.

2.2.3.2. Regla de Propagación y tipo de regla de implicación

Una segunda regla en la construcción del OET es la regla de la propagación. La regla de la propagación indica que cuando un objeto D es dependiente de la existencia de un objeto M, el último estará por defecto implicado en todos los eventos en que D esté implicado.

Esto significa que si la columna de un objeto D dependiente está marcada para un evento, la misma columna se deberá marcar en el objeto maestro M para el mismo evento.

Además, el tipo de regla de la implicación indica que un objeto dependiente no puede comenzar a existir antes que su maestro, un evento crear para una clase dependiente es un tipo de evento "de crear" o "de modificación" para la clase principal. Un tipo de evento "de modificación" para una clase dependiente es también un tipo de evento "de modificación" para su clase principal. Y finalmente, puesto que un dependiente no puede sobrevivir después de su maestro, un tipo de evento "finalizar" para un

dependiente es un tipo de evento "finalizar" o un tipo de evento "de modificación" para su maestro.

Para saber que tipo de participación tiene un objeto maestro sobre los objetos dependientes en el OET se colocará el tipo de evento precedido por una "A/" (de adquirido) y el último por un "O/" (de poseído u owned).

La ejecución y verificación de la propagación de manera manual es una tarea que desperdicia mucho tiempo, especialmente para proyectos grandes. Un case-tool sin embargo, puede generar fácilmente todas las participaciones propagadas.

El OET se puede modificar independientemente del EDG, pero en este caso, la consistencia se hace cumplir siempre que sea posible. Al agregar un tipo de objeto, el OET agregará el mismo tipo de objeto dentro del EDG aunque no será relacionado con otros tipos de objeto (en el EDG).

Los eventos se pueden agregar en el OET y para ellos podemos agregar los métodos poseídos (OWNED), que

serán propagados automáticamente. Los métodos adquiridos no pueden ser agregados o ser quitados.

2.2.3.3. Detección de posibles trayectorias redundantes

Las trayectorias que se ensamblan en el EDG ocurren cuando un objeto maestro se puede relacionar con un objeto dependiente siguiendo dos diversas trayectorias.

Por ejemplo, la administración de un proyecto es extendido con la facturación como se muestra en la figura 2.3. Durante su asignación a un proyecto, cada empleado puede colocar las horas realizadas para el proyecto.

Este registro de tiempo es incluido en una factura al fin del mes como línea de la factura.

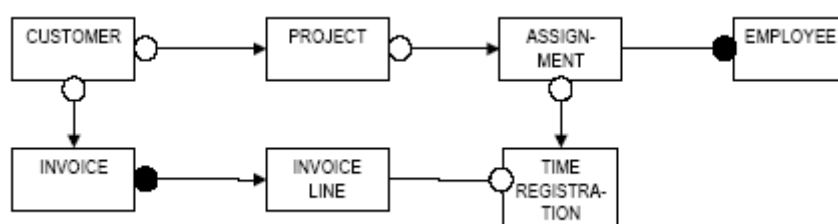


Figura 2.3 EDG extendido para la administración de proyecto

Yendo a lo largo de las relaciones de dependencia de la existencia desde el objeto dependiente al objeto

maestro, el objeto CLIENTE puede ser relacionado de dos maneras desde la clase LÍNEA de la FACTURA:

LINEA FACTURA»FACTURA»CLIENTE;

Y

LINEA FACTURA»TIMEREGISTRATION»ASSIGNMENT»
PROYECTO»CLIENTE.

Aplicando la regla de la propagación en el OET se identifican estas clases de uniones: las uniones (de ruta) conducen a las propagaciones múltiples en el OET. En el ejemplo, el objeto CLIENTE adquirirá los eventos de la línea de la factura dos veces, una vez a través de cada trayectoria. Identificar como es la trayectoria es importante puesto que uno debe contestar a la pregunta si uno o dos clientes están implicados en una línea de factura, en otras palabras ¿el cliente para quien el trabajo fue hecho (es decir, el cliente está conectado con el proyecto conectado con la línea de la factura vía la asignación y el timeregistration) es la misma persona a la que enviamos la factura? Si éste es el caso, la participación doble es substituida por una sola

participación y una limitante (un invariante) se agrega en la clase LÍNEA de la FACTURA:

```
self.INVOICE.CUSTOMER  
= self.TIME_REGISTRATION.ASSIGNMENT.PROJECT.CUSTOMER
```

2.2.3.4. Regla del ciclo de vida por defecto

La regla del alfabeto también indica que el FSM que define el comportamiento de un objeto P debe contener todos y solamente los eventos para los cuales haya una ' C ', ' M ' o ' E ' en la columna de P que se encuentra en el OET. Además, la secuencia impuesta por el FSM no debe violar el ciclo de vida por defecto de los eventos “de crear”, “de modificar” y “de finalizar”. Por lo tanto, según estas reglas, un FSM por defecto puede ser generado para cada objeto. El FSM puede ser refinado agregando nuevos eventos y estados del objeto.

Por ejemplo, un FSM puede ser derivado automáticamente del OET para el objeto TIME_REGISTRATION. Este FSM puede ser refinado más a fondo, por ejemplo para asegurar que un registro del tiempo no pueda ser modificado una vez que se haya facturado. La herramienta case asegura

que en cualquier momento, un evento "crear" sea usado solamente para una transición que sale del estado inicial, de que un evento "modificar" esté asociado solamente a las transiciones entre los estados intermedios y que un evento "finalizar" esté asociado solamente a las transiciones que terminan en un estado final.

Integridad

Tradicionalmente, el modelar se ve como el mapeo de un área o de una parte del mundo real.

Desde esta perspectiva, la validez (Mapeo de un área) significa que todas las declaraciones hechas por el modelo son correctas y relevantes al problema, mientras que lo completo (Parte del mundo real) significa que el modelo contiene todas las declaraciones sobre el dominio que son correctas y relevantes.

Las exigencias del consumidor sirven como punto de referencia para comprobar que un modelo se encuentra completo. Por lo tanto, cuando un usuario

no tiene más requerimientos se considera a menudo que el modelo está completo.

Desafortunadamente, los usuarios no entienden a menudo los modelos de datos. Es por esto que es casi imposible de comprobar que un modelo se encuentre completo si solo fue revisado por los usuarios.

La generación automática de las especificaciones que se pueden deducir de las que ya existen simplificará la comprobación para saber si los requerimientos han sido completados y esto está conectado con la semántica de las técnicas para modelar conceptualmente. Por ejemplo, si reconsideramos el diagrama de la clase de figura 2.3 es semánticamente correcto pero incompleto pues algunos requisitos relevantes no fueron identificados ni fueron incorporados explícitamente en el modelo. Es posible agregar una nota o un estereotipo para expresar las diferencias entre las dos asociaciones o bien para expresar la diferencia en el modelo del comportamiento.

El punto importante está, sin embargo, en que la técnica de diagramación "no hace cumplir" a un ingeniero en pensar en la diferencia de los requisitos: no ayuda en descubrir el estado incompleto del modelo. Transformando este gráfico en un EDG, el estado incompleto es resuelto, dando por resultado un modelo que sea más completo semanticamente. En el ejemplo de la administración de proyectos, la transformación del diagrama de clases a un EDG conduce a la creación del objeto ASIGNACIÓN por el ejemplo de la administración de proyectos.

Posteriormente, la regla del alfabeto requiere la definición de eventos de tipo "crear" y de tipo "finalizar" para el objeto ASIGNACIÓN, para saber a quien se asigna y a quien se quita proyectos. Estos eventos permiten especificar bajo qué condiciones se permite que se asignen y se quiten proyectos al empleado. La generación automática de estos dos eventos ayuda a completar el modelo. En el UML original, nada le señalará al ingeniero de requerimientos que considere modelar estos eventos.

La regla de propagación se muestra a continuación. El objeto dependiente no puede existir fuera de la vida del objeto maestro y cualquier cosa que sucede al objeto dependiente también afecta al objeto maestro, por lo menos indirectamente.

Notificando al objeto maestro de la ocurrencia de los eventos en sus objetos dependientes, la clase principal puede hacer una cierta contabilidad (e.g. en el objeto EMPLEADO cuenta el número de proyectos que un empleado ha trabajado), o hacer cumplir algunos requisitos (e.g. el objeto PROYECTO puede fijar como condición previa para el evento "asignar" que el estado del proyecto no se debe "cerrar").

La regla de propagación ilustra cómo la generación automática de las participaciones del objeto-evento hace a las especificaciones más completas: propagando un tipo de evento en las participaciones, todos los lugares posibles para las definiciones de los limitantes (constraints) y además se identifica la reunión de la información.

De esta manera, se invita al ingeniero de requerimientos que considere todos los elementos para la inclusión de reglas de negocio potenciales. Cuando se ha hecho el levantamiento de todos los requerimientos, algunas de las células marcadas no pudieron tener ningún cuerpo del limitante (constraint) o del método asociado a ellas.

Esas participaciones se pueden quitar fácilmente antes de la implementación.

Finalmente, el OET proporciona un mecanismo automático para identificar la unión de la trayectoria, que alternadamente conduce a la identificación de requisitos relevantes en el dominio.

2.3. Herramientas de MERODE⁶

2.3.1. El gráfico de la dependencia de existencia

Veamos el diagrama de clase en la figura 2.4. Representa una situación donde un documento tiene exactamente un

⁶ Monique Snoeck, Guido Dedene, Object-Oriented Enterprise Modelling with MERODE

código liberatorio. Cada código liberatorio puede ser aplicado para uno (1) o varios documentos.

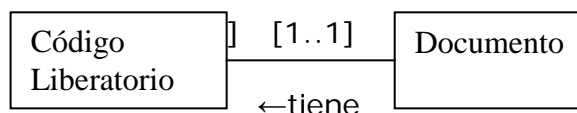


Figura 2.4 Liberación de Impuestos

Aunque la asociación parece estar correcta en su representación gráfica, hay una cierta diferencia substancial en la semántica. De hecho, cada código liberatorio es aplicado a un documento a la vez, pero los documentos dentro de un cierto plazo pueden tener varios códigos liberatorios.

Es decir la asociación "tiene" es modificable. Por lo tanto el diagrama en la figura 2.4 se puede considerar semánticamente incompleto: algunas declaraciones relevantes sobre el dominio no se han expresado. En MERODE, se requiere transformar un diagrama de la clase en un gráfico de la dependencia de la existencia (EDG). En tal gráfico, todos los objetos son solamente las asociaciones directas relacionadas que expresan dependencia de la existencia. Según las definiciones formales en MERODE, una clase D es dependiente de la existencia de una clase M si y solamente si la vida de cada ocurrencia de la clase D se

encaja en la vida de una sola y siempre de la misma ocurrencia de la clase M. D se llama la clase dependiente y es dependiente de la existencia de M, llamado la clase maestra. Una manera más informal de definir dependencia de la existencia es como sigue: si cada objeto de una clase D refiere siempre al mínimo uno, el máximo uno y siempre la misma ocurrencia de la clase M, después D es dependiente de la existencia de M. La dependencia de la existencia es equivalente a la noción de la entidad débil. Para evitar la confusión con un diagrama estándar de la clase de UML, las aplicaciones de MERODE tienen una notación propietaria con los puntos y flechas para definir la cardinalidad de la relación de la dependencia de la existencia. Este cardinalidad define cuántas ocurrencias del objeto dependiente pueden ser dependientes de un objeto maestro en un momento de tiempo pues la cardinalidad de la clase principal es siempre exactamente una (se asocia cada dependiente a exactamente un maestro), sólo la cardinalidad para el dependiente necesita ser especificada.

Una punta de flecha significa que el maestro puede tener varios dependientes simultáneamente mientras que una línea recta limita la cardinalidad máximo a uno.

Un punto blanco significa que eso de tener un dependiente es opcional para el maestro, mientras que un punto negro impone un límite mínimo de uno (el maestro tiene por lo menos un dependiente en cualquier momento).

La transformación del diagrama de la clase de figura 2.4 da lugar al EDG de figura 2.5. La relación "tiene" no representa dependencia de la existencia. Un código liberatorio puede existir fuera del contexto de un documento y un documento puede existir fuera del contexto de un código liberatorio. Cuando una asociación no expresa dependencia de la existencia, la asociación se basa en un objeto que sea dependiente de la existencia de todos los objetos que participan en la asociación.

Esto significa que la asociación "tiene" está dada sobre un objeto "ASOCIACIÓN DOC-COD.LIB", que es dependiente de la existencia del DOCUMENTO y CODIGO LIBERATORIO. MERODE llama a este tipo de clase intermedia una clase "contrato", modela qué puede suceder durante el período de tiempo en el que se relacionan un código liberatorio y un documento.

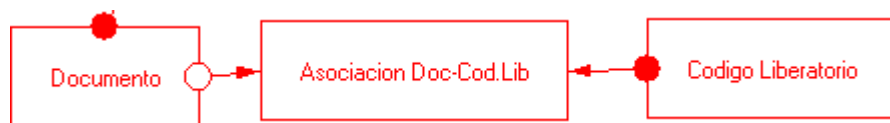


Figura 2.5. Gráfico de la dependencia de la existencia por el ejemplo de la liberación de impuesto.

2.3.2. La tabla de objeto-evento

En el caso de modelamiento conceptual orientado a objetos, los requisitos del dominio serán formulados en términos del negocio o los tipos de objetos de la empresa, asociaciones entre éstos y el comportamiento de los objetos del negocio. La definición del comportamiento deseado del objeto es una parte esencial en el proceso de la especificación. En un lado, tenemos que considerar el comportamiento de objetos individuales. Estos tipos de comportamiento serán especificados como métodos y diagramas de estado para las clases del objeto. Por otra parte, los objetos tienen que colaborar y trabajar recíprocamente.

Las técnicas típicas para modelar aspectos de la interacción de objetos son diagramas de interacción o cartas de la secuencia, y diagramas de colaboración. En la mayoría de los enfoques orientado a objetos, los eventos se consideran tan dependientes a los objetos, porque sirven solamente como un disparador para el método de un objeto. Las

interacciones de los mismos objetos se modelan por medio de diagramas de secuencia y/o de colaboración.

En contraste, MERODE sigue un enfoque orientado a eventos que da a los eventos el mismo nivel de importancia de los objetos, y que los reconoce como parte fundamental de la estructura del modelo.

Un evento del negocio se define como unidad atómica de acción que representa algo que sucede en el mundo real, tal como la creación de un nuevo código liberatorio, de una colocación a un documento, etc. Los eventos del negocio reflejan cómo los objetos del dominio vienen dentro de la existencia (los eventos que crean), cómo se modifican los objetos del dominio (los eventos de modificación), y cómo desaparecen del universo del discurso (los eventos de finalización). La interacción del objeto puede ahora ser modelada definiendo qué objetos están implicados concurrentemente en qué eventos. Las participaciones del Objeto-evento se denotan por medio de una tabla del objeto-evento (OET). La tabla de Objeto-Evento es una tabla con una fila para cada evento y una columna para cada objeto. Cada celda indica si un objeto está inmerso en un evento o

no. Cada celda de la tabla contiene un espacio en blanco, una C (crea ocurrencias), una M (modifica ocurrencias) o una E (finaliza ocurrencias).

Cuando un objeto participa en un evento, pone un método en ejecución para que defina el efecto del evento en el objeto. En la ocurrencia del evento todos los métodos correspondientes a los objetos que participan se ejecutan en paralelo.

Así, en vez de modelar una secuencia compleja de las invocaciones del método, ahora se asume que todos los métodos están siendo ejecutados concurrentemente. El OET para el ejemplo de la liberación de impuestos se da en la tabla 1. Las reglas que gobiernan la construcción de esta tabla se describen en la sección siguiente.

2.3.3. Máquinas de estado finito

Finalmente, el ciclo vital de cada clase de objeto de la empresa se modela por medio de una máquina de estado finito (FSM). Los eventos de la tabla del objeto-evento se utilizan como disparadores para las transiciones en la máquina de estado finito.

Como ejemplo, la figura 2.6 demuestra a FSM para el CODIGO LIBERATORIO. De igual manera, un FSM puede ser

definido para las clases CODIGO LIBERATORIO, ASOCIACION
DOC-COD.LIB y DOCUMENTO.

Procesos	Exportacion	Importacion	Pais	Convenio	Cliente	Documento	Asociacion Doc-Cod-Lib	Codigo Liberatorio	Producto	Partida Arancelaria	Regimen	Reg-Export	Reg-Import	Asoc-Exp-Reg	Asoc-Imp-Reg	Factura	Asoc-Proc
crear_proceso	O/C	I/C	I/C	A/M		A/M			A/M							A/M	
cr_Exportacion		S/C		A/M		A/M			A/M							A/M	
cr_Importacion			S/C	A/M		A/M			A/M							A/M	
fin_proceso	O/E	I/E	I/E	A/M		A/M			A/M							A/M	
crear_pais				O/C													
fin_pais				O/E													
crear_convenio				A/M	O/C												
fin_convenio				A/M	O/E												
crear_cliente						O/C											
fin_cliente						O/E											
fin_exportacion		O/E		A/M		A/M			A/M							A/M	
modificar_datos		O/C	O/C	A/M		A/M			A/M							A/M	
				A/M		A/M			A/M							A/M	
generar_dau		O/C	O/C	A/M		A/M			A/M							A/M	
				A/M		A/M			A/M							A/M	
fin_importacion			O/E	A/M		A/M			A/M							A/M	
generar_dui				O/C	A/M	A/M			A/M							A/M	
generar_dav			O/C	A/M		A/M			A/M							A/M	
crear_documento						O/C											
fin_documento						O/E											
cambio en documento						O/M											
crear_asoc.doc-cod						A/M	O/C	A/M	A/M	A/M							
fin_asoc.doc-cod						A/M	O/E	A/M	A/M	A/M							
crear_codigo								O/C	A/M	A/M							
fin_codigo								O/E	A/M	A/M							
cambio_en_codigo								O/M	A/M	A/M							
crear_producto									O/C								
fin_producto									O/E								
crear_partida									A/M	O/C							
fin_partida									A/M	O/E							
cambio_partida									A/M	O/M							
crear_regimen											O/C	I/C	I/C				
cr_Reg_Export												S/C					
cr_Reg_Import													S/C				
fin_regimen											O/E	I/E	I/E				
crear_asoc.export-reg		A/M		A/M		A/M			A/M			A/M		O/C		A/M	
fin_asoc.export-reg		A/M		A/M		A/M			A/M			A/M		O/E		A/M	
fin_asoc.import-reg			A/M	A/M		A/M			A/M				A/M		O/E	A/M	
crear_asoc.import-reg			A/M	A/M		A/M			A/M				A/M		O/C	A/M	
crear_factura																O/C	
fin_factura																O/E	

Tabla 2.1 Tabla del Objeto-Evento del sistema

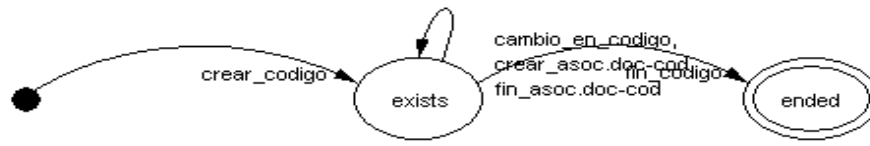


Figura 2.6 FSM del Código Liberatorio

CAPÍTULO 3

3. ANÁLISIS DEL SISTEMA

3.1. Requerimientos funcionales

De acuerdo a entrevistas a los usuarios finales del sistema, y a investigaciones realizadas que adicionalmente completan el esquema de trabajo que deberá desarrollar el mismo, se describen a continuación los requerimientos que la aplicación debe satisfacer desde el punto de vista del negocio.

Se deberá desarrollar un sistema del tipo aplicación ejecutable que sea capaz de solicitar a un Agente de Despacho, conocedor del área de declaración aduanera, la información pertinente a cualquier

trámite de importación o exportación de mercadería, sujeto a cualquiera de los regímenes ya establecidos por la Corporación Aduanera Ecuatoriana (C.A.E.), acogiéndose a los estándares impuestos por la Institución mencionada. Posteriormente, esa información deberá ser guardada adecuadamente en una base de datos de manera que en cualquier momento pueda ser recuperada enteramente para consultas, y bajo ciertas condiciones, editada o cambiada. Finalmente, en base a la información solicitada previamente, el sistema debe estar en capacidad de generar los documentos concernientes al trámite en cuestión a fin de completar la declaración aduanera. Para ello, la información solicitada debe ser concisa y no redundante, para que no necesite ninguna información adicional para generar los mencionados documentos. Esto no siempre se da, ya que por lo menos en documentos como el DUI o el DAU, se necesita información de sus respectivos refrendos, que ineludiblemente se ingresa luego de que son generados.

Adicionalmente, el sistema debe permitir la actualización de información compartida por todos los usuarios. Tal es el caso de la información de los clientes, por ejemplo. Se debe brindar la posibilidad de añadir y modificar los datos de un cliente y esta

nueva información estar disponible para que los digitadores la usen en sus trámites.

A continuación se detalla la funcionalidad del sistema en términos de las tareas fundamentales que éste debe realizar.

El sistema deberá cumplir con los siguientes requisitos:

PARA LAS TAREAS DE LOS AGENTES: El sistema debe cumplir con las siguientes tareas eventualmente encomendadas por los agentes, declarantes o digitadores:

Ingreso de información para el trámite aduanero. El sistema debe solicitar al operador todos los datos necesarios, como por ejemplo: el cliente, el proveedor, la mercadería a importar o exportar el régimen al cual se acoge, el banco a realizar el pago de impuestos, etc. que servirán para crear todos los documentos correspondientes a la declaración aduanera.

Consulta de un trámite. En cualquier momento y en cualquier circunstancia un digitador puede buscar un trámite guardado y examinar todos y cada uno de los datos ingresados, así como los

documentos generados que hayan sido guardados en el sistema, sin necesidad de volverlos a generar.

Modificación de la información ingresada. Bajo la condición de que la declaración no haya sido aceptada por la C.A.E. usando la información hasta ese momento ingresada, con el cual se da por finalizado el trámite aduanero en cuestión, el agente puede cambiar cualquier dato en el trámite, manteniendo la suficiente coherencia con el resto de información. Ello implica que la información tiene que estar interrelacionada entre sí desde el principio del trámite hasta el final.

Generación de documentos declaratorios. El sistema, en base de la información obtenida por el declarante, como ya se mencionó anteriormente, debe generar los documentos correspondientes que se necesita enviar a la CAE para su posterior refrendo. Entre estos documentos imprimibles están: el Documento Único de Importación (D.U.I.), La Declaración Aduanera Única (D.A.U.), y la Declaración Aduanera al Valor (D.A.V.). Estos documentos tienen formatos preestablecidos por la CAE y el sistema debe respetarlos al generarlos.

Generación de archivos para el Intercambio Electrónico con la CAE. Como un requerimiento específico del sistema, adicionalmente a los documentos impresos, se requiere generar un conjunto de archivos de texto con un formato definido por la CAE que contenga la información del trámite (al que se lo conoce también como DAU electrónico), y que esté listo para un posterior envío por e-mail a la aduana.

A NIVEL DE LOS ADMINISTRADORES: El sistema además de las tareas descritas, debe permitir como privilegios adicionales a los administradores las siguientes opciones:

Ingreso de nuevos clientes. El sistema debe incluir la posibilidad de añadir o modificar la información de clientes que posteriormente se utilizarán en algún trámite. La información que se requiere del cliente es la siguiente: Nombre, dirección, teléfono, fax, dirección e-mail, ciudad de residencia, CIIU (Clasificación Industrial Internacional Uniforme), y sector al que pertenece. Adicionalmente se debe indicar nombre de la empresa agente que realiza siempre la declaración aduanera a este cliente.

Ingreso de nuevos proveedores. Se debe ingresar la información de nuevos proveedores, o modificar la información de proveedores

existentes. La información concerniente al proveedor que se requiere es la siguiente: Nombre, dirección, teléfono(s), fax, e-mail, RUC, y ciudad de residencia, además del nombre del beneficiario del giro.

Ingreso de nuevos productos. La mercadería declarada tiene ciertas características comunes en todos los trámites que deben ser agrupadas por el sistema bajo un mismo ente; el cual, debe poder ser añadido y modificado, tal como en el caso de los clientes y proveedores, por un usuario administrador. La idea es ahorrar tiempo al operador de no ingresar la misma información que es común para un mismo tipo de productos.

Modificación de información de clientes. Así como es posible añadir un cliente nuevo en el sistema, también debe ser posible cambiar alguna información desactualizada de un cliente, tal como la dirección, un número de teléfono, etc. y estar disponible para que los demás usuarios puedan hacer uso de ella. Debe también actualizar la información de los clientes en trámites anteriores ya guardados en el sistema.

Modificación de información de proveedores. Si existe alguna información de los proveedores que necesite ser cambiada, como la

dirección o el teléfono del proveedor, o incluso un cambio en el nombre del beneficiario del giro, el sistema debe permitir modificarlo.

3.2. Requerimientos No Funcionales

Entre los requerimientos no funcionales que debe cumplir el sistema están:

Toda la información ingresada en cualquier momento, durante el ciclo de vida de un trámite, desde que se inicia hasta que culmina, debe ser coherente y consistente entre sí; tanto en lo que tiene que ver con la lógica del proceso en sí, como en lo que tiene que ver con consideraciones matemáticas y semánticas.

Se debe mantener una actualización constante de la información usada de manera general por todos los trámites. Tal es el caso de la información de clientes, proveedores y productos.

Los datos presentados a los usuarios deben ser correctos y no prestarse a confusión; especialmente en lo que concierne con el cálculo de impuestos y aranceles para la autoliquidación de tributos.

El sistema debe mantener la debida seguridad para que ningún usuario pueda realizar más tareas de las que su rol se lo admite. Para el caso de este sistema, los administradores tienen control total de cualquier funcionalidad del sistema, en cambio que los digitadores, agentes o declarantes, solo pueden controlar todo lo que tenga que ver con los trámites aduaneros que el sistema realiza; esto es, la creación, consulta, ingreso y modificación (mientras no haya sido cerrado) de los trámites.

Como un complemento a lo antes mencionado, cada funcionalidad que el sistema permita efectuar, debe contar con la mayor facilidad y simplicidad de uso, y en tal virtud aplique los conceptos de visibilidad, permisividad y retroalimentación visual; con el fin de suprimir en lo posible en los usuarios del sistema la necesidad de ser un experto en despacho aduanero, ni mucho menos en computación.

3.3. Análisis de la Interacción Hombre-Máquina

El sistema a desarrollarse, dentro de lo que concierne a su interfaz con el usuario, debe brindar facilidad y rapidez de uso a los despachadores de aduana en el momento de efectuar la declaración; los cuales se pueden clasificar en dos grandes grupos:

- Los primeros son aquellos que muy poco trabajan con una computadora, y si la utilizan no sería más que para realizar otras actividades que no sean relacionadas con su trabajo como despachador (leer correo electrónico, navegar por Internet, etc.). En cuyo caso, tendrían una leve noción de un ambiente de trabajo basado en ventanas y formularios; por tanto, el sistema debe facilitar la recolección de información que estos reciben, guiando al usuario a través de ellos sin impedirle la libertad de buscar otras maneras de trabajar con él.
- Los segundos son aquellos usuarios experimentados en sistemas basados en formularios, tales como secretarias y digitadores; los cuales muy poco se preocupan de cómo muestran la información en la pantalla, mientras tenga la suficiente rapidez para pedirla y procesarla. Generalmente estos usuarios por este motivo prefieren usar sólo el teclado y no el ratón, y esperan emplear la menor cantidad de combinaciones de teclas, que no sean las estrictamente necesarias para ingresar cada dato.

Tomando en cuenta lo anterior, el sistema debe cumplir eficientemente con los principios de usabilidad que dicta una buena interacción hombre-máquina, los cuales se analizan a continuación:

Consistencia: El sistema debe permitir al usuario aprender a realizar tareas nuevas, basado en experiencias previas con otras tareas. Para esto, las opciones deben tener patrones definidos y consistentes que hagan que el usuario se acostumbre a ellos y espere que todo el sistema responda de la misma manera. Tal es el caso de la ubicación de los componentes de las ventanas, barras de herramientas, etc.

Familiaridad: El sistema no debe hacer que el usuario que ha usado otras aplicaciones tenga que aprender de nuevo o modificar su percepción del mundo para acoplarse al programa. Este criterio no solo se refiere a reconocer dentro del sistema opciones comunes vistas en otros programas, sino también a identificar elementos que un despachador conoce en su trabajo diario, plasmados lo más parecido posible a la realidad en el sistema.

Flexibilidad: Debe existir en el sistema más de una manera de ejecutar cualquier opción. Con esto el usuario buscará la más rápida o cómoda a su juicio. El cumplimiento de este principio deberá resolver el problema de complacer a los dos tipos de usuarios descritos anteriormente.

Visibilidad: Ninguno de los dos usuarios mencionados espera buscar la opción que necesita para una tarea diaria, dentro de otras opciones; ya sea por desconocimiento del usuario novato, o por

necesidad de rapidez del usuario experimentado. Por ende, el sistema debe mostrar siempre las opciones más frecuentes y agrupadas en las tareas más comunes en la pantalla principal.

Tiempo de respuesta y retroalimentación: Al percatarse de que una acción dada por el usuario no ofrece el resultado que se espera en un tiempo prudencial, éste podría confundirse, especialmente si se trata de un usuario novato, y rechazar el programa. Lo mismo ocurre con el usuario experimentado, con más importancia al tiempo de respuesta; quien si bien puede acostumbrarse a cierto aspecto del sistema que no le agrada, tenderá a hacer comparaciones negativas con la forma como trabajaba anteriormente.

3.4. Modelos de Análisis

Los modelos y diagramas de análisis que se presentan a continuación permiten obtener una visión más clara del funcionamiento del Sistema de Despacho Aduanero.

3.4.1. Modelo MERODE

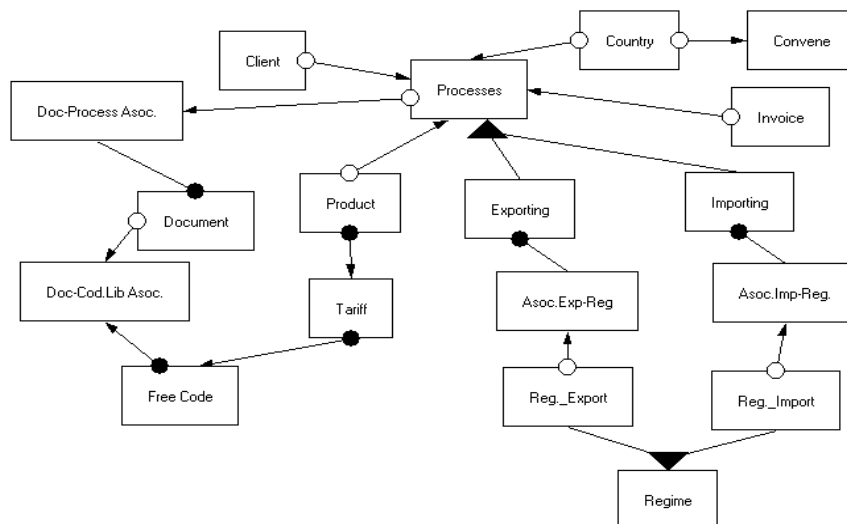


Figura 3.1 Modelo de Objetos del Sistema

En la figura 3.1 tenemos el modelo de objetos realizado con la técnica de MERODE, y del cual podemos sacar muchas conclusiones y restricciones:

El modelo consta de 18 objetos que se encuentran relacionados entre sí. Existen objetos que pueden ser maestros o dependientes.

A continuación se detallan las condiciones de los objetos, las relaciones y restricciones que tienen cada uno. Muchos objetos aparecen como maestros y dependientes a la vez.

Procesos:

Un objeto procesos puede tener muchos objetos de tipo Asoc. Doc-Proceso. Un objeto Procesos no puede finalizar mientras tenga objetos de tipo Asoc. Doc-Proceso.

Además, un objeto Procesos existe como dependiente en muchas relaciones del modelo. Un objeto procesos puede solo existir en el contexto de exactamente un objeto de tipo país. Un objeto procesos puede solo ser creado después que el objeto País al que se refiere haya sido creado. Un objeto procesos puede solo existir en el contexto de exactamente 1 objeto de tipo cliente. Un objeto procesos puede solo ser creado después que el objeto Cliente al que se refiere haya sido creado. Un objeto procesos puede solo existir en el contexto de exactamente 1 objeto de tipo Producto. Un objeto procesos puede solo ser creado después que el objeto Producto al que se refiere haya sido creado. Un objeto procesos puede solo existir en el contexto de exactamente 1 objeto de tipo Documento. Un objeto procesos puede solo ser creado después que el objeto Documento al que se refiere haya sido creado. Un objeto procesos puede solo existir en el contexto de exactamente 1 objeto de tipo Factura. Un objeto procesos puede solo ser creado después que el objeto Factura al que se refiere haya sido creado.

País:

Un objeto País puede tener muchos objetos de tipo Convenio. Un objeto País no puede ser finalizado mientras tenga dependencia de objeto Convenio. Un objeto País puede tener muchos objetos de tipo Procesos. Un objeto País no puede ser finalizado mientras tenga dependencia de objeto Procesos.

Convenio:

Un objeto Convenio puede solo existir en el contexto de exactamente 1 objeto de tipo País. Un objeto Convenio puede solo ser creado después de que el objeto País al que se refiere haya sido creado.

Cliente:

Un objeto Cliente puede tener muchos objetos de tipo Procesos. Un objeto Cliente no puede ser finalizado mientras tenga dependencia de objeto Procesos.

Exportación:

Un objeto Exportación debe tener 1 objeto de tipo Asoc.Exp-

Reg. Un objeto Exportación no puede ser finalizado mientras tenga dependencia de objetos Asoc.Exp-Reg.

Importación:

Un objeto Importación debe tener 1 objeto de tipo Asoc.Imp-Reg. Un objeto Importación no puede ser finalizado mientras tenga dependencia de objetos Asoc.Imp-Reg.

Documento:

Un objeto Documento debe tener muchos objetos de tipo Procesos. Un objeto Documento no puede ser finalizado mientras tenga dependencia de objetos Procesos. Un objeto Documento puede tener muchos objetos de tipo Asociación Doc-Cod.Lib. Un objeto Documento no puede ser finalizado mientras tenga dependencia de objetos Asociación Doc-Cod.Lib.

Asociación Doc-Cod.Lib:

Un objeto Asociación Doc-Cod.Lib puede solo existir en el contexto de exactamente 1 objeto de tipo Documento. Un objeto Asociación Doc-Cod.Lib puede solo ser creado después de que el objeto Documento haya sido creado. Un objeto Asociación Doc-Cod.Lib puede solo existir en el

contexto de exactamente 1 objeto de tipo Código Liberatorio. Un objeto Asociación Doc-Cod.Lib puede solo ser creado después de que el objeto Código Liberatorio haya sido creado.

Código Liberatorio:

Un objeto Código Liberatorio debe tener muchos objetos de tipo Asociación Doc-Cod.Lib. Un objeto Código Liberatorio no puede ser finalizado mientras tenga dependencia de objetos Asociación Doc-Cod.Lib.

Además de que el objeto Código Liberatorio exista como maestro en algunas relaciones hay en otras en las cuales aparece como dependiente.

Un objeto Código Liberatorio puede solo existir en el contexto de exactamente 1 objeto de tipo Partida Arancelaria.

Un objeto Código Liberatorio puede solo ser creado después de que el objeto Partida Arancelaria al que se refiere haya sido creado.

Producto:

Un objeto Producto puede tener muchos objetos de tipo

Procesos. Un objeto Producto no puede ser finalizado mientras tenga dependencia de objetos Procesos.

Un objeto Producto puede tener muchos objetos de tipo Partida Arancelaria. Un objeto Producto no puede ser finalizado mientras tenga dependencia de objetos Partida Arancelaria.

Partida Arancelaria:

Un objeto Partida Arancelaria puede tener muchos objetos de tipo Código Liberatorio. Un objeto Partida Arancelaria no puede ser finalizado mientras tenga dependencia de objetos Código Liberatorio.

Además de que el objeto Partida Arancelaria exista como maestro en algunas relaciones, también existe en otras relaciones como dependiente.

Un objeto Partida Arancelaria puede solo existir en el contexto de exactamente 1 objeto de tipo Producto. Un objeto Partida Arancelaria puede solo ser creado después de que el objeto Producto al que se refiere haya sido creado.

Reg._Export:

Un objeto Reg._Export puede tener muchos objetos de tipo Asoc.Exp-Reg. Un objeto Reg._Export no puede ser

finalizado mientras tenga dependencia de objetos Asoc.Exp-Reg.

Reg._Import:

Un objeto Reg._Import puede tener muchos objetos de tipo Asoc.Imp-Reg. Un objeto Reg._Import no puede ser finalizado mientras tenga dependencia de objetos Asoc.Imp-Reg.

Asoc.Exp-Reg:

Un objeto Asoc.Exp-Reg puede solo existir en el contexto de exactamente 1 objeto de tipo Reg._Export. Un objeto Asoc.Exp-Reg puede solo ser creado después de que el objeto Reg._Export al que se refiere haya sido creado. Un objeto Asoc.Exp-Reg puede solo existir en el contexto de exactamente 1 objeto de tipo Exportación. Un objeto Asoc.Exp-Reg puede solo ser creado después de que el objeto Exportación al que se refiere haya sido creado.

Asoc.Imp-Reg.:

Un objeto Asoc.Imp-Reg. Puede solo existir en el contexto de exactamente 1 objeto de tipo Reg._Import. Un objeto Asoc.Imp-Reg. Puede solo ser creado después de que el

objeto Reg._Import al que se refiere haya sido creado. Un objeto Asoc.Imp-Reg. puede solo existir en el contexto de exactamente 1 objeto de tipo Importación. Un objeto Asoc.Imp-Reg. . Puede solo ser creado después de que el objeto Importación al que se refiere haya sido creado.

Factura:

Un objeto Factura puede tener muchos objetos dependientes de tipo Procesos. Un objeto Factura no puede ser finalizado mientras tenga dependencia de objetos Procesos.

Asoc. Doc-Proceso:

El objeto Asoc. Doc-Proceso existe como dependiente en muchas relaciones del modelo. Un objeto Asoc. Doc-Proceso puede solamente existir en el contexto de exactamente 1 objeto de tipo proceso. Un objeto Asoc. Doc-Proceso puede ser solamente creado después de que el objeto Procesos al que se refiere haya sido creado. Un objeto Asoc. Doc-Proceso puede solamente existir en el contexto de exactamente un objeto de tipo Documento. Un objeto Asoc. Doc-Proceso puede ser solamente creado luego que el objeto Documento al que se refiere haya sido creado.

Restricciones

Las restricciones para cada objeto del modelo del sistema realizado con la técnica de MERODE son representadas por medio de Máquinas de Estado Finito. A continuación se describe cada máquina de estado finito de los objetos que se encuentran en el modelo:

Cliente.- El objeto cliente tiene un ciclo de vida por defecto. Al comenzar se crea el cliente dentro del sistema. En el momento en que ya existe el cliente, este será capaz de crear un proceso, este proceso podrá ser de exportación o de importación, de acuerdo a que proceso el cliente haya seguido se crearán diferentes asociaciones que podrían ser asociaciones de exportación con regimenes de exportación o la asociación de importación con regimenes de importación. Los clientes también pueden realizar modificaciones respecto a los documentos que se generarán de acuerdo al proceso realizado.

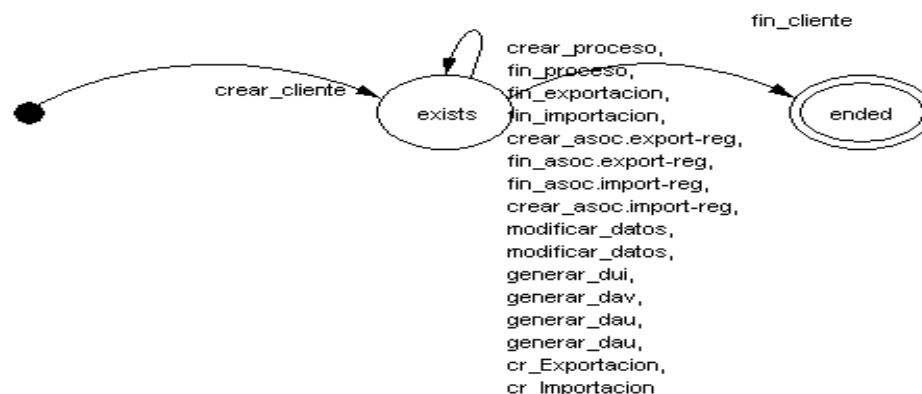


Figura 3.2 FSM del Cliente

País.- El objeto país tiene un ciclo de vida por defecto. Al comenzar el objeto país se crea en donde el cliente se encuentra. En el momento que este existe se creará un proceso por petición de un cliente y se creará la lista de convenios que existe con respecto al otro país con el cual se va a realizar el proceso, si existe algún convenio se tendrán que ejecutar las ordenes que se encuentran descritas en dicho convenio. En el momento que un proceso es creado, de acuerdo a que proceso se realice, se crearán diferentes asociaciones que ser de exportación con regimenes o de importación con regimenes. En el momento que finalice el proceso, se pasa al estado de fin del país, es decir en ese momento se destruye el objeto país.

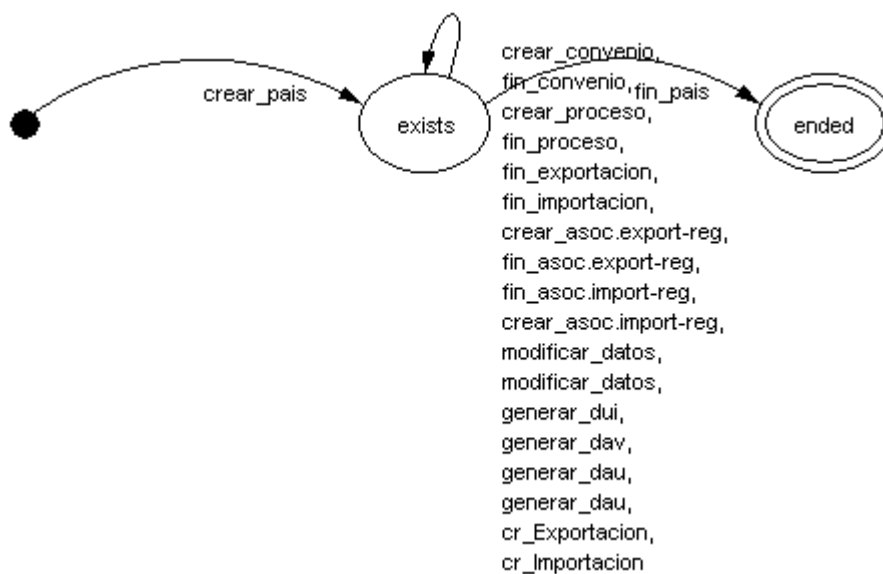


Figura 3.3 FSM del Objeto Pais

Convenio.- El objeto convenio tiene un ciclo de vida por defecto. El objeto convenio se crea con el evento crear convenio. En el momento que ya existe el objeto convenio, espera el tiempo que dura un proceso para ver si es necesario o no y luego el objeto se destruye.

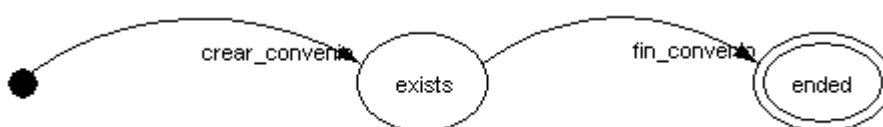


Figura 3.4 FSM del Convenio

Factura.- El objeto factura tiene un ciclo de vida por defecto. Al comienzo de este objeto se crea la factura por medio del evento crear factura. Cuando la factura existe se pueden dar diferentes eventos dentro de esta maquina de estado finito.

La factura existe dado que es un objeto que tiene un nivel de importancia considerable dentro de lo que es un proceso. Luego de que el objeto factura existe se crea el proceso, de acuerdo a que proceso se cree se crearán las relaciones de exportación con regimenes y de importación con regimenes. Si algún dato esta mal dentro de lo que al proceso relacionado con la factura se podrá hacer la generación de documentos relacionados con los procesos o/y modificación de datos de cualquiera de estos. Cuando se este llegando al fin de esta maquina de estado finito el objeto se destruirá con el evento fin_factura.

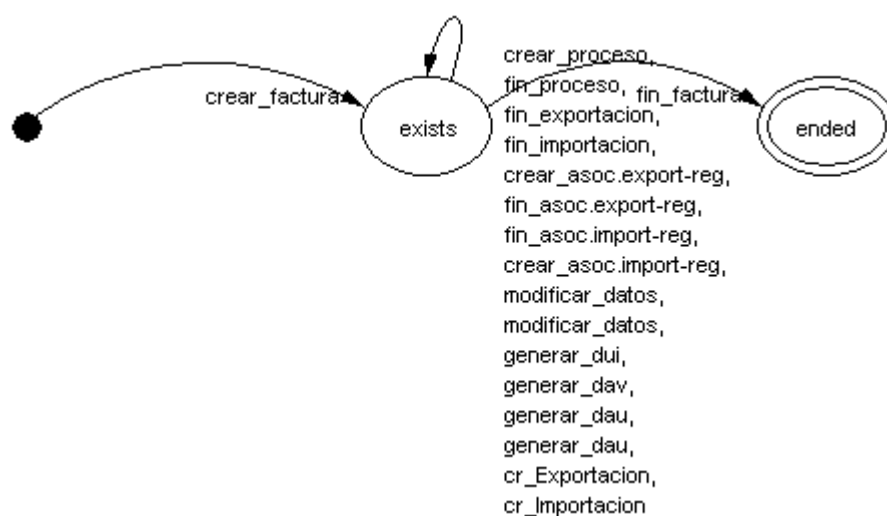


Figura 3.5 FSM del Objeto factura

Régimen.- El objeto Régimen tiene un ciclo de vida por defecto. El objeto régimen se crea con el evento crear régimen. En el momento que ya existe el objeto régimen,

espera el tiempo que dura un proceso para ver si es necesario o no y luego el objeto se destruye.

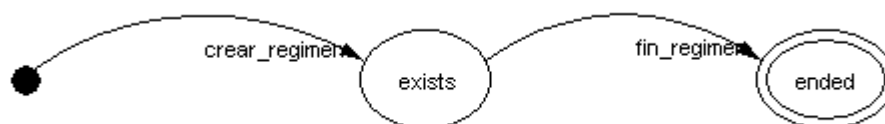


Figura 3.6 FSM del objeto Régimen

Reg Export.- El objeto Reg.Export tiene un ciclo de vida por defecto. Este objeto se crea mediante el evento crear reg. Export. En el momento que el objeto ya existe, se pueden realizar diferentes eventos como lo es la creación de asoc.export con regimenes. Cuando se ha realizado todas las tareas con este evento se pasa al evento de fin asoc.export con regimenes para luego poder cerrar este objeto con el evento fin de régimen.

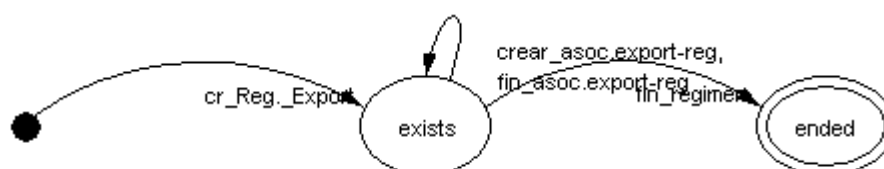


Figura 3.7 FSM del objeto Reg.Export

Reg.Import.- El objeto Reg.Import tiene un ciclo de vida por defecto. Este objeto se crea mediante el evento crear reg. Import. En el momento que el objeto ya existe, se pueden realizar diferentes eventos como lo es la creación de

asoc.import con regimenes. Cuando se ha realizado todas las tareas con este evento se pasa al evento de fin asoci.import con regimenes para luego poder cerrar este objeto con el evento fin de régimen.

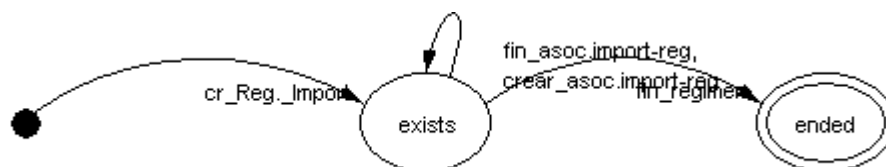


Figura 3.8 FSM del objeto Reg.Import

Asoc.Exp-Reg.- El objeto Asoc.Exp-Reg tiene un ciclo de vida por defecto. El objeto Asoc.Exp-Reg se crea con el evento crear Asoc.Exp-Reg. En el momento que ya existe el objeto Asoc.Exp-Reg, espera el tiempo que dura un proceso para ver si es necesario o no y luego el objeto se destruye por medio del evento fin asoci.export-reg y así se llega al fin de la maquina de estado finito para el objeto Asoc.Exp-Reg.

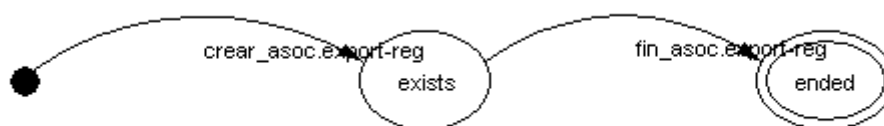


Figura 3.9 FSM del objeto Asoc.Export-Reg

Asoc.Imp-Reg.- El objeto Asoc.Imp-Reg tiene un ciclo de vida por defecto. El objeto Asoc.Imp-Reg se crea con el evento crear Asoc.Imp-Reg. En el momento que ya existe el

objeto Asoc.Imp-Reg, espera el tiempo que dura un proceso para ver si es necesario o no y luego el objeto se destruye por medio del evento fin asoci.import-reg y así se llega al fin de la maquina de estado finito para el objeto Asoc.Imp-Reg.

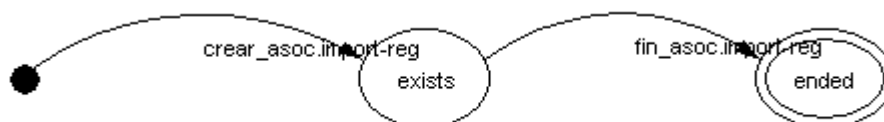


Figura 3.10 FSM del objeto Asoc.Import-Reg

Producto.- El objeto Producto tiene un ciclo de vida por defecto. El objeto se crea con el evento crear producto. Luego que este objeto ha sido creado, es decir, que ya existe tiene varias opciones de eventos para ejecutar. Luego del objeto creado se ejecuta el evento de crear proceso, según que proceso se siga, es decir, exportación o importación, se crea el código, después se crea la partida en la cual se encuentre un producto. En el momento de tener todos estos pasos realizados, se crea la relación asociación exportación y regimenes o relación asociación importación y regimenes. Ya habiendo creado se pueden ejecutar otras clases de eventos mientras el objeto producto este existiendo, eventos como lo son cambio en la partida, cambio en el código, o modificar datos que me servirá para cambiar los datos referentes a los productos dentro de los documentos generados como lo son

el DUI, dau, dav, etc. El objeto se destruye luego de que cambie del estado existir al estado finalizado, llega a este estado final en el momento que se ha ejecutado el evento fin de producto.

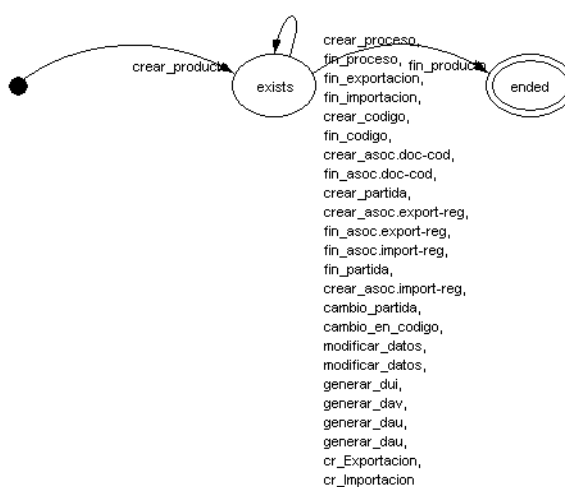


Figura 3.11 FSM del objeto Producto

Partida Arancelaria.- El objeto llamado partida arancelaria tiene un ciclo de vida por defecto. El objeto se crea gracias al evento crear partida. En el momento que el objeto es creado, pasa al estado "existe"; en este estado puede quedarse para realizar diversas clases de eventos como lo son: crear código, crear asoc.doc-cod, también se pueden realizar eventos para realizar modificaciones como lo son cambio en la partida y cambio en el código. Este objeto se termina en el momento que los eventos que existen para finalizar eventos

creadores se ejecuten. En el instante que no exista ya eventos ejecutándose, el objeto podrá pasar al estado de "finalizado" mediante la ejecución del evento fin partida que cerrará el objeto de Partida Arancelaria.

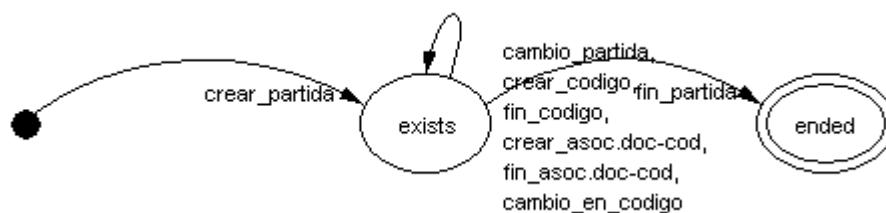


Figura 3.12 FSM del objeto Partida

Código Liberatorio.- El objeto código tiene un ciclo de vida por defecto. En el comienzo, el objeto código Liberatorio se crea mediante por el evento crear código. Cuando el objeto ha sido creado pasa al estado "existe", donde se podrán ejecutar varios eventos como lo es el crear asociación doc con código liberatorio. También se puede ejecutar el evento cambio en código, el cual se usará en el momento que se desea algún cambio dentro de los códigos liberatorios. Para poder pasar del estado existe al estado finalizado, se ejecuta el evento fin código.

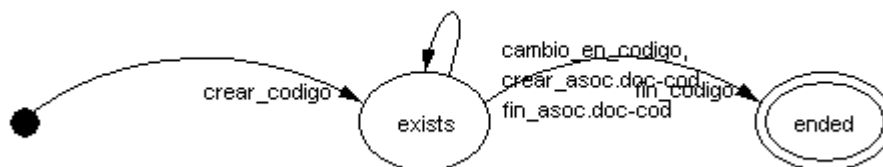


Figura 3.13 FSM del objeto Código

Asoc.Doc-Proceso.- El objeto Asoc.Doc-Proceso tiene un ciclo de vida por defecto. Asociación Doc-Cod.Lib.- El objeto asociación Doc-Cod.Lib tiene un ciclo de vida por defecto. En el comienzo, el objeto asociación Doc-Cod.Lib es creado mediante el evento crear asoc.doc-cod. En el momento en el cual el objeto es creado pasa al estado de "existe" en el cual se queda un momento corto y luego se ejecuta el evento fin asoc.doc-cod llevando así al objeto al estado "finalizado", el cual indica que la vida del objeto ha finalizado.

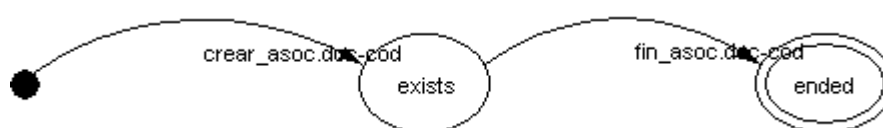


Figura 3.14 FSM del objeto Asoc.Doc-Proceso

Documento.- El objeto Documento tiene un ciclo de vida por defecto. En el comienzo, el objeto Documento se crea mediante el evento llamado crear documento, en el momento que el objeto Documento ha sido creado cambia al

estado "existe". En este estado, se pueden ejecutar diversas opciones de eventos como lo es el de crear asoc. Doc -cod, que lo que hace es crear la asociación entre documento y código liberatorio, y también el evento llamado cambio en documento, que se utilizará en caso que se desee realizar algún cambio en los atributos del objeto Documento. Para poder pasar del estado "existe" al estado "finalizado", se deberá ejecutar el evento fin asoc.doc-cod que cerrará al objeto Documento.

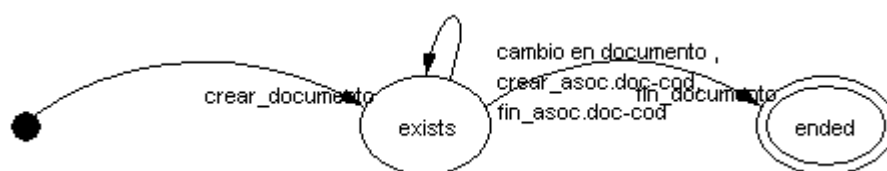


Figura 3.15 FSM del objeto Documento

Importación.- El objeto importación es creado en primera instancia por el evento crear importación. En el momento que el objeto importación ha sido creado, el objeto pasa al estado "existe". En este estado se pueden ejecutar varios tipos de eventos como lo son ingresar datos, si no se puede ejecutar algún evento de generación de documento por falta de algún dato, se ejecutará el evento ingresa datos; también existe el evento de modificar datos, con el cual se podrá modificar algún dato que se encuentre dentro del proceso de

Importación, el cual es representado por el objeto importación. El evento generación de DUI y el evento generación de DAU se ejecutan para que el objeto pase a cualquiera de los dos estados DUI's y DAU's. En el estado DUI se pueden ejecutar tres eventos que nos trasladarán a dos diferentes estados. El evento de modificar datos nos regresará al estado "existe", el evento generación de DAV que creara en el estado DUI una instancia de DAV y el evento generación de DAU que nos trasladará al estado DAU. En este estado se pueden ejecutar los eventos de modificar datos que nos regresará al estado de "existe", que viene hacer nuestro primer estado dentro de esta máquina de estado finito para el objeto Importación y el evento enviar que nos trasladará al estado de "enviado". En este estado solo existe el evento de recibir, el cual hace que el estado cambie a "recibido". En este estado existen dos diferentes caminos que escoger, si el objeto lleva los datos correctos se ejecutará el evento aceptar y nos llevará al estado "finalizado", en el cual el objeto se cerrará. En el caso que no sea satisfactoria la respuesta se ejecutará el evento de modificar datos, el cual nos llevará al estado de "existe" y seguirá la máquina ejecutándose hasta el momento que se llegue al estado "finalizado".

También se pudo haber escogido otro camino, en el instante en que ha sido creado el objeto Importación y se encuentra en el estado "existe" se puede ejecutar el evento generación de DAU, el cual hará que el objeto cambie de estado al estado DAU. En este estado existen dos caminos diferentes, de los cuales se escogerá uno según las características que en ese momento tendrá el objeto Importación. Se puede ejecutar el evento modificar datos el cual nos regresará al estado anterior llamado "existe" y el evento generación de DUI, el cual nos llevará al estado DUI. Dentro de este estado se ejecutarán los eventos generación de DAV, que nos creará una instancia de DAV dentro del DUI y el evento enviar que nos llevará al estado "enviado". En este estado se ejecutará el evento recibir, el cual nos llevará al evento "recibido". En este estado existen dos diferentes caminos que escoger, si el objeto lleva los datos correctos se ejecutará el evento aceptar y nos llevará al estado "finalizado", en el cual el objeto se cerrará. En el caso que sea satisfactoria la respuesta se ejecutará el evento de modificar datos, el cual nos llevará al estado de "existe" y seguirá la máquina ejecutándose hasta el momento que se llegue al estado "finalizado".

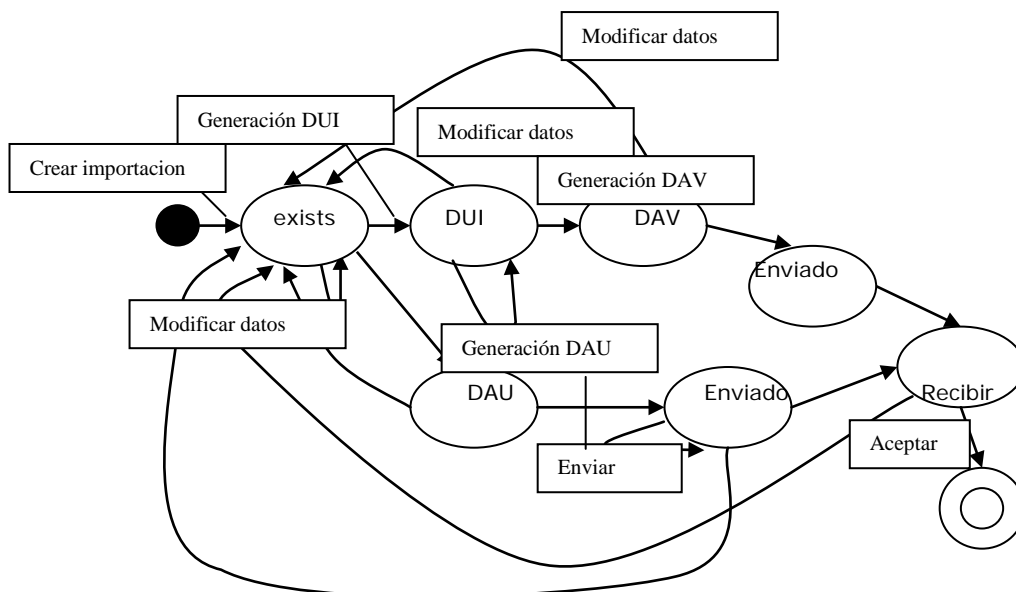


Figura 3.16 FSM del objeto Importación

Exportación.- El objeto exportación es creado en primera instancia por el evento crear exportación. En el momento que el objeto exportación ha sido creado, el objeto pasa al estado "existe". En este estado se pueden ejecutar varios tipos de eventos como lo son ingresar datos, si no se puede ejecutar algún evento de generación de documento por falta de algún dato, se ejecutará el evento ingresa datos; también existe el evento de modificar datos, con el cual se podrá modificar algún dato que se encuentre dentro del proceso de Importación, el cual es representado por el objeto exportación. El evento generación de DAU se ejecuta para que el objeto pase al estado DAU. En este estado existen dos

caminos diferentes, de los cuales se escogerá uno según las características que en ese momento tendrá el objeto Exportación. Se puede ejecutar el evento modificar datos el cual nos regresará al estado anterior llamado "existe". El evento enviar que nos llevará al estado "enviado". En este estado se ejecutará el evento recibir, el cual nos llevará al evento "recibido". En este estado existen dos diferentes caminos que escoger, si el objeto lleva los datos correctos se ejecutará el evento aceptar y nos llevará al estado "finalizado", en el cual el objeto se cerrará. En el caso que no sea satisfactoria la respuesta se ejecutará el evento de modificar datos, el cual nos llevara al estado de "existe" y seguirá la máquina ejecutándose hasta el momento que se llegue al estado "finalizado".

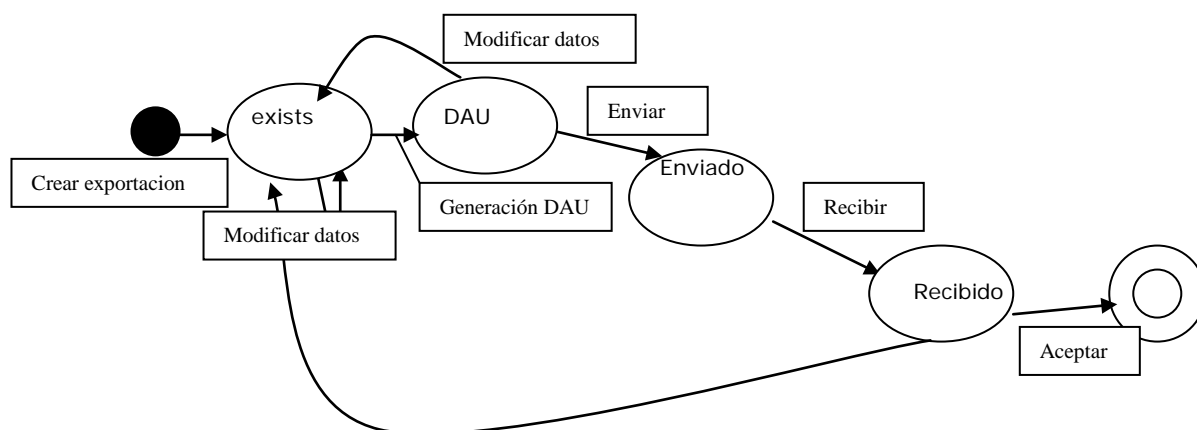


Figura 3.17 FSM del objeto Exportación

3.4.2. Casos de uso y escenarios

Debido a que uno de los propósitos de este proyecto de tesis es desarrollar una aplicación que permita administrar contenidos de los formularios necesarios para realizar un proceso aduanero y además se utilizan las técnicas de MERODE, que son orientadas a objetos, se ha optado por realizar un sistema orientado a objetos en el cual los objetos encapsulen el comportamiento del sistema a través de sus atributos y operaciones.

Para conseguir este objetivo se debe utilizar también UML (Unified Modeling Language – Lenguaje de Modelamiento Unificado) el cual es un lenguaje de modelamiento de propósito general diseñado para especificar, visualizar y documentar los elementos de un sistema.

La notación UML proporciona casos de uso y escenarios para el modelamiento de un sistema. Un caso de uso es el conjunto de transacciones en un sistema cuyo objetivo es producir un resultado con valor medible.

Los escenarios describen los eventos a los que el sistema responderá para proveer la funcionalidad descrita en el caso de uso.

A continuación se describen los casos de uso para el sistema de despacho aduanero AX2:

1. Usuario ingresa al sistema.
2. Administrador ingresa un cliente.
3. Administrador ingresa un proveedor.
4. Digitador ingresa información de banco.
5. Digitador ingresa información de transporte.
6. Digitador ingresa información de mercadería.
7. Digitador ingresa información de procesos precedentes.
8. Digitador genera orden de trabajo.
9. Digitador modifica información de banco.
10. Digitador modifica información de transporte.
11. Digitador modifica información de mercadería.
12. Digitador modifica información de procesos.
13. Digitador modifica información de cliente.
14. Digitador modifica información de proveedor.
15. Digitador realiza consulta de DAU.
16. Digitador realiza consulta de orden de trabajo.

17. Digitador realiza consulta de DUI.

18. Digitador realiza consulta de DAV.

Nombre: 1. Usuario ingresa al sistema.

Descripción: Un digitador o administrador del sistema intenta ingresar al sistema a través de su nombre de usuario y contraseña.

Notas:

- Los digitadores y administradores del sistema sólo pueden ser creados por un administrador del sistema.

Escenarios:

1.1. Acceso autorizado a un administrador del sistema.

1.2. Acceso autorizado a un digitador.

1.3. Acceso no autorizado a una persona cuyo nombre de usuario es incorrecto.

1.4. Acceso no autorizado a una persona cuyo nombre de usuario es correcto, pero la contraseña es inválida.

Nombre: 2. Administrador ingresa un cliente.

Descripción: Un administrador ingresa los datos de un nuevo registro de un cliente aduanero.

Escenarios:

- 2.1. El ingreso de un nuevo cliente se produce satisfactoriamente.
- 2.2. No se produce el ingreso porque ya existe el número de cédula del cliente.
- 2.3. No se produce el ingreso porque el número de cédula no posee 10 dígitos.
- 2.4. No se produce el ingreso porque no se han ingresado todos los datos obligatorios.

Nombre: 3. Administrador ingresa un proveedor.

Descripción: Un administrador ingresa los datos de un nuevo registro de un proveedor.

Escenarios:

- 3.1. El ingreso de un nuevo proveedor se produce satisfactoriamente.
- 3.2. No se produce el ingreso porque ya existe el número de RUC del proveedor.
- 3.3. No se produce el ingreso porque el número de RUC no posee 13 dígitos.
- 3.4. No se produce el ingreso porque no se han ingresado todos los datos obligatorios.

Nombre: 4. Digitador ingresa información de banco.

Descripción: Un digitador ingresa los datos de un nuevo banco en el cual se realizan los pagos de los trámites aduaneros.

Notas:

- Se debe especificar cuál o cuáles son las agencias centrales de los bancos con el nombre matriz

Escenarios:

4.1. El ingreso de un nuevo banco para realizar los pagos se produce satisfactoriamente.

4.2. No se produce el ingreso porque ya existe los datos del banco en la base de datos.

4.3. No se produce el ingreso porque la información del banco esta incompleta.

Nombre: 5. Digitador ingresa información de transporte.

Descripción: Un digitador ingresa los datos de un medio de transporte por el medio del cual ingresaran o saldrán productos de la aduana y del país.

Notas:

- Se debe especificar cuál o cuáles son las líneas de transportes, que cantidad de transporte tienen y la bandera del país al cual pertenece.

- Además, se debe especificar el medio del transporte en el cual se utilizará.

Escenarios:

5.1. El ingreso de un nuevo transporte se produce satisfactoriamente.

5.2. No se produce el ingreso porque ya existe los datos del transporte en la base de datos.

5.3. No se produce el ingreso porque la línea a la cual pertenece el transporte no fue ingresada.

5.4. No se produce el ingreso porque la bandera del país al cual pertenece el transporte no fue ingresada.

5.5. No se produce el ingreso porque el medio al cual pertenece el transporte no fue ingresado.

Nombre: 6. Digitador ingresa información de mercadería

Descripción: Un digitador ingresa los datos de la mercadería o producto que están dentro de un trámite aduanero

Notas:

- Se deben especificar las unidades comerciales con las cuales se tratará a un producto. Podrá ser unidades, Kg., libras, etc.

- Además, se debe especificar el modelo y año en el cual fue creado el producto. En caso de ser frutas no se llenara estos datos.
- El FOB o valor unitario del producto debe estar en dólares para poder realizar la transacción.

Escenarios:

6.1. El ingreso de un nuevo producto o mercadería se produce satisfactoriamente.

6.2. No se produce el ingreso del producto porque ya existen los datos del mismo en la base de datos.

6.3. No se produce el ingreso porque el año en el cual fue creado no fue ingresado.

6.4. No se produce el ingreso porque el país al cual pertenece o de origen del producto no fue ingresado.

6.5. No se produce el ingreso porque la unidad comercial del producto no fue ingresada.

6.6. No se produce el ingreso porque la subpartida dentro de la cual se encuentra declarado el producto no fue ingresada.

6.7. No se produce el ingreso porque el estado en el cual se encuentra el producto no fue ingresado.

6.8 No se produce el ingreso porque el valor del producto no fue ingresado.

Nombre: 7. Digitador ingresa información de procesos precedentes

Descripción: Un digitador ingresa los datos de los procesos precedentes de otro proceso.

Notas:

- No todos los procesos tienen procesos precedentes dentro de lo que es el despacho aduanero.
- Un proceso puede ser precedente de uno o varios procesos.

Escenarios:

7.1. El ingreso de un nuevo proceso precedente de un proceso se produce satisfactoriamente.

7.2. No se produce el ingreso del proceso precedente porque ya existen los datos del mismo en la base de datos.

Nombre: 8. Digitador genera orden de trabajo.

Descripción: Un digitador genera una orden de trabajo por medio del sistema.

Notas:

- Para generar la orden de trabajo se tuvo que haber ingresado datos.

Escenarios:

8.1. La generación de la orden de trabajo se produce satisfactoriamente.

8.2. No se produce la generación de la orden de trabajo porque los datos están incompletos.

8.3. No se produce la generación de la orden de trabajo porque no se ha finalizado algún proceso de almacenamiento de información.

Nombre: 9. Digitador modifica información de banco.

Descripción: Un digitador modifica los datos de un registro de un banco.

Notas:

- No se puede cambiar el número de identificación del banco.
- Se muestran todos los datos del banco.

Escenarios:

9.1. La modificación de los datos del banco se produce satisfactoriamente.

9.2. No se presentan datos porque no existe el número de identificación previamente ingresado.

9.3. No se produce la modificación de datos de la información del banco porque no se han ingresado todos los datos obligatorios.

Nombre: 10. Digitador modifica información de transporte.

Descripción: Un digitador modifica los datos de un registro de un transporte.

Notas:

- No se puede cambiar el número de identificación del transporte.
- Se muestran todos los datos del transporte.

Escenarios:

10.1. La modificación de los datos del transporte se produce satisfactoriamente.

10.2. No se presentan datos porque no existe el número de identificación previamente ingresado.

10.3. No se produce la modificación de datos de la información del transporte porque no se han ingresado todos los datos obligatorios.

Nombre: 11. Digitador modifica información de mercadería.

Descripción: Un digitador modifica los datos de un registro de la mercadería o producto.

Notas:

- No se puede cambiar el número de identificación de la mercadería o producto.
- Se muestran todos los datos de la mercadería o producto.

Escenarios:

11.1. La modificación de los datos de la mercadería se produce satisfactoriamente.

11.2. No se presentan datos porque no existe el número de identificación previamente ingresado.

11.3. No se produce la modificación de datos de la información de la mercadería o producto porque no se han ingresado todos los datos obligatorios.

Nombre: 12. Digitador modifica información de procesos.

Descripción: Un digitador modifica los datos de un registro del proceso para poder realizar el despacho aduanero.

Notas:

- No se puede cambiar el número de identificación del proceso.
- Se muestran todos los datos vinculados con el proceso.

Escenarios:

12.1. La modificación de los datos del proceso se produce satisfactoriamente.

12.2. No se presentan datos porque no existe el número de identificación previamente ingresado.

12.3. No se produce la modificación de datos de la información del proceso porque no se han ingresado todos los datos obligatorios.

Nombre: 13. Digitador modifica información de cliente.

Descripción: Un digitador modifica los datos de un registro de un cliente que haya cambiado algún dato acerca de su información.

Notas:

- No se puede cambiar el número de identificación del cliente
- Se muestran todos los datos vinculados con el cliente.

Escenarios:

13.1. La modificación de los datos del cliente se produce satisfactoriamente.

13.2. No se presentan datos porque no existe el número de identificación previamente ingresado.

13.3. No se produce la modificación de datos de la información del cliente porque no se han ingresado todos los datos obligatorios.

Nombre: 14. Digitador modifica información de proveedor

Descripción: Un digitador modifica los datos de un registro de un proveedor que haya cambiado algún dato acerca de su información.

Notas:

- No se puede cambiar el número de identificación del proveedor.
- Se muestran todos los datos vinculados con el proveedor.

Escenarios:

14.1. La modificación de los datos del proveedor se produce satisfactoriamente.

14.2. No se presentan datos porque no existe el número de identificación previamente ingresado.

14.3. No se produce la modificación de datos de la información del proveedor porque no se han ingresado todos los datos obligatorios.

Nombre: 15. Digitador realiza consulta de DAU.

Descripción: Un digitador consulta los datos registrados en el sistema que forman un DAU.

Notas:

- Se muestra el número de identificación del DAU.
- Se muestra los datos que pertenecen al DAU con respecto al transporte.
- Se muestra los datos que pertenecen al DAU con respecto a la mercadería.
- Se muestra los datos que pertenecen al DAU con respecto al cliente.
- Se muestra los datos que pertenecen al DAU con respecto al proveedor.

Escenarios:

15.1. La consulta de DAU se produce satisfactoriamente.

15.2. No se presentan resultados porque no existen datos de mercadería para presentar.

15.3. No se presentan resultados porque no existen datos de cliente para presentar.

15.4. No se presentan resultados porque no existen datos de transporte para presentar.

Nombre: 16. Digitador realiza consulta de orden de trabajo.

Descripción: Un digitador consulta los datos registrados en el sistema que forman una Orden de trabajo.

Notas:

- Se muestra el número de identificación de la Orden de trabajo.
- Se muestra los datos que pertenecen a la Orden de trabajo con respecto a la mercadería.
- Se muestra los datos que pertenecen a la Orden de trabajo con respecto al cliente.

Escenarios:

16.1. La consulta de Orden de trabajo se produce satisfactoriamente.

16.2. No se presentan resultados porque no existen datos de mercadería para presentar.

16.3. No se presentan resultados porque no existen datos de cliente para presentar.

16.4. No se presentan resultados porque no existen datos del estado en el cual se encuentra la mercadería para presentar.

Nombre: 17. Digitador realiza consulta de DUI

Descripción: Un digitador consulta los datos registrados en el sistema que forman una DUI.

Notas:

- Se muestra el número de identificación del DUI.
- Se muestra los datos que pertenecen al DUI con respecto a la mercadería.
- Se muestra los datos que pertenecen al DUI con respecto al cliente.

Escenarios:

17.1. La consulta del DUI se produce satisfactoriamente.

17.2. No se presentan resultados porque no existen datos de mercadería para presentar.

17.3. No se presentan resultados porque no existen datos de cliente para presentar.

17.4. No se presentan resultados porque no existen datos del transporte para presentar.

Nombre: 18. Digitador realiza consulta de DAV.

Descripción: Un digitador consulta los datos registrados en el sistema que forman una DAV.

Notas:

- Se muestra el número de identificación del DAV.

- Se muestra los datos que pertenecen al DAV con respecto a la mercadería.
- Se muestra los datos que pertenecen al DAV con respecto al cliente.

Escenarios:

18.1. La consulta del DAV se produce satisfactoriamente.

18.2. No se presentan resultados porque no existen datos de mercadería para presentar.

18.3. No se presentan resultados porque no existen datos de cliente para presentar.

18.4. No se presentan resultados porque no existen datos del transporte para presentar.

CAPÍTULO 4

4.DISEÑO DEL SISTEMA

4.1. Modelos de Diseño

4.1.1. Diseño de la Arquitectura del Sistema

La arquitectura de este sistema está basada en la arquitectura Cliente – Servidor de tres capas (three-tier), la cual está compuesta por la capa de presentación, la capa de lógica del negocio, y la capa de datos, tal como se muestra en la Figura 4.1. Este diseño es ampliamente aceptado debido a la facilidad de mantenimiento y crecimiento que proporciona.

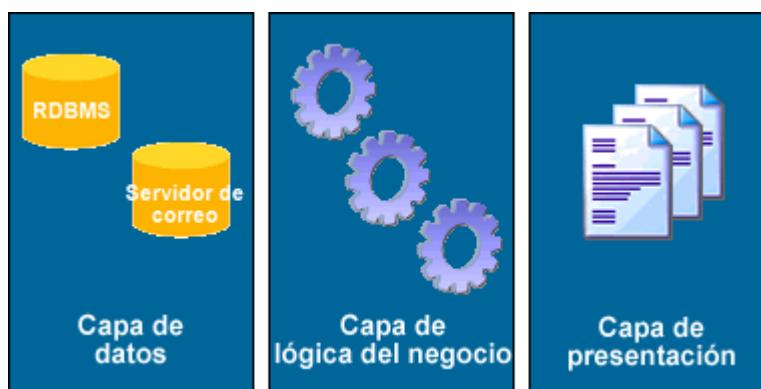


Figura 4.1. Arquitectura Cliente – Servidor de 3 capas

La capa de datos establece la forma mediante la cual se administra el almacenamiento, actualización y consulta de la información, y se permite que la misma sea persistente guardando su integridad. Esta capa está formada por los sistemas administradores de bases de datos relacionales (RDBMS), y por los esquemas de datos que son propios de cada aplicación.

La capa de lógica del negocio determina el comportamiento del sistema, el cual es definido por los componentes que modelan la lógica del negocio de las organizaciones. Estos componentes reciben los requerimientos de la capa de presentación y llevan a cabo las acciones necesarias en la capa de datos para manipular la información del sistema. Está compuesta por los servidores de aplicación. Este

esquema permite una integración más sencilla y eficaz con sistemas externos.

La capa de presentación proporciona todos los elementos que constituyen la interfaz con el usuario, lo cual incluye aquello con lo que el usuario puede enviar y recibir información desde y hacia la capa de lógica de negocio para su procesamiento, como por ejemplo las pantallas de las aplicaciones, el modelo de navegación del sistema y las aplicaciones para cada modo de acceso.

El nivel de procesamiento está compuesto por un computador que ejecutará el servidor de aplicaciones y el servidor Web. Los requerimientos de hardware y software para que el servidor pueda desempeñarse de la mejor manera son:

- Memoria: 512 MB o superior.
- Procesador: Pentium IV 2 GHz o superior.
- Capacidad de disco duro: 80 GB.
- Sistema Operativo: Windows XP Professional
- .NET Framework 1.1

Por otro lado, la conexión con el nivel de almacenamiento se realiza por medio de ODBC, leyendo de un archivo la cadena de conexión específica, lo que permite, que el sistema pueda comunicarse y trabajar con cualquier motor de base de datos.

4.1.2. Modelo Lógico de la Base de Datos

En el diseño propuesto para la base de datos del sistema existen 75 tablas, las que pueden ser agrupadas en 3 clases:

4.1.2.1. Tablas Maestras

Se refiere a aquellas tablas cuya existencia y registros que en ella se encuentran cumplen con estándares administrados y difundidos por la aduana para que sean usados por todos los despachadores, navieras, consolidadores y demás operadores de comercio exterior. Tienen un alto porcentaje de lecturas a sus registros pero un bajísimo nivel de escrituras, sólo cuando la CAE decida cambiar algún registro. Las tablas maestras en el modelo de diseño son:

Maestras		
Aforo	Manifest	TPCI
Agent	Money	TPNG
Aso_Document_Type	Observation_Type	Transport_Agency
Bank	Operation_Type	Transport_Line
Beneficiary	Payment	Transport_Way
Cargo	Physical_Units	Treatment
CIIU	Port	Veh_Body
Class	Processing_Type_Code	Veh_Class
Commerce_Operator_Type	Prov_cond	Veh_Color
Country	Regime	Veh_Fuel
Deposit	Sector	Veh_Traction
Destination_Type	Selection_Code	Veh_Trademark
Dispatch	Sending_type	Veh_Transmission
Doc_Type	State	Warehouse
Duty	Tariff_game	Warrantor
Exon_Cert		Warranty_type
TOTAL: 47 tablas		

Tabla 4.1: Tablas maestras de la base de datos del Sistema

4.1.2.2. Tablas Transaccionales

Se denominan así a aquellas tablas que se ven afectadas en el proceso de grabación de trámites y la mayoría siempre van a ser modificadas cuando se añade o se edite un trámite. Por tanto tienen un alto porcentaje de escrituras y un medio de lecturas. Un gran número de ellas son las tablas tipo asociación

que resultaron del modelo de análisis. Estas tablas son las siguientes:

Transaccionales		
Bank_Agency_Process_detail	Doc_type_detail Free_Code-Doc_Detail	Process
Country_Agree_Detail	Invoice	Process_Line_Agency_Detail
Container	News	Product_Invoice_Detail
Doc_Process_Detail	Preceding_Process_detail	Work_Order
TOTAL: 12 tablas		

Tabla 4.2: Tablas transaccionales de la base de datos del Sistema

4.1.2.3. Tablas Híbridas

Se ha dado en llamar así a las tablas que tienen características de los dos tipos anteriores. No son completamente transaccionales porque tienen un nivel de escrituras más bajo y no forman necesariamente parte del proceso de grabación de un trámite; ni tampoco son maestras porque no son emitidas por la CAE sino creadas como apoyo para el sistema. Se refieren a las tablas que sirven como parámetros de uso para los trámites, tales como la tabla de productos, entre otras. Tienen un nivel medio tanto de lecturas como de escrituras. A continuación la lista de ellas.

Híbridas		
Agent_Client_Detail	DAV	Product
Agreement	Document	Provider
Ben_Prov_Detail	Dui	Transport_Vehicle
Client	free_code	Vehicle
DAU	Observation	Warranty
TOTAL: 15 tablas		

Tabla 4.3: Tablas híbridas

4.1.2.4. Diseño de las Relaciones

A efectos de describir las relaciones de las tablas dentro del modelo de diseño de la base de datos, se las ha separado en secciones representativas de los entornos de análisis usados en el sistema.

La tabla Process es la más importante del modelo, ya que ella concentra la mayor cantidad de relaciones con las demás tablas y a través suyo se puede llegar a otras secciones del modelo. Esta tabla puede compararse como la raíz de un gran árbol de relaciones entre tablas, siendo sus ramas principales las ramas transaccionales, seguidas de las pseudotransaccionales, para acabar en las hojas de las tablas maestras.

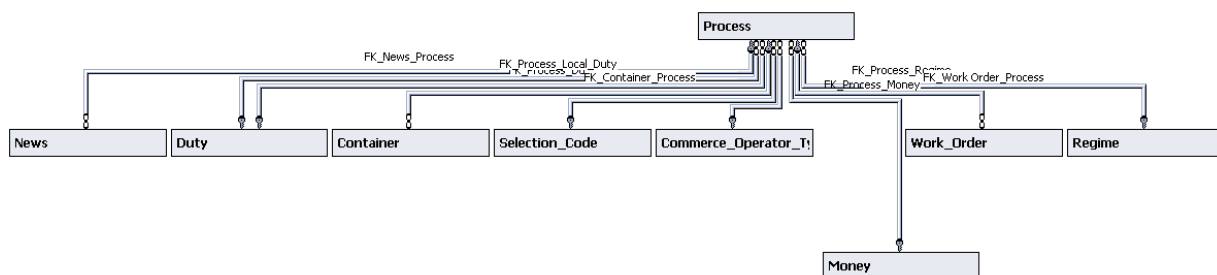


Figura 4.2 Relación de la Tabla Process

Unidas a la tabla Process se encuentran algunas de las tablas maestras que describen características generales del trámite. Adicionalmente, se relacionan directamente a ella todo documento declaratorio que en un trámite se realice: Órdenes de trabajo, DUIs, DAUs, DAVs, etc.

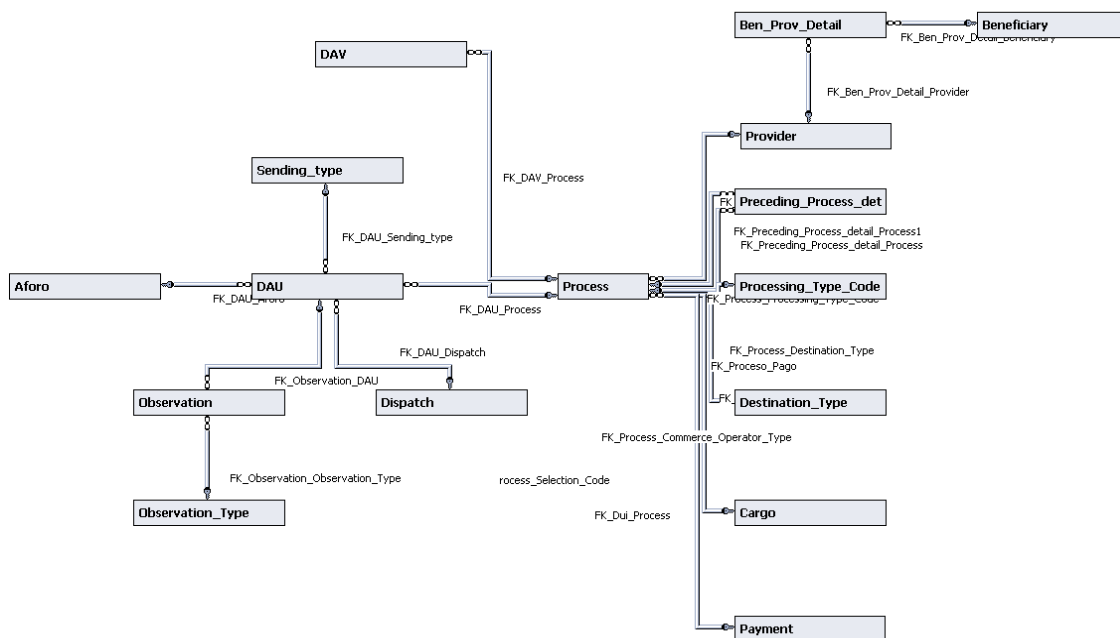


Figura 4.3 Relaciones de las Tablas DUI, DAV, DAU

El proceso indica a qué proveedor se realizó la importación a través de su identificador, y a través de este se obtiene la información del beneficiario del giro. También se obtiene la información del cliente al que pertenece el trámite en cuestión incluyendo su identificador hacia la tabla de clientes, que tiene toda la información que le corresponde, y por medio de este el agente o declarante usado. Se incluye información general como el sector, CIU y ciudad de residencia; así como también el tipo de documento de identificación usado por el cliente (cédula, pasaporte, RUC, etc.)

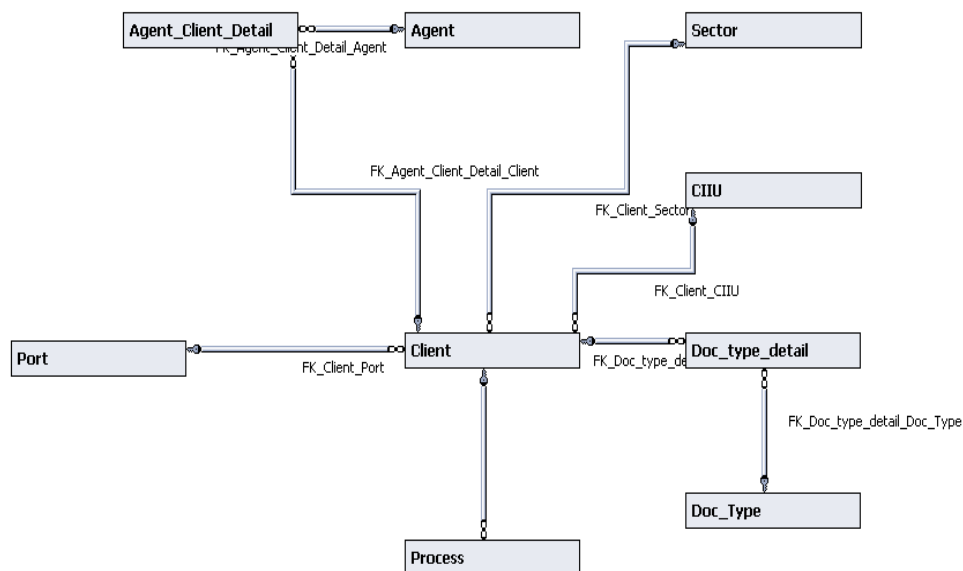


Figura 4.4 Relación de la Tabla Cliente

La sección bancaria se presenta a continuación. La tabla que define la relación contiene la información de la agencia bancaria, la ciudad en la que se encuentra y la fecha en la que se efectuó el pago.

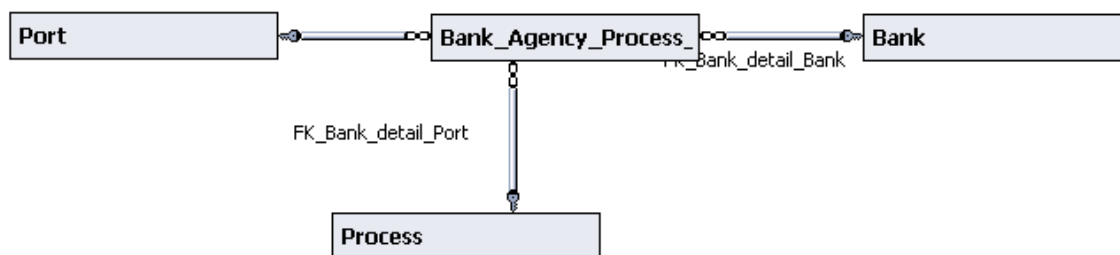


Figura 4.5 Relación de la Tabla Bank

La información concerniente al transporte se indica como sigue. A través de esta sección se tiene acceso a la agencia, línea vehículo y vía de transporte usado en el viaje que corresponde al trámite especificado, incluyendo el tipo de tratamiento utilizado.

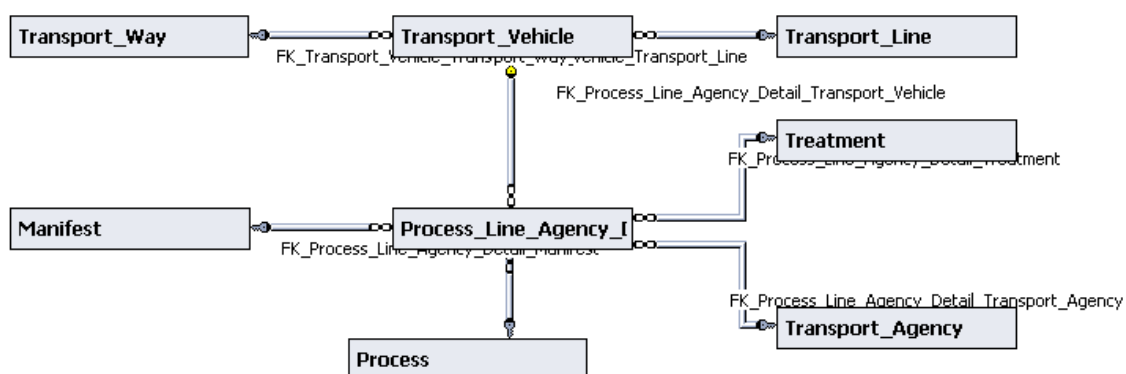


Figura 4.6 Relaciones de la Tabla Transport

La información de la mercadería implica un mayor número de tablas relacionadas para describirla. La tabla Invoice contiene los datos principales de la factura comercial: número y fecha de la factura, pesos, cantidades, costos, etc. totales, ubicación de la mercadería dentro de almacenes y/o depósitos. Luego se realiza el detalle de la mercadería, en la que se describe el estado y país de adquisición. Finalmente, los datos que hablan acerca del producto, incluyendo la subpartida arancelaria a la que pertenece, se encuentran en la tabla Product.

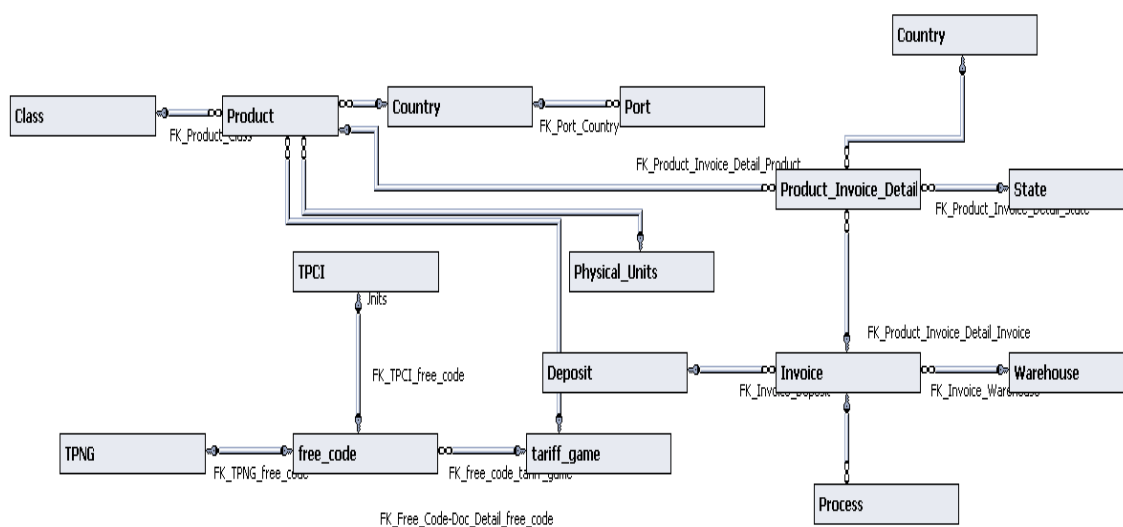


Figura 4.7 Relaciones de la Tabla Invoice

Si el producto corresponde a un vehículo, se describe toda la información correspondiente a ese producto en la tabla Vehicle: chasis, transmisión, tracción,

combustible, marca, clase y color, son características requeridas para declarar un vehículo.

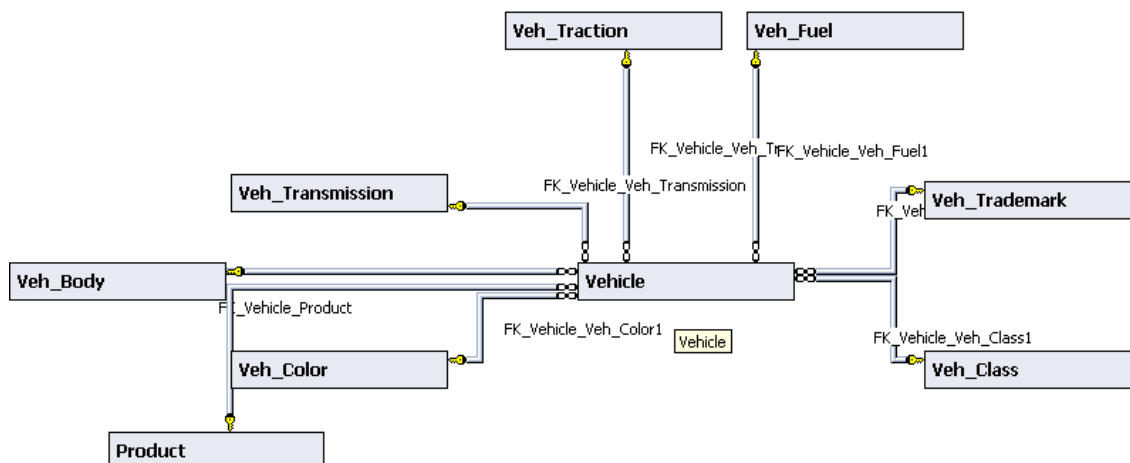


Figura 4.8 Relaciones de la Tabla Vehicle

En el caso en el que se acompañe a un trámite documentos tipificados por la aduana, se adjunta al proceso como una relación entre este y la tabla Document; la cual, no sólo contiene la información de un documento específico de acompañamiento, sino que almacena la información de un documento de apoyo o sustento para un código liberatorio de aranceles.

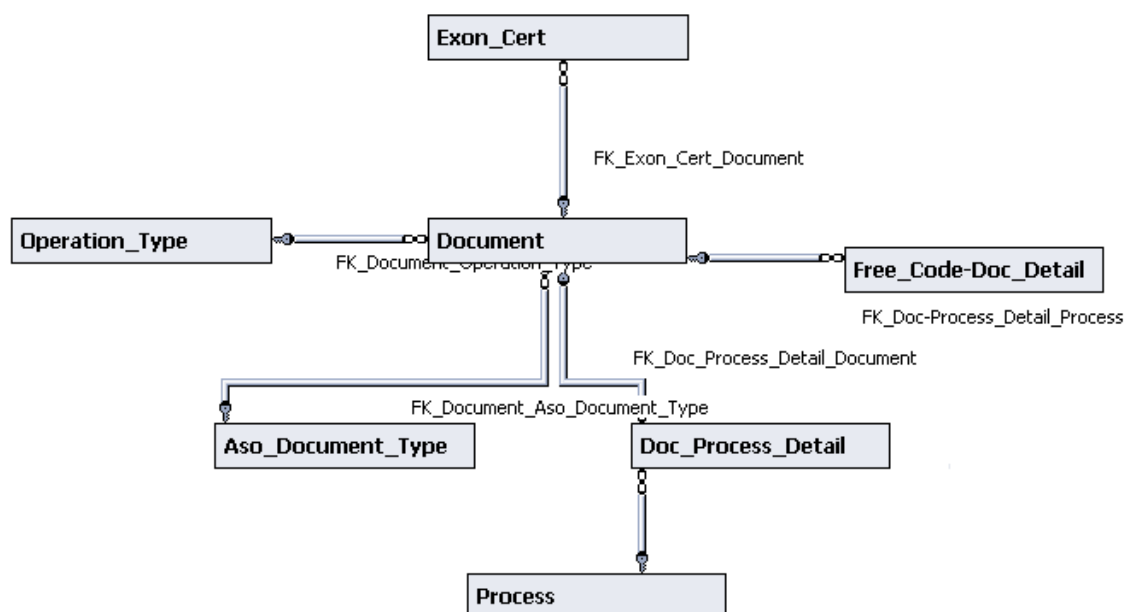


Figura 4.9 Relaciones de la Tabla Document

El Documento Único de Importación describe la información de garantías, cuya información se asenta en el apartado I de este documento. Por ende, cuando se añade un DUI al proceso, es posible describir la información de la garantía por medio de la tabla Warranty, que funciona como una tabla tipo asociación entre las tablas Warrantor (Garante), Warranty_type (el tipo) y la moneda en la que se expresa dicha garantía.

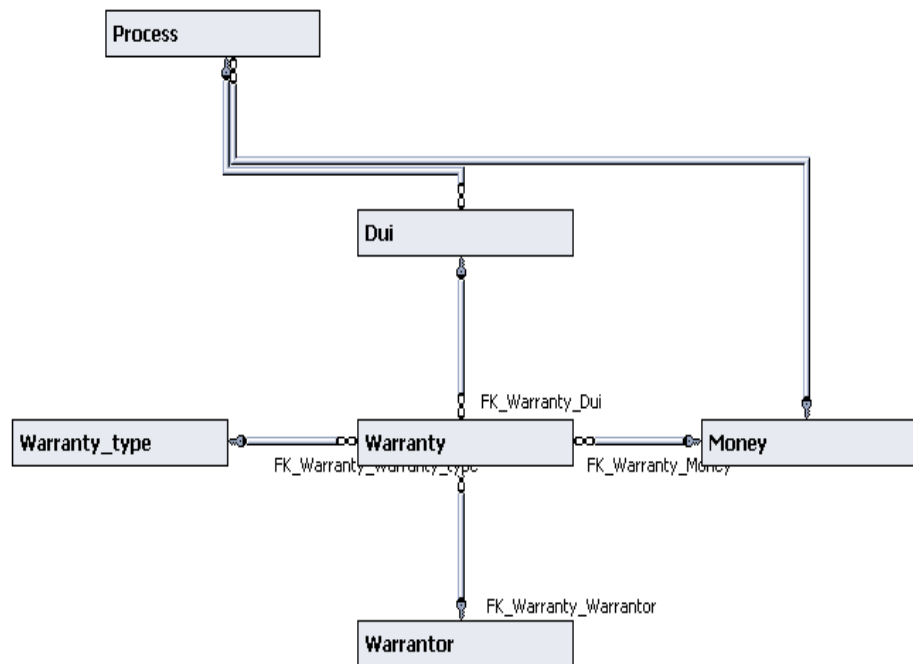


Figura 4.10 Relaciones de la Tabla Warranty

Finalmente, estas secciones se unen para dar como resultado el modelo de diseño de la base de datos usada por el sistema en la presente tesis. La siguiente figura representa dicho modelo.

4.2. Tipología de los Campos

Los campos que constan en los documentos manejados en el ambiente aduanero tienen dos características en común: su obligatoriedad y aplicabilidad. Estos dos factores son dependientes de los cambios en el Reglamento de Procedimientos Aduaneros, que a su vez, deben reflejar lo que estipula la Ley Orgánica de Aduanas. Dentro del sistema también los controles pueden ser identificados por medio de esta clasificación:

4.2.1. Mandatorios

Se definen como mandatorios a aquellos campos que obligatoriamente deben ser llenados para la declaración aduanera. No importan las condiciones ni el régimen al que se acoja, se trata de campos de uso común en todas las declaraciones. El trámite no debe ser grabado ni generar los documentos correspondientes sino están llenos estos campos.

4.2.2. Condicionales

Son condicionales los campos cuyo llenado está supeditado a la ocurrencia de ciertas circunstancias especificadas en la Ley. No quiere decir que su llenado es opcional. El hecho que no estén llenados no impide que se pueda grabar un

trámite o proseguir con el workflow del mismo, pero ello no exime de que se exija su llenado si las circunstancias de las que depende se cumplen.

4.2.3. No Aplicables

Su aplicación queda discontinuada debido a reformas en el Reglamento de Aduanas, sin embargo se mantiene dentro del sistema por compatibilidad hacia atrás en lo que se refiere al ámbito de la declaración aduanera. También son susceptibles a verificación pero cualquier observación hacia ellos no impide continuar con el proceso de declaración.

4.3. Diseño de los módulos del Sistema

A continuación se da una clasificación de los formularios que están supeditados a un trámite, desde el punto de vista de la información que requieren o devuelven al usuario.

Definidos los tipos de formularios y los campos de los mismos, a continuación se presenta un diagrama de diseño de las clases involucradas.

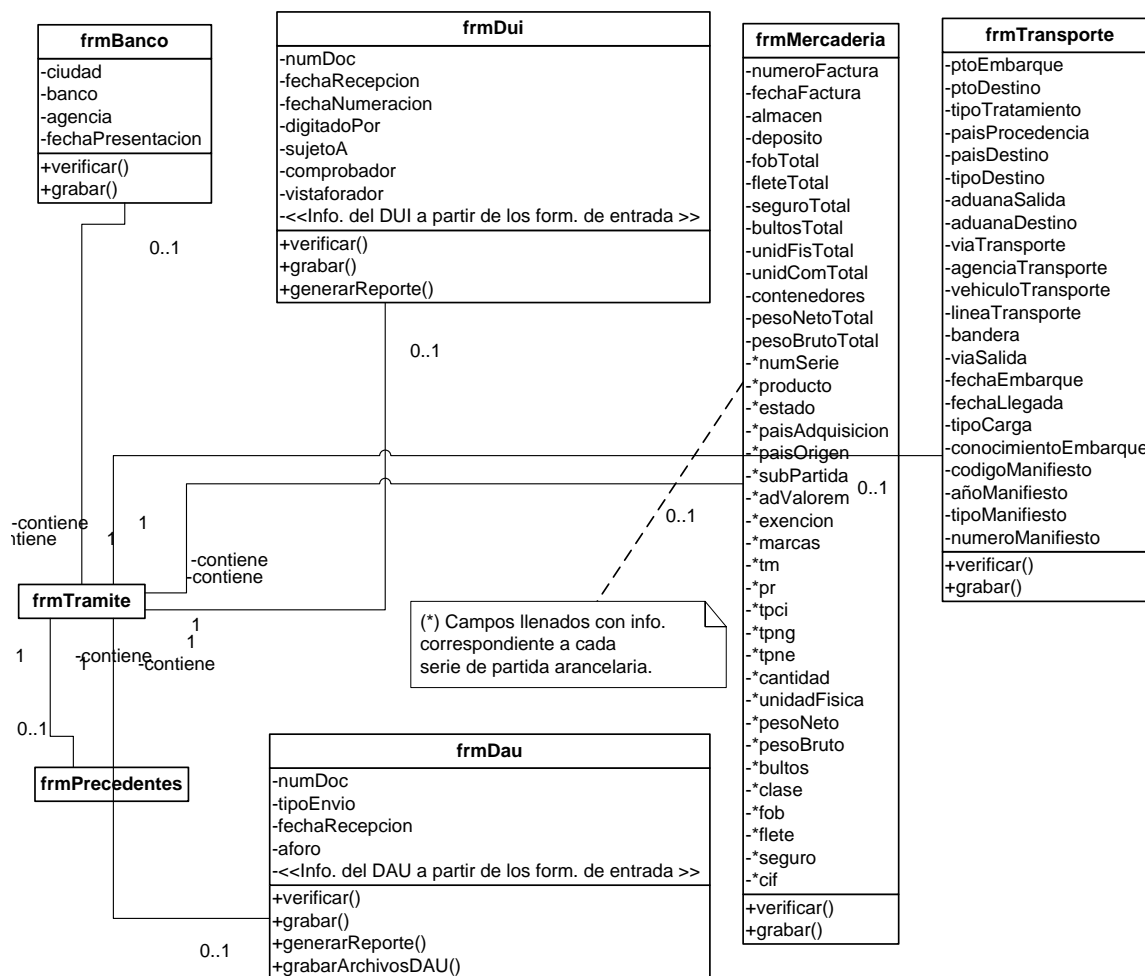


Figura 4.12 Formularios que Intervienen en un trámite

4.3.1. Trámite

En el ámbito del sistema se define como un trámite a todo proceso de declaración aduanera que implica importación o exportación de mercadería, que tiene un remitente y un destinatario, desde o hacia el Ecuador, hacia o desde el extranjero. Se involucran también: el modo de transportación de dicha mercadería, el banco en el que se

pagan los tributos arancelarios, y los trámites precedentes si tuviese.

El sistema define al trámite como el conjunto de información general del mismo, junto con información de entrada específica a clasificada por el siguiente orden: Orden de Trabajo, Banco, Transporte, Mercadería, y Precedentes. Estas secciones están claramente diferenciadas entre sí, de manera que el usuario centre su atención a un contexto de información en un momento dado.

Esta información de entrada se abstrae en clases respectivas que responden al usuario como formularios de entrada y tienen un comportamiento autónomo en cuanto a tareas como verificación y grabación, pero controladas o iniciadas por una clase que represente la información general del trámite, y sin la cual los otros formularios no pueden existir.

Dicha información incluye los siguientes atributos:

- Nombre del trámite
- Cliente
- Proveedor
- Régimen
- Aduana

- Forma de pago
- Moneda
- Descripción
- Tipo de operador de comercio
- Código de selección
- Tipo de procesamiento
- Estado

El atributo estado puede tener uno de los siguientes valores:

- **Nuevo:** El trámite ha sido recién creado y ninguno de sus datos se ha guardado.
- **Ingresando datos:** El trámite ha sido guardado y se permite modificaciones a la información de entrada.
- **Declarando:** El primer documento declaratorio se ha ordenado a ser generado. Se permiten cambios a la información de entrada pero los formularios de salida deben volver a recopilar los datos para actualizar sus cambios.
- **Finalizado:** El trámite junto con todos sus documentos de entrada y salida quedan como registro y constancia para el usuario en modo sólo-lectura.

Una vez que toda la información de entrada haya sido recopilada y se empiece a declarar, al trámite se añade la

información de los documentos declaratorios (DUI, DAU, etc.) que no redunde la información de entrada; tal es el caso de los refrendos respectivos de cada documento. A continuación se presenta un diagrama de clases que ilustra la dependencia de los formularios al trámite principal.

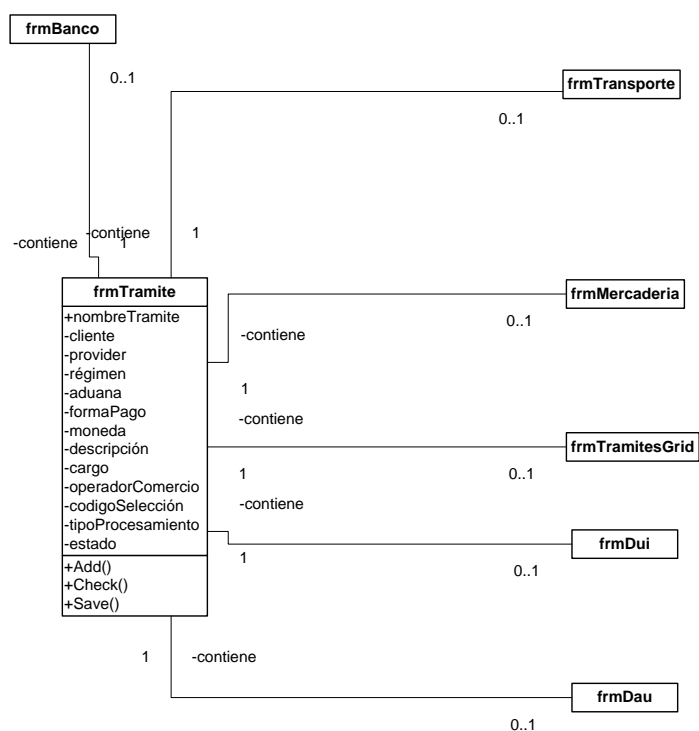


Figura 4.13 Formularios que Intervienen en un trámite

4.3.2. Componentes de Entrada

Se dedican específicamente a recibir información del usuario con el fin de utilizarla en uno o varios documentos declaratorios. Esta información se pide en el momento de

crearse un trámite nuevo junto con el resto de información general y permite modificarse una vez grabados. Estos formularios son: Banco, Transporte, Mercadería y Precedentes.

La información bancaria consta de los siguientes atributos:

- Banco donde se ha hecho el depósito.
- Ciudad donde está ese banco.
- Agencia bancaria dentro de esa ciudad.
- Fecha de aceptación.

La información de transporte comprende lo siguiente:

- Puerto de embarque
- País de procedencia
- Puerto de destino
- País de destino
- Tipo de tratamiento
- Tipo de destino
- Aduana de salida
- Aduana de destino
- Vía de transporte
- Agencia de transporte
- Línea de transporte

- Vehículo de transporte
- Bandera
- Vía de Salida
- Fecha de embarque
- Fecha de Llegada
- Tipo de carga
- Conocimiento de embarque
- Código del manifiesto
- Tipo de manifiesto
- Año del manifiesto
- Número de manifiesto.

La información de la mercadería incluye lo siguiente:

En lo que concierne a la factura:

- Número de factura
- Fecha de la factura
- Almacén
- Depósito
- FOB total (en moneda de la transacción y en dólares)
- Flete total
- Seguro total
- Total de bultos
- Total de unidades físicas

- Total de unidades comerciales
- Total de contenedores
- Peso neto y bruto totales.

Por cada serie de partidas arancelarias correspondiente a dicha mercadería:

- Número de serie
- Producto
- Estado
- País de adquisición
- País de origen
- Subpartida arancelaria
- Ad Valórem
- Exención
- Marcas y números
- T.M. y P.R.
- TPCI, TPNG, y TPNE si hubiese
- Cantidad con su unidad física
- Peso neto y bruto
- Cantidad de bultos
- Clase
- FOB
- Flete
- Seguro

- CIF

Los trámites precedentes se refieren a aquellos que por consideraciones relacionadas con el tipo de mercadería preceden al trámite en cuestión; como por ejemplo el cuero importado como materia prima para la exportación de zapatos, carteras y demás artículos fabricados con dicho cuero. También se definen como precedentes aquellas partes de un solo trámite visto desde el punto de vista del cliente; como es el caso de los despachos parciales de mercadería.

4.3.3. Componentes Generados o de Salida

Se refieren a la abstracción de los formularios escritos que componen a la declaración aduanera y constituyen el resultado de todo el ingreso de datos efectuado anteriormente por el usuario. Estos formularios recopilan la información correspondiente recibida a través de los formularios de entrada (y grabada ya en la base de datos) y la muestran de una manera útil y que sea reconocible fácilmente por el usuario por lo que representan. Dentro de las clases que pertenecen a este tipo de formularios están: frmDui, frmDau, y frmDav. Estos formularios están diseñados para recolectar la información del trámite que le

compete y que se encuentra grabada para presentarla en modo sólo-lectura al usuario. La información recopilada sólo puede modificarse por medio de los formularios de entrada, y si el estado del trámite lo permite. La única información que puede grabarse aquí será el refrendo del documento, que queda como una constancia del mismo dentro la base de datos.

4.3.4. Componentes de Entrada/Salida

Se refieren a formularios que a la vez pueden recibir información útil para los formularios de salida, y extraer información de otras fuentes como la base de datos o formularios de entrada para mostrar información relacionada bajo cierto punto de vista. Un ejemplo de este tipo de formulario es el formulario de la Orden de Trabajo; el que recibe datos importantes como el número de la orden de trabajo, nombre del cliente, del proveedor, etc., y a la vez muestra información que aparece en un formulario escrito de una orden de trabajo y que es recopilada a partir de la información guardada. Por tanto esta la propiedad de salida de este tipo de formularios no está habilitada si el estado del trámite es nuevo.

4.3.5. Componente de Verificación



Antes que un trámite sea almacenado en la base de datos, es necesario verificar toda la información ingresada, buscando que todos los campos que se consideran mandatorios; es decir, que deben llenarse obligatoriamente, efectivamente lo estén; o que no existan incoherencias en todo el conjunto de información; por ejemplo, que un régimen de importación corresponda a puertos de salida extranjeros y de llegada ecuatorianos, etc.

Cada uno de los cuatro formularios básicos de ingreso de datos; esto es, Banco, Transporte, Mercadería y Precedentes, tiene incluida la opción "Verificar". Adicionalmente, en la Barra de Herramientas principal y en el menú también existe la misma opción, con la diferencia que en este caso se verifica el trámite completo.

Cuando se ordena grabar cualquier formulario, primeramente se ejecuta la subrutina de verificación, con el fin de asegurar que los datos a grabarse estén correcta y completamente ingresados. Cuando se ordena la verificación de un formulario cualquiera, se le efectúa solo a ese formulario. Cuando se ordena la verificación del trámite completo, lo que

primero se hace es ejecutar la subrutina de verificación de los campos del formulario principal del trámite, y acto seguido, se ejecutan en orden las respectivas subrutinas de verificación de los formularios que contenga en ese momento.

Cuando se encuentra alguna situación de no completitud o incoherencia en cualquier formulario, esta se visualiza en forma de una Tarea Pendiente al usuario añadida al Panel de Tareas que cada formulario incluye.

Existen dos tipos de tareas a considerar: los "Errores", representados por el icono , que no son más que situaciones que no pueden ser pasadas por alto sin ser contempladas; y las "Advertencias", cuyo símbolo es , que son tareas que informan situaciones anormales o inusuales, pero que en todo caso no impedirán que el formulario pueda grabarse.

El formulario Banco chequea que todos sus campos, que son mandatorios, hayan sido llenados. En todos los formularios este tipo de tareas se las considera un error, y como tal aparecerá en el panel de tareas. El formulario Transporte

además de chequear campos mandatorios no llenados, verifica que en caso de que el trámite sea una importación por ejemplo, los puertos de salida y destino sean siempre uno extranjero y uno ecuatoriano respectivamente, en cuyo caso se dispara un error según sea el caso. El formulario Mercadería después de confirmar que los campos mandatorios sean llenados, cotejará los valores numéricos ingresados, a saber:

- Los valores del FOB, peso neto y bruto, número de bultos y cantidad sean mayores que cero; en otras palabras, se declare algún producto.
- El valor del seguro calculado por el porcentaje establecido en ese momento por el usuario debe ser mayor o igual al 2% del valor del FOB.

Para cada uno de los casos anteriores, se añade una advertencia de no cumplirse.

El formulario Transporte realiza los siguientes chequeos adicionales:

- Un régimen que corresponda a una importación o exportación debe corresponder respectivamente a un viaje desde o hacia el Exterior, hacia o desde el Ecuador.

- Un régimen de importación debe corresponder a un tipo de destino 'EN EL PAIS', y un régimen de exportación debe corresponder a un tipo de destino 'EN EL EXTRANJERO'.
- Una vez que el usuario indica el tipo de destino es conveniente especificar las aduanas de salida y destino.

En las dos primeras verificaciones se añade un error al panel de tareas, en la última se añade una advertencia.

4.3.6. Componente de Grabación

Cuando el usuario ejecuta la acción de grabar a cualquier formulario, primeramente este debe hacer un chequeo o verificación de sí mismo antes de ser grabado. Si como resultado el formulario llega a mostrar solo advertencias, puede grabarse; pero si tiene por lo menos un error, se impide la grabación del formulario hasta que el usuario no complete con estas tareas pendientes.

Una vez pasada la etapa de chequeo, existe un orden de grabación y no se aplica solamente al formulario que llamó al método, el único es el de Mercadería. Si el usuario ordena grabar el formulario Banco o Transporte, el sistema en

primer lugar graba la información de los campos del formulario Mercadería. Luego los del formulario Trámite que correspondan a la tabla Process: Forma de pago, moneda, aduana, régimen, tipo de operador de comercio, código de selección, tipo de procesamiento, descripción y estado; y adicionalmente lee los campos de: Puerto de embarque, de destino, aduana local y aduana foránea que le pertenecen al formulario de Transporte y que también son campos de la tabla Process. Luego se llama a grabar el formulario de Banco, en el que se encuentran los datos de la agencia, el nombre del banco y la fecha de pago. Por último se graba el formulario de Transporte.

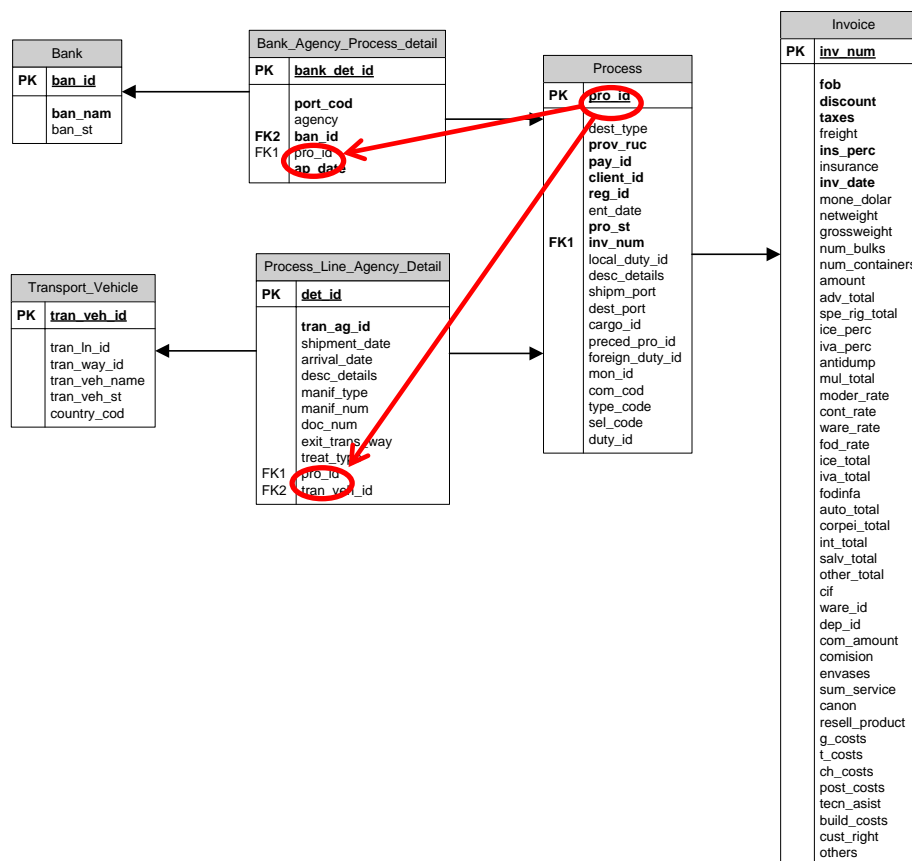


Figura 4.14: Relaciones principales entre tablas

Este orden de grabación ocurre debido al diseño de relaciones de las tablas Process – Bank y Process – Transport_Vehicle, como puede verse en la figura 4.9. Ambas relaciones tienen una multiplicidad de muchos a muchos y por tanto utilizan las tablas tipo asociación Bank_Agency_Process_detail y Process_Line_Agency_Detail

respectivamente; lo que ocasiona que el `pro_id` de la tabla `Process` se copie a estas dos tablas. Cuando se grabe un trámite, si se añadieran registros primeramente a estas dos tablas y luego a la tabla `Process`, seguidamente tendría que hacerse una operación `update` a las otras dos tablas para registrar el `pro_id` en los registros respectivos que se añadieron previamente, lo cual es una pérdida de eficiencia en cuanto a ejecución de sentencias SQL. Adicionalmente se debe buscar la atomicidad de la operación de grabación de trámites; en otras palabras, se debe guardar todo el trámite o no se guarda nada.

4.4. Diseño de Interacción Hombre-Máquina

El sistema introduce el concepto de Trámite para describir a todo proceso realizado por el despachador a un cliente en un momento determinado. Esto implementa familiaridad al agente de aduana quien ve reflejado lo que existe en la realidad dentro del sistema, usando términos que él entiende. Normalmente el despachador debe recolectar toda la información que necesita antes de declarar; en el sistema, de igual manera, se debe suministrar toda la información a los componentes de entrada (y luego guardarla) antes de comenzar a generar los documentos declaratorios por medio de los componentes de salida.

El sistema muestra al trámite como un contenedor no sólo de información general, sino de todos sus componentes de entrada y salida, mostrados con imágenes representativas (Image-Documents). Esto favorece la visibilidad de la extensa información que contiene el trámite: se oculta dentro de estas imágenes la información específica dentro de un contexto; y a la vez se indica implícitamente al usuario que se puede mostrar dicha información al abrir ese componente; ya sea con clic/doble clic (configurable) o en su menú contextual, la opción Abrir (flexibilidad).

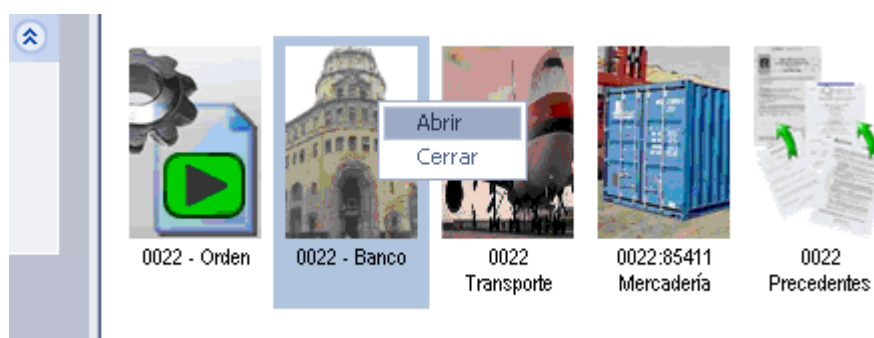



Figura 4.15: Opciones del Sistema

El usuario asociará a un trámite con el símbolo  y se refiere a todo lo que tenga relación con éste, manteniendo consistencia en la simbología.

Se ha conseguido familiaridad emulando la manera como el Explorador de Windows ofrece las tareas más comunes al usuario. Se utiliza un panel izquierdo orientado a las tareas que el usuario esté realizando en un momento dado. Cuando el sistema no tiene

ningún trámite abierto, este panel induce al usuario a crear uno nuevo o abrir uno existente; y cuando existe por lo menos uno, las tareas más comunes son las de los documentos que se pueden declarar a partir de ese trámite.



Figura 4.16: Familiaridad del Sistema

Existen tres maneras posibles para restringir al usuario ciertas opciones dependiendo de la situación: ocultar opciones, deshabilitarlas y habilitarlas pero presentando mensajes de advertencia o de error. Como puede verse en la figura, el panel izquierdo cambia de opciones cuando existe un trámite nuevo o

recuperado; esto se da debido a que si no existe ningún trámite en la pantalla, el usuario solo debe pensar en crear uno, y luego continuar con la declaración. El uso del cambio de opciones se da sólo en este caso, ya que no se aconseja salvo necesidad. Cuando se tiene un trámite abierto y por ende las opciones presentadas en el panel izquierdo son documentos declaratorios y adicionales, el usuario puede hacer uso de ellos solo si el trámite ha sido grabado; por lo tanto se dejan habilitadas estas opciones, y de no cumplirse con esta condición, se pregunta al usuario si se desea grabar antes de continuar. Ciertas opciones de la barra de menús y de herramientas que se aplican a un trámite abierto se deshabilitan cuando no hay ninguno, lo que acompaña al hecho de no ver un trámite en la pantalla. Esta técnica es muy usada por todas las aplicaciones que usan formularios MDI.

Se ha guardado consistencia en lo que tiene que ver con el formato de los formularios que componen el trámite.

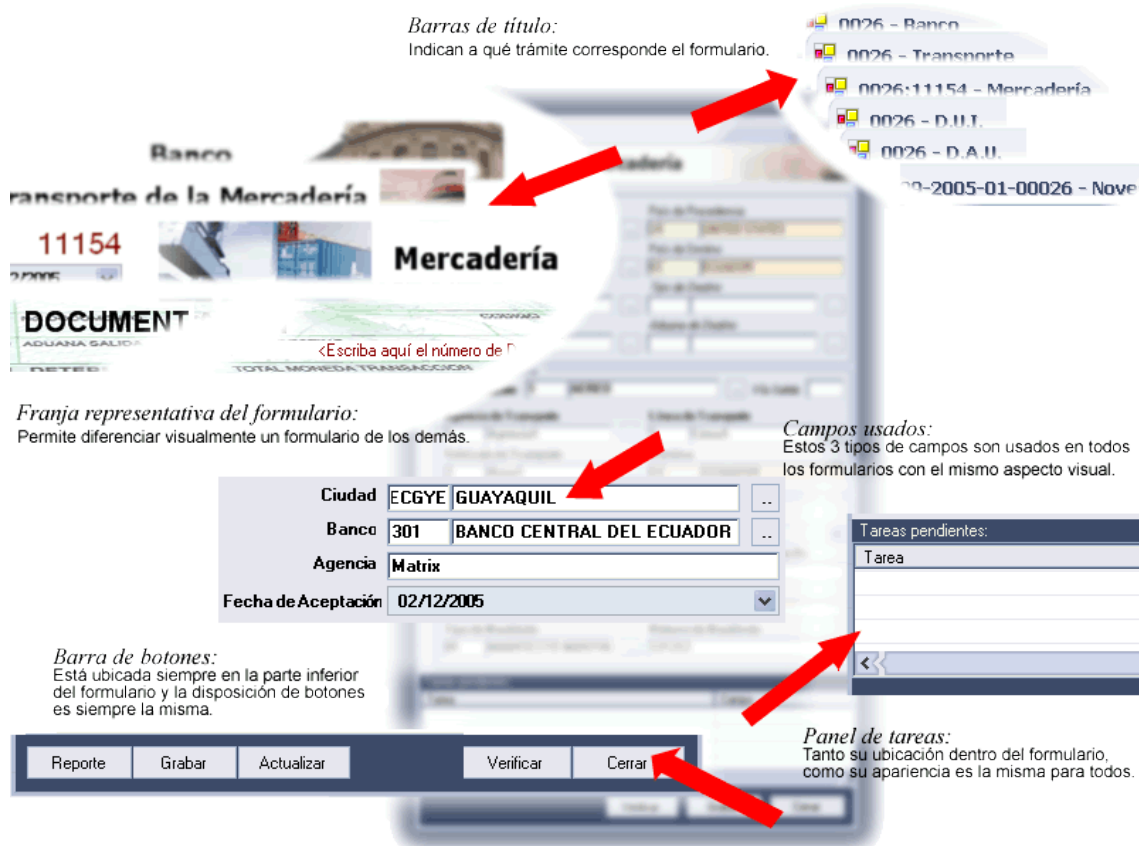


Figura 4.17: Consistencia entre formularios

El sistema no preocupa al usuario de cómo se graba un trámite en la base de datos, ni de cómo afectan las relaciones entre las tablas a la grabación entera. Existe una opción general para grabar un trámite siempre visible en la barra de herramientas y menús; y cada formulario tiene la alternativa de un botón propio de grabación, ofreciendo flexibilidad al usuario.

El panel de tareas en la parte inferior de cada formulario ha sido siempre una opción muy útil que los entornos de programación ofrecen a los programadores para depurar los programas. Este

concepto se ha introducido en el sistema debido a su gran utilidad y facilidad de uso. Haciendo doble clic en una tarea se conduce al campo de donde esta procede.

Con el fin de aplicar flexibilidad y agilizar la navegación entre trámites y entre sus componentes, se ha implementado dos alternativas (adicionales al ya conocido menú Ventana), respectivamente: En la barra de herramientas se encuentra un selector de los trámites que se encuentren abiertos; al dar clic en uno de ellos muestra el trámite seleccionado encima de los demás formularios.



Figura 4.18: Selector de Trámites

Para navegar entre los componentes principales del trámite, se cuenta con una barra de botones identificada con el nombre del trámite y pegada a la ventana maximizada del formulario, simulando las pestañas de una carpeta (familiaridad). Al presionar un botón de esta barra se abre el componente correspondiente de

ese trámite. Sus íconos son similares a los componentes que representan, guardando consistencia.

Debido a que la mayoría de los comandos del usuario se ejecutan en el cliente con una rapidez relativa (inclusive el de verificación, que podría ser el más lento de todos, el que durante su ejecución, produce cambios visibles en los paneles de tareas), el tiempo de respuesta es muy corto y la retroalimentación es inmediata; a excepción de cuando se abre y se graba un trámite, en el que se concentra la mayor cantidad de transacciones a la base de datos y el tiempo de respuesta depende de la velocidad de comunicación cliente-servidor. Sin embargo, mientras cualquiera de estas operaciones se está ejecutando, se ha preocupado de mantener la "fluidez" del sistema, en el sentido de permitir que se sigan procesando eventos de pantalla, como por ejemplo el repintado de la ventana. De esta manera, el usuario no pensará que el sistema se ha "inhibido" en cuanto vea que la ventana se congela al ejecutar estos comandos.

4.4.1. Interacción con la Base de Datos

Debido a que en el sistema que se desarrolla como parte de este proyecto de tesis, los dos más importantes requerimientos de software a utilizarse son: Microsoft Visual Basic .NET 2003, para todo lo que tiene que ver con la

programación de la aplicación cliente; y Microsoft SQL Server 2000 o superior como motor de la base de datos; se ha buscado maximizar la eficiencia de estos dos componentes, tanto por las características individuales como por su trabajo en conjunto, puesto que los dos fueron creados por la misma compañía de software.

Existen varias alternativas para trabajar con bases de datos, las cuales a continuación se citan:

1. Manejo directo de los datos mediante ejecución de sentencias de Transact-SQL.
2. Ejecución de Stored Procedures personalizados a cada tarea.
3. Empleo de DataSets como característica inherente a la programación bajo la plataforma .NET.

Las dos primeras son aplicables independientemente de la base de datos o el lenguaje de alto nivel a usarse. La ejecución de operaciones a la base de datos directamente, ya sea con sentencias SELECT, INSERT, DELETE o UPDATE tiene ventajas específicas, las cuales se citan a continuación:

Funcionalidad extra. Hay algunas operaciones, tales como la ejecución de comandos DDL, que se pueden llevar a cabo ejecutando comandos de datos adicionales.

Mayor control sobre la ejecución. Usando comandos, se tiene un control más directo sobre cómo y cuándo una declaración SQL o un stored procedure se ejecuta y qué deriva de los resultados o valores de retorno.

Menor overhead.⁷ Por medio de la lectura o escritura a la base de datos, se puede obviar el hecho de guardar datos en un dataSet; debido a que el dataSet requiere memoria, haciendo que se reduzca el overhead en la aplicación. Esto es especialmente evidente en situaciones donde se intenta usar los datos sólo una vez, e.g. mostrar resultados de búsqueda en una página Web.

Menos programación en algunas instancias. En pocos casos, particularmente en aplicaciones Web, hay una programación adicional requerida para salvar el estado de un dataSet. Por ejemplo, en los formularios de páginas Web, la página es recreada con cada vez que se regresa a la misma; a menos que se añada programación para salvar y restaurar el estado del dataSet, este también es descartado y recreado

⁷ Overhead en términos generales significa un gasto; en este caso, de procesador y memoria.

junto con la página. Para el caso presente, no se trata de una aplicación Web.

Sin embargo, la tercera alternativa presenta las siguientes ventajas; y dadas las herramientas que van a utilizar, es la opción más conveniente. El uso de dataSets en la presente aplicación se justifica por las siguientes razones:

Trabajo con múltiples tablas. Un dataSet puede contener múltiples tablas de resultados, los que mantiene como objetos discretos. Se puede trabajar con las tablas individualmente o navegar entre ellas como tablas madre-hija.

Manipulación de datos desde múltiples fuentes. Las tablas en un dataSet pueden representar datos desde fuentes muy diferentes (por ejemplo, desde diferentes bases de datos, archivos XML, hojas de cálculo, y así, todas en el mismo dataSet). Una vez que los datos están en el dataSet, se pueden manipular y relacionar en un formato homogéneo como si hayan venido de una única fuente.

Mover datos entre capas en una aplicación distribuida. Manteniendo los datos en el dataSet, se puede fácilmente

moverlos entre la capa de presentación, la de negocios y la capa de datos de la aplicación.

Intercambio de datos con otras aplicaciones. Un `dataSet` provee una poderosa vía de intercambiarse con otros componentes de la aplicación y con otras aplicaciones. Los `dataSets` incluyen soporte extensivo para características tales como serialización de datos como XML y leyendo y escribiendo esquemas XML.

Data Binding (Vínculo de datos). Si se está trabajando con formularios, es usualmente más fácil vincular controles a los datos en un `dataSet`, que programáticamente cargar los valores de los datos en el control después de ejecutar un comando.

Mantenimiento de registros para reuso. Un `dataSet` permite trabajar con los mismos registros repetidamente sin requerir la base de datos. Usando las facilidades del `dataSet`, se pueden filtrar y ordenar registros, y se puede usar el `dataSet` como una fuente de datos si se necesita.

Facilidad de programación. Cuando se trabaja con un `dataSet`, se puede generar un archivo de clase que representa su estructura como objetos (por ejemplo, una tabla `Customers` en el `dataSet` puede ser accedido como el objeto `dataSet.Customers`). Esto lo hace más fácil, más

claro, y menos propenso a errores al programar con ellos, y es soportado por herramientas de Visual Studio como IntelliSense, el Data Adapter Configuration Wizard, etc.

CAPÍTULO 5

5. IMPLEMENTACION Y PRUEBAS

5.1. Descripción de las Estructuras Principales

5.1.1. Estructura del DATASET

Existen dos tipos de dataSets, a saber: Typed Datasets (no existe una traducción literal, pero se trata de un dataSet basado en un tipo de datos, o "tipeado"), y los Untyped Datasets. Un Typed DataSet es un dataSet que es primero derivada de la clase base System.Data.DataSet y luego usa información de un archivo de esquema XML (XML Schema Definition file, un archivo .xsd) para generar una nueva clase. Visual Basic tiene mayor soporte de herramientas para

este tipo, lo que hace la programación más fácil y menos propensa a errores.

La información del esquema (tablas, columnas, etc.) es generada y compilada dentro de esta nueva clase `dataSet` como un conjunto de objetos y propiedades. Debido a que un `typed dataSet` hereda de la clase base `DataSet`, la clase `typed` asume toda la funcionalidad de la clase base y puede ser usada con métodos que toman una instancia de una clase `DataSet` como parámetro.

Un `Untyped Dataset`, en contraste, no tiene un esquema correspondiente integrado. Se trata de una variable del tipo `System.Data.DataSet`. Como su contraparte `typed`, un `untyped dataSet` contiene tablas, columnas, etc., pero estas son expuestas como colecciones.

En el sistema se usa un `typed dataSet`, para aprovechar al máximo las ventajas ya descritas; en especial la facilidad de programación. Por ejemplo, considérese el siguiente par de sentencias, la una usando un `typed` y la otra un `untyped dataSet`, nótese la simplicidad de código en el primer caso.

Dim s As String

```
s = dsCustomersOrders1.Customers(0).CustomerID
```



```
s=CType(dsCustomersOrders1.Tables("Customers").Rows(0)
. _ Item("CustomerID"), String)
```

Básicamente el dataSet contiene 19 clases heredadas de la clase base DataTable que representan a las tablas transaccionales del modelo de diseño de la base de datos. Cada dataTable creado por el dataSet, debe ser administrado por un respectivo objeto del tipo SqlDataAdapter (por tratarse de una base basada en SQL Server) que lee datos del dataSet e inserta o actualiza datos del mismo. Estos dataAdapters contienen internamente cuatro objetos del tipo SqlCommand que representan a las cuatro operaciones posibles que se pueden realizar a cada dataTable. Estos sqlCommandCommands tienen como parte de sus variables miembro un objeto de tipo String llamado CommandText, los cuales están basados en el siguiente formato:

DataSet.DataTable.SelectCommand.CommandText:

“SELECT field1, field2, ... , fieldN FROM [Table]”

DataSet.DataTable.InsertCommand.CommandText:

“INSERT INTO [Table] (field1, field2,..., fieldN) VALUES (@value1, @value2, ... , @valueN); _

```
SELECT field1, field2, ... , fieldN FROM [Table] WHERE  
(fieldPK = @valuePK)”
```

DataSet.DataTable.UpdateCommand.CommandText:

```
“UPDATE [Table] SET field1 = @value1, field2 = @value2,...,  
fieldN = @valueN WHERE (field1 = @OriginalValue1) AND  
(field2 = @OriginalValue2) AND ... AND (fieldN =  
@OriginalValueN); _  
SELECT field1, field2, ... , fieldN FROM [Table] WHERE  
(fieldPK = @valuePK)”
```

DataSet.DataTable.DeleteCommand.CommandText:

```
“DELETE FROM [Table] WHERE (field1 = @OriginalValue1)  
AND (field2 = @OriginalValue2) AND ... AND (fieldN =  
@OriginalValueN)”
```

Adicionalmente los dataAdapters tienen una Collection llamada Parameters, en donde se copian como atributos cada uno de los campos del dataTable que manejan a través de los commandTexts citados anteriormente.

5.1.2. DATA BINDING

Una característica aprovechada del uso de dataSets en el modelo de implementación de la Base de Datos y el framework de .NET es el hecho de que se puede vincular el valor de una propiedad de un objeto al valor de una propiedad de un control. En el caso presente este objeto es un campo de un dataTable que pertenece al dataSet del sistema (Ax2DataSet1, que es una instancia de la clase Ax2DataSet). Toda clase heredada de System.Windows.Forms.Control tiene una propiedad llamada DataBindings. Se trata de una colección (System.Windows.Forms.ControlBindingsCollection) de objetos de la clase Binding. Esta clase representa el simple enlace entre el valor de propiedad de un objeto y el valor de propiedad de un control.

Cuando se construye una instancia de Binding, se debe especificar tres ítems: el nombre de la propiedad del control a vincular, la fuente de datos, y el camino de navegación que resulta en una lista o una propiedad en la fuente de datos.

Primero, se debe especificar el nombre de la propiedad del control a vincular. Por ejemplo para mostrar información en un control TextBox, se especifica la propiedad Text.

Segundo, se puede especificar una instancia de cualquiera de las siguientes clases como una fuente de datos:

- Cualquier clase que implemente `IBindingList` o `ITypedList`. Estas clases incluyen: `DataSet`, `DataTable`, `DataGridView`, o `DataGridViewManager`.
- Cualquier clase que implemente `IList` para crear una colección indexada de objetos. La colección debe ser creada y llenada antes de la creación del objeto `Binding`. Los objetos en la lista deben ser todos del mismo tipo; de otro modo se arrojaría una excepción.
- Una clase que implemente `IList` que sea `strongly typed` (que su tipo sea explícitamente definido; y en caso de arreglos, su dimensión sea fija).

En el caso presente se utiliza la primera alternativa, específicamente un `dataTable`.

Tercero, se debe especificar la ruta de navegación, la que puede ser un string vacío, un nombre de propiedad, o una jerarquía de nombres delimitada por puntos. Si se deja la ruta de navegación como un string vacío, el método `ToString` será llamado al objeto fuente de datos en cuestión.

Si el data source (fuente de datos) es DataTable, que puede contener múltiples instancias de DataColumn, la ruta de navegación debe ser usada para resolver a una columna específica.

Se requiere una ruta de navegación delimitada por puntos cuando el data source está dado a un objeto que contiene múltiples objetos dataTable, como un dataSet o un dataGridViewManager (que no es el caso presente). También se la puede usar para vincular a un objeto cuyas propiedades retornan referencias a otros objetos (tales como una clase con propiedades que retornan otros objetos de clase). Por ejemplo, las siguientes rutas describen campos de datos válidos):

“Size.Height”

“Suppliers.CompanyName”

“Regions.regionsToCustomers.CustomerFirstName”

“Regions.regionsToCustomers.customersToOrders.ordersToDetails.Quantity”

Se puede vincular de manera simple a cualquier objeto que derive de la clase System.Windows.Forms.Control, por ejemplo, los siguientes controles Windows:

Button	DomainUpDown	ListBox	ProgressBar	TextBox
CheckBox	GroupBox	ListView	RadioButton	TreeView
CheckedListBox	HScrollBar	MonthCalendar	RichTextBox	VScrollBar
ComboBox	Label	NumericUpDown	ScrollBar	
DateTimePicker	LinkLabel	PictureBox	StatusBar	

Tabla 5.1: Controles derivados de System.Windows.Forms.Control

Nota. Solo la propiedad SelectedValue del control ComboBox, CheckedListBox, y ListBox es vinculable de manera simple.

En el sistema presentado como parte de la presente tesis, los formularios de entrada implementan data binding hacia sus controles. Este data binding se efectúa cuando se carga por primera vez el formulario; esto es, dentro del método Private Sub <N. form.>_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles MyBase.Load, en el que se llama al método Private Sub <N. form.>.BindData() del formulario. A continuación un ejemplo de la implementación de este método en la clase frmBanco. En este ejemplo dt es una instancia de la clase Ax2DataSet.Bank_Agency_Process_detailDataTable y contiene todo el esquema de la tabla Bank_Agency_Process_detail del modelo de la base de datos.

```
Me.ucCiudad.DataBindings.Add("Code", dt, "port_cod")
```

```
Me.ucBanco.DataBindings.Add("Code", dt, "ban_id")
```

```
Me.txtAgencia.DataBindings.Add("Text", dt, "agency")
```

```
Me.dtpApDate.DataBindings.Add("Value", dt, "ap_date")
```

Los objetos ucCiudad y ucBanco son de la clase UserControl1, y el atributo Code referencia al campo Text de uno de los textboxes que posee. El objeto.dtpApDate es un objeto de la clase ColectorFechaHora, y el atributo Value fija o devuelve la fecha en el control. TxtAgencia es de la clase base TextBox.

El formulario Mercadería contiene una descripción de los productos que una factura contiene, por lo que tiene 2 dataTables para representar a los productos de la factura y a la factura en sí. Los controles que detallan a la mercadería están vinculados con los campos de la tabla Product_Invoice_Detail, y el usuario puede moverse entre rows de esta tabla por medio de un objeto C1.Win.C1Input.C1DbNavigator del paquete Component One®.

Todo formulario tiene una colección de contextos de enlace (BindingContext). Los elementos de esta colección Form.BindingContext son heredadas de la clase abstracta System.Windows.Forms.BindingManagerBase, cuyas clases hijas más conocidas son System.Windows.Forms.CurrencyManager y System.Windows.Forms.PropertyManager. Esta clase, una vez derivada, obtiene información de los vínculos u objetos Binding que procedan de una misma fuente de datos. Se puede acceder a un item de la colección BindingContext por su índice, que viene a ser la fuente de datos. Por ejemplo, form.BindingContext(dt) obtiene la información de vínculos que salen del dataTable dt. Pero quizá la propiedad más importante de la clase BindingManagerBase sea Position, la cual, fija o devuelve la posición actual en la lista subyacente a la que los controles vinculan. En el caso del dataTable, Position apunta al DataRow actual al cual leer o escribir.

El formulario Mercadería por ejemplo tiene 2 dataTables (sus instancias se llaman dtm y dt), que serán fuentes de datos para algunos controles respectivamente; por tanto se trabajan aquí con 2 miembros de la colección BindingContext del formulario. La primera fuente de datos, que proviene de

la tabla `Product_Invoice_Detail` de la base de datos, provee información de la mercadería declarada para una factura comercial. La segunda fuente, de la clase `Ax2DataSet.InvoiceDataTable`, proviene de la tabla `Invoice`, y contiene información de la factura en sí.

Se sabe a qué fila del `dataTable dtm` (la que detalla la mercadería) variando el atributo `integer frmMercaderia.BindingContext(dtm).Position`, y los bindings de los controles se establecen a los campos del `dataRow` apuntado por este índice `Position`. Por tanto, al moverse entre registros por medio del objeto `C1DbNavigator`, el atributo `Position` cambia y los valores de los atributos (`Text`, `Code`, `Value`, etc.) de los controles vinculados que corresponden a la información de la mercadería (`ucProdCode`, `txtProdDesc`, etc.), mostrarán los datos del respectivo ítem apuntado.

La idea principal del `data binding` es que cualquier cambio en el control se refleje en la fuente de datos vinculada (en este caso el `datatable`), y viceversa. Esto implica en que se debe hacer un control dinámico sobre la dirección en la que efectúa este paso de información para no caer en un molesto ciclo.

5.2. Descripción de Procedimiento del Trámite

Cada trámite se representa visualmente como un formulario MDIChild dentro del formulario principal, en el que consta cierta información general, junto con íconos de los componentes de información requeridos para completar la declaración (los formularios Banco, Transporte, Mercadería y Precedentes), y de los documentos que posteriormente serán generados por el usuario una vez llenada toda la información solicitada en los formularios anteriores.

Los formularios pueden llenarse en cualquier orden, pero antes de comenzar a generar los documentos declaratorios y reportes, se debe completar toda la información solicitada y posteriormente grabarse para que el trámite se asiente en la base de datos. La información para los documentos declaratorios se obtiene mediante consultas a la base de datos y no hacia los formularios.

El acceso a los formularios se facilita mediante una estructura de datos llamada Trámite que contiene referencias específicas a cada uno de ellos.

A continuación un diagrama de clases en el que se muestra las dos maneras a las que se puede acceder a los formularios. Nótese que

la forma de acceso por la clase Trámite es más directa y requiere menos typecastings que a través de la colección de ImgDocs.

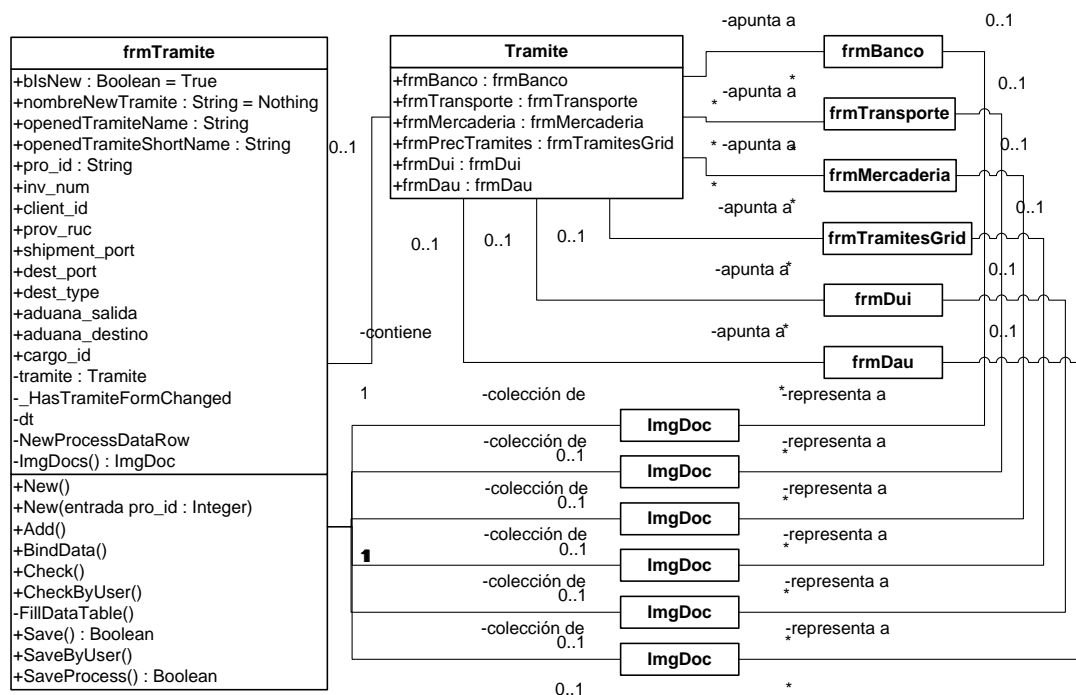


Figura 5.5: Instancias involucradas en el acceso a los formularios de entrada

5.2.1. IMAGE-DOCUMENTS

Esta clase describe la representación visual de los formularios que conciernen al trámite en forma de íconos asociables. Consta de un PictureBox cuyo atributo Image es el que contiene la imagen del formulario, y un Label que contiene algún identificador de dicho formulario.

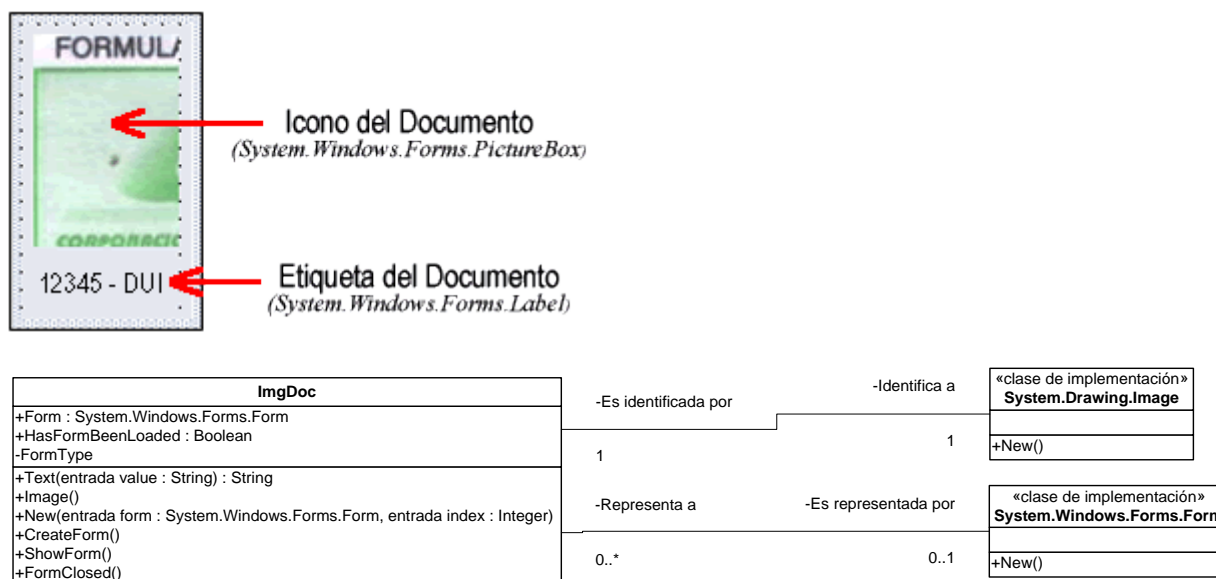


Figura 5.6: Image-Document

Contiene métodos para controlar la creación y cierre de su formulario. `CreateForm` valida si el formulario ha sido cerrado o dispuesto (en estado `Disposed`), en cuyo caso vuelve a crear la instancia dependiendo de su `FormType`. Si un formulario de trámite se cierra, se disponen sus `imgDocs`; y cada `imgDoc` cierra o dispone su formulario.

Durante la creación de un formulario de trámite, se declara la creación de los `imgDocs`, por ende de sus respectivos formularios, mediante una sola sentencia inline; un ejemplo se cita a continuación:

```
frm.Tramite.frmBanco = frm.Add(New imgDoc(New frmBanco(frm.pro_id), 0)).Form
```

Esta clase tiene la función adicional de re-instanciar el formulario al que apunta si este no existe o haya sido cerrado. Una vez que un formulario se cierra, su método Dispose es llamado casi inmediatamente y sus recursos serán devueltos al sistema operativo, y a partir de ese momento dicha instancia no puede reabrirse, ni acceder a sus controles. Lo único que se puede hacer es crear otra instancia que se refiera a la misma información que tenía la anterior. Una instancia de la clase ImgDoc tiene el método CreateForm, que verifica si existe su formulario (.form = Nothing), o si ha sido cerrado (.form.isDisposed = True). De no darse ninguna de estas situaciones el método no se hace nada y el método termina. Si una de estas condiciones se cumple, se crea una nueva instancia del tipo de datos del formulario apuntado, usando el parámetro que inicialmente se le envió a la anterior instancia si hubiese. El atributo correspondiente de la estructura Trámite del frmTramite al cual el imgDoc pertenece apunta entonces a la nueva instancia.

5.2.2. Reportes Generados por el Sistema

Un formulario de salida puede ordenar mostrar uno o varios reportes construidos con los mismos datos. Estos reportes fueron creados usando Crystal Reports™ y mostrado por un formulario de la clase frmRpt; que contiene internamente un arreglo de reportes en forma de instancias de la clase CrystalDecisions.CrystalReports.Engine.ReportClass, que son enviados por un formulario de salida (por ejemplo, los formularios A, B, C y Vehículos que pertenecen al DAU), y mostrados a través de una barra de botones en la que se elige qué reporte mostrar dentro del arreglo recibido en su constructor, y un CrystalDecisions.Windows.Forms.CrystalReportViewer, que es el que efectivamente los muestra.

Los reportes que son mostrados a través de esta clase son: los formularios impresos bajo estándar A, B y C del DUI y del DAU; así como los de vehículos y contenedores del DAU, y los correspondientes de la DAV. Cada conjunto de reportes son enviados como un arreglo por su respectivo formulario.

5.3. Creación de las Instancias de los Formularios

Los cuatro formularios de entrada: Banco, Transporte y Mercadería tienen dos constructores para dos escenarios diferentes.

El primero de ellos ocurre cuando se crea un trámite nuevo y la información de ese formulario será llenada por primera vez. Este constructor no lleva argumentos. Existe una variable privada boolean llamada `bIsNew` cuyo valor es `True` al salir de este constructor. Adicionalmente, el formulario tiene una variable miembro privada heredada de la clase base `System.Data.DataRow` que refleja el nombre la tabla transaccional a la que representa y que será llenada con nueva información recibida por el usuario. Se instancia mediante la función `Public Function <N. tabla>DataTable.New<N. tabla>Row()`. A continuación los formularios en mención y los nombres de las clases `dataRows` referidas:

frmBanco	Ax2DataSet. Bank_Agency_Process_detailRow
frmTransporte	Ax2DataSet.Process_Line_Agency_DetailRow
frmMercaderia	Ax2DataSet.InvoiceRow

Tabla 5.3: Clases heredadas de DataRow para el sistema

El segundo escenario ocurre cuando se abre un trámite grabado previamente y es llamado cada formulario en consecuencia para su posterior modificación. Los formularios de entrada tienen una variable miembro privada heredada de la clase base `System.Data.DataTable` que refleja a la tabla transaccional a la que

representa. A continuación los formularios en mención y los nombres de las clases dataTables referidas:

frmBanco	Ax2DataSet. Bank_Agency_Process_detailDataTable
frmTransporte	Ax2DataSet.Process_Line_Agency_DetailDataTable
frmMercaderia	Ax2DataSet.InvoiceDataTable

Tabla 5.4: Clases heredadas de DataTable para el sistema

Esta tabla es llenada mediante una consulta SQL a través de la modificación del commandText de la variable miembro selectCommand de su respectivo dataAdapter. Esta modificación consiste en variar este string de manera que devuelva solo el registro que le corresponda al trámite en cuestión. Considérese las cuatro siguientes sentencias en el ejemplo más sencillo, con la clase frmBanco:

Dim strCommand As String

Dim numRowsRetrieved As Integer

**strCommand = CType(Ax2.DataAdapters("Agency"),
Data.SqlClient.SqlDataAdapter).SelectCommand.CommandText**


```

CType(Ax2.DataAdapters("Agency"),
Data.SqlClient.SqlDataAdapter).SelectCommand.CommandText      =
strCommand & " WHERE pro_id = " & pro_id

```

```

numRowsRetrieved      =      CType(Ax2.DataAdapters("Agency"),
Data.SqlClient.SqlDataAdapter).Fill(dt)

```

```

CType(Ax2.DataAdapters("Agency"),
Data.SqlClient.SqlDataAdapter).SelectCommand.CommandText      =
strCommand

```

El `commandText` es modificado para que devuelva solo el registro en el que coincida el `pro_id` de la tabla `Process` que le corresponda a ese trámite. Luego la variable es dejada como estaba originalmente. El resultado de esta operación es un `dataTable` con un solo `dataRow`. Sólo por considerar el escenario no exitoso se chequea que el número de rows recuperadas, en cuyo caso la variable `blsNew` es `True`; pero en todo caso la variable debería ser `False`.

5.3.1. Creación de un Trámite Nuevo

Existe un único método (en el formulario principal) para la creación de un nuevo trámite que se encarga de todas las

tareas que dicha creación implica. Primero se crea la instancia de frmTramite nueva llamando a su constructor sin argumentos, con lo que ya se indica que se trata de un trámite a llenarse por primera vez (`bIsNew = True`). Luego se crean cada uno de los formularios de entrada mediante la sentencia inline que se refirió en la sección que se trata acerca de los Image-Documents, para añadirlos a la estructura Tramite del frmTramite. Cada formulario es instanciado llamando a su constructor por defecto. Finalmente, las herramientas que puede utilizar el usuario en la barra de botones y menús son habilitadas, que entre otras están: la verificación, grabación y creación de los documentos de salida pertinentes. A continuación, un diagrama de interacción de objetos que explica la creación de un trámite nuevo.

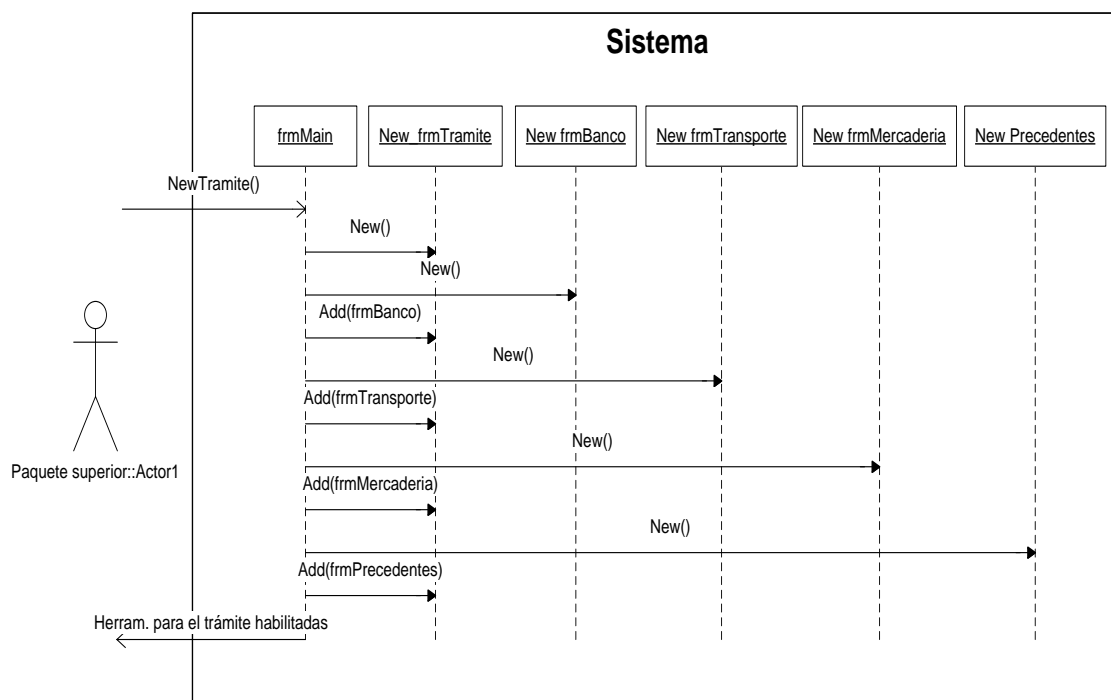


Figura 5.7: Creación de un nuevo trámite

5.3.2. Apertura de un Trámite

En caso de que el usuario decida abrir un trámite existente, el método que se encarga de esta tarea, en primer lugar abre una instancia de la clase frmTramitesGrid, en cuyo constructor se envía el parámetro enumeración eAction = Open, para abrir un trámite existente. Después que el usuario elige el trámite a abrir, se crea la instancia de frmTramite enviando como parámetro al constructor el pro_id del registro de la tabla Process que se buscó en la base de datos y que corresponde al trámite escogido. El mismo parámetro se envía a los formularios de Banco,

Transporte, y Precedentes (instancia de frmTramitesGrid que sobreentiende que enviarle el pro_id al constructor implica eAction = ShowPrec). Al único que se le envía otro parámetro es al formulario de Mercadería, al que previamente se encontró el inv_num de la tabla Invoice que corresponda en la relación Process-Invoice, y que se encuentra en la tabla Process. Cada vez que se crean cada una de las instancias de los formularios, estas son añadidas al atributo Tramite que referencia a los formularios, nuevamente mediante la sentencia inline que adicionalmente crea a los imgDocs respectivos. El diagrama de interacción de objetos para esta operación sigue a continuación:

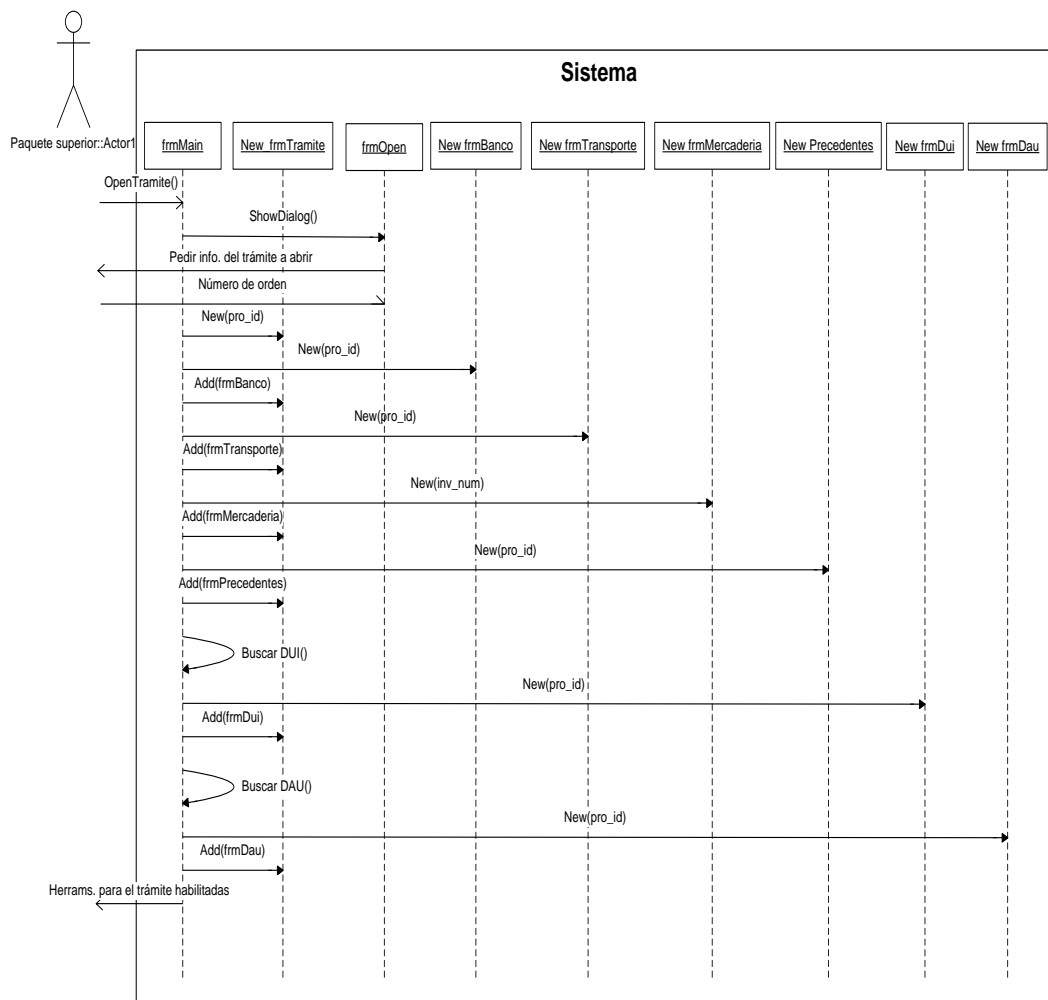


Figura 5.8: Apertura de un trámite

5.3.3. Verificación y Grabado de Datos

Cuando el usuario presiona el botón "Verificar" dentro de un formulario, solamente se verifica ese formulario; mientras que si se presiona el botón "Verificar" de la barra de herramientas, se verifica el trámite completo.

Cuando la orden de verificación se la da al trámite, en primer lugar se verifican los controles que frmTramite tenga dentro

de sí. De haber tareas pendientes en ellos, se agrega una tarea por cada control en el panel de tareas de frmTramite. Luego se verifican todos los formularios existentes en el trámite siempre y cuando el usuario haya hecho cambios en algún control (con la propiedad readOnly form.HasFormChanged, que es True cuando se dispara el evento TextChanged de los textboxes en el formulario, o ValueChanged en el caso de las instancias de ColectorFechaHora), o el formulario es nuevo (reflejado por la propiedad form.IsNew, que es False si se ha instanciado sin cargar datos de la base de datos, mostrando un formulario vacío, o si se han grabado los datos dentro del este).

Cada formulario tiene un criterio general o específico para verificarse. En todo caso, con el fin de mejorar su rendimiento, todos omiten cualquier verificación si ya ha sido verificado antes y no ha habido cambios desde entonces. La variable privada _alreadyChanged toma el valor True apenas el formulario ha sido verificado y retorna a False cuando el usuario ha efectuado cambios en los valores de sus controles.

Cuando un formulario añade alguna tarea dentro de sí, adicionalmente agrega una tarea en su frmTramite, que

indica que hay tareas pendientes en ese formulario. Toda tarea en el panel de tareas apunta al control que tiene dicha tarea, con el fin de acceder a ella rápidamente. Una tarea que indica tareas pendientes en un formulario apunta a ese formulario.

En el siguiente diagrama de interacción de objetos, se muestra el escenario siguiente: se trata de un trámite nuevo, en el que ningún formulario ha sido abierto a excepción del formulario banco y transporte, al que se han llenado algunos datos, aunque existen campos mandatorios que no han sido llenados. El formulario banco no ha sufrido ningún cambio. El usuario ha ordenado verificar el trámite completo.

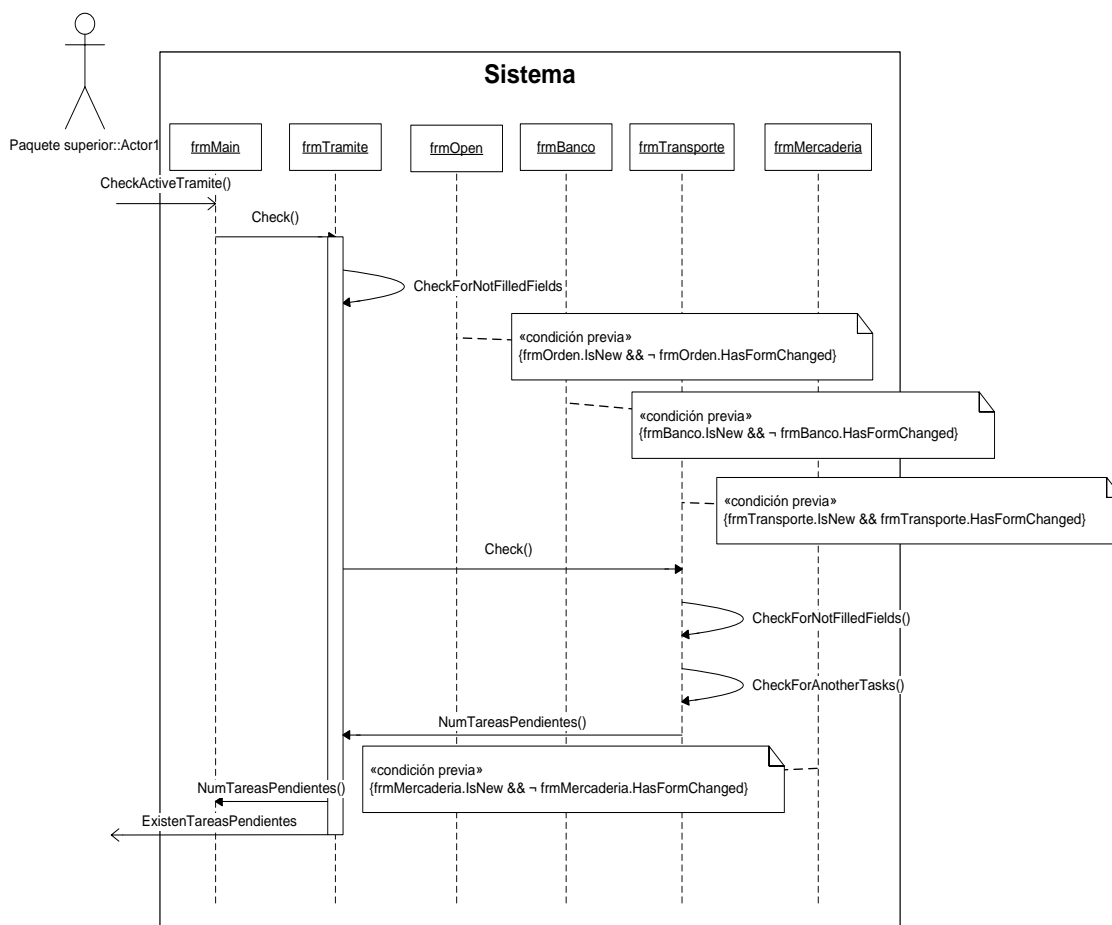


Figura 5.9: Trámite Completo

A fin de almacenar los datos ya sea nuevos o modificados de vuelta a la tabla de la base de datos, se utiliza el método `Update` del respectivo `DataAdapter` y se envía como parámetro el `DataTable` con los datos modificados. La interfaz o firma de este método es la siguiente:

```
Overloads Public Function Update(ByVal dataTable As
DataTable) As Integer
```


Este método llama a las respectivas declaraciones INSERT o UPDATE) para cada registro o fila insertado o modificado en el especificado dataTable. Declaraciones que contengan la sentencia DELETE nunca serán llamadas porque un ningún registro perteneciente a un trámite puede ser borrado.

Cuando se llama al método Update, el dataAdapter examina la propiedad RowState de cada row en el argumento recibido y ejecuta las declaraciones SQL requeridas iterativamente para cada row.

Cuando se usa Update, el orden de ejecución es el siguiente:
Los valores en el DataRow son copiados a los valores de los parámetros.

El evento RowUpdating se dispara.

El command se ejecuta.

Si el command es fijado a FirstReturnedRecord, para luego el primer resultado se ubique en el DataRow.

Si hay parámetros de salida, se ubican en el DataRow.

El evento RowUpdated se dispara.

El método AcceptChanges del DataRow es llamado.

El valor de retorno del método Update es el número de filas actualizadas exitosamente. Aunque todo este proceso ocurre cada vez que se grabe un formulario, es totalmente transparente incluso para el programador.

Cada uno de los formularios de Banco, Transporte, Mercadería y Precedentes tiene como mínimo un dataTable (en el caso de Mercadería, dos) llenado dentro de su constructor con argumento con la información pertinente si se trata de un trámite ya grabado; y un dataRow, para cuando el trámite es nuevo, instanciado en el constructor sin parámetros. Si el valor de la variable boolean blsNew es True, la información de los controles del formulario son ubicados en los respectivos atributos del dataRow; y luego ese dataRow se añade al dataTable mediante el método heredado DataTable.Rows.Add(ByVal DataRow). Finalmente, se llama al método Update enviando como parámetro el dataTable y cambiando el valor de la variable blsNew a False, en el escenario exitoso de esta operación. Si el usuario ordena a grabar un formulario determinado por primera vez -exceptuando el formulario de mercadería que no necesita un pro_id para grabarse, llama a grabarse la información del trámite antes de continuar con el proceso. Si la orden de

grabación del usuario va al trámite y el trámite es nuevo, antes de guardar la información del frmTramite, se manda a guardar al formulario de Mercadería para obtener el inv_num de la tabla Invoice, necesario al momento de guardar la información del trámite que pertenece a la tabla Process. Finalmente, se graba cada uno de los formularios que haya sido abierto y tenga cambios. El comando falla cuando no se completa uno de los pasos mencionados.

5.4. PLAN DE PRUEBAS

Con el fin de asegurar que el sistema funcione de acuerdo a lo esperado se hizo el siguiente plan de pruebas, en base a las operaciones más relevantes que se realizan con el sistema.

Número	1
Nombre de la prueba	Creación de un trámite.
Instrucciones para la prueba	<p>1.- Un usuario debe ingresar la información de componente. No importa en el orden que en cada se vaya llenando los componentes siempre y cuando se llenen los campos mandatorios.</p> <p>2.- La información que ingresó el usuario debe ser consistente con el proceso que se está realizando (IMPORTACION/EXPORTACION).</p>
Comportamiento aceptable	<p>Si se llenaron los campos mandatorios, el sistema no emitirá ningún mensaje de error.</p> <p>De lo contrario, se presentará un mensaje indicando que campo mandatorio falta por llenar.</p>

Número	2
Nombre de la prueba	Edición de un trámite. Apertura de un trámite guardado.
Instrucciones para la prueba	1.- El usuario debe seleccionar el trámite que desea editar. Puede utilizar como criterios búsqueda de los trámites el número de orden, proveedor, cliente. 2.- Una vez que encontró el trámite, el usuario puede abrir el trámite y cambiar la información y los campos mandatorios no deberán quedar vacíos.
Comportamiento aceptable	En el momento que se abra un trámite que ha sido guardado se mostrará toda la información correspondiente en cada uno de los componentes del sistema,

Número	3
Nombre de la prueba	Verificación de datos de un trámite.
Instrucciones para la prueba	1.- El usuario en el momento de hacer la verificación de los datos hará clic en el botón de Verificación.
Comportamiento aceptable	El sistema ejecutará un proceso que se encargará de verificar que todos los campos mandatorios del trámite que se realizó se encuentren llenos. En caso de que se encuentre algún campo obligatorio vacío el sistema enviará en forma de mensaje el nombre del campo que debe ser llenado.

Número	4
Nombre de la prueba	Grabación de un trámite.
Instrucciones para la prueba	1.- El usuario hace clic en el botón "Grabar". 2.- El sistema, antes de que se realice la

	<p>grabación de cualquiera de los datos que se encuentra en los componentes de un trámite verifica automáticamente que se encuentren completos todos los datos necesarios para poder realizar el trámite.</p> <p>3.- El sistema grabará la información general del trámite (campos obligatorios) y luego la información que se encuentra dentro de cada componente de manera secuencial.</p>
Comportamiento aceptable	<p>El sistema en caso de encontrar que existen tareas pendientes (llenar algún componente o algún campo obligatorio) se encuentran vacíos, impedirá que se realice la grabación y emitirá un mensaje de error indicando que tarea se encuentra pendiente. En caso de que al momento de realizar la verificación automática no se encuentra ninguna tarea pendiente, el trámite será guardado.</p>

Número	5
Nombre de la prueba	Generación los documentos del trámite para realizar declaración.
Instrucciones para la prueba	<p>1.- El usuario realiza la generación de los componentes de salida correspondientes a los documentos declaratorios (DUI, DAU, DAV). Se mostrará documentos por pantalla.</p> <p>2.- El usuario luego que ha visto los documentos por pantalla podrá visualizar como será impreso cada documento.</p> <p>3.- Luego el usuario generará los archivos de intercambio electrónico de datos (IED) con la CAE. Este archivo se generará con datos que se encuentran en el documento DAU del trámite.</p>
Comportamiento	En el momento que el usuario realice la

aceptable	generación de los documentos para el trámite, se mostrarán los documentos en pantalla. Al momento de generar los archivos de intercambio se crearán archivos de texto en la unidad C del computador.
------------------	--

Número	6
Nombre de la prueba	Realizar seguimiento de un trámite.
Instrucciones para la prueba	1.- El usuario ingresa comentarios acerca del proceso del llenado del trámite.
Comportamiento aceptable	En el momento que el usuario abra algún trámite se mostrará en la pantalla principal que comentarios se han ingresado en el proceso.

5.5. Resultados de las pruebas

En esta sección se presentarán los resultados que se obtuvieron al realizar las pruebas anteriormente mencionadas, en el sistema de despachador aduanero. En cada caso, el grado de éxito se ha medido en comparación al comportamiento aceptable descrito para la prueba.

Número	1
Nombre de la prueba	Creación de un trámite.
Resultados	En primera instancia no se llenaron dos

	<p>campos obligatorios por lo cual en el cuadro de tareas pendientes sale el error de que campos faltan por llenar.</p> <p>En un segundo intento se llenaron todos los campos obligatorios y se pudo continuar con el proceso.</p>
--	--

Número	2
Nombre de la prueba	Edición de un trámite. Apertura de un trámite guardado.
Resultados	La respuesta que obtuvo el usuario en el momento que el hizo clic sobre un trámite anterior es que se le mostrara todos los datos que habían sido llenados en su momento para ese trámite.

Número	3
Nombre de la prueba	Verificación de datos de un trámite.
Resultados	Se hicieron 2 intentos, el primero no se llenaron algunos campos obligatorios con el cual salió el mensaje de error que se deben llenar todos los campos. En el segundo intento se llenaron todos los campos y se continuò con el proceso.

Número	4
Nombre de la prueba	Grabación de un trámite.
Resultados	En el momento de hacer clic sobre "Grabar" el sistema comienza a validar que los campos obligatorios no esten vacíos y al terminar se puede ya crear cualquier documento de trámite (DUI, DAU, DAV).

Número	5
Nombre de la prueba	Generación de los documentos del trámite para realizar declaración.
Resultados	El usuario pudo ver los documentos del trámite por pantalla y los pudo imprimir.

	Se verificó que los archivos para el intercambio electrónico se generen y se encuentren en el disco C del computador.
--	---

Número	6
Nombre de la prueba	Realizar seguimiento de un trámite.
Resultados	Se pudo ingresar comentarios acerca del proceso. Se verificó en las tablas y se vio que si se guardó el comentario, la fecha y el número de trámite.

CONCLUSIONES Y RECOMENDACIONES

Como resultado del desarrollo de esta tesis, tenemos un sistema que se fundamenta básicamente en el dominio de la empresa (despachador aduanero); esto facilita en el mantenimiento y actualización del sistema. Además, tiene una interacción mas amigable que otros sistemas de despacho aduanero por lo que los trámites de importación y exportación se agilitan y se hace más eficiente el proceso que realizan los despachadores aduaneros.

Por medio de este sistema, los despachadores aduaneros podrán realizar el llenado de datos en los formularios de aduana respectivos de una manera más rápida y organizada.

Este sistema puede implementarse en empresas despachadoras de cualquier tamaño, y soportar futuros crecimientos, lo que lo hace escalable.

El modelo de MERODE (EDG) sirvió como punto de arranque para el modelo de la Base de Datos lo cual facilitó mucho su creación. Además, MERODE ayudó a identificar de una manera clara los objetos de la empresa.

Los FSM que se realizaron sirvieron para ver de una forma mas clara cada proceso del sistema.

Como recomendación después del desarrollo de esta tesis es que en la implementación de un sistema se debe usar una herramienta que sea multiplataforma como JAVA para que el sistema pueda ser utilizado en cualquier lado sin tener ninguna restricción por Sistemas Operativos.

Se debe dar seguimiento al sistema desarrollado para verificar la facilidad de mantenimiento que proporciona la herramienta.

En próximas tesis en las que se vaya a usar la metodología MERODE se debe implementar el uso de métricas de tiempos estimados vs. Tiempos reales; y de errores (de dominio, de funcionalidad y de presentación) vs.

alguna medida de tamaño del sistema para probar de una forma mas cuantitativa la metodología.

APÉNDICES

APÉNDICE A

DICCIONARIO DE DATOS

Aforo:

- Af_desc.- tipo varchar, longitud 20. Af_desc es el nombre del aforo. No puede ser null.
- Af_id.- Primary Key. Es tipo integer. Almacena el id del aforo. No es null.
- Aforo.-La tabla de Aforo es una tabla maestra del sistema. En la cual consta dos tipos de datos: el código del aforo y el nombre del aforo.

Agente:

- Ag_cod.- tipo varchar, longitud 5. Ag_cod es el número de código único que tiene cada agente.

- Ag_id. - Primary key. Es tipo integer. Almacena el id del agente. No es null.
- Ag_nam.- tipo varchar, longitud 20. Ag_nam es el nombre del agente. No permite valores null.
- Ag_st.- tipo char, longitud 1. Ag_st es el estado del agente, este puede ser activo(A) o inactivo (D).
- Ag_type.- tipo integer. Ag_type define que tipo de agente es.
- Agent.-En la tabla agent se tienen todos los datos del agente que interviene en un proceso aduanero.

Agent_Client_Detail:

- Agent_Client_Detail.-En la tabla Agent_Client_Detail se almacenan los datos que son detalles de la relación Client-Agent, la cual es una relación muchos a muchos.
- Agent_Client_Detail.Det_id.- Primary key. Es de tipo identity, el cual se aumenta en 1 de forma automática cuando se ingresan los datos del ag_id y client_id.

Agreement:

- Agree_id.- Primary key. Este campo es de tipo integer. Agree_id es el id del convenio con el cual se puede hacer referencia a un convenio.

- Agree_des.- Es de tipo varchar, longitud 50. Agree_desc es la descripción del convenio. Es el cual donde se encuentran ciertos datos del convenio.
- Agree_name.- Es de tipo varchar, longitud 50. Agree_name es el nombre que lleva el convenio.
- Agreement.-En la tabla agreement se encuentran todos los convenios que pueden existir en un proceso aduanero. Es una tabla maestra del sistema.

Aso_Document_Type:

- Aso_cod.- Primary key. Es de tipo char, longitud 2. Es el código asignado a un cierto tipo de documento asociado.
- Aso_name.- tipo varchar, longitud 50. Es el nombre del tipo de documento asociado. Tiene asignado un código único para cada tipo de documento asociado.
- Aso_Document_Type.-En la tabla Aso_Document_Type se encuentra información acerca de los tipos de documentos asociados. Es una tabla maestra del sistema.

Bank:

- Ban_id.- Primary key. Es de Tipo varchar, longitud 4. Es el código de un banco.

- Ban_nam.- Tipo varchar, longitud 20. Es el nombre del banco el cual tiene un código único asignado.
- Ban_st.- tipo char, longitud 1. Es el estado que se encuentra el banco con la empresa para poder realizar cobros de procesos aduaneros. Puede ser activo o inactivo.
- Bank.-La tabla bank contiene los datos de los bancos en los cuales se pueden realizar pagos para continuar con algún proceso aduanero.

Bank_Agency_Process_Detail:

- Agency.- tipo varchar, longitud 10. Es el nombre de la agencia bancaria. Esta relacionado con un código de banco. En una agencia bancaria se pueden realizar varios pagos.
- Ap_date.- tipo datetime, longitud 8. Es la fecha de aprobación del pago, es decir, la fecha en la se realizó el pago.
- Bank_Agency_Process_detail.-La tabla Bank_Agency_Process_detail contiene los detalles de la relación banco – agencia – proceso. Los procesos aduaneros pueden ser pagados en cualquier banco de la ciudad, y un banco posee una o muchas agencias.

Ben_prov_detail:

- Ben_Prov_detail.-En la tabla Ben_Prov_detail se encuentran los detalles de la relación entre proveedores y beneficiarios.
- Ben_Prov_detail.Det_id.- Primary key, tipo integer. Es la clave del detalle que existe en la relación beneficiario proveedor.

Beneficiary:

- Ben_id.- Primary key. Tipo varchar, longitud 4. Es el código único que tiene cada beneficiario.
- Ben_name.- Tipo varchar, longitud 50. Es el nombre del beneficiario. Tiene un código por cada beneficiario.
- Ben_st.- Tipo char, longitud 1. Es el estado en el que se encuentra el beneficiario, puede ser activo o inactivo.
- Beneficiary.-La tabla beneficiary tiene todos los datos del beneficiario en la transacción.

Cargo:

- Cargo.-La tabla cargo es la tabla que tiene todos los tipos de carga.
- Cargo_id.- Primary key. Tipo varchar, longitud 2. Es el código único del cargo.
- Cargo_name.- Tipo varchar, longitud 30. Es el nombre del cargo.

- Cargo_st.- Tipo char, longitud 1. Es el estado en el que se encuentra el cargo. Puede ser activo o inactivo.

CIIU:

- CIIU.-La tabla CIIU contiene todos los CIIU que podría ser un cliente. Es una tabla maestra del sistema.
- CIIU_cod.- Primary key. Tipo char, longitud 4. Es el código del CIIU.
- CIIU_desc.- Tipo varchar, longitud 50. Es el nombre que tiene el CIIU.
- CIIU_st.- Tipo char, longitud 1. Es el estado en el que se encuentra el CIIU. Puede ser activo o inactivo.

Class:

- Class.-Contiene todos los datos de las clases existentes. Es una tabla maestra del sistema
- Cls_cod.- Primary key. Es el código con el cual se puede identificar una clase.
- Cls_name.- Tipo varchar, longitud 10. Es el nombre que lleva la clase.
- Cls_st.- Tipo char, longitud 1. Es el estado en el que se encuentra la clase. Puede ser activo o inactivo.

Client:

- Client.- En la tabla Cliente se guarda los datos de los clientes frecuentes de la compañía.
- Client_add.- tipo varchar, longitud 50. Client_add almacena las direcciones de cada cliente. Acepta valores null.
- Client_fax.- tipo varchar, longitud 10. Client_fax almacena el número de fax del cliente. Puede ser null.
- Client_Id.- Primary Key. Tipo integer, Almacena el id del cliente. No puede ser null.
- Client_mail.- tipo varchar, longitud 50. Client_mail almacena el correo electrónico del cliente. Soporta valores nulos.
- Client_name.- tipo varchar, longitud 50. Client_name almacena los nombres de los clientes.
- Client_phone.- tipo varchar, longitud 10. Client_phone almacena el número de teléfono del cliente. Puede ser null.
- Client_st.- tipo char, longitud 1. Client_st es el estado en el que se encuentra el cliente. Puede ser activo (A) o desactivo (D).

Comerce_Operator_Type:

- Com_name.- Tipo varchar, longitud 50. Es el nombre del tipo de operador de comercio.
- Com_type.- Primary key. Tipo varchar, longitud 2. Es el código del tipo de operadores de comercio que existen.

- Commerce_Operator_type.-La tabla Commerce_operator_type contiene todos los datos de todos los tipos de operadores de comercio. Es una tabla maestra del sistema.

Container:

- Cont_id.- Primary key. Tipo integer. Es un código de tipo numérico que se le asigna a cada contenedor dentro del sistema.
- Cont_num.- Tipo varchar, longitud 15. Es un código de tipo numérico que se le asigna a cada a nivel de la aduana.
- Container.-La tabla container tiene todos los datos acerca de los contenedores. Es una tabla maestra en el sistema.
- Marks.- tipo varchar, longitud 60. Es la descripción de la marca del container
- Precinto_num.- Tipo varchar, longitud 15. Es el número del precinto en el cual se encuentra el container.
- Situation.- tipo varchar, longitud 50. Es el nombre del lugar en donde se encuentra ubicado el container.

Country:

- Country_id.- Primary key. Tipo varchar, longitud 6. Es un código numérico que se le asigna a cada país en el sistema.

- Country_cod.- tipo varchar, longitud 3. Es el código que se les asigna a los países en los procesos aduaneros.
- Country_name.- Tipo varchar, longitud 20. Es el nombre del país.
- Country_st.- tipo char, longitud 1. Es el estado en el que se encuentra el país. Puede ser activo o inactivo.
- Country.-La tabla country es la que contiene todos los datos necesarios de los países a nivel mundial.

Country_Agree_Detail:

- Country_Agree_detail.-La tabla country_agree_detail se llena con los detalles de la relación que existe entre el país y algún convenio. Esta tabla surge dado que entre país y convenio hay una relación muchos a muchos.
- Id_det_conv_pais.- Primary key, tipo integer. Es el identificador del detalle de la relación entre convenio y país. Se va incrementando de manera automática en valor de uno (1).

DAU:

- DAU (Documento Aduanero Único).-La tabla DAU lleva valores que son importantes para el proceso aduanero. Uno de los valores que lleva es el código de proceso. Esta relacionado con los dos tipos de procesos.

- Dau_id.- Primary key. Tipo char, longitud 10. Es el código con el cual se identifica un dau. Es una cadena string de 10 caracteres, el cual esta compuesta por el código de la aduana mas los 2 últimos números del año mas un numero secuencial.
- Dau_num.- tipo numeric, longitud 9. Es el numero secuencial que forma parte del id del DAU.
- Dis_type.- tipo varchar, longitud 4. Es el código del tipo de despacho con el cual se ha realizado el proceso.
- Omc_val_met.- Método de valoración de la Organización Mundial de Comercio.
- Reception_date.- tipo datetime. Es la fecha en la cual se ha hecho la recepción del documento por parte de la aduana.
- Refrend_num.- tipo varchar, longitud 20. Es el número con el refrenda un dau de manera física.

DAV:

- DAV.-La tabla DAV lleva todos los datos de los documentos DAV. El DAV es un documento que solo se relaciona con las importaciones.
- DAV.Cont_num.- tipo varchar, longitud 40. Es el número del contrato, el cual se ha firmado para realizar algún trámite extra dentro de un proceso aduanero. Soporta valores NULL.

- DAV.Cont_date.- tipo datetime, longitud 8. Es la fecha en la cual se hizo el contrato para realizar todos los tramites de importacion. Soporta valores NULL.
- DAV.Decl_date.- tipo small datetime, longitud 4. Es la fecha en la que se ha hecho la declaración.
- DAV.Item_total.- tipo integer, longitud 4. Es el numero de ítems que se estan declarando dentro de un DAV que pertenece a un proceso de Importación.
- DAV.Res_duty.- tipo varchar, longitud 15. En este campo se almacena el valor o el numero de resolucio de la Aduana. Soporta valores NULL.
- DAV.Res_duty_date.- tipo datetime, longitud 8. En este campo se almacena la fecha en la cual fue expedido el número de resolucio de la aduana. Soporta valores NULL.
- DAV.Send_num.- tipo integer, longitud 4. Es el campo en el cual se indica el valor de envios que se realizan dentro de un trámite o proceso aduanero.
- Dav_id.- primary key. Tipo varchar, longitud 10. Es el código que lleva la DAV. Esta formado por una cadena de 10 caracteres la cual es la unión de el código de la aduana mas los dos últimos dígitos del año mas el numero secuencial del dav.
- Question 6.1.- tipo bit, longitud 1. En este campo se almacena el valor si o no como respuesta a la pregunta de que si Se utilizó

algún intermediario en la transacción comercial?. El numero 1 significará SI y el numero cero será el NO.

- Question 7.1.- tipo bit, longitud 1. En este campo se almacena el valor si o no como respuesta a la pregunta de que si Existe vinculacion con el proveedor?. El numero 1 significará SI y el numero cero será el NO.
- Question 7.2.- tipo bit, longitud 1. En este campo se almacena el valor si o no como respuesta a la pregunta de que si Ha influido la vinculacion en el precio de las mercancías importadas?. El numero 1 significará SI y el numero cero será el NO.
- Question 7.3.- tipo bit, longitud 1. En este campo se almacena el valor si o no como respuesta a la pregunta de que si Existen pagos indirectos relativos a la mercaderias?. El numero 1 significará SI y el numero cero será el NO.
- Question 7.4.- tipo bit, longitud 1. En este campo se almacena el valor si o no como respuesta a la pregunta de que si Existen cánones o derechos de licencias relativas a las mercaderias importadas que Ud. Esta obligado a pagar directa o indirectamente como condicion de venta?. El numero 1 significará SI y el numero cero será el NO.
- Question 7.5.- tipo bit, longitud 1. En este campo se almacena el valor si o no como respuesta a la pregunta de que si Está la

venta condicionada por un acuerdo según el cual una parte del producto de cualquier reventa, cesión o utilización posterior de las mercancías importadas se revierta directa o indirectamente a su proveedor extranjero?. El número 1 significará SI y el número cero será el NO.

- Question 7.6.- tipo bit, longitud 1. En este campo se almacena el valor si o no como respuesta a la pregunta de que si Existen restricciones para la cesión o utilización de las mercancías por el importador, de acuerdo a lo señalado en el artículo 1 del acuerdo del valor GATT?. El número 1 significará SI y el número cero será el NO.
- Question 7.7.- tipo bit, longitud 1. En este campo se almacena el valor si o no como respuesta a la pregunta de que si Depende la venta o el precio, de condiciones o contraprestaciones en relación a las mercancías a valorar?. El número 1 significará SI y el número cero será el NO.
- Question 7.8.- tipo bit, longitud 1. En este campo se almacena el valor si o no como respuesta a la pregunta de que si Puede determinarse el valor de las condiciones o contraprestaciones ?. El número 1 significará SI y el número cero será el NO.
- Question 8.5.- tipo bit, longitud 1. En este campo se almacena el valor si o no como respuesta a la pregunta de que si Tiene

carácter estimativo o provisional los casilleros 8.2.4 y 8.2.5. El numero 1 significará SI y el numero cero será el NO.

- Incoterm.- tipo varchar, longitud 3. Es el tipo de termino comercial que se esta utilizando dentro de un proceso aduanero.
- Naturaleza.- tipo varchar, longitud 2.
- Total 8.1.- tipo Money, longitud 8. Es el campo en el cual se almacena el valor que se debe pagar como total de los importes incluidos tales como precio factura, pagos indirectos, descuentos retroactivos entre otros.
- Total 8.2.- tipo Money, longitud 8. Es el campo en el cual se almacena el valor que se debe pagar como total de los pagos tales como comisiones, corretajes, salvo comisiones compra, envases y embalajes, bienes y servicios suministrados por el importador, cánones y derechos de licencia, producto de cualquier reventa, gastos de entrega, gastos de transporte, gasto de carga y gasto seguro.
- Total 8.3.- tipo Money, longitud 8. Es el campo en el cual se almacena el valor que se debe pagar como total de algunos valores tales como gasto de entregas posteriores a la importación, intereses, asistencia técnica, derechos de aduana y otros gastos.

- Total 8.4.- tipo Money, longitud 8. Es el campo en el cual se almacena el valor total que se pagará, obtenido por medio de la sumatoria de los valores Total 8.1, Total 8.2 y Total 8.3.
- Sending_way.- tipo bit, longitud 1. Es la forma de como ha sido enviada la carga. Puede ser parcial o completa. Este campo puede coger valores de 0 o 1.
- Num_containers.- Tipo integer, longitud 4. Es el número de container en el cual se encuentra la mercadería.

Deposit:

- Dep_cod.- tipo varchar, longitud 5. Es el código que tiene cada depósito para realizar los procesos aduaneros. Este el código que se le debe enviar a la CAE.
- Dep_id.- Primary key. Tipo integer, longitud 4. Es el código que tiene cada depósito dentro del sistema.
- Dep_nam.- tipo varchar, longitud 30. Es el nombre que lleva cada depósito.
- Dep_st.- tipo char, longitud 1. Es el estado en el que se encuentra cada depósito. Puede ser activo o inactivo.
- Deposit.- La tabla deposit guarda todos los datos de los depósitos existentes dentro de un proceso aduanero.

Destination_Type:

- Dest_name.- tipo varchar, longitud 20. Es el nombre que lleva cada deposito, esta relacionado con un solo código.
- Dest_type.- Primary key. Tipo char, longitud 1. Es el código único que tiene un tipo de destino.
- Destination_Type.-Es la tabla en donde se encuentran todos los tipos de destinos del sistema. Es una tabla maestra del sistema.

Dispatch:

- Dis_name.- tipo varchar, longitud 30. Es el nombre que lleva cada tipo de despacho.
- Dis_type.- Primary key. Tipo varchar, longitud 4. Es el código único que tiene cada tipo de despacho y sirve para poder hacer referencia a cada uno de ellos.
- Dis_st.- tipo char, longitud 1. Es el estado en el que se encuentra cada tipo de despacho. Puede ser activo o inactivo.
- Dispatch.-La tabla dispatch contiene todos los tipos de despachos existentes para poder realizar un proceso aduanero.

Doc_Process_Detail:

- Doc_date.- tipo char, longitud 10. Es la fecha en la que el documento fue expedido.

- Doc_num.- tipo char, longitud 10. Es el número que lleva cada documento. Por cada tipo de documento es un número único.
- Doc_Process_Detail.-La tabla Doc_Process_Detail contiene los detalles de la relación proceso- documentos. Esta tabla surge dado que la relación entre proceso y documento es de muchos a muchos.
- Doc_Process_Detail.Description.- tipo varchar, longitud 50. Es la descripción de cada relación proceso- documento.
- Doc_Process_Detail.Det_id.- Primary key. Tipo integer. Se incrementa de manera automática en uno (1). Es el id con el cual se reconoce cada detalle de la relación.
- Emitter_entity.- tipo varchar, longitud 40. Es el nombre de la entidad que ha emitido el documento. Este campo soporta valores null.
- Exp_date.- tipo datetime, longitud 8. Es la fecha máxima de validez del documento. Luego de esta fecha el documento no será valido para poder realizar un proceso aduanero.

Doc_Type:

- Doc_type_st.- tipo char, longitud 1. Es el estado en el que se encuentra el tipo de documento. Puede ser activo o inactivo.
- Doc_type_id.- Primary key. Tipo varchar, longitud 2. Es el código con el cual se identifica un tipo de documento.

- Doc_type_nam.- Tipo varchar, longitud 30. Es el nombre de cada tipo de documento. Puede ser cedula, RUC o pasaporte.
- Doc_Type.-La tabla Doc_type es la tabla en donde se encuentran todos los tipos de documentos de la persona o empresa que es cliente.

Doc_Type_Detail:

- Doc_type_detail.-La tabla Doc_type_detail tiene los datos que resultan de la relación que existe entre tipo de documento y cliente. Esta relación es de muchos a muchos.
- Doc_type_num.- tipo varchar, longitud 13. Es el número que lleva cada tipo de documento para cliente. Para cliente existe solo un número por cada tipo de documento que tenga.

Document:

- Doc_id.- Primary key. Tipo integer, longitud 4. Es el código con el cual se identifica un documento.
- Doc_name.- Tipo varchar, longitud 20. Es el nombre que lleva cada documento de acompañamiento.
- Document.-La tabla de Document lleva todos los documentos que se necesitan para todos y cada uno de los procesos aduaneros. Pueden ser certificados, exoneraciones, permisos, etc. Es una tabla maestra del sistema.

- Document.Doc_des.- Tipo char, longitud 50. Es la descripción para cada documento.
- Document.Doc_type.- Tipo char, longitud 10. Es el tipo de documento.
- Document.Doc_st.- Tipo char, longitud 1. Es el estado en el que se encuentra un documento de acompañamiento. Puede ser activo o inactivo.

DUI:

- DUI (Documento Único de Importación).-La tabla DUI almacena todos los datos de los DUI's que serán ingresados al sistema.
- Dui_num.- Primary key. Tipo numeric. Es el número que tiene cada DUI. Es un número secuencial y único.
- DUI.Reception_date.- Tipo datetime, longitud 8. Es la fecha en la que se ha recibido el DUI por parte de la CAE.
- Numbering_date.- Tipo datetime, longitud 8. Es la fecha en la que se le asigno un número al DUI.
- Subject_to.- Tipo varchar, longitud 20. Es el campo en el cual se almacenara el tema al con el cual se puede dar una descripción un poco general del proceso que se va a realizar.
- Vistaforador.- Tipo varchar, longitud 20. Es el nombre de la persona que hace la revisión de la mercadería que ha ingresado por medio de un proceso aduanero.

- Checker.- Tipo varchar, longitud 20. En este campo se almacena el valor puesto como nombre del chequeador que se encuentra dentro del proceso aduanero.
- Num_part_disp.- Tipo integer, longitud 4. Número de despachos parciales. Tratándose de cambios de régimen y cuando proceda, se indicará el número de orden del despacho que se declara.
- Typed_by.- Tipo varchar, longitud 20. Es el nombre de la persona que ha tipado el refrendo del DUI.

Duty:

- Duty.-La tabla Duty contiene todos los datos de las aduanas que se encuentran dentro del país. Es una tabla maestra del sistema.
- Duty_cod.- Tipo varchar, longitud 50. Es el código de la aduana. Un código es asignado de forma única a cada aduana.
- Duty_id.- Primary key. Tipo integer, longitud 4. Es la clave primaria para cada aduana dentro del sistema.
- Duty_nam.- Tipo varchar, longitud 50. Es el nombre que lleva cada aduana.
- Duty_st.- Tipo char, longitud 1. Es el estado en el que se encuentra cada aduana. Puede ser activo o inactivo.

Exon_Cert:

- Exon_cert.-La tabla Exon_cert contiene todos los datos que se refiere a los certificados de exoneración. Es una tabla maestra del sistema dado que existen diferentes tipos de exoneraciones.
- Exon_cert_cod.- Primary key. Tipo integer, longitud 4. Es el código que se utiliza para identificar a cada certificado de exoneración.
- Exon_cert_desc.- Tipo varchar, longitud 100. Es el nombre o descripción de cada certificado de exoneración.
- Exon_cert_st.- Tipo char, longitud 1. Es el estado en el que se encuentra el certificado de exoneración. Puede ser activo o inactivo.

Free_Code:

- Cod_type.- Tipo varchar, longitud 50. Es el tipo de código que lleva cada código liberatorio.
- Free_code.-La tabla free_code contiene todos los datos de los códigos liberatorios que existen dentro de los procesos aduaneros. Es una tabla maestra del sistema.
- Free_cod_des.- Tipo varchar, longitud 50. Es el nombre del código liberatorio.
- Free_code_id.- Primary key. Tipo varchar, longitud 4. Es el código que lleva cada código liberatorio.

Free_Code-Doc_Detail:

- Free_code-Doc_Detail.-La tabla free_code-Doc_detail es la tabla que contiene todos los datos de la relación free code y documentos de acompañamiento. La relación entre estas dos tablas es de muchos a muchos. Un código liberatorio puede tener uno o varios documentos de acompañamiento, mientras que un documento de acompañamiento puede tener uno o varios códigos liberatorios.
- Free_code-Doc_Detail.Description.- tipo varchar, longitud 50. Es la descripción que existe dentro de la relación entre el código liberatorio y el documento de acompañamiento.

Invoice:

- Mone_dolar.- Tipo Money, longitud 8. Es el valor de cambio a dólares.
- ware_rate.- Es la tasa de almacenaje que se aplica al valor CIF de la mercadería declarada.
- Cif.- Tipo money, longitud 8. Es la suma del FOB total, el flete y el seguro total de la mercadería.
- Canon.- Tipo money, longitud 8. En este campo se almacena el valor que se paga por concepto de canones que se encuentra dentro del total 8.2.

- Com_amount.- Tipo integer, longitud 4. Es la cantidad de mercadería declarada expresada en unidades comerciales.
- Amount.- Tipo integer, longitud 4. Es la cantidad de mercadería que se encuentra dentro del proceso.
- Adv_total.- Tipo Money, longitud 8. Es el total de valor ad valorem que se cobrara por el proceso aduanero realizado.
- Antidump.- Tipo integer, longitud 4. Es la tasa Antidumping cargada al valor CIF de la mercadería declarada.
- Auto_total.- Tipo money, longitud 8. Representa el total de autoliquidación que el importador debe pagar, sumando los aranceles e impuestos de ley.
- Commission.- Tipo money, longitud 8. En este campo se almacena el valor perteneciente al total 8.2 y que se llama comisiones, corretaje, salvo de comisiones de compra. Soporta los valores NULL.
- cont_rate.- Es el porcentaje o tasa de control, usada entre los impuestos que el importador debe pagar para las mercancías que se despachen bajo los Regímenes Especiales de: Depósito Aduanero Comercial, público o privado; Almacén Libre (Ind Bond o Duty Free); Admisión Temporal con Reexportación en el Mismo Estado; y Zona Franca.

- Corpei_total.- Tipo money, longitud 8. En este campo se almacena el total a pagar a la Corpei por motivo de realizar un proceso aduanero.
- Cust_right.- Tipo money, longitud 8. En este campo se almacenará el valor que se pagan por derechos de licencia. Este valor forma parte del total 8.2.
- Discount.- Tipo Money. Es el valor que se descontará si algún caso lo amerita.
- Envases.- Tipo money, longitud 8. En este campo se almacena el valor que forma parte del total 8.2 y se refiere a lo que son los envases y embalajes que se le brinda al importador.
- fod_rate.- Tipo integer, longitud 4. Es la tasa que debe pagar el importador por concepto del fondo para el INNFA (Fodinfra).
- Fodinfra.- Tipo money, longitud 8. Es el valor en términos de la moneda de transacción del impuesto para el fondo del INNFA (Fodinfra).
- Freight.- Tipo money, longitud 8. Es el valor que el cliente a de tener que pagar por cuestión de flete. Este valor es proporcional al valor base de la importación.
- GrossWeight.- Tipo numeric, longitud 9. Es el peso bruto de la mercadería que se encuentra descrito en la factura.

- ICE_perc.- Tipo integer, longitud 4. Es el porcentaje ICE (Impuesto a Consumos Especiales) que se cobrará dependiendo que tipo de mercadería esta dentro del proceso aduanero.
- Ice_total.- Tipo money, longitud 8. En este campo se almacenará el valor total del ice que es el impuesto a consumos especiales que se pagará por el proceso aduanero.
- Insurance.- Tipo money, longitud 8. Es el valor que el cliente a de tener que pagar por motivo del aseguramiento del producto. Este valor es proporcional al valor base de la importación.
- Int_total.- Tipo money, longitud 8. En este campo se almacenará el valor del total de todos los intereses que se pagaron o se pagarán por la mercaderia que se esta trayendo por medio de un proceso aduanero.
- Inv_date.- Tipo datetime, longitud 8. Es la fecha en la cual ha sido expedida la factura. No soporta valores null.
- Inv_num.- Primary key. Tipo de datos varchar, longitud 20. Es el numero de la factura que ha llevado el cliente para poder realizar el proceso aduanero. No soporta valores null.
- Invoice.-La tabla invoice es la que contiene todos los datos de las facturas ingresadas en el sistema.
- IVA_perc.- Tipo integer, longitud 4. Es el porcentaje IVA (Impuesto al Valor Agregado) que se utiliza para imponerle a la mercadería que se encuentra dentro del proceso aduanero.

- Iva_total.- Tipo money, longitud 8. En este campo se almacenará el valor total del iva que es el impuesto al valor agregado que se pagará por el proceso aduanero.
- moder_rate.- Tasa de modernización. Es uno de los impuestos que se le atribuyen al CIF declarado de la mercadería.
- mul_total.- El porcentaje total al CIF por concepto de multas establecidas en los Artículos 94 y 96 de la Ley Orgánica de Aduanas por cometimiento de contravención aduanera y falta reglamentaria, tipificadas en los Artículos 93 y 95 de la misma Ley, respectivamente.
- Netweight.- Tipo numeric, longitud 9. Es el peso neto de la mercadería que se encuentra descrita en la factura.
- Num_bulks.- Tipo integer, longitud 4. Es el numero de bultos que se esta importando o exportando.
- Other_total.- Tipo money, longitud 8. En este campo se almacenará el valor de otros conceptos a pagar, se encuentra en el lugar 9.4 del DAV.
- Others.- Tipo money, longitud 8. En este campo se almacenará el valor de otros conceptos a pagar que se encuentra dentro del total 8.3 con el nombre de otros gastos.
- Post_costs.- Tipo money, longitud 8. En este campo se almacena el valor que forma parte del total 8.2 y se debe pagar

por el concepto de gastos de entrega posteriores a la importación como lo sería el transporte.

- Resell_products.- Tipo money, longitud 8. Es el campo en el cual se almacena el valor que se deberá pagar en caso de que el producto que se este importando tenga un producto de reventa cualquiera, cesión o utilización posterior que revierta al proveedor extranjero.
- Spe_rig_total.- Tipo money, longitud 8. Es el campo en el cual se almacena el valor si llegara a existir de lo que se define como derechos especiales sobre algunas clases de productos que se encuentran dentro de los procesos aduaneros.
- Sum_services.- Tipo money, longitud 8. Es el total de la suma de los valores de los servicios que se ha recibido para poder mover un proceso aduanero.
- Tecn_asist.- Tipo money, longitud 8. Es el campo en el cual se encuentra el valor que se debe o se debio pagar por la contratación de asistencia técnica para realizar el armado o la instalación de algún equipo importado por un cliente.
- Taxes.- Tipo Money. Es el valor que se sumará al FOB en caso de que existan impuestos.

Manifest:

- **Manif_name.-** Tipo varchar, longitud 21. Es el nombre que lleva un tipo de manifiesto.
- **Manif_type.-** Primary key. tipo varchar, longitud 2. Es el código con el cual se identifica un tipo de manifiesto para poder realizar el proceso aduanero.
- **Manifest.-**La tabla manifest es la tabla en donde se almacenan todos los datos de los manifiestos para poder realizar un proceso aduanero. Es una tabla maestra del sistema.

Money:

- **Mon_id.-** Primary key. Tipo integer. Es el código que se le asigna a una moneda por parte del sistema AX2.
- **Mon_cod.-** Tipo varchar, longitud 10. Es el código único que se le asigna a una moneda de manera estándar. Vale acotar que este código se utiliza para realizar los procesos aduaneros.
- **Mon_nam.-** Tipo varchar, longitud 20. Es el nombre con el cual se identifica a una moneda.
- **Mon_st.-** Tipo char, longitud 1. Es el estado en el cual se encuentra un tipo de moneda. Puede ser activo o inactivo.
- **Money.-**La tabla Money es la que guarda todos los datos acerca de todas las monedas que se aceptan en el proceso aduanero. Es una tabla maestra del sistema.

Observation:

- Obs_id.- Primary key. Tipo integer, longitud 4. Es el código que se utiliza para las observaciones dentro del sistema.
- Obs_cont.- tipo varchar, longitud 250. Es el contenido que se encuentra dentro de una observación.
- Observation.-La tabla observation tiene todos los datos referentes a las observaciones que se tienen dentro de un proceso aduanero.
- Seq_num.- tipo varchar, longitud 4. Es el número secuencial que se utiliza para diferenciar las observaciones del sistema.

Observation_Type:

- Obs_code.- Primary key. Tipo char, longitud 1. Es el código que lleva un tipo de observación. Con este código se hace referencia al tipo de observación que se relaciona con un proceso aduanero.
- Obs_name.- Tipo varchar, longitud 50. Es el nombre que lleva un tipo de observación en el sistema.
- Observation_type.-La tabla observation_type es la tabla que contiene todos los tipos de observaciones que existen para realizar un proceso aduanero. Es una tabla maestra del sistema.

Operation_Type:

- Oper_cod.- Primary key. Tipo integer. Es el código con el cual se relaciona un tipo de operación dentro del sistema con un proceso.
- Oper_name.- Tipo varchar, longitud 30. Es el nombre que lleva un tipo de operación.
- Operation_type.-En la tabla Operation_type se llevan los datos de los tipos de operaciones que se encuentran dentro de los procesos aduaneros. Es una tabla maestra del sistema.

Payment:

- Pay_id.- Primary key. Tipo varchar, longitud 3. Es el código con el cual se identifica una forma de pago dentro del sistema.
- Pay_cod.- Tipo varchar, longitud 3. Es el código con el cual se identifica una forma de pago en un proceso aduanero.
- Pay_name.- Tipo varchar, longitud 50. Es el nombre que lleva una forma de pago dentro del sistema.
- Pay_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra una forma de pago dentro del sistema. Puede ser activo o inactivo, cabe recalcar que en el sistema solo se pueden usar formas de pagos que se encuentran en estado activo.
- Payment.-En la tabla payment se llevan los datos de todas las formas de pago que se encuentran en un proceso aduanero.

Physical_units:

- Phy_cod.- Tipo varchar, longitud 4. Es el código que lleva una unidad física dentro de los procesos aduaneros.
- Phy_eng_name.- Tipo varchar, longitud 20. Es el nombre de la unidad física traducida al inglés.
- Phy_id.- Primary key. Tipo integer. Es el código que lleva una unidad física dentro del sistema.
- Phy_nam.- tipo varchar, longitud 20. Es el nombre que lleva una unidad física dentro del sistema.
- Phy_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra una unidad física. Puede ser activo o inactivo.
- Physical Units.-La tabla Physical_Units es en donde se encuentran todos los datos de las unidades físicas que se utilizan para un proceso aduanero.

Port:

- Port.-En la tabla Port se encuentran todos los datos referentes a todos los puertos que se encuentran a nivel mundial y en los que se realizan trámites aduaneros.
- Port_cod.- Primary key. Tipo varchar, longitud 6. Es el código asignado a cada puerto a nivel mundial. Es el campo con el cual se puede hacer referencia a un puerto y cualquiera de sus datos.

- Port_id.- Tipo varchar, longitud 6. Es la numeración que se le asigna a cada puerto dentro del sistema. No son números repetidos.
- Port_nam.- Tipo varchar, longitud 30. Es el nombre que lleva cada puerto.
- Port_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra un Puerto. Puede ser activo o inactivo.

Preceding_Process_Detail:

- Expir_date.- Tipo datetime. Es la fecha final o de caducidad que existe en esta relación.
- Prec_pro_id.- Tipo integer, longitud 4. Es el ID del proceso predecesor de otro proceso. Soporta valores null dado que un proceso puede ser que no tenga predecesor.
- Pret_pro_det_id.- Primary key, tipo numeric. Es el código que lleva el detalle de la relación Proceso- Proceso.
- Preceding_Process_detail.-La tabla preceding_process_detail es la tabla en la cual se encuentra todos los datos que son tratados como detalles de la relación Proceso-Proceso.
- Start_date.- Tipo datetime. Es la fecha de inicio que hay en esta relación.

Process:

- Ent_date.- Tipo datetime, longitud 8. Es la fecha en la que se comienza a realizar el proceso aduanero.
- Pro_id.- Primary key. Tipo integer. Es el código del proceso aduanero. Es un valor entero que se va incrementando en valor de uno de manera secuencial y automática.
- Pro_st.- Tipo char, longitud 1. Es el estado en el que se encuentra el proceso aduanero. Puede ser activo o inactivo.

- Process.-La tabla Process es la tabla en la cual se encuentran todos los datos principales de los procesos aduaneros. Es la tabla principal del sistema.
- Process.Des_details.- Tipo varchar, longitud 50. Es el detalle del proceso.
- Dest_port.- Tipo varchar, longitud 6. Es el código del puerto que será el destino de la mercadería.
- Shipm_port.- Tipo varchar, longitud 6. Es el código del puerto de embarque de la mercadería la cual se encuentra dentro del proceso aduanero.

Process_Line_Agency_details:

- Arrival_date.- Tipo datetime, longitud 8. Es la fecha en la que se dice que llegará la mercadería.
- Process_Line_Agency_Detail.-En la tabla Process_Line_Agency_Detail se encuentran los datos que resultan detalles de la relación en Proceso, la línea de transporte y la agencia de transporte.
- Process_Line_Agency_Detail.Desc_details.- Tipo varchar, longitud 50. Es la descripción de los detalles de la relación por la cual esta tabla es creada. Soporta valores null.
- Process_Line_Agency_Detail.Det_id.- Primary key. Tipo integer, longitud 4. Es el código que se le asigna a un detalle de esta

relación de manera automática, incrementándose cada vez que se ingresa un nuevo detalle en el valor de uno (1).

- `Process_Line_Agency_Detail.Doc_num.`- Tipo `numeric`. Es el número del documento que viene acompañando a la Mercadería. Puede ser BL's, Guía aérea, etc.
- `Shipment_date.`- Tipo `datetime`, longitud 8. Es la fecha en la que se ha realizado el embarque de la mercadería al transporte que la llevara a su destino.
- `Manif_num.`- Tipo `integer`, longitud 4. Es el número de manifiesto con el cual se relaciona un proceso.
- `Exit_trans_way.`- Tipo `tinyint`, longitud 1. Vía de transporte de Salida.

Processing_Type_Code:

- `Processing_Type_Code.`-La tabla `Processing_Type_Code` es la tabla en la que se llevan los códigos de tipos de procesos que se encuentran dentro de cualquier despacho aduanero. Es una tabla maestra del sistema.
- `Processing_Type_Code.Type_code.`- Tipo `varchar`, longitud 2. Es el código del tipo de proceso. Este código sirve para llamar cualquier tipo de proceso dentro de un proceso aduanero.

- Processing_Type_Code.Type_nam.- Tipo varchar, longitud 50. Es el nombre que lleva un tipo de proceso relacionado con un código único.

Product:

- Naladisa.- Tipo char, longitud 1. Es la que indica si el producto se encuentra dentro del convenio naladisa.
- Nandina.- Tipo char, longitud 1. Es la que indica si el producto se encuentra dentro del convenio nandina, que es el Convenio de Países andinos.
- Prod_des.- Tipo varchar, longitud 30. Es la descripción que tiene el producto.
- Prod_id.- Primary Key. Tipo char, longitud 10. Es el código que se le asignará a un producto a medida que se ingrese al sistema.
- Prod_price.- Tipo money, longitud 8. Es el precio que lleva el producto.
- Product.-La tabla Product es la tabla en la cual se ingresarán todos los datos de los productos que pasarán por un proceso aduanero.
- Trademark.- Tipo varchar, longitud 10. Es la marca del producto. En el caso de que se frutas no lleva marca.

Product_invoice_Detail:

- FOB.- Tipo Money. Es el valor que le toca pagar al cliente por los productos que importa o exporta.
- Product_Invoice_Detail.-Es la tabla en la cual se encuentra detallada la factura con los productos para los cuales se quiere agilizar los trámites. Es el resultado de la relación entre producto y factura. Esta es una relación de muchos a muchos.
- Product_Invoice_Detail.Det_id.- Tipo integer, longitud 4. Es el ID que tienen los datos de los detalles de la relación producto factura. Es un campo incremental en uno, el cual se va ir incrementando cada vez que se ingresen datos.
- Serial_num.- Tipo integer, longitud 4. Es el número de serie que lleva cada producto.

Prov_cond:

- Cond_id.- Tipo char, longitud 1. Es el código que tiene una condición de proveedor dentro del sistema.
- Cond_name.- Tipo varchar, longitud 50. Es el nombre que lleva una condición de proveedor dentro del sistema.
- Prov_cond.-Es la tabla en la cual se encuentran las condiciones que puede tener un proveedor para poder realizar una transacción aduanera. Es una tabla maestra del sistema.

Provider:

- Prov_addr.- Tipo varchar, longitud 50. Es la dirección del proveedor.
- Prov_fax.- Tipo varchar, longitud 10. Es el fax del proveedor.
- Prov_mail.- Tipo varchar, longitud 50. Es el mail del proveedor por medio del cual se puede intentar tener contacto con el proveedor.
- Prov_name.- Tipo varchar, longitud 20. Es la razón social del proveedor.
- Prov_ruc.- Primary key. Tipo char, longitud 13. Es el número de ruc que tiene cada proveedor para poder realizar ventas.
- Prov_phone.- Tipo varchar, longitud 20. Es el teléfono del proveedor.
- Prov_phone2.- Tipo varchar, longitud 20. Es el segundo teléfono del proveedor.
- Prov_st.- Tipo char, longitud 1. Es el estado en el que se encuentra el proveedor. Este puede ser activo o inactivo.
- Provider.- Es la tabla en la cual se encuentran datos de los proveedores comunes dentro del sistema. Estos cuando se encuentran por primera vez en un proceso aduanero se ingresan sus datos.

Regime:

- Reg_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra un régimen en un momento determinado. Puede ser activo o inactivo.
- Regime.-En la tabla regime se encuentran todos los regimenes que se necesitan o se encuentran dentro de los procesos aduaneros. Es una tabla maestra del sistema.
- Reg_cod.- Tipo varchar, longitud 2. Es el código del régimen que se relaciona con un nombre y un código de régimen. Este código es el que se usa para hacer referencia a un régimen dentro de un proceso aduanero.
- Reg_id.- Primary key. Tipo integer, longitud 4. Es el id del régimen que se encuentra dentro del sistema. Se incrementa automáticamente.
- Reg_nom.- Tipo varchar, longitud 200. Es el nombre que lleva el régimen.

Sector:

- Sec_id.- Primary key. Tipo varchar, longitud 3. Es el id del sector de la sociedad al cual el cliente pertenece.
- Sec_nom.- Tipo varchar, longitud 100. Es el nombre del sector de la sociedad al cual el cliente pertenece. Puede ser: SECTOR PÚBLICO, Gobierno Central, Gobierno Local (Municipios y

Consejos Provinciales), Seguridad Social, Empresa Pública, Institución Financiera, Otros Organismos, SECTOR PRIVADO, Empresa no Financiera, Institución Financiera, Compañía de Seguro, Institución sin fin de lucro, Productor Individual, Persona Natural.

- Sec_st.- Tipo char, longitud 1. Es el estado del sector el cual puede ser activo o inactivo.
- Sector.-En la tabla Sector se encuentran los nombres de los sectores de la sociedad que pueden estar involucrados en un proceso aduanero.

Selection_code:

- Sel_code.- Primary key. Tipo char, longitud 1. Es el ID del código de selección, con el cual se puede hacer referencia en cualquier proceso aduanero. Puede ser: R, N, V.
- Sel_name.- Tipo varchar, longitud 50. Es el nombre que lleva un código de selección dentro de un proceso aduanero. El código de selección puede ser: CANAL ROJO, CANAL NARANJA, CANAL VERDE. Cada uno de estos nombres se relaciona con un ID.
- Selection_Code.-En la tabla Selection_Code se encuentran los códigos de selección con los cuales se dice en que tipo de canal se encuentra el proceso.

Sending_Type:

- Send_code.- Primary key. Tipo char, longitud 2. Es el ID que tiene un tipo de envío dentro del sistema. Puede tener los valores de 09 ó 53.
- Send_name.- Tipo varchar, longitud 14. Es el nombre que tiene el tipo de envío dentro del sistema. Puede ser: envío Original, Prueba. Estos nombres se relacionan con los ID del tipo de envío.
- Sending_Type.-En la tabla sending_type se encuentra el tipo de envío que puede tener la documentación de manera electrónica, que se ha pedido para realizar un proceso aduanero. Es una tabla maestra del sistema.

State:

- St_id.- Primary Key. Tipo integer, longitud 4. Es el ID que tiene un estado del producto dentro del proceso.
- St_name.- Tipo varchar, longitud 10. Es el nombre del estado en el cual se encuentra un producto.
- St_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra un "estado" de un producto. Puede ser activo o inactivo.
- State.-En la tabla state se encuentran los datos de los estados físicos en los cuales se puede encontrar el producto en el

momento en que se vaya a realizar un proceso aduanero. Es una tabla maestra del sistema.

Tariff_Game:

- Ad_val.- Tipo nvarchar, longitud 4. Es el valor de ad valorem que acompaña a una tarifa arancelaria. Es la cantidad de dinero que se tendrá que aumentar según la partida arancelaria que se aplique al proceso aduanero.
- Tar_des.- Tipo varchar, longitud 256. Es la descripción de una tarifa arancelaria, la cual se relaciona con un tar_id.
- Tar_id.- Primary key. Tipo decimal. Es el código con el cual se identifica a una partida arancelaria.
- Tar_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra la partida arancelaria. Puede ser activo o desactivo.
- Tariff_game.-Es la tabla en la cual se encuentran las partidas arancelarias necesarias para realizar cualquier proceso aduanero. Es un maestro del sistema.

Transport_Agency:

- Transport_Agency.-Es la tabla en la cual se almacena los datos de la agencia de transporte que se utilizan para traer y llevar mercadería dentro de un proceso aduanero. Es una tabla maestra del sistema.

- Transport_Agency.Tran_des.- Tipo varchar, longitud 50. Es la descripción que tiene una agencia de viaje. Soporta valores null.
- Transport_Agency.Tran_id.- Primary key. Tipo varchar, longitud 4. Es el código con el cual se identifica una agencia de transporte dentro del proceso aduanero.
- Transport_Agency.Tran_nam.- Tipo varchar, longitud 30. Es el nombre de las agencias de transportes que se relaciona con un código identificador llamado Tran_id.
- Transport_Agency.Tran_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra la agencia de transporte. Puede ser activo o Inactivo. Soporta valores null.

Transport_Line:

- Tran_In_id.- Primary key. Tipo integer, longitud 4. Es el código identificador de la línea de transporte con el cual se identifica que línea de transporte se relaciona con un proceso aduanero. Es un número que se aumenta secuencial mente en el valor de 1.
- Tran_In_st.- Tipo char, longitud 1. Es el estado en el cual se encuentran las líneas de transporte dentro del sistema o dentro de los procesos aduaneros. Puede ser activo o inactivo.

- Tran_name.- Tipo varchar, longitud 30. Es el nombre que lleva una línea de transporte dentro del sistema. Se encuentra relacionado con un código identificador de Tran_In_id.
- Transport_Line.-En esta tabla se encuentra los datos de la línea de transporte que se utiliza dentro de un proceso aduanero. Es una tabla maestra del sistema.

Transport_Vehicle:

- Tran_veh_id.- Primary Key. Tipo integer, longitud 4. Es el código identificador de un vehículo dentro del sistema y dentro de un proceso aduanero. Es un número secuencial de manera incremental en el valor de 1.
- Tran_veh_name.- Tipo varchar, longitud 10. Es el nombre del vehículo que va a transportar la mercadería dentro de un proceso de despacho aduanero.
- Tran_veh_st.- Tipo char, longitud 1. Es el estado en el que se encuentra el vehículo dentro de los procesos aduaneros. Puede ser activo o inactivo.
- Transport_Vehicle.-En esta tabla se encuentran los datos de los vehículos que existen para poder hacer algún traslado dentro de un proceso aduanero. Esta tabla es creada por la relación entre transport_line y transport_way que es una relación de muchos a muchos, es decir que, una línea de transporte puede tener uno

o muchas vías de transporte y una vía de transporte puede tener una o muchas líneas de transporte.

Transport_Way:

- Tran_way_id.- Tipo varchar, longitud 3. Es el código identificador de la vía de transporte dentro del proceso aduanero. Es un único código que no se puede repetir.
- Tran_way_nam.- Tipo varchar, longitud 20. Es el nombre que tiene una vía de transporte. Este puede ser: Aéreo, Marítimo, Terrestre.
- Transport_way.-En la tabla Transport_way están los datos de las vías de transporte que existen dentro del sistema y dentro de los procesos aduaneros. Es un maestro del sistema.

Treatment:

- Treat_type.- Primary key. Tipo char, longitud 1. Es el código identificador del tipo de tratamiento que existe dentro de los procesos aduaneros.
- Treat_name.- Tipo varchar, longitud 11. Es el nombre que tiene cada tipo de tratamiento. Puede ser Normal, Diplomático o Donación.

- Treatment.-En la tabla treatment se guardan los datos de los tipos de tratamiento que se le puede dar al proceso aduanero. Es un maestro del sistema.

Veh_body:

- Veh_body.-En esta tabla se encuentran los datos específicos del cuerpo de los vehículos que se pueden importar. Es un maestro del sistema.
- Veh_body_cod.- Primary Key. Tipo varchar, longitud 5. Es el código identificador de cada cuerpo que puede tener un vehículo.
- Veh_body_name.- Tipo varchar, longitud 10. Es el nombre que tiene un cuerpo de vehículo. Esta relacionada con un código identificador Veh_body_cod.
- Veh_body_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra un cuerpo de vehículo. Puede ser activo o inactivo.

Veh_Class:

- Veh_Class.-En esta tabla se encuentran los datos de las clases de vehículos que están permitidos para importar. Es un maestro del sistema.

- Veh_cls_cod.- Primary key. Tipo varchar, longitud 5. Es el código identificador de las clases de vehículos que son permitidos dentro del proceso aduanero.
- Veh_cls_name.- Tipo varchar, longitud 10. Es el nombre que tiene una clase de vehículo. Esta relacionada con un código identificador Veh_cls_cod.
- Veh_cls_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra una clase de vehículo. Puede ser activo o inactivo.

Veh_Color:

- Veh_Color.-En esta tabla se encuentran los datos de los colores de los vehículos que están permitidos para ser importados. Es un maestro del sistema.
- Veh_color_cod.- Primary key. Tipo varchar, longitud 5. Es el código identificador de los colores de los vehículos que son permitidos dentro del proceso aduanero.
- Veh_color_name.- Tipo varchar, longitud 10. Es el nombre que tiene un color de un vehículo. Esta relacionado con un código identificador Veh_color_cod.
- Veh_color_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra un color de un vehículo. Puede ser activo o inactivo.

Veh_Fuel:

- Veh_Fuel.-En esta tabla se encuentran los datos de los tipos de gasolinas de los vehículos que están permitidos para ser importados. Es un maestro del sistema.
- Veh_fuel_cod.- Primary key. Tipo varchar, longitud 5. Es el código identificador de los tipos de gasolinas de los vehículos que son permitidos dentro del proceso aduanero.
- Veh_fuel_name.- Tipo varchar, longitud 10. Es el nombre de la gasolina que acepta un vehículo. Esta relacionado con un código identificador Veh_fuel_cod.
- Veh_fuel_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra un tipo de gasolina que acepta un vehículo. Puede ser activo o inactivo.

Veh_Traction:

- Veh_Traction.-En esta tabla se encuentran los datos de los tipos de tracción de los vehículos que están permitidos para ser importados. Es un maestro del sistema.
- Veh_traction_cod.- Primary key. Tipo varchar, longitud 5. Es el código identificador de los tipos de tracción de los vehículos que son permitidos dentro del proceso aduanero.

- Veh_traction_name.- Tipo varchar, longitud 10. Es el nombre de la tracción que existe en un vehículo. Esta relacionado con un código identificador Veh_traction_cod.
- Veh_traction_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra un tipo de tracción que tiene un vehículo. Puede ser activo o inactivo.

Veh_Trademark:

- Veh_Trademark.-En esta tabla se encuentran los datos de las marcas de los vehículos que están permitidos para ser importados. Es un maestro del sistema.
- Veh_trd_cod.- Primary key. Tipo varchar, longitud 5. Es el código identificador de las marcas de los vehículos que son permitidos dentro del proceso aduanero.
- Veh_trd_name.- Tipo varchar, longitud 10. Es el nombre de la marca que existe en un vehículo. Esta relacionado con un código identificador Veh_trd_cod.
- Veh_trd_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra una marca que tiene un vehículo. Puede ser activo o inactivo.

Veh_Transmission:

- Veh_Transmission.-En esta tabla se encuentran los datos de los tipos de transmisiones de los vehículos que están permitidos para ser importados. Es un maestro del sistema.
- Veh_transm_code.- Primary key. Tipo varchar, longitud 5. Es el código identificador de los tipos de transmisiones de los vehículos que son permitidos dentro del proceso aduanero.
- Veh_transm_name.- Tipo varchar, longitud 10. Es el nombre del tipo de transmisión que tiene un vehículo. Esta relacionado con un código identificador Veh_transm_code.
- Veh_transm_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra un tipo de transmisión que tiene un vehículo. Puede ser activo o inactivo.

Vehicle:

- Vehicle.-Es la tabla en la cual se encuentran los datos de los vehículos que se pueden importar, con sus respectivas características. Es una tabla maestra del sistema.
- Veh_id.- Primary key. Tipo varchar, longitud 5. Es el código identificador de los tipos de transmisiones de los vehículos que son permitidos dentro del proceso aduanero.

Warehouse:

- Ware_id.-Tipo integer, longitud 4. Primary key. Es el código identificador de una bodega dentro del sistema.
- Ware_cod.- Tipo varchar, longitud 5. Es el código con el cual se identifica una bodega que se este relacionando con un proceso aduanero.
- Ware_nam.- Tipo varchar, longitud 30. Es el nombre de la bodega, el cual esta relacionado con un código de bodega para poder ser identificada y relacionada con un código aduanero.
- Ware_st.- Tipo char, longitud 1. Es el estado en el cual se encuentra una bodega dentro del sistema. Puede ser activo o inactivo.
- Warehouse.- Tabla en la cual constan los datos de las bodegas que están permitidas para tener las mercaderías que llegan o que ingresan por medio de una importación. Es una tabla maestra del sistema.

Warranty_Type:

- W_type_id.- Tipo Integer, longitud 4. Primary key de la tabla
- warranty_type. Es el código identificador de un tipo de garantía específico.

- **W_type_name.**- Tipo varchar, longitud 20. Es el nombre con el cual se identifica un tipo de garantía y se encuentra relacionado con un **w_type_id**.
- **Warranty_type.**- En esta tabla se encuentran los datos de los tipos de garantías que existen y que se pueden aplicar a un proceso aduanero. Es un maestro del sistema.

Work_Order:

- **Com_doc.**- Tipo bit, longitud 1. Este campo indica si la documentación necesaria para hacer el retiro o el envío de la mercadería se encuentra completa o incompleta. Puede ser 1 o 0. El 1 es si la respuesta es afirmativa y el cero es en caso contrario, es decir cuando la mercadería es incompleta.
- **Cont_open.**- Tipo bit, longitud 1. Me indica si un contenedor va a ser abierto o no. Este campo puede ser 1 o 0. El 1 es si la respuesta es afirmativa y el cero en caso contrario.
- **Gui_num.**- Tipo varchar, longitud 10. Es el campo en el cual se ingresa o representa el número de guía.
- **Office_pay.**- Tipo bit, longitud 1. Este campo indica si todo el trámite se ha pagado todo el trámite aduanero en la oficina . Puede ser 1 o 0. El 1 es si la respuesta es afirmativa y el cero es en caso contrario, es decir cuando el pago no ha sido en la oficina o si todavía no se ha realizado.

- Ref.- Tipo varchar, longitud 50. Es el número de referencia dentro de la orden de trabajo. Soporta valores NULL.
- Reg_num.- Tipo varchar, longitud 50. Es el numero de registro el cual se encuentra en la orden de trabajo.
- Retiro.- Tipo bit, longitud 1. Este campo indica si se puede hacer el retiro del sello que tenga un contenedor o no.
- Puede ser 1 o 0. El 1 es si la respuesta es afirmativa y el cero es en caso contrario.
- Tim_num.- Tipo numeric, longitud 9. Es el numero de timbrada que se registra en la orden de trabajo.
- Wk_year.- Tipo char, longitud 4. Es el campo en el cual se almacena el año en el cual se esta realizando un proceso aduanero especifico. Este valor es muy importante para muchas de las operaciones dentro de un proceso aduanero.
- Wor_id.- Tipo varchar, longitud 10. Primary key de la tabla work_order. Es el codigo identificador de una orden de trabajo el cual consta de 10 caracteres, que se componen de la siguiente manera: 5 numeros mas 1 letras mas los dos ultimos dígitos del año
- Wor_num.- Tipo varchar, longitud 6. Es la parte numérica del código identificador de una orden de trabajo.

- Work_Order.- Es la tabla en la cual se almacenan los datos pertenecientes o relacionados con una orden de trabajo y que es necesario para realizar un proceso aduanero.
- Work_Order.Aforo.- Tipo bit, longitud 1. Me indica si la mercadería que se encuentra dentro del proceso va a ser aforada o no. Este campo puede ser 1 o 0. El 1 es si la respuesta es afirmativa y el cero en caso contrario.

RELACIONES

Las relaciones establecidas entre las entidades del sistema son:

- Relación Agent-Agent_Client_Detail.- relación con diferenciación 1:M, en la cual un agente se puede encontrar en uno o varios detalles relacionado con distintos clientes.
- Relación Client-Agent_Client_Detail.- relación con diferenciación 1:M, en la cual un cliente se puede encontrar en uno o varios detalles relacionado con distintos agentes.
- Relación Bank_Agency_Process_detail-Bank.- relación con diferenciación 1:M, en la cual un banco puede tener una o varias agencias dentro de una misma ciudad. Un banco puede estar solo en una ciudad o en varias ciudades. El campo Ban_id se encuentra en la tabla como foreign key.
- Relación Bank_Agency_Process_detail-Port.- relación con diferenciación 1:M, en la cual un banco puede tener una o varias

agencias dentro de una misma ciudad y un banco puede estar en una o varias diferentes con diferente cantidad de oficinas y sucursales. Es el código del puerto en el cual se encuentra la agencia bancaria.

- Relación Bank_Agency_Process_detail-Process.- relación con diferenciación 1:M, en la cual un proceso puede estar relacionado con uno solo detalle de la relación Banco-Ciudad, y los detalles de la relación Banco-Ciudad pueden estar relacionados con uno o varios procesos. El campo Pro_id se encuentra en la tabla como foreign key.
- Relación Ben_Prov_detail-Beneficiary.- relación con diferenciación 1:M, en la cual un beneficiario se puede encontrar en uno o varios detalles relacionados con uno o varios proveedores. El campo Ben_id se encuentra como foreign key de la tabla Beneficiary.
- Relación Ben_Prov_detail-Provider.- relación con diferenciación 1:M, en la cual un proveedor se puede encontrar en uno o varios detalles relacionado con distintos beneficiarios. El campo Prov_ruc se encuentra en la tabla como foreign key.
- Relación Client-CIIU.- relación con diferenciación 1:M, en la cual un cliente puede tener un solo CIIU, pero un CIIU puede tener uno o varios clientes a la vez. El campo Client_CIIU se encuentra en la tabla como foreign key.

- Relación Client-Sector.- relación con diferenciación 1:M, en la cual un cliente puede pertenecer a un solo sector, pero un sector puede tener uno o varios clientes a la vez. El campo Client_Sector se encuentra en la tabla como foreign key.
- Relación Client-Port.- relación con diferenciación 1:M, en la cual un cliente puede pertenecer a una sola ciudad, pero una ciudad puede tener uno o varios clientes a la vez. El campo Client_City se encuentra en la tabla como foreign key.
- Relación Country_Agree_detail-Agree: relación con diferenciación 1:M, en la cual un convenio puede estar relacionado con uno o varios detalles de la relación Country_Agree_detail, pero un detalle de la relación Country_Agree_detail puede estar relacionado con un solo convenio. El campo Agree_id se encuentra en la tabla como foreign key.
- Relación Country_Agree_detail-Country: relación con diferenciación 1:M, en la cual un país puede estar relacionado con uno o varios detalles de la relación Country_Agree_detail, pero un detalle de la relación Country_Agree_detail puede estar relacionado con un solo país. El campo Country_cod se encuentra en la tabla como foreign key.
- Relación DAU-Aforo.- relación con diferenciación 1:M, en la cual un aforo puede estar en uno o muchos DAU's pero un DAU

puede tener solo un aforo. Af_id se encuentra en la tabla como foreign key.

- Relación DAU-Process.- relación con diferenciación 1:M, en la cual un proceso puede tener uno o varios DAU's pero un DAU puede pertenecer o estar en un solo proceso. El campo Pro_id se encuentra en la tabla como foreign key.
- Relación DAU-Sending_type.- relación con diferenciación 1:M, en la cual un tipo de envío puede estar en uno o varios DAU's, y un DAU puede tener solo un tipo de envío a la vez. Send_type se encuentra en la tabla como foreign key. Puede ser 09 o 53.
- Relación Doc_Process_Detail –Document.- relación con diferenciación 1:M, en la cual un documento esta relacionado con uno o varios procesos definidos. El campo de la tabla llamado Doc_id es un foreign key.
- Relación Doc_Process_Detail –Process.- relación con diferenciación 1:M, en la cual un proceso esta relacionado con uno o varios detalles de la relación Doc_Process_Detail-Process, esto quiere decir que un proceso puede tener uno o varios documentos y un detalle de la relación Doc_Process_Detail se relaciona solo con un proceso definido. El campo de la tabla llamado Pro_id es un foreign key.
- Relación Doc_type-Client.- relación con diferenciación de M:M, en la cual un cliente puede tener uno o varios tipos de

documentos personales, y un tipo de documento puede ser tenido por uno o varios clients. Esta tabla es creada por la relación M:M que existe la cual se la divide en relaciones con diferenciación de 1:M las cuales son Client-Doc_type_Detail y Doc_type-Doc_type_Detail. Los campos Doc_id y Client_id son los foreign keys en esta tabla.

- Relación Document-Aso_Document_Type.-relación con una diferenciación de 1:M, en la cual un documento puede tener un solo tipo de documentos asociados, y un tipo de documento asociado puede tener uno o varios documentos. El campo de la tabla llamado Aso_type es foreign key.
- Relación Document-Operation_type.- relación con una diferenciación de 1:M, en la cual un documento puede tener un solo tipo de operación, y un tipo de operación puede tener uno o varios documentos. El campo de la tabla llamado Oper_type es foreign key.
- Relación Free_code-Documents.- relación con diferenciación de M:M, en la cual un código liberatorio puede relacionarse con uno o muchos documentos, y un documento puede relacionarse con uno o muchos códigos liberatorios. Esta relación con diferenciación de M:M se divide en dos relaciones con diferenciación de 1:M, las cuales son relación Free_code-Doc_Doc_Detail con un campo llamado Free_cod_id el

cual es foreign key y la segunda relación Document-Free_Code_Detail con el campo Doc_id como foreign key.

- Relación Free_code-Tariff_game.- relación con diferenciación de 1:M, en la cual un código liberatorio tiene solo una tarifa arancelaria, pero una tarifa arancelaria puede tener uno o varios códigos liberatorios. El campo de la tabla llamado Tar_id es el foreign key.
- Relación Invoice-Deposit.-relación con diferenciación de 1:M, en la cual la mercadería de una factura puede estar en un solo deposito y en un deposito puede existir mercadería de una o muchas facturas. El campo llamado Dep_id es el foreign key en esta tabla.
- Relación Invoice-Warehouse.- relación con diferenciación de 1:M, en la cual la mercadería de una factura puede estar en una única bodega y en una bodega puede existir mercadería de una o muchas facturas. El campo llamado Ware_id es el foreign key en esta tabla.
- Relación Observation_type-DAU.- relación con diferenciación de M:M, en la cual un tipo de observación puede estar en uno o varios DAU's, y un DAU puede tener uno o varios tipos de observaciones. Por esta relación con diferenciación de M:M, se forman dos relaciones con diferenciación de 1:M, la relación Observation-DAU que pasa como foreign key al campo DAU_id y

la relación Observation-Observation_type la cual pasa como foreign key al campo Obs_type.

- Relación Port- Country.- relación con diferenciación de 1:M, en la cual un puerto puede pertenecer solo a un país pero un país puede tener 1 o varios puertos. El campo llamado Country_cod es el foreign key de esta relación.
- Relación Preceding_Process_detail-Process.- relación con diferenciación M:M, en la cual un proceso puede tener uno o muchos procesos predecesores y un proceso predecesor puede estar relacionado con uno o varios procesos actuales.
- Relación Process-Cargo.-relación con diferenciación de 1:M, en la cual un proceso puede tener un solo cargo, pero un cargo puede estar en uno o más procesos. En la tabla existe un campo llamado Cargo_id el cual es el foreign key.
- Relación Process-Client.-relación con diferenciación de 1:M, en la cual un proceso exclusivo puede tener un solo cliente, pero un cliente puede estar en uno o más procesos a la vez. En la tabla existe un campo llamado Client_id el cual es el foreign key.
- Relación Process-Commerce_Operator.-relación con diferenciación de 1:M, en la cual un proceso puede tener un solo operador de comercio, pero un operador de comercio puede estar en uno o más procesos a la vez. En la tabla existe un campo llamado Com_cod el cual es el foreign key.

- Relación Process-Destination_type.- relación con diferenciación de 1:M, en la cual un proceso puede tener un solo tipo de destino, pero un tipo de destino puede estar en uno o más procesos a la vez. En la tabla existe un campo llamado Dest_type el cual es el foreign key.
- Relación Process-Duty.- relación con diferenciación de 1:M, en la cual un proceso puede tener una sola aduana local, pero una aduana local puede estar en uno o más procesos a la vez. En la tabla existe un campo llamado Local_duty_id el cual es el foreign key, con el cual se representa la aduana a la cual llegara la mercadería de cierto proceso aduanero.
- Relación Process-Invoice.- relación con diferenciación de 1:M, en la cual un proceso puede tener una o muchas facturas, pero una factura puede estar en un único proceso. En la tabla existe un campo llamado Inv_num el cual es el foreign key.
- Relación Process-Manifest.-relación con diferenciación de 1:M, en la cual un proceso puede tener un solo tipo de manifiesto, pero un tipo de manifiesto puede estar en uno o más procesos. El campo en la tabla llamado Manif_type es el foreign key.
- Relación Process-Money.-relación con diferenciación de 1:M, en la cual un proceso puede tener solo un tipo de moneda para realizar los pagos debidos, pero un tipo de moneda puede estar

en uno o varios procesos. El campo llamado Mon_id es el foreign key.

- Relación Process-Payment.- relación con diferenciación de 1:M, en la cual un proceso puede tener solo un tipo o forma de pago para realizar los pagos, pero un tipo o forma de pago puede estar en uno o varios procesos. El campo llamado Pay_id es el foreign key.
- Relación Process-Port.-relación con diferenciación de 1:M, en la cual un proceso puede tener solo un puerto o aduana de salida, pero un puerto o aduana de salida puede estar en uno o varios procesos. El campo llamado Foreign_duty_id es el foreign key.
- Relación Process-Precedent_Process.- relación con diferenciación de 1:M, en la cual un proceso puede tener solo un proceso precedente, pero un proceso precedente puede estar en uno o varios procesos. El campo llamado Preced_pro_id es el foreign key.
- Relación Process-Provider.-relación con diferenciación de 1:M, en la cual un proceso puede tener solo un proveedor a la vez, pero un proveedor puede estar en uno o varios procesos. El campo llamado Prov_ruc es el foreign key.
- Relación Process-Regime.- relación con diferenciación de 1:M, en la cual un proceso puede tener solo un régimen que lo rija,

pero un régimen puede estar en uno o varios procesos. El campo llamado Reg_id es el foreign key.

- Relación Process-Selection_Code.- relación con diferenciación de 1:M, en la cual un proceso puede tener solo un código de selección, pero un código de selección puede estar en uno o varios procesos. El campo llamado Sel_code es el foreign key.
- Relación Process-Processing_Type_Code.- relación con diferenciación de 1:M, en la cual un proceso puede tener solo un código de tipo de proceso, pero un código de tipo de proceso puede estar en uno o varios procesos. El campo llamado Type_code es el foreign key.
- Relación Process-Transport_Line-Transport_Agency.-Es una relación ternaria la cual dice que un proceso puede tener una o varias líneas de transporte y una línea de transporte puede tener una o varias agencias de transporte. Tiene como foreign keys a Pro_id, Tran_ag_id.
- Relación Process-Transport_Line-Transport_Agency-Vehicle.- relación la cual indica el vehículo que puede pertenecer a una o varias líneas de transporte y a una o varias agencias de transporte que traslada la mercadería que se encuentra dentro de un proceso aduanero. Tiene como foreign key a Tran_veh_id.
- Relación Process-Treatment.- relación con diferenciación de 1:M, en la cual un proceso puede tener un solo tipo de

tratamiento, pero un tipo de tratamiento puede estar en uno o más procesos. El campo en la tabla llamado Treat_type es el foreign key.

- Relación Product-Class.-relación con diferenciación de 1:M, en la cual un producto puede tener una sola clase, pero una clase puede tener uno o varios productos. Tiene como foreign key al campo Cls_cod.
- Relación Product-Country.- relación con diferenciación de 1:M, en la cual un producto puede tener un solo país de origen, pero un país de origen puede tener uno o varios productos. Tiene como foreign key al campo Origin_country.
- Relación Product-Country.-relación con diferenciación de 1:M, en la cual un producto puede tener un país de adquisición y un país de adquisición puede tener uno o varios productos. El campo llamado Adq_country_id es el foreign key.
- Relación Product-Deposit.- relación con diferenciación de 1:M, en la cual un producto puede tener un solo lugar de depósito y un depósito puede tener uno o varios productos. El campo llamado Dep_id es el foreign key.
- Relación Product-Invoice.-relación con diferenciación de M:M, en la cual un producto puede estar en una o muchas facturas, y en una factura puede haber uno o muchos productos. En esta tabla los campos que son foreign keys son Prod_id e Inv_num.

- Relación Product-Physic_Units.-relación con diferenciación de 1:M, en la cual un producto puede tener una sola unidad física, pero una unidad física puede tener uno o varios productos. Tiene como foreign key al campo Phy_cod
- Relación Product-State.- relación con diferenciación de 1:M, en la cual un producto puede tener un solo estado y un estado puede estar uno o varios productos. El campo llamado Prod_st es el foreign key.
- Relación Product-Tariff_game.- relación con diferenciación de 1:M, en la cual un producto puede tener una sola tarifa arancelaria, pero una tarifa arancelaria puede tener uno o varios productos. Tiene como foreign key al campo Tar_id
- Relación Product-Warehouse.- relación con diferenciación de 1:M, en la cual un producto puede estar en una bodega y una bodega puede tener uno o varios productos. El campo llamado Ware_id es el foreign key.
- Relación Provider-Prov_cond .-relación con diferenciación de 1:M, en la cual un proveedor puede tener solo un tipo de condición de proveedor, y una condición de tipo de proveedor puede tener uno o varios proveedores. El campo llamado Prov_cond es el foreign key que indica cual es la condición del proveedor.

- Relación Provider-Port.- relación con diferenciación de 1:M, en la cual un proveedor viene de una sola ciudad, y una ciudad puede tener uno o varios proveedores. El campo llamado Prov_city es el foreign key que indica cual es la ciudad del proveedor.
- Relación Transport_Vehicle-Transport_Line.- relación con diferenciación de 1:M, en la cual un vehículo pertenece a una línea de transporte, pero una línea de transporte tiene uno o más vehículos. El campo llamado Tran_In_id es el foreign key que indica a línea pertenece el vehículo.
- Relación Transport_Vehicle-Transport_Way.-relación con diferenciación de 1:M, en la cual un vehículo pertenece a una vía de transporte, pero una vía de transporte tiene uno o más vehículos. El campo llamado Tran_way_id es el foreign key que indica a vía pertenece el vehículo.
- Relación Transport_Vehicle-Country.- relación con diferenciación de 1:M, en la cual un vehículo pertenece a un país específico, pero un país puede tener uno o varios vehículos. El campo llamado Country_cod es el foreign key que indica cual es la bandera que tiene el vehículo.
- Relación Work_Order-Process.- Relación con diferenciación de 1:M, en la cual una orden de trabajo puede tener uno o varios procesos, pero un proceso solo puede tener una sola orden de

trabajo. En esta relacion existe como foreign key el campo Pro_id.

VISTAS

- ax2_Freight_i (@weight_i [numeric], @weight_t [numeric], @freight_t [float]).-Se encarga de calcular el valor del flete prorrateado a una partida determinada, cuyo peso individual se envía como parámetro. También se envía el peso total con el fin de sacar el porcentaje del flete total ponderado al peso. El flete total también es un argumento recibido.
- ax2_getAgentCod (@id varchar(6)).-Recibe como parámetro un varchar de 6 caracteres como máximo que representa el campo Id interno de la tabla Agent y devuelve el campo código que le corresponde en la misma tabla (ag_cod), a fin de que sea usado como identificador del usuario dentro del sistema.
- ax2_getAgentId(@code varchar(5)).-Recibe como parámetro un varchar de 5 caracteres máximo que representa el campo código de la tabla Agent, y devuelve el Id real correspondiente dentro la tabla (ag_id). Este es el Id que estará involucrado en las relaciones que tengan que ver con esta tabla en la base de datos.
- ax2_getCountryCod(@id varchar(6)).-Recibe como parámetro un varchar de 6 caracteres como máximo que representa el

campo Id interno de la tabla Country y devuelve el campo código que le corresponde en la misma tabla (country_cod), a fin de que sea usado como identificador del usuario dentro del sistema.

- ax2_getCountryId(@code varchar(3)).-Recibe como parámetro un varchar de 5 caracteres como máximo que representa el campo código de la tabla Country, y devuelve el Id real correspondiente dentro la tabla (country_id). Este es el Id que estará involucrado en las relaciones que tengan que ver con esta tabla en la base de datos.
- ax2_getDepositCod(@id int).-Recibe como parámetro un entero que representa el campo Id interno de la tabla Deposit y devuelve el campo código que le corresponde en la misma tabla (dep_cod), a fin de que sea usado como identificador del usuario dentro del sistema.
- ax2_getDepositId(@code varchar(5)).-Recibe como parámetro un varchar de 5 caracteres máximo que representa el campo código de la tabla Deposit, y devuelve el Id real correspondiente dentro la tabla (dep_id). Este es el Id que estará involucrado en las relaciones que tengan que ver con esta tabla en la base de datos.
- ax2_getMoneyCod(@id int).-Recibe como parámetro un entero que representa el campo Id interno de la tabla Money y

devuelve el campo código que le corresponde en la misma tabla (mon_cod), a fin de que sea usado como identificador del usuario dentro del sistema.

- ax2_getMoneyId(@code varchar(10)).-Recibe como parámetro un varchar de 10 caracteres como máximo que representa el campo código de la tabla Money, y devuelve el Id real correspondiente dentro la tabla (mon_id). Este es el Id que estará involucrado en las relaciones que tengan que ver con esta tabla en la base de datos.
- ax2_getPaymentCod(@id int).-Recibe como parámetro un entero que representa el campo Id interno de la tabla Payment y devuelve el campo código que le corresponde en la misma tabla (pay_cod), a fin de que sea usado como identificador del usuario dentro del sistema.
- ax2_getPaymentId(@code varchar(3)).-Recibe como parámetro un varchar de 3 caracteres como máximo que representa el campo código de la tabla Payment, y devuelve el Id real correspondiente dentro la tabla (pay_id). Este es el Id que estará involucrado en las relaciones que tengan que ver con esta tabla en la base de datos.
- ax2_Seguro_i (@freight_i [numeric],@fob_i [money]).-Calcula la cantidad de seguro que le corresponda a los parámetros que recibe. Este es un porcentaje de la suma del F.O.B. y el flete.

El porcentaje del seguro es un valor que varía dependiendo del criterio de la empresa despachadora. En nuestro caso el valor por defecto es 20% (0.2).

- NUMERO_ORDEN.- Es el stored procedure el cual va generando un número secuencial para las órdenes de trabajo, el cual si es la primera orden de trabajo va a revisar si es cero si lo es devolverá como primer número al 0001.

APÉNDICE B

CONTROLES DESARROLLADOS PARA EL SISTEMA

En el sistema se han utilizado como es lógico, los componentes comunes que vienen como parte del Framework de .NET. Estos son: Form, Button, Label, CheckBox, ComboBox, ListBox, TabControl, GroupBox, TextBox, ImageList, PictureBox, RadioButton y Panel. Sin embargo, a fin de mejorar la funcionalidad del sistema, encapsular operaciones en común de sus controles, y adaptarlos a las necesidades de la aplicación, se han heredado a algunos controles las siguientes clases:

BINDABLE CHECKBOX

Hereda de `System.Windows.Forms.CheckBox` e implementa funcionalidad adicional para que sea capaz de vincularse a una fuente de datos. En esta clase se añade el atributo privado `_chkStr` que viene a ser la representación string de la propiedad `CheckState` del checkbox. Este atributo puede tomar 3 valores: "1", si la propiedad `CheckState` toma el valor de `CheckState.Checked`; "0" si es `Unchecked`; y "-1" si es `Indeterminate`. Este atributo viene

acompañado de su par de propiedades públicas `CheckedStringValue(value As String) As String` y `Shadows CheckState() As Windows.Forms.CheckState`, las que devuelven el valor de `_chkStr` y `CheckState`, respectivamente; y fijan el valor de `CheckState` después de fijar el valor de `_chkStr`, y el valor de `_chkStr` después de cambiar el valor de `CheckState`, respectivamente. Todo esto a fin de mantener la correspondencia entre estas dos propiedades.

BindableCheckBox
<code>-_chkStr : String</code>
<code>+New()</code>
<code>+CheckedStringValue(entrada value : String) : String</code>
<code>+CheckState()</code>

Figura 5.1: BindableCheckBox

La finalidad de usar esta clase es poder usar declaraciones del tipo `Me.bindableCheckBox1.DataBindings.Add ("CheckedStringValue", dataTable1, "")` en un formulario que lo tenga entre sus componentes.

EDIT TEXTBOX

Esta clase hereda de `System.Windows.Forms.TextBox` y contiene funcionalidades adicionales para trabajar en un ambiente en donde el contenido de un campo de texto depende del valor de otro campo del mismo tipo en otro formulario; se pueda diferenciarlos

por su obligatoriedad y aplicabilidad y provea cierto control de su apariencia y cambios en su atributo Text.

Tiene la opción de vincularse a una instancia de su misma clase con la finalidad de que el valor en el campo Text de la clase vinculada siempre sea copiado en el campo Text de esta instancia, y se verifique cada cambio de esta propiedad en la instancia vinculada mediante un handler a su evento TextChanged. Tiene un atributo System.Windows.Forms.Label que será quien lo identifique en el formulario ante el usuario (por ejemplo, "Nombre del destinatario:"). Un objeto de esta clase puede definirse como mandatorio, condicional o no aplicable seleccionando la propiedad Observación, cuyo valor es una enumeración que contiene estos tres posibles valores. Un cambio de esta propiedad afecta al tipo de letra (fuente) de esta instancia y del label vinculado, de acuerdo al estándar citado en secciones previas. Finalmente, cualquier cambio en el campo Text de esta clase, siempre que la propiedad ReadOnly de la instancia sea False, hace que el formulario añada un "*" a su título, lo que a su vez hace que el formulario del trámite también añada un "*", indicando que hay cambios en el trámite y que debe ser grabado.

Pero la característica más útil es que lo expuesto anteriormente puede actuar en modo de diseño; ya que el IDE de Microsoft Visual

Studio puede precompilar las clases dependientes del proyecto (Build y Rebuild, en Visual Studio), y al momento de trabajar con ellas en modo de diseño, sus propiedades (Property) aparecen y actúan tal como fueron programadas en el panel de propiedades del control.

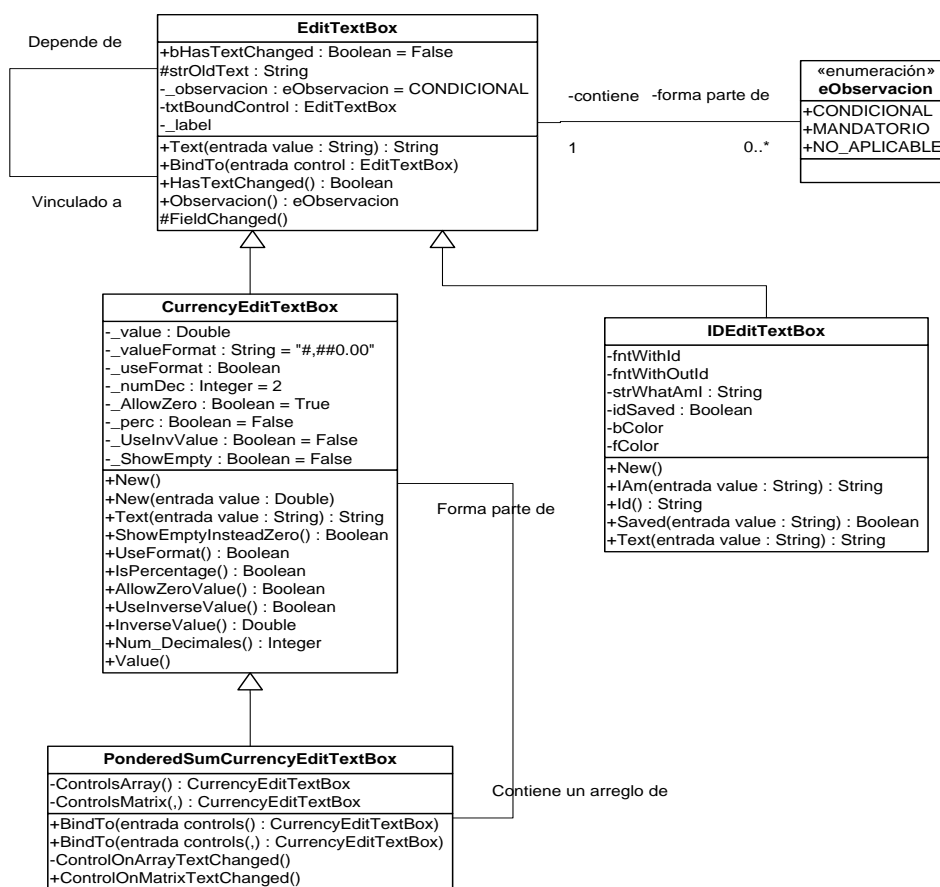


Figura 5.2: EditTextBox y sus clases derivadas

CURRENCY EDIT TEXTBOX.- Es una clase heredada de EditTextBox que está pensada para recibir solo valores numéricos y

tratarlos de ese modo. Al principio manejaba sólo valores monetarios pero su uso se lo amplió, para poder representar cualquier valor numérico. Por ello tiene las propiedades Num_Decimales, cuyo valor por defecto es 2; e IsPercentage(), el cual añade siempre un “ %” al valor ingresado si su valor es True, además de que su valor no será X sino X/100 (por ejemplo; para 20%, su valor será 0.2). Internamente tiene como atributo el valor Double del campo String Text accedido por la propiedad readOnly Value, por ello al momento de introducir un valor debe poder efectuarse un typecasting a Double antes de aceptar ese ingreso, de lo contrario un mensaje es devuelto al usuario para que lo ingrese correctamente. Tiene propiedades adicionales, tales como: ShowEmptyInsteadZero, que muestra un string vacío si el valor actual es cero; UseFormat, que aplica o no el formato basado en el número de decimales especificado, por ejemplo “20.50” en lugar de “20.5”; UseInverseValue, que define si usa X o 1/X como su valor; y AllowZeroValue, que permite o no el ingreso del valor cero, esto se da especialmente si la propiedad anterior es True. Los valores se deben ingresar mediante la propiedad Text, y leídos mediante la propiedad Value, que da el valor Double del texto ingresado. Todas estas propiedades actúan incluso en modo de diseño.

PONDERED SUM CURRENCY EDIT TEXTBOX.- Su propósito es ampliar las capacidades de la clase anterior, heredando de la misma, y vinculando ya no a un único control sino a un arreglo unidimensional o bidimensional según sean las necesidades.

Se cuenta internamente con un arreglo y una matriz de `currentEditTextBoxes`, los que se deberían usar excluyentemente. Se establece el vínculo mediante una llamada al método `Public Overloads Sub BindTo(ByVal ParamArray controls() As CurrencyEditTextBox)`, o (exclusivo) `Public Overloads Sub BindTo(ByVal controls(,) As CurrencyEditTextBox)` de esta clase. Una vez terminada la ejecución de este método, cada miembro de cualquiera de los dos arreglos que dispare el evento `TextChanged` será captado y el valor de la propiedad `Text` (y por ende `Value`) de la instancia de esta clase cambiará de la siguiente manera: Si se está trabajando con el arreglo, el valor será la suma de las propiedades `Value` de cada miembro; y si se trabaja con la matriz, el valor será la suma ponderada de sus elementos, es decir, la multiplicación de todos los elementos de una misma fila y la suma de los resultados de esas multiplicaciones en cada fila. A continuación un ejemplo que explica lo anterior, usando tanto el arreglo como la matriz.

ControlsArray(i).Value	ControlsMatrix(i,j).Value			
7	5	8	2	$5 \times 8 \times 2 = 80$
3	2	3	6	$2 \times 3 \times 6 = 36$
2	3	4	9	$3 \times 4 \times 9 = 108$
5	7	1	3	$7 \times 1 \times 3 = 21$
Value = 17 (Suma)	(Suma) Value =			245

Tabla 5.2: Ejemplos de instancias de PonderedSumCurrentEditTextBox

Esta clase es útil por ejemplo en la Autoliquidación de tributos, en donde se necesita sacar un porcentaje del CIF para un determinado impuesto y sumar con el valor obtenido con otro impuesto, y así sucesivamente para sacar el total. Solo basta con escribir al momento de cargar el formulario declaraciones del tipo:

Me.txt2.BindTo(New CurrencyEditTextBox(), {{Me.txt02, Me.txt2Cif}})

Me.txt3.BindTo(New CurrencyEditTextBox(), {{Me.txt03, Me.txt2Cif}, {Me.txt03, Me.txt1}, {Me.txt03, Me.txtC}})

ID EDIT TEXTBOX.- Esta es una clase heredada de `EditTextBox` que se usa para mostrar y editar identificadores en los formularios; tales como números de factura, números de DUI y DAU, etc. Cuando no tiene un valor ingresado mostrará en su propiedad `Text` el mensaje: "Escriba el <N. del identificador>.". Lo que va en <N. del identificador> viene dado por la propiedad `Public Property IAm(value As String) As String`; que le dice a la instancia de qué se trata el ID que representa, por ejemplo "número de factura". El valor del ID como tal se lo obtiene de la propiedad `Public ReadOnly Property Id() As String`, si no se ha ingresado ninguno esta propiedad es nula (`Nothing` en Visual Basic). Se sobrecarga entonces la propiedad `Text` de su clase base para implementar esta característica. Si se ha grabado el formulario en el que se encuentra este objeto, se fija la propiedad `Saved` en `True` para indicar que cuando el usuario intente cambiar el ID dando clic en el objeto, devuelva un mensaje al usuario advirtiéndole que está a punto de cambiar el identificador del formulario que acaba de grabar y preguntándole si desea continuar.

USERCONTROL

Es una clase utilizada exclusivamente para esta aplicación y es heredada de la clase `Eikon.ekControls.ekTextCodeName.TextCodeName` que pertenece

al paquete de controles de la empresa Eikon. Sus componentes más importantes son dos textboxes y un botón. La idea principal es que el texto del segundo textbox sea la descripción del código que se ingresa en el primero; y si el usuario no se acuerda del código, tiene la opción de usar el botón para llamar a un diálogo en el que elige el código a utilizar. La propiedad que corresponde al campo Text del primer textBox se llama Code. Cuando un código ingresado en este textBox no es uno conocido, ambos textboxes quedan con string vacío después de cambiar el foco o presionar Enter. La búsqueda del código se la hace mediante un query a la base de datos. El tipo de query a efectuar se lo elige mediante el atributo `TypeTableSearch` que describe la enumeración `MySearch.eTipoTableSearch` (algunos de cuyos elementos son `M_COUNTRY`, `M_COUNTRY_AGREE_DETAIL`, `M_DEPOSIT`, `M_PORT`, `M_DUTY`, `M_DESTINATION_TYPE`, `M_DISPATCH`, etc). Cada una de estos elementos se refiere a una tabla maestra dentro del modelo. Tiene exactamente las mismas funcionalidades adicionales que la clase `EditTextBox`: Vínculo a otro `UserControl1`; un `Label` como atributo; propiedad `HasTextChanged`; un tipo de observación mandatorio, condicional o no aplicable, etc.

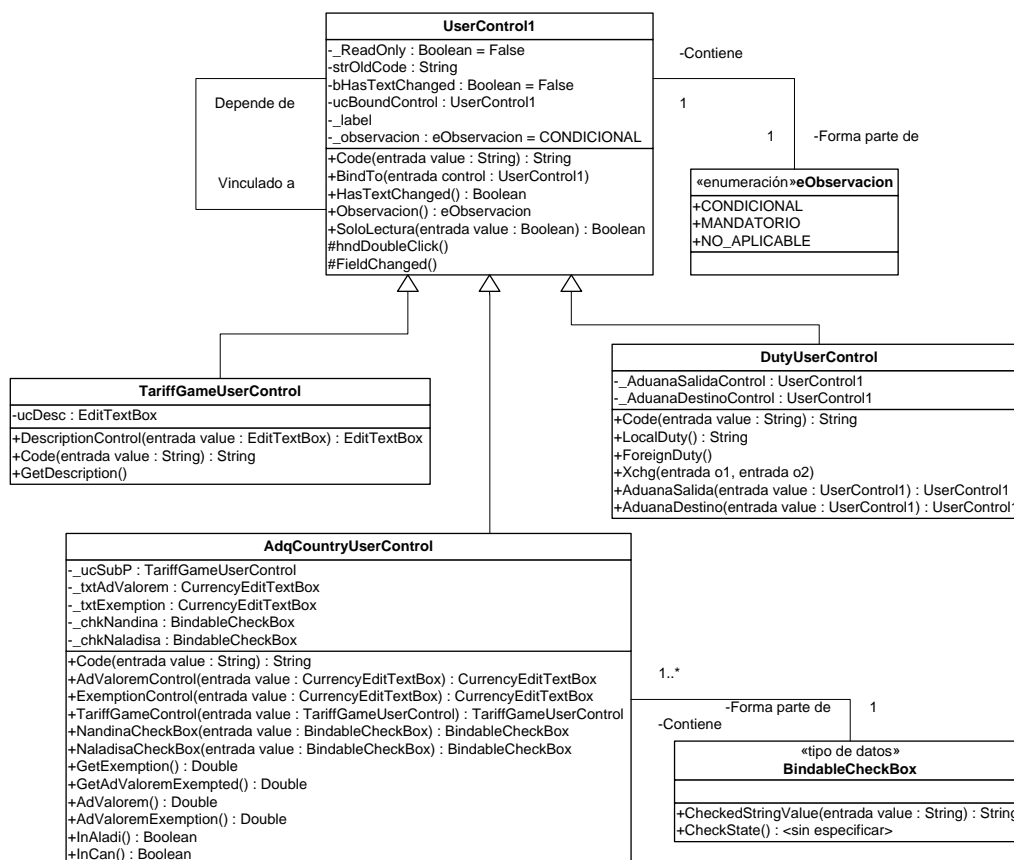


Figura 5.3: UserControl

ADQCOUNTRYUSERCONTROL.- Esta clase está desarrollada para que su instancia sea el control que describa el país de adquisición de un ítem de mercadería. El formulario de mercaderías tiene dos checkboxes que indican que el ítem pertenece a la CAN y/o a la ALADI; en cuyo caso existe un porcentaje de exención de aranceles (dependiente del país de adquisición y de la partida arancelaria) que esta clase se encarga de consultar. Contiene referencias hacia ambos checkboxes (que en realidad son `BindableCheckBox`s); dos

editTextBoxes, uno para recibir el porcentaje de ad valorem (su evento TextChanged es controlado con un handler dentro de la clase), y uno para fijar el porcentaje exento de aranceles calculado por esta clase.

DUTYUSERCONTROL.- El formulario de transporte necesita de este control debido a que existen tres campos que están relacionados entre sí: Cuando la propiedad Code del userControl1 que especifica el Tipo de destino corresponde a 'EN EL PAIS', el atributo TypeTableSearch del userControl1 ucAduanaSalida debe ser M_PORT (un puerto) y el del userControl1 ucAduanaDestino, M_DUTY (una aduana local). Pero si el tipo de destino es 'EN EL EXTRANJERO' se deben invertir los valores de estos dos atributos; es decir, ucAduanaSalida debe ser una aduana local y ucAduanaDestino un puerto. Entonces, la instancia de esta clase es la que describe el tipo de destino y tiene 2 referencias a los dos userControl1 de salida y destino, y verifica si su propia propiedad Code cambia, para intercambiar los valores de los respectivos atributos TypeTableSearch de estos dos controles.

TARIFFGAMEUSERCONTROL.- Este control se usa porque siempre que en un formulario se encuentre algún userControl1 que define la subpartida arancelaria, un editTextBox debe mostrar a qué descripción detallada corresponde esa subpartida. Este userControl1 tiene una referencia a ese editTextBox y cada vez que

cambia la subpartida ingresada en el campo Code de esta clase, se realiza una iterativa consulta en la tabla Tariff_Game de partidas arancelarias, buscando la descripción respectiva, llegando de partida a subcapítulo, y de subcapítulo a capítulo, para finalmente mostrar ese detalle en el editTextBox.

COLECTORFECHAHORA

Hereda de Infragistics.Win.UltraWinEditors.Ultra DateTimeEditor del paquete Net Advantage™ de Infragistics®. Se trata de un control que permite ingresar una fecha en una caja de texto o desplegar un calendario para elegirla (i.e., consiste de ambos componentes). Tiene exactamente las mismas funcionalidades adicionales que la clase EditTextBox y UserControl1: Vínculo a otro ColectorFechaHora; un Label como atributo; propiedad HasTextChanged; un tipo de observación mandatorio, condicional o no aplicable, etc.

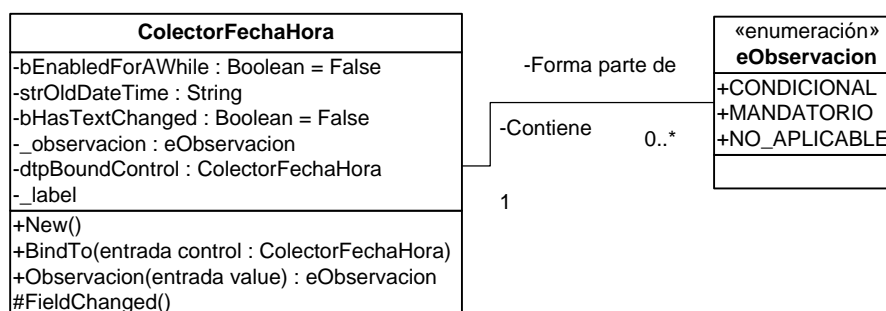


Figura 5.4: ColectorFechaHora

APÉNDICE C

CONCEPTOS GENERALES EN UN PROCESO ADUANERO

Definición de un proceso aduanero

Proceso aduanero se lo conceptualiza como un conjunto secuencial, repetitivo y ordenado, bajo ciertas condiciones, de actividades de la operativa aduanera en las que intervienen funcionarios y operadores de comercio exterior, tecnología adecuada, materiales, equipamiento, organizadas coherentemente para brindar un servicio ejerciendo control, de forma tal que satisfaga los requerimientos del usuario.

Los procesos más importantes y que son objeto de análisis son los de importación y exportación; sin embargo, en ambos casos, siempre se dan dos procesos que también son de gran importancia: El proceso de embarque y el proceso de despacho.

Proceso de embarque y proceso de despacho

En el caso de una importación, el embarque se da en el momento en que el cliente o importador se ha puesto de acuerdo y comprometido en la compra de mercancías al proveedor en el exterior; y ya sea contratando los servicios de un Agente Afianzado de Aduanas o por su propia cuenta, realiza todos los trámites pertinentes para embarcar la mercadería desde el puerto del país de adquisición.

En el caso de la exportación se hacen los mismos trámites para enviar la mercadería desde el país local hacia el exterior, donde el cliente en otro país espera desembarcar la mercadería.

Luego de que la nave de transportación lleva la mercadería al puerto de destino, se requiere un proceso de nacionalización de los productos desembarcados en ese país. Este es el proceso de despacho objeto principal de interés y análisis y es llevado a cabo comúnmente por despachadores de aduana o por los mismos clientes, quienes directamente tratan con las autoridades de la aduana.

Agentes que intervienen en un proceso aduanero

Agencia Naviera

Entidad de la línea naviera en el puerto de origen y/o destino que se encarga de efectuar las diligencias correspondientes al correcto embarque y desembarque de las mercancías enviadas desde y hacia el puerto en el que se encuentra.

Línea Naviera

Empresa encargada de enviar mercadería de un destino a otro contratando los servicios de medios de transporte debidamente certificados y confiables; motivo por el cual, cada línea naviera tiene su propia lista de buques, en el caso de viajes vía marítima o aviones de carga, en el caso de vía aérea.

Armador

Persona física o jurídica propietaria del buque, o la que, sin serlo, lo tiene en fletamento. En cualquiera de los casos, es el que acondiciona el buque para su explotación, obteniendo rendimiento del flete de las mercancías o transporte de pasajeros. Persona bajo cuyo nombre y responsabilidad directa gira la expedición o la gestión de la explotación del buque. Algunas veces el armador es la misma línea naviera.

Agente o Despachador de Aduanas

Es la persona natural o jurídica cuya licencia otorgada por el Gerente General de la Corporación Aduanera le faculta a gestionar de manera habitual y por cuenta ajena, el despacho de las mercancías, debiendo para el efecto firmar la declaración aduanera⁸

El Agente de Aduana tendrá el carácter de Fedatario Aduanero en cuanto que la Aduana tendrá por cierto que los datos que consignan en las declaraciones aduaneras que formulen, guarden conformidad con los antecedentes que legalmente le deben servir de base, sin perjuicio de la verificación que puede practicar el Gerente Distrital. El Agente de Aduana que interviene en el despacho de las mercancías es responsable solidario de la obligación tributaria aduanera, sin perjuicio de la responsabilidad penal que legalmente corresponda.

El otorgamiento y suspensión de la licencia de agente de aduana así como sus obligaciones se determinan en el Reglamento de la Ley.

Proceso de Embarque

Definiciones

Itinerario

⁸ Art. 120 de la Ley Orgánica de Aduanas.

Un itinerario es una ruta que debe recorrer un buque debido a un encargo de la línea naviera para llevar mercadería almacenada en uno o varios contenedores a un destino o varios.

Se puede ver a un itinerario como el conjunto de destinos en términos de los puertos de los países a los que debe llegar un buque transportando la mercadería solicitada por el cliente. Dentro de un buque podrían caber más de un contenedor, y dentro de cada uno mercadería de distinta procedencia a ser entregada en uno o varios puertos especificados en el manifiesto que es entregado al momento del viaje.

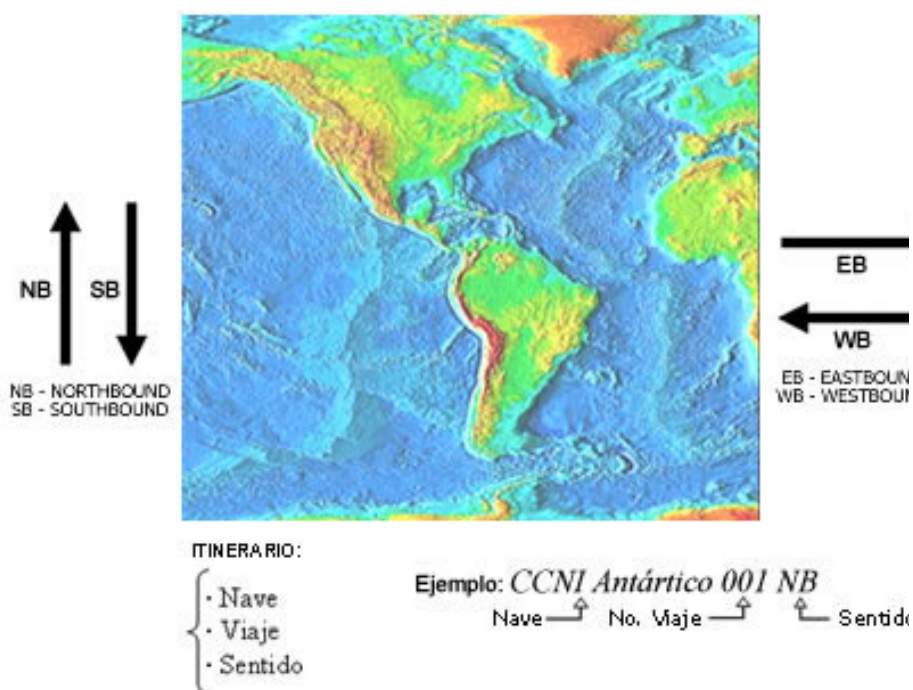


Figura 1 Itinerario

Para la descripción de un itinerario se toman en cuenta 3 elementos (figura 1): El nombre de la nave, el número de viaje, y el sentido del viaje.

Generalmente, la mercadería viaja de un puerto a otro que se encuentra en la ruta de un buque, pero no siempre es así. En términos prácticos, se elige una naviera lo más económica posible. Un buque se dice que tiene una línea regular cuando tiene una ruta definida. En caso de que una mercadería necesite llegar a un sitio que no está en ningún puerto dentro de la ruta regular de un buque, se contrata el envío de la mercadería al puerto más conveniente y se contrata los servicios de otro medio de transporte, ya sea otro buque con otra ruta regular o cualquier otro medio de transporte, por ejemplo, terrestre. A esta alternativa se le llama trasbordo.

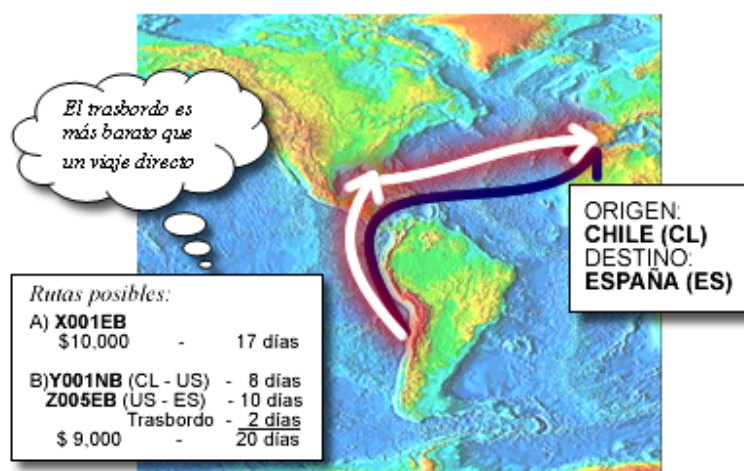


Figura 2 Traslado**Traslado**

Se denomina traslado a la operación que consiste en enviar la carga de un puerto enrutado a un destino no enrutado.

De manera general, se opta por enviar la carga por una ruta comercialmente común. En la figura 2 se muestra una ruta común usada para llevar mercadería desde cualquier país de Latinoamérica hacia Europa. En el ejemplo se compara el costo de un envío por medio del buque X, de carga desde Chile en un viaje directo pasando a través del Canal de Panamá hacia España; contra el uso del buque Y que envía la mercadería desde Chile a través del Canal de Panamá y hacia el puerto de Miami, en los Estados Unidos; para posteriormente usar un traslado de la mercadería desde Miami hacia Europa para llegar a su destino en España. En el ejemplo, puede verse que a pesar

de que un viaje toma menos tiempo que un trasbordo, es más económico y por tanto es de uso común por la mayoría de la gente. En realidad, un cliente no tiene que preocuparse de cómo llevar la carga, ya que la naviera se encarga de elegir la mejor ruta, y contratar los servicios de quienes correspondan para tal efecto.

La solicitud para el trasbordo se la presentará ante el Gerente del Distrito de ingreso en el plazo de 24 horas de llegado el medio de transporte. La solicitud debe ser efectuada por el transportista y antes de la descarga de las mercancías.⁹

El Gerente Distrital, bajo su vigilancia, autorizará el transbordo con destino al exterior por una sola vez en el mismo viaje y se efectuará:

- Directamente de un medio de transporte a otro;
- Con descarga a tierra; y,
- Previo almacenamiento temporal en una bodega en espera de su destino ulterior.

En el caso de los dos primeros, el transbordo se deberá efectuar dentro de las 24 horas siguientes a su autorización. En el literal c) dentro de los 5 días siguientes a partir de su autorización.

⁹ Tomado del art. 34 del Reglamento General a la Ley Orgánica de Aduanas.

Cuando la operación de trasbordo obedezca a caso fortuito o de fuerza mayor, el Gerente Distrital concederá un plazo prudencial adicional para esta operación, tomando en cuenta las circunstancias.

Las mercancías extranjeras objeto de trasbordo no están sujetas al pago de tributos, pero sí al de tasas por servicios aduaneros.

Importación

Cuando se realiza una importación siempre existe una declaración explícita de los términos bajo los cuales la mercadería va a arribar al puerto de destino. En el momento de la llegada o el arribo de la carga, se realiza lo que se llama una libre plática entre la Aduana, un representante de la Agencia Naviera, y comúnmente un Inspector Sanitario. Una vez realizada la libre plática, se realiza lo que se conoce como DRM, el documento de recepción del medio de transporte. En él se encontrará información referente más a la nave que a la carga que transporta. Finalmente, se efectúa el desembarco físico de la mercadería de la nave.

Exportación

En el caso de la exportación se debe especificar la mercadería que se va a embarcar en la nave antes que esta zarpe. Para ello, se llenan dos documentos, a saber: La lista de embarque y el Manifiesto de Carga. El último nombrado es de vital importancia porque es una constancia que refleja explícitamente lo que se está transportando, y debe concordar con lo que físicamente está en el buque en el momento del embarque, y es una constancia de verificación en el momento del desembarque en el destino. Luego se realiza el proceso físico de exportación y el documento que generado para este caso es el Documento de Zarpe (DSM).

En una operación de importación o exportación existen personas que se dedican a estibar o desestibar; es decir, a acomodar de una manera eficiente la carga dentro de los contenedores que viajan en la nave; y en destino, desembarcarla.

Documentos a enviarse

Una vez que se conocen los elementos que intervienen en el proceso de embarque, es posible comprender como funciona y como se relaciona con el proceso de despacho, que es el objeto de análisis.

Especificando un proceso de importación el cliente, quién es en este caso el exportador que se encuentra en el extranjero y desea enviar la mercadería a nuestro país, contrata los servicios de una línea naviera a través de la agencia naviera ubicada en ese país. En este punto el cliente se lo conoce como Embarcador o Shipper. El destinatario de la carga, en este país, recibe el nombre de Consignatario o Consignee. En el país de origen, se debe detallar la mercadería que se envía a través de un documento de transporte; para el caso del uso de un medio marítimo, este documento se conoce como B/L (Bill of Lading, Conocimiento de embarque marítimo).

En destino, la aduana de nuestro país debe recibir los B/Ls para constatar al momento del desembarco de que la mercadería declarada efectivamente es la que físicamente consta en la nave. Hoy en día, los B/Ls pueden y deben ser enviados con antelación a la fecha de arribo de la nave.

El embarcador debe declarar el Régimen al cual se acoge el embarque. En secciones posteriores se hace un análisis extensivo de lo que tiene que ver con los Regímenes Aduaneros.

En los documentos de transporte correspondientes a la mercadería, se debe especificar tres elementos: el puerto y país de origen, el tránsito

de la mercadería y el puerto y país de destino. El tránsito quiere decir el último puerto de trasbordo de la mercadería.

Para la Aduana del Ecuador, la declaración para destinos que no tienen un Distrito Aduanero se realiza de la siguiente manera: Suponiendo que el destino se encuentre en Ambato, el cual no es un puerto, se registra su nacionalización en el puerto en el que desembarcó, por ejemplo Guayaquil (GYE).

En cuanto a los roles de los agentes involucrados en este proceso, desde el punto de vista de nuestra aduana, se puede remarcar lo siguiente:

Para importaciones: el cliente es el consignatario (consignee).

Para exportaciones: el cliente es el embarcador (shipper).

En secciones anteriores se hizo referencia al término Manifiesto. A continuación se especifica este concepto y su relación con lo antes visto.

El Manifiesto es el documento que contiene una descripción lo más específica posible basada en un formato ya definido, de toda la mercadería que es embarcada en el medio de transporte hacia un destino específico. Este documento es emitido en el origen y debe

concordar con la mercadería entregada en cada uno de los destinos y puertos a los cuales la nave trasborda.

El documento de embarque, B/L para el caso de una transportación marítima, es un documento en el que se describe los datos más relevantes de la mercadería que se embarca en el medio de transporte por un cliente determinado hacia un destino.

Con lo expresado anteriormente, se puede inferir que existe un manifiesto por cada itinerario de una nave conteniendo un resumen de todos los documentos de embarque que esta lleva consigo.

A continuación considérese el ejemplo ya citado: Si la nave sale de Chile hacia España, pasando por medio del Canal de Panamá y haciendo trasbordo en Miami, existen 2 puertos de destino y 1 de trasbordo, lo que hacen 3 puertos. Por tanto, debería haber 3 manifiestos conteniendo la descripción de la carga.

Por otro lado, se tiene la siguiente distribución de la mercadería que se encuentra en la nave: 10 clientes que transportan mercadería a la Ciudad de Panamá, por tanto hay 10 B/Ls a Ciudad de Panamá; 5 B/Ls a Miami; y 2 B/Ls a España. En consecuencia, la nave debe llevar consigo al momento del embarque 17 B/Ls, y en el manifiesto

declararse que se transporta la mercadería que consta en esos 17 B/Ls.

En nuestro país, se acostumbra a llamar a los B/Ls como los manifiestos.

Contenedores y Proceso de tarja¹⁰

Cuando la mercadería ha llegado al puerto que solicitado, se produce una inspección de que toda la mercadería declarada ha llegado efectivamente, tarjando la Lista de Descarga y la Lista de Tarja que es entregada por la naviera al representante de la Autoridad Portuaria.

Con respecto a los contenedores, sus dimensiones son agrupables en dos tipos: los de 12 pies y los de 14 pies de largo; ya que su ancho y altura son similares. Su clasificación principalmente se basa en la carga que transporta. A continuación una lista de los principales tipos de contenedores:

¹⁰ **Tarjar.** (*N. del t.*) tr. Señalar o rayar en la tarja lo que se va sacando fiado, o lo que se cuenta. Tachar (borrar lo escrito). || puntear (compulsar una lista). Biblioteca de Consulta Microsoft® Encarta® 2005. © 1993-2004 Microsoft Corporation. Reservados todos los derechos.

DV) Dry Van (Seco); RF) Refrigeración; OT) OpenTop (tapa abierta);
 VE) Ventilado (tiene en los extremos rejillas para la circulación de aire dentro); HC) High Cube (mayor altura); FR) Flat Ran; IT) Isotank

Consolidadores

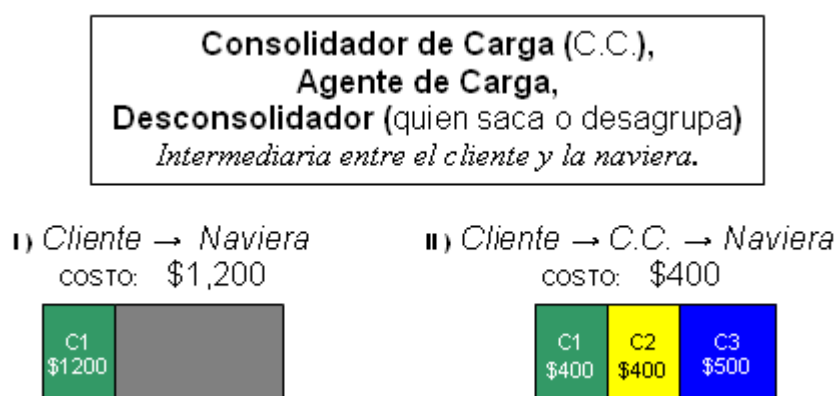


Figura 3 Labor de un Consolidador de Carga

La acción de acomodar la mercadería en los contenedores se la puede realizar de dos maneras, como se muestra en la figura 2.3. La primera de ellas resulta al adquirir un contenedor para almacenarla, cuando el cliente trata directamente con la naviera. Esto puede resultarle muy costoso y convertirse en un gasto inútil. De acuerdo al ejemplo de la figura, el cliente compra un contenedor única y exclusivamente para almacenar una mercadería que puede que difícilmente va a ocupar todo su volumen, a un costo muy elevado. La

segunda, por otro lado se da al contratar los servicios de un Consolidador o Agente de Carga, quien a su vez puede ser contratado por otros clientes para ocupar un contenedor y almacenar sus respectivas mercaderías. Entonces un mismo contenedor puede ser ocupado por uno más clientes y el costo por cliente es mucho menor.

Entonces, el consolidador de carga contrata a la aerolínea por peso y espacio y el cliente de la naviera ahora es el consolidador (y el consignatario se convierte en el embarcador o shipper). Es ahora el agente de carga el que envía a la Aduana y al cliente los B/Ls o AWBs. Esta documentación, junto con las demás documentación ya mencionada debe ser enviada antes que se produzca la libre plática. El cliente adicionalmente recibe del consolidador B/Ls o AWBs hijas amparadas en el B/L o AWB Master que se envió a la Aduana. El consignatario que recibe la carga es el agente de carga desconsolidadora en el país de destino.

Zonas Aduaneras

La Zona Primaria es la parte del territorio aduanero en la que se habilitan recintos para la práctica de los procedimientos aduaneros; la Zona Secundaria es la parte restante del territorio aduanero.

En la zona primaria, el Gerente General de la Corporación Aduanera Ecuatoriana será la máxima autoridad aduanera y ejercerá el control a través de los órganos administrativos, operativos y de vigilancia señalados en la Ley.

El Gerente General podrá establecer en la zona secundaria, perímetros fronterizos de vigilancia especial.¹¹

El embarque por línea aérea

Existe una estrecha similitud entre los diversos medios de transporte y el tratamiento de estudio que a ellos se puede someter. En todo caso, la importancia que se dará en el presente análisis se enfoca a estos dos medios (marítimo y aéreo). En el ámbito aéreo, la Línea Aérea, generalmente es la dueña del avión y existe una oficina de ella en cada aeropuerto. El documento de embarque al que se referirá en el caso de transporte aéreo es la Guía Aérea (AWB, Airway Bill, en inglés), el que es comparable con el B/L del medio marítimo.¹²

¹¹ Art. 3 de la Ley Orgánica de Aduanas.

¹² El documento de embarque del medio de transporte terrestre es la **Carta de Porte**.

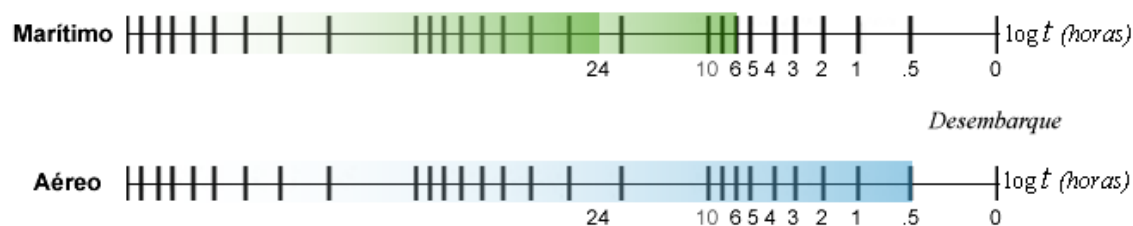


Figura 4 Plazo de entrega de documentación dependiendo de la vía de transporte.

En el medio de transporte marítimo, el tiempo de espera que se permite la entrega de los documentos requeridos para el desembarque o despacho, entre ellos los manifiestos, documentos de embarque, el envío electrónico de los documentos declaratorios y los ya mencionados DRM y DSM son de 24 y hasta de unas 6 horas antes del arribo de la nave (figura 2.4). En cambio que en el medio aéreo, debido a que los viajes son más rápidos, se exige que se entregue toda la documentación hasta una media hora antes del arribo al aeropuerto del avión que transporte la carga. La entrega tardía de cualquier documento pasado el tiempo establecido incurriría en una multa cobrada al importador, o al despachador, si se hace el trámite por medio de este agente. ¹³

¹³ Conviene diferenciar las funciones que realiza la Aduana de las que realiza la Autoridad Portuaria (Aeroportuaria, en el caso de aéreo). La Autoridad Portuaria (o Aeroportuaria) se encarga de la logística del puerto (o del aeropuerto), es decir de la infraestructura. En cambio que la Aduana se encarga de controlar las operaciones importación y exportación como tales.

Proceso de Despacho

En las presentes secciones se estudiarán los detalles más importantes de la actividad se realiza una vez que la mercadería ha sido embarcada desde un puerto de origen y cuáles son los requisitos que se necesitan cumplir para nacionalizar correctamente en el destino. Como es lógico, el análisis se efectuará desde el punto de vista del Ecuador; definiendo como importación al proceso de traer mercadería desde cualquier parte del mundo a nuestro país; y exportación, el proceso de llevar mercancía desde el Ecuador hacia el resto del mundo. Por tanto, lo expuesto a continuación se basa en la Ley Orgánica de Aduanas del Ecuador (en adelante, LOA), así como tomando en cuenta los Convenios Internacionales que mantiene el Ecuador con los demás países del mundo.

El sistema a desarrollarse y bajo la cual se ampara la presente tesis se encarga precisamente de brindar las facilidades necesarias para que un agente despachador de aduanas emita los documentos pertinentes utilizados en la declaración aduanera y cumplir con los requisitos de nacionalización y/o exportación de mercaderías.

Aforo o verificación en destino

Aforo es el acto administrativo de determinación tributaria, mediante el cual el distrito aduanero procede a la revisión documental o al reconocimiento físico de la mercancía, para establecer su naturaleza, cantidad, valor y clasificación arancelaria.

Los aforos se realizarán por parte de la administración aduanera o por las empresas contratadas o concesionadas y se efectuará en destino, conforme a las disposiciones que dicten para el efecto la Corporación Aduanera Ecuatoriana (CAE). Las empresas contratadas o concesionadas serán responsables solidarias con el importador en los términos previstos en los contratos de concesión que estuvieren vigentes.

El aforo físico en destino es obligatorio en los siguientes casos:

Cuando la mercancía venga con certificado de inspección en origen y se active el mecanismo de selección aleatoria;

Cuando el declarante no acepte las observaciones formuladas por la aduana a su declaración;

Cuando el Gerente Distrital conociere o presumiere del sometimiento de un ilícito aduanero;

En los casos en que la verificación en origen no sea exigible;

Cuando lo solicitare el declarante; y,

En los demás que establezca el Directorio de la Corporación Aduanera Ecuatoriana.

Procede al aforo documental cuando la importación venga con certificado de inspección en origen y no se active el mecanismo selectivo aleatorio.

Asignación del canal de despacho

El sistema informático selecciona el canal de despacho o tratamiento a aplicar mediante un modelo probabilístico, que clasifica las declaraciones como de bajo, mediano o alto riesgo, teniendo en cuenta lo dispuesto por las disposiciones legales, así como el criterio de aleatoriedad.

Se asignará cualquiera de los canales que se detallan a continuación, las que deben ser respetadas, bajo responsabilidad:

ACEPTACIÓN DIRECTA: Declaración sujeta a aforo de documentos electrónicos.

Las Declaraciones seleccionadas a esta vía, se sujetarán al aforo de documentos electrónicos, que se realiza mediante el cruce de los datos contenidos en los formatos electrónicos DAUHDR01, DAUDET01, DAVHDR01, DAVDET01 y DAVDET02 con los del formato DAUDOCAS. No se requiere revisión de documentos impresos, ni reconocimiento físico, previo a la salida de la mercancía. Acerca de los formatos electrónicos, se profundizará más adelante.

La CAE remitirá el número de refrendo y la autorización de pago con el desglose de la liquidación.

Esta vía permitirá la salida de la mercancía luego de cancelados los tributos que correspondan, en el banco corresponsal.

El Agente de Aduana será el fedatario de todos los documentos correspondientes, los mismos que podrán estar sujetos a las auditorías que pueda realizar la CAE.

CANAL DOCUMENTAL: Declaración sujeta a aforo de documentos electrónicos y de documentos impresos.

Estas Declaraciones son seleccionadas por el sistema informático aduanero durante el proceso de validación y se informará de tal condición al Agente de Aduana, junto al número de refrendo.

Se deberá presentar los documentos de acompañamiento en ventanilla, junto con la declaración impresa y firmada por el Agente de Aduana, para dar inicio al aforo documental.

Además del aforo de los documentos electrónicos, el personal de aduana realizará el aforo de documentos impresos, que consiste en la revisión material de documentos físicos tales como la declaración y sus documentos de acompañamiento.

Cancelados los tributos y confirmado el pago, se autoriza la salida de la mercancía.

Cuando el personal encargado determine que las declaraciones seleccionadas a esta vía requieren de aforo físico, se comunicará al jefe inmediato, quien autoriza el requerimiento y envía la orden al Departamento de Aforo, para su realización.

El Agente de Aduana será el fedatario de todos los documentos correspondientes a la declaración, los mismos que serán

entregados a ellos siguiendo el procedimiento designado por la CAE.

CANAL AFORO FÍSICO POR ADUANA: Declaración sujeta a aforo de documentos electrónicos, documentos impresos y reconocimiento físico de la mercancía:

Estas Declaraciones son seleccionadas por el sistema informático aduanero, durante el proceso de validación y se informará de tal situación al Agente de Aduana, junto al número de refrendo.

Se deberá presentar los documentos de acompañamiento en ventanilla, junto con la declaración impresa y firmada por el Agente de Aduana, para dar inicio al aforo documental.

El personal de aduana realiza el aforo documental, cotejando los documentos presentados, con la información existente en el sistema.

El personal de aduana es el que realiza el aforo físico de las mercancías, a continuación ingresa al sistema informático el resultado del mismo, como máximo, hasta 24 horas después de concluido el aforo.

De haberse producido ajustes en el valor de la mercancía, el sistema calcula la liquidación, la imprime y se entrega la liquidación al Agente de Aduana para el pago correspondiente.

Cancelados los tributos y confirmado su pago se autoriza la salida de la mercancía.

El Agente de Aduana será el fedatario de todos los documentos correspondientes a la declaración, los mismos que serán entregados a ellos siguiendo el procedimiento designado por la CAE.

CANAL AFORO FÍSICO POR VERIFICADORA: Declaración sujeta a aforo de documentos electrónicos, documentos impresos y reconocimiento físico de la mercancía (Canal Rojo).

i Estas Declaraciones son seleccionadas por el sistema informático aduanero durante el proceso de validación y se informará de tal situación al Agente de Aduana, junto al número de refrendo.

Se deberá presentar los documentos de acompañamiento en ventanilla, junto con la declaración aduanera impresa y firmada por el Agente de Aduana, para dar inicio al aforo documental.

El personal de la aduana realiza el aforo documental, cotejando los documentos presentados, con la información existente en el sistema.

El personal de la empresa verificadora realiza el aforo documental, cotejando los documentos presentados, con la información existente en el sistema.

El personal de la empresa verificadora realiza el aforo físico de las mercancías, a continuación ingresa al sistema informático el resultado del mismo, como máximo, hasta 24 horas después de concluido el aforo.

De haberse producido ajustes en el valor de la mercancía, el sistema calcula la liquidación, la imprime y se entrega la liquidación al Agente de Aduana, para el pago correspondiente.

Cancelados los tributos y confirmado su pago, se autoriza la salida de la mercancía.

El Agente de Aduana será el fedatario de todos los documentos correspondientes a la declaración, los mismos que serán entregados a ellos por la CAE siguiendo el procedimiento designado por la CAE.

En el caso de declaraciones que estén en la vía directa, no se exigirá la presentación de la declaración impresa.

En el caso de la vía documental o vía física por aduana se presentará la declaración impresa de acuerdo a lo indicado, en la ventanilla de recepción en el departamento de Importaciones del Distrito.

Se considerará como fecha de presentación de la declaración la de aceptación de la transmisión electrónica.

En caso de no haber observaciones, el funcionario asignado, sumilla la hoja de ruta en la casilla correspondiente a Aforo, anotando la fecha, su código, firma y sello.

De no ser conforme, existir errores o no presentarse algún documento exigible por ley, se procede de la siguiente manera: Se

determinará la infracción y de acuerdo a ello el procedimiento a seguir, según los Artículos 80 - 92 de la LOA.

En caso de que el trámite sea asignado a un Aforo de documentos impresos, el trámite pasará directamente a la etapa de liquidación.

Si es que el trámite es asignado a aforo físico, sea éste por aduana o verificadora, el sistema distribuirá la información correspondiente al aforador asignado, para que la misma sea utilizada en el momento de la constatación física;

La empresa verificadora recibirá la información de los trámites que le han sido asignados para la realización del aforo físico;

El Agente de Aduana debe presentarse en el local de almacenamiento, para que el Funcionario de Aduana designado por el sistema, efectúe el reconocimiento físico de la mercancía en los lugares habilitados para ello y contando con la presencia del importador o uno de sus representantes;

Los resultados del aforo deberán ser registrados en el sistema, en el caso de tratarse de Aforos realizados por empresas verificadoras.

Este informe deberá ser presentado máximo 24 horas después de realizado el aforo e ingresado al sistema;

Si hubieran diferencias menores o iguales al 10% del valor FOB declarado, el sistema ajustará la liquidación de acuerdo a lo constatado en el Aforo Físico.

Si como resultado del Aforo Físico, se encontraran diferencias de naturaleza, cantidad, valor o clasificación arancelaria, que repercutan en los tributos y que la diferencia entre autoliquidación y dicho cálculo, fuera superior al 10% del valor FOB declarado, dichos casos serán puestos a consideración de la Gerencia Distrital por presunción de delito.

Toda importación, cuyo valor sea superior a \$4,000 USD, deberá contar con el correspondiente certificado de verificación en origen, excepto las importaciones destinadas al sector diplomático y consular; las mercaderías declaradas en tránsito aduanero con destino al exterior; el equipaje acompañado de viajero; los envíos o paquetes postales, cuyo valor CIF o FOB, en su caso, no exceda del límite que se establece en el Reglamento de esta Ley, transportados por cualquier clase de correo, incluidos los denominados correos rápidos; mercancías destinadas al uso o

consumo doméstico entre poblaciones fronterizas y los productos de pesca en alta mar.

Toda mercadería proveniente de zonas francas, puertos libres, puertos de transferencia y, en general de los denominados paraísos fiscales, ingresados vía terrestre, marítima, fluvial o aérea, será obligatoriamente sometida a aforo físico en destino.

Los certificados de inspección en origen emitidos por las compañías verificadoras tienen la categoría de instrumento público.

Cuando se trate de un trámite con aceptación directa, el Sistema comunicará al usuario el número de Refrendo o Aceptación y el Desglose de la liquidación para que éste pueda acercarse a las instituciones bancarias y efectuar el pago correspondiente o autorizar el débito bancario;

Para los casos en que se realice un Aforo Físico o documental, el sistema realizará el cálculo de los tributos a liquidarse, el mismo que será impreso para su posterior cancelación en los bancos autorizados.

El art. 49 de la LOA cita lo siguiente al respecto:

Art. 49 Autorización de Pago.- El Distrito autorizará el pago de los tributos aduaneros:

Cuando el aforo documental sea firme y no se haya activado el mecanismo de selección aleatoria; y,

Si realizado el aforo físico no aparecieren observaciones que formular a la declaración.

Cancelados los Tributos y confirmado el pago se autoriza la salida de la mercancía; tal como lo estipula el art. 50 de la Ley:

Art. 50 Entrega de la mercancía.- Procede la entrega de las mercancías:

Cuando se han pagado los tributos al comercio exterior;

Cuando se ha garantizado el pago de los tributos al comercio exterior; y,

Cuando se ha presentado el certificado liberatorio otorgado por la autoridad competente.

Ordenada la entrega de la mercancía ninguna autoridad podría retenerla, salvo orden judicial que disponga lo contrario.

Los Almacenes Temporales y las Garitas de Aduana permiten el retiro de la mercancía de los recintos, previa verificación de la

cancelación, mediante consulta realizada en la página WEB de la Aduana (<http://www.aduana.gov.ec> o <http://www.cae.gov.ec>), constatándose lo siguiente: el número de declaración o Refrendo; el número de autorización de salida; y el número de contenedor o marcas y números;

Mercadería cuyo valor FOB es menor de US \$ 4000 y la verificación física se realiza en el Ecuador, los valores que se paga a la verificadora en el país de destino son los siguientes:

Desde	Hasta	% del FOB con un mínimo de \$10
-	50	25
50	99	30
99	149	35
149	199	40
199	1,999	US\$ 80
1,999	3,999	US\$ 120
3,999	-	US\$ 200

Tabla 1 Costo de una verificadora ecuatoriana

Aranceles, Partidas y Subpartidas Arancelarias

Definición de Arancel

Es un sistema de gravámenes que se aplica a la importación de las mercaderías para proteger la industria nacional o el área de una

integración, genera recursos fiscales, orienta la estructura, el consumo o sirve para corregir los desequilibrios de la balanza de pagos¹⁴.

ARANCEL EXTERNO COMÚN

De conformidad con el artículo 92 del Acuerdo de Cartagena mediante Decisión 370 de noviembre de 1994, los países miembros de la Comunidad Andina aprobaron el Arancel Externo Común, a partir del 6 de marzo del 2002, se adoptó un nuevo arancel basado en la Decisión No 507 de la Comisión de la Comunidad Andina el mismo que fue publicado en el Registro Oficial No 547 del 3 de abril del 2002.

Los tributos que se pagan en las importaciones a través del arancel se denominan derechos arancelarios y son el resultado de multiplicar el valor CIF en dólares por la tarifa arancelaria propia de cada mercadería.

Estructura del Arancel

Está integrado por 8 dígitos los cuales no se pueden cambiar a nivel de la Comunidad Andina, y a esta estructura se denomina la Subpartida Arancelaria Regional.

¹⁴http://www.sica.gov.ec/agronegocios/nuevas%20agroexportaciones/negocios%20ecuador/importar/mecanismos_imp.htm

A nivel de 6 dígitos, esta codificación no la pueden cambiar los países por disposición de la Organización Mundial de Comercio (OMC), y se denomina Subpartida Arancelaria Internacional.

A nivel de cuatro dígitos se denomina la Partida Arancelaria.

A nivel de dos dígitos se denomina Capítulo.

El Ecuador tiene Subpartida a nivel de 10 dígitos y se denomina Subpartida Arancelaria Nacional.

Subpartida Arancelaria Nacional	0000.00.00.00
Subpartida Arancelaria Regional	0000.00.00
Subpartida Arancelaria Internacional	0000.00
Partida Arancelaria	0000
Capítulo	00

Tabla 2 Estructura del Arancel

1	DIGITO VERIFICADOR	PUDELECO EDITORES S.A.	ARANCEL NACIONAL DE ADUANAS (IMPORTACIONES)	RESTRICCIONES Y AUTORIZACIONES PREVIAS			NOTAS			
				IMPUESTO AL VALOR AGREGADO (IVA) %	DERECHOS ARANCELARIOS (% AD-VALOREM)	UNIDAD FISICA				
2				3	4	5	6	7		
PARTIDA Y PRODUCTO										
			22.07	Alcohol etílico sin desnaturalizar con grado alcohólico volumétrico superior o igual al 80% vol; alcohol etílico y aguardiente desnaturalizados, de cualquier graduación.....						
8			2207.10.00	- Alcohol etílico sin desnaturalizar con grado alcohólico volumétrico superior o igual a 80% vol.....		15	12	0003	1002	
0			2207.20.00	- Alcohol etílico y aguardiente desnaturalizados, de cualquier graduación.....		15	12	0003	1002	
			22.08	Alcohol etílico sin desnaturalizar con grado alcohólico volumétrico inferior al 80% vol; aguardientes, licores y demás bebidas espirituosas.						
			2208.20	- <i>Aguardiente de vino o de orujo de uvas:</i>						
				- - <i>De vino (Por ejemplo: «coñac», «brandys», «pisco», «singani»):</i>						
5			2208.20.21	- - - Pisco		20	12	0003	1002	
8			2208.20.22	- - - Singani		20	12	0003	1002	
4			2208.20.29	- - - Los demás		20	12	0003	1002	
5			2208.20.30	- De orujo de uvas («grappa» y similares).....		20	12	0003	1002	
8			2208.30.00	- Whisky		20	12	0003	1002	
0			2208.40.00	- Ron y demás aguardientes de caña		20	12	0003	1002	
2			2208.50.00	- «Gin» y ginebra		20	12	0003	1002	
4			2208.60.00	- Vodka		20	12	0003	1002	
			2208.70	- <i>Licores:</i>						
9			2208.70.10	- - De anís		20	12	0003	1002	
2			2208.70.20	- - Cremas		20	12	0003	1002	
3			2208.70.90	- - Los demás		20	12	0003	1002	
			2208.90	- <i>Los demás:</i>						
3			2208.90.10	- - Alcohol etílico sin desnaturalizar con grado alcohólico volumétrico inferior a 80% vol.....		15	12	0003	1002	
6			2208.90.20	- - - Aguardientes de ágaves (tequila y similares).....		20	12	0003	1002	
				- - - <i>Los demás aguardientes:</i>						
8			2208.90.42	- - - - De anís		20	12	0003	1002	
4			2208.90.49	- - - - Los demás		20	12	0003	1002	
7			2208.90.90	- - Los demás		20	12	0003	1002	
8			2209.00.00	Vinagre y secedáneos del vinagre obtenidos a partir del ácido acético..				20	12	0003

RESTRICCIONES Y AUTORIZACIONES PREVIAS		NOTAS	
0003	Previa autorización del MINSALUD	1002	El alcohol y productos alcohólicos están gravados con el Impuesto a los Consumos Especiales (Art. 71 y 78 de la Ley de Régimen Tributario Interno), además deben cumplir con requisitos específicos para su importación. Véase la Nota de subpartida 1 del Capítulo 22.

Figura 5 Fragmento del Arancel Nacional

Costo, Seguro y Flete

En previas secciones ya se ha hecho referencia a los términos FOB y CIF. A continuación se especificarán estos términos y otros relacionados, ya que todos los que aquí se detallan formarán parte de la Factura Comercial y la autoliquidación que realiza el importador o el Agente de Aduanas.

Incoterms

La Cámara de Comercio Internacional (CCI), estableció los Términos de Comercio Internacional desde 1980 con el fin de identificar bajo un

mismo estándar los parámetros de uso común en el ámbito aduanero. Los términos que más se están utilizando en el país son los de la primera versión 1.980, sin embargo la versión 2000 sigue ganando más adeptos, motivo por el cual se enfoca los dos esquemas.

Versión INCOTERMS 1980

Precio EXW. Comprende el precio del producto, más el primer embalaje, más el certificado de calidad peso o medida, determina el precio ex fábrica, las siglas EXW, en inglés *Ex work*, y significa que el comprador o importador adquiere la mercadería a un costado de la fábrica o lugar de expendio o producción.

Precio FOT-FOR. Del inglés free on truck o free on rail-road, libre sobre camión o ferrocarril al valor ex - fábrica se debe añadir un segundo o un tercer envase o embalaje con la finalidad de que la mercadería (dependiendo de la misma) se encuentre debidamente protegida, más el gasto de acarreo que puede ser manual o a través de una máquina para subir la mercadería de la fábrica al camión o al ferrocarril.

Precio FAS. Free along side ship, libre a un costado del barco, al precio FOT- FOR se suma los gastos que se producen tales como transporte

interno en el país exportador, tributos internos en país exportador y el gasto de descarga del camión al muelle.

El valor FOB. Free on board, libre a bordo al precio FAS se añade los siguientes gastos enunciativos más no limitativos: comisiones, corretajes, derechos arancelarios pagados en el exterior por el envío de la mercadería, derechos consulares, gasto de carga, gasto de estiba, certificados (Origen, sanitario, fitosanitario...), etc.

El valor C y F. Al valor FOB se añade el flete desde el país exportador al país importador y tenemos el valor Costo y Flete (C y F), Cost and freight.

El CIF. Al valor Costo y flete se suma el seguro que se debe adquirir en nuestro país y se obtiene el valor CIF (Cost Insurance and Freight), Costo Seguro y Flete.

En esta terminología se debe utilizarse sin importar el tipo de medio de transporte que se utilice. Será en la nota de pedido y en la factura comercial y en el contrato que se identificará plenamente el medio de transporte (aéreo, marítimo o terrestre).

Versión INCOTERMS 2000

Aquí se agrupa los términos por la letra y el medio de transporte (A – aérea, T – terrestre, M – marítima) que se va a utilizar.

Grupo	Símbolo	Simbología en	Simbología en español	Vías
Grupo de la E. <i>De salida.</i>	EXW	<i>Ex work</i>	Ex fábrica, lugar determinado.	A-T-M
Grupo de la F. <i>Sin pago del transporte principal por parte de exportador.</i>	FAS	<i>Free along side Ship</i>	Franco a un costado del barco, puerto convenido.	M
	FOB	<i>Free on board</i>	Franco a bordo, puerto	M
	FCA	<i>Free carrier</i>	Franco lugar convenido.	A-T
Grupo de la C. <i>Con pago del transporte principal por parte del exportador.</i>	CFR	<i>Cost and freight</i>	Costo y flete, puerto de destino convenido.	M
	CIF	<i>Cost Insurance and Freight</i>	Costo Seguro y flete, puerto de destino convenido.	M
	CPT	<i>Carriage paid to</i>	Transporte pagado hasta, lugar de destino convenido.	T-A
	CIP	<i>Carriage and Insurance</i>	Transporte y seguro pagado hasta lugar de destino	T-A
Grupo de la D. <i>De Entrada</i>	DAF	<i>Delivered at frontier</i>	Entregado en frontera, lugar de destino convenido.	T
	DES	<i>Delivered ex slip</i>	Entregado en barco, puerto de destino convenido.	M
	DEQ	<i>Delivered Ex quay</i>	Entregado en muelle, puerto de destino convenido.	M
	DDU	<i>Delivered duty unpaid</i>	Entregado en lugar de destino convenido sin derechos	A-M-T
	DDP	<i>Delivered duty paid</i>	Entregado en lugar de destino convenido con derechos pagados	A-M-T

Tabla 3 Incoterms versión 2000

Variables aduaneras dependientes e independientes

En esta sección se ahondará un poco en las variables ya referenciadas con el objeto de abstraer entre sí los valores que son ingresados por el usuario; esto es, las variables independientes, de aquellas que deben ser calculadas.

Se conoce que el valor FOB es el costo de la mercadería que se importa o se exporta, al que todavía no se han aplicado ningún derecho arancelario. En otras palabras; sea p el precio de la mercadería y q la cantidad de la misma, el valor FOB de esa mercadería viene dado por $FOB = pq$. Sea también i una de las n series de partidas arancelarias de las mercaderías que se traen o se envían al exterior; entonces el FOB de esa partida $FOBi = piqi$.

El Flete es el costo total que representa el transporte de la carga por cualquier medio. En la declaración es necesario el desglose por serie de partida arancelaria; por tanto para un valor de flete total dado, el flete por partida será una ponderación al peso W de este valor, esto es,

$$Flete_i = \frac{W_i}{W_T} Flete_T$$

El Seguro es una responsabilidad de las despachadoras de aduanas, ante cualquier eventualidad que afecte a la mercadería, y según políticas previstas por estas empresas. Generalmente, dentro de estas políticas está ofrecer un porcentaje de seguro a la mercadería al valor Costo y Flete; i.e., para un porcentaje de seguro 100s% (generalmente son del orden del 20%, 40%, etc.), el valor monetario del seguro estará dado por:

$$Seguro = s(Flete_T + pq)$$

Donde pq se sabe que es el FOB total. Entonces, para una serie de partida arancelaria i , el seguro de esa partida será:

$$Seguro_i = s(Flete_i + p_i q_i)$$

Despejando las variables dependientes de esta ecuación en valores ingresados, para cada una de las n series declaradas se tiene,

$$Seguro_i = s \left(\frac{W_i}{W_T} Flete_T + p_i q_i \right)$$

$$Seguro_i = s \left(\frac{W_i}{\sum_k^n W_k} Flete_T + p_i q_i \right)$$

El flete y el seguro en el caso de las exportaciones vienen incluidos en el precio de los artículos; por tanto, no se habla de flete o seguro en estos casos.

El valor CIF, como se ha visto anteriormente, es la suma del FOB, el flete y el seguro. Esto constituye la Base Imponible de la mayor parte de los impuestos o tributos de los que se asientan en las declaraciones aduaneras. En el caso de las importaciones, el CIF viene dado por la suma del FOB, flete y seguro, pero en las exportaciones el CIF tiene el mismo valor del FOB. Igual que ocurre con el flete y el seguro, el CIF también necesita ser desglosado en función de las partidas que se importen o exporten. Por tanto, uniendo las dos ecuaciones anteriores, se tiene una expresión para el valor CIF de una partida i :

$$CIF_i = p_i q_i + \frac{W_i}{\sum_k^n W_k} Flete_T + s \left(\frac{W_i}{\sum_k^n W_k} Flete_T + p_i q_i \right)$$

$$CIF_T = \sum_j^n \left(p_i q_i + \frac{W_i}{\sum_k^n W_k} Flete_T + s \left(\frac{W_i}{\sum_k^n W_k} Flete_T + p_i q_i \right) \right)$$

Obviamente, el valor total CIF debe coincidir con la suma de los valores CIF individuales para cada una de las partidas a declararse.

Un parámetro importante a tomar en cuenta es el Ad Valórem, que se conceptualizará posteriormente. Aritméticamente, el Ad Valórem es un porcentaje aplicado al CIF de cada serie de partida arancelaria declarada, como parte de los tributos arancelarios que se debe pagar. Sea $100\alpha_i\%$ tal porcentaje, que depende del tipo de mercadería en

cuestión. El ad valorem para cualquier partida arancelaria, ya expresado en unidades monetarias, aquí viene denotado por Adv_i , además de que, nuevamente, para dejarlo en términos de variables independientes se combina con la ecuación anterior y se tiene lo siguiente:

$$\begin{aligned}
 Adv_i &= \alpha_i CIF_i \\
 &= \alpha_i \left(p_i q_i + \frac{W_i}{\sum_k^n W_k} Flete_T + s \left(\frac{W_i}{\sum_k^n W_k} Flete_T + p_i q_i \right) \right) \\
 &= \alpha_i p_i q_i + \alpha_i \frac{W_i}{\sum_k^n W_k} Flete_T + \alpha_i s \frac{W_i}{\sum_k^n W_k} Flete_T + \alpha_i s p_i q_i \\
 &= \alpha_i p_i q_i + \alpha_i s p_i q_i + \alpha_i \frac{W_i}{\sum_k^n W_k} Flete_T + \alpha_i s \frac{W_i}{\sum_k^n W_k} Flete_T \\
 &= (1+s) \alpha_i p_i q_i + (1+s) \alpha_i \frac{W_i}{\sum_k^n W_k} Flete_T \\
 Adv_i &= \alpha_i (1+s) \left(p_i q_i + \frac{W_i}{\sum_k^n W_k} Flete_T \right)
 \end{aligned}$$

El ad valorem total, que va en la liquidación de tributos, no es más que la suma de los ad valorems respectivos a cada una de las series de partidas arancelarias de las mercancías. Esto es,

$$Adv_T = (1+s) \sum_i^n \alpha_i \left(p_i q_i + \frac{W_i}{\sum_k^n W_k} Flete_T \right)$$

Derechos Arancelarios y Para-Arancelarios

Derechos Arancelarios

Están determinados por el arancel, se entiende como tal el documento de carácter legal en el que constan todas las mercaderías, que en su clasificación sigue una norma internacional denominada Sistema Armonizado de Clasificación y Designación de la mercancía, conocido comúnmente como Sistema Armonizado. El tributo a cancelar debido a este derecho arancelario se lo conoce simplemente como AD VALÓREM (del lat. Según el valor).

Cuando la Comunidad Andina adoptó el Sistema Armonizado tomó el nombre de NANDINA, Nomenclatura Armonizada para el Grupo Andino y los países que lo integran son Ecuador, Colombia, Venezuela, Bolivia y Perú, con estos países cuando se importan productos originarios no se pagan ni derechos arancelarios ni cláusula de salvaguardia (Resolución No 129 del COMEXI publicado en el Registro Oficial N° 510 del 6 de febrero del 2002).

Como norma de excepción se tendría la comercialización con el Perú, que se debe ver si el producto que se va a comercializar está o no negociado con rebaja arancelaria.

Cuando el Sistema Armonizado adoptó ALADI Asociación Latinoamericana de Integración los productos originarios de esta región y que están negociados deben pagar los derechos arancelarios con un trato de preferencias de aranceles.

Los países de ALADI son: Venezuela, Colombia, Ecuador, Perú, Bolivia, Chile, Argentina, Uruguay, Paraguay, Brasil, México y Cuba. Venezuela, Colombia, Perú y Bolivia son miembros de la Comunidad Andina y las ventajas de desgravación son totales.

Con los países como de Chile, Uruguay, Cuba, el Ecuador se ha firmado Acuerdos Bilaterales y algunos son de Complementación Económica.

Derechos Para-Arancelarios

Se dividen en dos grupos: cualitativos y cuantitativos.

DERECHOS PARA-ARANCELARIOS CUALITATIVOS

Cualitativos son normas y procedimientos de carácter administrativo que se debe cumplir para importar una mercadería. Estos son: Certificados de Origen, Zoosanitario, Fitosanitario,

Ictiosanitario, de Inspección, de Calidad, Registro Sanitario, Autorizaciones y Prohibiciones.

Certificado de Origen

Es un documento que nos envía el exportador a petición del importador y nos indica que la mercadería es producida y fabricada en ese país, y sirve para que el importador tenga un trato de preferencias de aranceles por parte de las autoridades ecuatorianas. Es un documento necesario cuando se importa de los países de la CAN y ALADI.

El Certificado de Origen es un documento obligatorio entregado bajo las siguientes condiciones:

US\$ 4 000.00 de FOB en adelante; Consumo; Depósito Comercial Público; Depósito Comercial Privado; Almacenes libres y Especiales; Postal y Correo Rápido (según valor)

Están exentos de entregar dicho documento si se cumple que:

FOB menor a US\$ 4,000.00; Armas y Material Bélico – FF.AA. – PP.NN.; Periódicos, Libros y Revistas, materiales impresos, tarifa cero; Sellos de correos, timbres fiscales, etc.; Ley diplomáticos; Combustibles; Sector Público – SOLCA; Aéreos de Carácter Urgente

(según valor); Envíos de Socorro; Fétretos; Efectos Personales de viajeros; Donaciones; Muestras sin valor comercial; Regímenes Aduaneros Especiales; Importación temporal con reexportación en el mismo estado; Importación temporal para perfeccionamiento activo; Depósitos industriales; Reposición con franquicia arancelaria; Zona franca; Maquila; Ferias internacionales; Tránsito internacional; Exportación temporal con reimportación en el mismo estado; Exportación temporal para perfeccionamiento pasivo.

Certificado Zoosanitario

Indica que los animales o productos del reino animal en estado fresco, refrigerado, congelado o en agua salada no tienen enfermedades, la autoridad que se encarga de la revisión de estos productos es el Ministerio de Agricultura y Ganadería (MAG - SESA).

Certificado Fitosanitario

Indica que los vegetales en estado fresco, refrigerado y congelado no tienen enfermedades, la autoridad nacional que revisa en el MAG - SESA Servicio Ecuatoriano de Sanidad Agropecuario.

Certificado Ictiosanitario

Indica que los productos acuáticos no tienen enfermedades, la autoridad que revisa es el MICIP (Dirección General de Pesca).

Certificado de Inspección

Se debe realizar una Solicitud de Inspección a las Compañías Verificadoras de acuerdo con las disposiciones legales vigentes, si la mercadería: Tiene un valor FOB de US \$ 4,000 o más se debe efectuar la verificación en el lugar de embarque o de origen de la mercancía, el importador debe pagar por este servicio un mínimo de US\$ 180 más el IVA; a veces se paga el 50% al presentar la solicitud y el otro 50% cuando la verificadora entrega el Certificado de Inspección. En la solicitud de inspección se adjunta la nota de pedido, póliza de seguro y el Documento Única de Importación debidamente aprobado por el banco corresponsal. Las verificadoras más importantes son SGS, VERITAS, COTECNA, e ITS.

Mediante Resolución No. 11-2002-R3 publicado en el Registro Oficial No 619 de 16 de Julio de 2002, el Directorio de la Corporación Aduanera Ecuatoriana (CAE), establece una tabla para el cobro por parte de las verificadoras.

Mercadería en General		
DESDE US\$	HASTA US\$	TARIFA % SOBRE VALOR FOB
-	100,000	0.70
100,000	500,000	0.65
500,000	1,000,000	0.55
1,000,000	2,000,000	0.40
2,000,000	-	0.30
2.- Graneles		
	500.000	0.30
500.000	1.000.000	0.25
1.000.000		0.20
3.- Tarifa mínima		
180 dólares		

Tabla 4 Cobro de verificadoras por Certificados de Inspección

Como carga a granel se tiene entre otras: pescado, hortalizas, frutas, café, té, especias, malta, aceites, cacao, tortas, sal, azufre, tierras, minerales, urea, gas, químicos, abonos, látex, algodón, lino, cáñamo, yute, chancaca, etc.

Se encuentran exentas de verificación en origen¹⁵:

¹⁵ Resolución N° 10-2001-R-I del Directorio de la Corporación Aduanera Ecuatoriana R.O. N° 607 de 28 de junio de 2002.

Mercancías de valor FOB inferior US \$ 4,000; Objetos y compras personales de viajeros; Menaje de casa y equipos de trabajo exentos de pago de impuestos; Aquellas importadas al amparo de la Ley de Inmunidades, Privilegios, Franquicias; Diplomáticas Consulares y de Organismos Internacionales; Importaciones realizadas por el sector público, y la sociedad de lucha contra el cáncer; Paquetes de correo

Tráfico fronterizo de la mercadería; Féretros o ánforas que contengan cadáveres y restos humanos; Envíos de socorro por catástrofes naturales o siniestros análogos, a favor de entidades; del sector público o de organizaciones privadas de beneficencia o de socorro, exentos del pago de impuestos según la Ley Orgánica de Aduanas.

Autorizaciones Previas

Mediante Resolución N° 183 el COMEXI expide la normativa que regula el procedimiento de licencias de importación publicado en el registro oficial edición especial N° 6 de 5 de mayo de 2003. Para importar determinados tipos de productos se requiere en forma previa la autorización de alguna entidad pública razón por la cual se debe conseguir tal permiso antes de realizar cualquier trámite. Mediante Resolución No. 0111 de la CAE (R.O. N° 47 de marzo de 2003).

Armas municiones y explosivos previa autorización del Ministerio de Defensa.

Productos sujetos a fiscalización con autorización del CONSEP (Consejo Nacional de Sustancias Estupefacientes y Psicotrópicas).

Residuos radioactivos previa autorización de Energía Nuclear.

Fauna y flora silvestre en proceso de extinción para fines científicos o de estudios previa autorización y cupo del Ministerio de Agricultura y Ganadería.

Certificado de calidad

Certifica que la mercadería se ha elaborado de acuerdo con las normas internacionales de fabricación internacional las cuales son las normas ISO 9000, 9001, 9002, 9003, 9004 o ISO 9000/2000, ISO 14000.

En la actualidad debido a la apertura de las fronteras y a las políticas de globalización, el crecimiento de la competencia en calidad es más intensa tanto en el campo industrial como en el tipo de vida de la sociedad, a punto que la calidad se ha constituido en el factor más crítico en el comercio mundial, esto debido a múltiples razones las cuales le podemos sintetizar en lo siguiente:

Demandas más exigentes; Competencia cada vez más agresiva; Exigencias de mayor rentabilidad; Responsabilidad en los productos; Leyes y regulaciones

Registro Sanitario

Antes de realizar la importación se debe solicitar al Ministerio de Salud que se aplique la calificación de los productos los cuales son los siguientes:

Los alimentos procesados o aditivos, medicamentos en general, productos naturales procesados, drogas, insumos o dispositivos médicos, productos médicos naturales y homeopáticos unisistas, cosméticos, productos higiénicos o perfumes, plaguicidas de uso doméstico, industrial o agrícola, fabricados en el Ecuador o en el exterior deberán contar con el Registro Sanitario, a esto se añade los licores.

Prohibiciones

Existen determinados tipos de mercadería que no se puede importar tales como la ropa usada, zapatos usados, clordano, aldrín, endrín, paratión etílico, neumáticos usados marfil trabajado y manufacturas de marfil, cueros de reptil y la mercadería de las posiciones 9601.90.00 o de la posición 4107.21.00 y 4103.20.00 y otros (anexo 2).

Son de libre importación a consumo todas las mercaderías excepto aquellas que constan en la Resolución N° 145 publicado en el Registro Oficial N° 647 de 23 de agosto de 2002 expedida por el Consejo de Comercio Exterior e Inversiones COMEXI, en la referida resolución se detallan las mercaderías de prohibida importación, como aquélla que siendo de permitida importación, esta sujeta a autorizaciones previa.

Se permite la importación de tractores nuevos, usados y remanufacturados, de igual manera se permite la importación de partes y piezas y accesorios de tractores siempre y cuando sean nuevos o remanufacturados.

Para el caso de las importaciones de tractores y de las partes y piezas y sus accesorios remanufacturados, se deberá presentar la certificación de la verificadora internacional en origen y la marca del producto que se desea importar deberá contar con un representante en el Ecuador que garantice el mantenimiento como los repuestos que eventualmente pueda requerirse Adicionalmente para efectos de poner en riesgo el ecosistema deberá exigirse que este tipo de importaciones cumpla con todas las normas de protección del medio ambiente vigente en el Ecuador.

DERECHOS PARA-ARANCELARIOS CUANTITATIVOS

Son tributos adicionales que se debe pagar por las mercaderías cuando éstas se importan al país, y declarados para su posterior liquidación. A continuación se citan los más importantes.

Tasa de FODINFA

El Fondo para el Desarrollo de la Infancia y la Familia, mediante Ley 92 publicada en Registro Oficial No 934 de 12 de mayo de 1988, está destinado a cubrir el costo de programas de inversión para la atención y cuidado de los niños ecuatorianos. Se cobra el 0.5 % del valor CIF y se exceptúa el pago en las importaciones de materias primas que se utilizan para la elaboración de fármacos para uso humano y veterinario.

Impuesto a los Consumos Especiales (ICE)

El modo de aplicación de este impuesto difiere del anterior en que la base imponible no es directamente el CIF, sino la que se expone a continuación:

Base Imponible ICE = CIF + (%Ad Valorem)CIF + 1.25
(%Fodinfra)CIF, donde %Fodinfra = 0.5 %

$$\text{Valor ICE} = \% \text{ICE} \times \text{Base Imponible ICE}$$

El porcentaje de este impuesto varía dependiendo del tipo de mercadería que se menciona. La siguiente tabla muestra los tipos de mercancías a los que se aplican este derecho para-arancelario con sus respectivas tasas de cobro.

Mercadería	Tarifa %
Cigarrillo rubio	77,25
Cigarrillo negro	18,54
Cerveza	30,90
Bebidas gaseosas	10,30
Alcohol y productos alcohólicos distintos de la cerveza	26,78
Vehículos motorizados de transporte terrestre de hasta 3,5 toneladas de carga	5,15
Aviones, avionetas, helicópteros excepto los que se utilizan para transporte de carga y personas motos acuáticas, tricars, cuadrones, yates y barcos de recreo	10,30

Tabla 5 Tasas de cobro del ICE

Impuesto al Valor Agregado (IVA)

$$\text{Valor IVA} = 12\% \times (\text{CIF})$$

En todos los casos en los cuales se reconozcan exoneraciones de derechos arancelarios o tarifa cero para el impuesto al valor agregado IVA, al momento de la nacionalización se efectuará

obligatoriamente el aforo físico independiente del valor declarado por las mercancías.

Tasa de control

Se aplica cuando la mercadería se despacha bajo los regímenes aduaneros especiales a excepción del depósito industrial, equivalente a US \$ 40.

Tasa de almacenaje

El Ecuador aplica varias tasas por servicios aduaneros, incluyendo las tasas de almacenaje. Las tasas de almacenaje se aplican a las mercancías depositadas en almacenes autorizados por la aduana. La Ley Orgánica de Aduanas dispone que las mercancías importadas sean depositadas en dichos almacenes mientras se llevan a cabo los trámites para su despacho. Las tasas de almacenaje son específicas y se fijan en función del peso de la mercancía. Las tasas de almacenaje aeroportuario oscilan entre siete dólares de los EE.UU. para embarques de menos de 50 kg hasta 750 dólares de los EE.UU. para embarques de más de 40 toneladas por cada siete días (a partir del octavo día, se cobra un recargo diario de medio centavo dólar de los EE.UU. por kg). Para la mercancía almacenada en los puertos, las tasas que se aplican a un contenedor de 40 pies almacenado entre 11 y 20 días se elevan

a 5,26 dólares de los EE.UU. (2,63 dólares de los EE.UU. para un contenedor de 20 pies).

Tasa de Inspección

Se aplicará a los servicios de inspección técnica ante solicitudes de autorización o renovación de depósitos comerciales, industriales, courier, almacenes temporales y otros servicios sujetos a autorización o concesión de acuerdo con la siguiente tabla:

Depósito Comercial	US \$ 300
Depósito Industrial	US \$ 375
Courier	US \$ 180
Almacén temporal	US \$ 300
Otros	US \$ 250

Tabla 6 Tasas de Inspección

Cuota redimible del CORPEI

(Corporación de Promoción de Exportaciones e Inversiones)

1,5 por mil sobre el valor FOB de las exportaciones del sector privado.

0,5 por mil del valor FOB de las exportaciones de petróleo y sus derivados.

0,25 por mil del valor FOB de las importaciones con un mínimo de \$ 5.

Las cuotas redimibles serán entregadas por los exportadores de bienes y servicios al momento de venta de divisas y por los importadores de las mercaderías y servicios al momento del pago de los tributos a las entidades bancarias.

Cuando se tenga US\$ 500 de valores pagados se cambia por certificado redimible al cabo de 10 años.

Privilegios Arancelarios

Se clasifican en tres tipos:

Trato Preferencial por Convenio Internacional (TPCI)

Es un privilegio al que un importador puede acogerse debido a que la mercadería proviene de un país al cual el Ecuador mantiene un convenio bilateral o multilateral; tal es el caso de la ALADI o el Pacto Andino. En este caso la ley establece un porcentaje de exoneración parcial o total de los derechos arancelarios que el importador debe pagar.

Trato Preferencial por Norma General (TPNG)

En este caso la preferencia se da por diversos motivos que al igual que el TPCI se encuentran debidamente tipificados en el código aduanero. Dependiendo del tipo de TPNG se establece la exoneración total o parcial del pago de los derechos arancelarios y para-arancelarios.

Trato Preferencial por Norma Específica (TPNE)

El privilegio al que se acoge aplicando este trato preferencial se basa en la exoneración de la entrega de documentos especificados en el código aduanero.

Regímenes Aduaneros

I. REGIMENES COMUNES

Art. 55 Importaciones a Consumo: La importación a consumo es el régimen aduanero por el cual las mercancías extranjeras son nacionalizadas y puestas a libre disposición para su uso o consumo definitivo.

Art. 56 Exportación a Consumo: La exportación a consumo es el régimen aduanero por el cual las

mercancías, nacionales o nacionalizadas, salen del territorio aduanero, para su uso o consumo definitivo en el exterior.

II. REGIMENES ESPECIALES

Art. 57 Tránsito aduanero: Es un régimen por el cual las mercancías son transportadas bajo control aduanero de una oficina Distrital a otra del país o con destino al exterior.

Art. 58 Importación Temporal con Reexportación en el mismo Estado: Es el régimen suspensivo del pago de impuestos, que permite recibir mercancías extranjeras en el territorio aduanero, para ser utilizados con un fin determinado durante cierto plazo y reexportadas sin modificación alguna, con excepción de la depreciación normal por el uso.

Art. 59 Importación Temporal para Perfeccionamiento Activo: Es el régimen suspensivo del pago de impuestos que permite recibir mercancías extranjeras en el territorio aduanero durante un plazo

determinado para ser reexportadas luego de un proceso de transformación, elaboración o reparación.

Art. 60 Depósito aduanero: Es el régimen suspensivo del pago de impuestos, por el cual las mercancías permanecen almacenadas por un plazo determinado en lugares autorizados y bajo control de la administración aduanera, en espera de su destino exterior.

Los depósitos aduaneros son: comerciales, públicos o privados, e industriales.

En los depósitos comerciales, las mercancías, de propiedad del concesionario o de terceros, permanecen almacenadas, sin transformación alguna.

En los depósitos industriales, las mercancías de propiedad del concesionario, se almacenan para su transformación.

Podrán ser concesionarios de depósito aduanero, las personas jurídicas, públicas o privadas, nacionales o extranjeras legalmente establecidas en el Ecuador.

Art. 61 Almacenes Libres y Especiales: Es el régimen liberatorio que permite, en puertos y aeropuertos internacionales, el almacenamiento y venta a pasajeros que salen del país, de mercancías nacionales o extranjeras, exentas del pago de impuestos.

Bajo el régimen de admisión temporal, podrán habilitarse almacenes especiales de mercancías, destinadas al aprovisionamiento, reparación y mantenimiento de naves, aeronaves y vehículos de transporte terrestre, internacionales.

Art. 62 Exportación Temporal con Reimportación en el mismo Estado: Es el régimen suspensivo del pago de impuestos que permite la salida del territorio aduanero de mercancías nacionales o nacionalizadas para ser utilizadas en el extranjero, durante cierto plazo con un fin determinado; y reimportadas sin modificación alguna, con excepción de la depreciación normal por el uso.

Art. 63 Exportación Temporal para Perfeccionamiento

Pasivo: Es el régimen suspensivo del pago de impuestos que permite la salida del territorio aduanero de mercancías nacionales o nacionalizadas, durante cierto plazo, para ser reimportadas luego de un proceso de transformación, elaboración o reparación.

Art. 64 Devolución Condicionada: Es el régimen por el cual se permite obtener la devolución total o parcial de los impuestos pagados por la importación de las mercancías que se exporten dentro de los plazos que señal el Reglamento de esta Ley, en los siguientes casos:

Las sometidas en el país a un proceso de transformación.

Las incorporadas a la mercancía; y,

Los envases o acondicionamientos.

Art. 65 Reposición con Franquicia Arancelaria: Es el régimen compensatorio por el cual se permite importar mercancías idénticas o equivalentes, sin el

pago de impuestos, en reposición de las importadas a consumo, que retornan al exterior después de haber sido sometidas a un proceso de transformación en el país, o se utilizaron para producir, acondicionar o envasar mercancías que se exportaron.

Art. 66 Zona Franca: Es el régimen liberatorio que por principio de extraterritorialidad, permite el ingreso de mercancías, libre de pago de impuestos, a espacios autorizados y delimitados del territorio nacional. Las mercancías ingresadas a zona franca no están sujetas al control de la administración aduanera.

Las zonas francas son comerciales o industriales:

Comerciales son aquellas en las cuales las mercancías admitidas permanecen sin transformación alguna, en espera de su destino ulterior; y,

Industriales son aquellas en que las mercancías se admiten para someterlas a operaciones autorizadas de transformación y perfeccionamiento, en espera de su destino ulterior.

Este régimen se regulará por las normas especiales contenidas en la Ley de Zonas Francas.

Art. 67 Régimen de Maquila: La Maquila es el régimen suspensivo del pago de impuestos, que permite el ingreso de mercancías por un plazo determinado, para luego de un proceso de transformación ser reexportadas.

El ingreso de las mercancías y la reexportación de los productos terminados, así como el tratamiento de los desperdicios es competencia del Gerente Distrital.

Art. 68 Ferias Internacionales: Es un régimen especial aduanero por el cual se autoriza el ingreso de mercancías de permitida importación con suspensión del pago de tributos, por un tiempo determinado, destinadas a exhibición en recintos previamente autorizados, así como de mercancías importadas a consumo con fines de degustación, promoción y decoración, libre del pago de impuestos, previo el cumplimiento de los requisitos y formalidades señaladas en el Reglamento.

III. REGIMEN PARTICULAR O DE EXCEPCION

Art. 69 Tráfico Postal Internacional y Correos Rápidos:

La importación o exportación a consumo de los envíos o paquetes postales, cuyo valor CIF o FOB, en su caso, no exceda del límite que se establece en el Reglamento de esta Ley, transportados por cualquier clase de correo, incluidos los denominados correos rápidos, se despacharán por la aduana mediante formalidades simplificadas. Los envíos o paquetes que excedan el límite establecido, se sujetarán a las normas aduaneras generales.

Art. 70 Tráfico Fronterizo: Es el régimen que, de acuerdo a los compromisos internacionales, permite el intercambio de mercancías destinadas al uso o consumo doméstico entre las poblaciones fronterizas, libre de formalidades y del pago de impuestos aduaneros. La Corporación Aduanera Ecuatoriana, de acuerdo a los compromisos internacionales, delimitará el área del territorio nacional en la que se aplicará este régimen.

Art. 71 Zona de Libre Comercio: Es el régimen que permite el intercambio de mercancías, libre del pago de impuestos aduaneros, entre países integrantes de una zona de territorio delimitado y de mercancías originarias de los mismos, sujeto a las formalidades aduaneras previstas en los respectivos convenios internacionales.

Declaración Aduanera

De acuerdo a lo que estipula el art. 43 de la LOA: ¹⁶

El propietario, consignatario o consignante, en su caso, personalmente, o a través de su representante legal o de un Agente de Aduana, presentará la declaración de las mercancías provenientes del extranjero o con destino a él, en la que solicitará el régimen aduanero al que se someterán.

En las importaciones, la declaración se presentará en la aduana de destino, desde siete días antes hasta quince días hábiles siguientes a la llegada de las mercancías.

¹⁶ Ley Orgánica de Aduanas, publicada en el R.O. No. 359 del 13 de julio de 1998; Reglamento General a la LOA, publicada en el R.O. No. 158 del 7 de septiembre del 2000; Leyes y Reglamentos Conexos, y Reformas a la Ley en los años 2000, 2001, 2002 y 2003.

En las exportaciones, la declaración se presentará en la aduana de salida, desde siete días antes hasta quince días hábiles siguientes al ingreso de las mercancías a la zona primaria aduanera.

En la importación y en la exportación a consumo, la declaración comprenderá la autoliquidación de los impuestos correspondientes.

La declaración aduanera es única y de carácter personal, consecuentemente se formulará independientemente por cada importador, exportador o pasajero, personalmente o a través de un Agente de Aduanas debidamente autorizado para el efecto y se presentará bajo el formato determinado por la Corporación Aduanera Ecuatoriana, sea a través de medios electrónicos o documentales de acuerdo a la normatividad que para el efecto dicte dicha Corporación.

Sólo podrá presentarse la declaración aduanera en el Distrito de Ingreso de las mercancías sin perjuicio de que posteriormente se solicite un cambio de régimen en el destino final luego de producido un tránsito aduanero.

Podrá presentarse una sola declaración, cualquiera que sea la cantidad de conocimientos de embarque, guías aéreas o cartas de porte,

siempre que concurren simultáneamente el mismo medio de transporte, viaje y régimen aduanero.

Términos para solicitar mercancía a despacho:

Desde 7 días antes de llegada de la mercancía.

Hasta 15 días hábiles siguientes a la llegada de la mercancía.

Dentro del plazo concedido a la mercancía sometida a los Regímenes suspensivos y liberatorios.

Tratándose de mercancía en abandono tácito y expreso, hasta antes del remate o adjudicación.

Sobre mercancía de importación prohibida y restringida:

Se puede importar todo tipo de mercancía, con excepción de aquellas que se encuentren prohibidas de acuerdo al COMEXI (Consejo de Comercio Exterior e Inversiones)¹⁷.

La mercancía restringida de acuerdo al COMEXI, puede ser objeto de importación siempre que cumplan con los requisitos exigidos por la normatividad legal específica para su internamiento al país.

¹⁷ Véase Ley de Comercio Exterior e Inversiones "LEXI", art. 10 para obtener detalles.

Requisitos que debe cumplir la mercancía amparada en una declaración:

Debe pertenecer a un solo consignatario.

Encontrarse depositada en un almacén temporal o depósito aduanero autorizado; salvo los casos de mercancía que se acoja a la desaduanización directa, y aquellos casos en los cuales la naturaleza de la mercancía u otras circunstancias excepcionales, no lo permitan.

Estar declarada en un solo Manifiesto de Carga.

Puede incluirse en una sola declaración, mercancía arribada en el mismo viaje del medio de transporte, que se encuentre manifestada en dos o más Documentos de Transporte, siempre y cuando estén destinadas al mismo consignatario y al mismo puerto de destino, inclusive, aquella consecuencia de transferencia.

En este último caso, deberá adjuntarse copia de los comprobantes de pago que acrediten dichas transferencias.

Presentada la declaración, la aduana verificará que ésta contenga todos los datos que contempla el formulario respectivo, los cotejará con los de los documentos de acompañamiento y comprobará la exactitud de la autoliquidación. Si no hay observaciones, se aceptará la declaración fechándola y otorgándole un número de validación para

el pago de los tributos. Efectuando el pago, se sujetará al mecanismo de selección aleatoria; si este se activa, se procederá al aforo físico; caso contrario sin más trámite se entregarán las mercancías.

En caso de existir observaciones a la declaración, se devolverá al declarante para que la corrija dentro de los tres días hábiles siguientes. Corregida la declaración, la aduana la aceptará.

Si el declarante no acepta las observaciones, la declaración se considerará firme, procederá al pago de los tributos y se someterá obligatoriamente al aforo físico de las mercancías.

La declaración aceptada por la aduana es definitiva, y no podrá ser enmendada ni anulada.

El declarante es personal y pecuniariamente responsable por la exactitud de los datos consignados en la declaración. En el caso de personas jurídicas, la responsabilidad recae en la persona de su representante legal (Art. 45 LOA).

A continuación se describirán los principales documentos que conforman la declaración aduanera.

Orden de Trabajo

Toda empresa que se dedique formalmente al despacho aduanero debe brindar a sus clientes un compromiso documental del trámite que va a realizar, así como el modo y condiciones en las que va a trabajar. Este documento también sirve como un respaldo y registro de los trámites que los empleados de la empresa realizan; permitiendo incluso sacar estadísticas por día laboral o por empleado de cuántos trámites se completan satisfactoriamente.

La forma y políticas de creación de una orden de trabajo varían dependiendo del modo de trabajo de la despachadora, pero como elemento en común se tiene que todas necesitan identificarse de alguna manera. Para la presente tesis, el sistema implementado tendrá el siguiente esquema de identificación:

Las órdenes de trabajo se identificarán con las letras "FA".

Se establece un tipo de trámite: A para los trámites en los que interviene la participación de la aduana (que es el que concita el mayor interés, ya que entre estos tipos de trámite está la declaración aduanera); y B o C para trámites específicos en los que no se necesita intervención de la aduana.

Un número secuencial de 4 dígitos que permitirá su identificación a efectos de determinar la fecha de presentación de la declaración. Este

número es único para cada despacho y el declarante deberá reiniciar desde 1 la numeración de despachos cada año.

Documento Único de Importación (DUI)

Era el único documento oficial que utilizan los importadores en la declaración de importación tanto a los Bancos Corresponsales del Banco Central previa al embarque de las mercancías para el otorgamiento del visto bueno, como en la declaración aduanera para la desaduanización de las mismas. Este documento entró en vigencia en todo el país a partir del 4 de septiembre de 1995, en sustitución de los documentos de ese entonces (Declaración de Importación y Documento Aduanero de Importación, DAI).

El DUI consta de tres formularios: A, B y C; para su llenado se utilizan los códigos aduaneros estipulados por la CAE.

En el formulario A se consigna la información general respecto de la importación que se declara, como por ejemplo: las oficinas del banco corresponsal del Banco Central y la Aduana, ante las cuales se presentan las declaraciones; embarcador o remitente de las mercancías; importador o consignatario de las mismas; declarante; transportista, etc. En este formulario también se incluyen los datos

globales para la determinación de la base imponible de los impuestos aduaneros y lo referente a los bultos que contienen las mercancías y sus pesos. Así mismo, este formulario sirve para declarar individualmente las mercancías que se clasifiquen en una sola subpartida arancelaria. Al final del formulario A, se incluyen los campos para las firmas del importador, declarante y vistaforador.

En el formulario B, en su primera parte se repite cierta información general que se consigna en el formulario A. El propósito fundamental del formulario B es permitir declarar más de una o a partir de la segunda subpartida arancelaria, aparte de la que se declara en el formulario A. Es decir, el formulario B solamente se presentará cuando se declare más de una subpartida arancelaria, por tanto, se presentarán cuantos formularios B sean necesarios.

De igual forma que en el formulario B, en el C se repite cierta información general que se incluye en el formulario A. El formulario C constituye básicamente la autoliquidación de los tributos por la importación que realiza el declarante ante la aduana y la liquidación de los mismos que por su parte realiza la aduana. Se llenará y presentará obligatoriamente, cualesquiera sea el régimen que se solicita y aunque no se declaren tributos.

Formulario Único de Exportación (FUE)

Era el único documento oficial para la exportación y salida de mercadería del Ecuador, y de la misma manera que ocurre con el DUI, sirve para la declaración aduanera en las exportaciones.

Declaración Aduanera Única (DAU)

Se estableció con la finalidad de unificar los documentos declaratorios existentes hasta el momento, el DUI y el FUE en una sola declaración que mantuviera como estándares las características más importantes de ambos documentos. La mayoría de los campos son los mismos que en el DUI y en el FUE, con la excepción de que por ejemplo, no existe la sección de Garantías que se encuentra en el DUI; y se resumen los datos de las series de partidas arancelarias que se declaran en los formularios A y B del DAU, con el fin de dar cabida a más series; en otras palabras, en un formulario B del DUI cabe la información de hasta 4 series, mientras que en un formulario B del DAU caben hasta 19 series de partidas arancelarias.

Declaración Andina al Valor (DAV)

Fue creada como un documento adicional válido para la declaración aduanera, que debe estar acompañado por sus respectivos DUIs o

FUEs, según sea el caso. Su aprobación y actualización estuvo a cargo de la Comunidad Andina de Naciones (CAN).

La DAV será remitida por encargo, por el Agente Afianzado de Aduana, no obstante, la responsabilidad de su contenido, será del importador. El importador podrá solicitar su código de usuario y clave de acceso, para consultar las DAV, que con su autorización han sido transmitidas por el Agente Afianzado de Aduana.

Documentos de Acompañamiento

Según consta en el art. 44 de la LOA, a la declaración aduanera se acompañarán los siguientes documentos:

Original o copia negociable del conocimiento de embarque, guía aérea o carta de porte;

Factura comercial y póliza de seguro expedida, que servirán de base para la declaración aduanera.

Certificado de inspección en origen o procedencia cuando sea el caso;

Certificado de origen cuando proceda;

Visto Bueno del Banco Central del Ecuador o de sus corresponsales, previo al embarque de las mercancías en las exportaciones a consumo; y,

Los demás exigibles por regulaciones expedidas por la Junta Monetaria.

Intercambio Electrónico de Datos con la C.A.E.

Se establece para el procedimiento de despacho, el uso de la Declaración Aduanera Única electrónica (DAU electrónica), la que se empleará para transmitir a la CAE los datos de los Documentos Únicos de Importación (DUI) y Formularios Únicos de Exportación (FUE). Este formato, sustituye al DUI y al FUE y su implantación constituye el primer paso para el cumplimiento de una Ley Supranacional, cual es la adopción de la Declaración Aduanera Única (DAU), cuya aprobación y actualización estará a cargo de la Comunidad Andina de Naciones (CAN).¹⁸

Se define como Documento Impreso al formulario de papel que contiene datos en casillas preestablecidas, y que es previamente foliado, firmado y sellado por el Agente de Aduanas para su entrega y aceptación por la CAE.

Análogamente, se define como documento electrónico, al "formato de datos electrónico", que contiene datos en los campos, preestablecidos,

¹⁸ Tomado del Manual de Procedimientos Sistema de Intercambio Electrónico de Datos "Cero Papeles". 2003. Carlos Ramos Díaz

y que es “firmado electrónicamente” por el Agente de Aduana para su transferencia y aceptación por el sistema informático de la CAE.

El Visto Bueno del Banco Central, previo al embarque de las mercancías de importación a consumo, señalado en el Art. 44, literal e) de la LOA, será obtenido electrónicamente por el importador y el número que esta institución le asigne, será ingresado en la DAU.

La transmisión de datos para declaraciones cuyo monto sea inferior a US\$ 2,000.00, podrá realizarse por los propios importadores, siempre y cuando se cumpla con los formatos y requisitos para el efecto.

El número de refrendo, servirá para que el Agente de Aduana, consulte e imprima la DAU, y cancele los tributos en los bancos autorizados.

Cuando el Agente de Aduana tenga conocimiento que algún bulto no ha sido embarcado o habiendo sido manifestado y no desembarcado, podrá solicitar el reconocimiento físico de la mercancía.

Documentos y archivos creados

Los distintos entes que participan en el proceso aduanero tienen que responder ante la Corporación Aduanera Ecuatoriana. En pro de la

automatización de los procedimientos aduaneros y aplicar Tecnología de Información, algunos de ellos deben enviar electrónicamente la información de los procesos que realizan. A continuación una descripción de las entidades que deben adoptar esta metodología y los nombres de los archivos que envían.

Entidad	Archivo	Comentario
Navieras	CUSCAR.ZIP	Detalle del B/L master.
Consolidadoras	CARDAT.ZIP	Detalle de los B/Ls hijos.
Almacenes	CARDAT.ZIP	-
Despachadoras	DAUDAT.ZIP	Compendio de la D.A.U.

Tabla 7 Archivos enviados por las entidades involucradas.

Estos archivos son archivos comprimidos en formato ZIP que contienen un conjunto de archivos de texto o XML dependiendo de a qué entidad le pertenece. Por ejemplo, el archivo DAUDAT.ZIP para que emiten los Couriers de Aduana debe contener archivos en formato XML; en cambio que el mismo DAUDAT.ZIP para los despachadores de aduanas, objeto de estudio para la presente tesis, debe contener archivos de texto que contengan líneas de campos separados con pipes (esto es, el carácter "|"). A continuación un fragmento de una

de las líneas de unos de los archivos contenidos dentro del archivo .ZIP:

109|2005|5656|10|1|ECUIO|301|Matriz

Cada uno de estos archivos cumple una función específica. El archivo DAUDAT.ZIP es el archivo que va a ser motivo de interés, ya que la presente tesis va dirigida al despacho de mercadería mediante la declaración aduanera; labor realizada por los despachadores de aduana. Este archivo comprimido, según estándar creado por el SEND (Sistema de Intercambio Electrónico de Datos aduaneros), de manera general consta de los siguientes archivos¹⁹:

DAUHDR01	Datos generales de la declaración aduanera de importación. Secciones A, B, D, E, F, G, H, I del DUI. Datos generales de la declaración para exportación.
DAUDET01	Datos de series o subpartidas nacionales. Sección J del DUI. Datos de series o subpartidas del FUE.
DAUDET02	Ampliación de datos de series o subpartidas nacionales (vehículos y otros que determine la CAE).
DAUDOCAS	Datos de los documentos de acompañamiento de la declaración

19 Según el punto 7 de las «Normas en relación a formatos, mensajes de datos e intercambio de datos» del Manual de Procedimientos del Sistema de Intercambio Electrónico de Datos «Cero Papeles». Carlos Ramos Díaz. 2003

DAUOBSER	Datos de las observaciones sobre datos generales, de series, etc.
DAUREGAP	Datos de las declaraciones precedentes, relacionadas con la DAU.
DAUCONTE	Datos de los contenedores asociados con la DAU.
DAVHDR01	Datos Generales de la Declaración Andina al Valor
DAVDET01	Datos de factura de la Declaración Andina al Valor
DAVDET02	Datos de ítem de la Declaración Andina al Valor
ENDCTROL	Datos de control de envío.

Tabla 8 Formatos de datos de acuerdo al SEND

Este archivo comprimido se envía a la Aduana vía correo electrónico con SSL (Secure Socket Layer) implementado, y la Aduana en respuesta envía también un archivo .ZIP de respuesta.

→	CUSCAR.ZIP
←	CUSRES.ZIP
→	CARDAT.ZIP
←	CARRES.ZIP
→	DAUDAT.ZIP
←	DAURES.ZIP

Tabla 9 Archivos de Respuesta emitidos por la Aduana

El "Mensaje de Datos" (el archivo .ZIP) que remite el Agente de Aduanas a la CAE, contendrá datos específicos del importador, agente, embarcador, certificados, medio de transporte, determinación de la base imponible, garantías y la mercancía que ingresa o sale del país.

El "Mensaje de Datos" de respuesta, que remite la CAE al Agente de Aduana, contendrá la identificación del "Mensaje de Datos" previamente enviado a la CAE, y, de ser el caso, una lista detallada de los errores detectados en el proceso de validación.

El contenido de los archivos de respuesta recibidos indica, en el caso del archivo CUSRES.ZIP por ejemplo, qué B/Ls tienen errores, y cuáles no. Las navieras envían a la Aduana el número del manifiesto seguido el código 09, si se trata de la primera declaración del trámite, y para una modificación a la declaración el código 21. Así por ejemplo, como se muestra en la figura, es enviado el código 00-09 indicando que se trata el manifiesto 00 y por ser el envío por primera vez, el código 09. Un error en la declaración de este envío implica que la Aduana responda con un error. A continuación una tabla con los tipos de envío existentes tipificados por la Aduana.

En el caso de rechazo, se precisarán los hallazgos para facilitar la corrección de los errores y acelerar el posterior reenvío. Los formatos

de datos se encuentran en el anexo 2. Con relación al casillero que señala la obligación de envío en el citado anexo, la clasificación "M" indica que es mandatario u obligatorio el envío de ese campo, en tanto que "C" significa que el envío está condicionado a la existencia de la situación y no a que el envío es opcional o potestativo.

Tipo Transmisión	Descripción	Tipo Envío	Descripción	Responsable
10 a 74	Transmisión del manifiesto electrónico	01	CANCELACIÓN: Para cancelar todo el DAU.	No activado
		09	ORIGINAL: Cuando se envía por primera vez el DAU.	Agente de Aduana
		17	CANCELACIÓN DETALLE: Cuando se desea cancelar una Serie o Ítem.	No activado
		20	REEMPLAZO DE CABECERA: Cuando se desea reemplazar datos generales del DAU.	No activado
		21	REEMPLAZO DE DETALLE: Cuando se desea reemplazar datos de una Serie o Ítem.	No activado
		53	PRUEBA: Cuando el envío se realiza durante el período de prueba.	Agente de Aduana

Tabla 10 Tipos de envío.

Corregido ese error, se vuelve a enviar el "Mensaje de datos" y la información aceptada es guardada en los registros de la Aduana no sin antes responder satisfactoriamente a la naviera. Acto seguido se envía una corrección a la información guardada, por lo que se envía un código 21. La información es modificada y se envía una respuesta satisfactoria.

Los archivos deberán ser empaquetados con los programas PKUNZIP v.4 o superior, WINZIP 7 o superior, o programas compatibles. El archivo resultante será el DAUDAT.ZIP que será enviado a la CAE por el Agente Afianzado de Aduana.

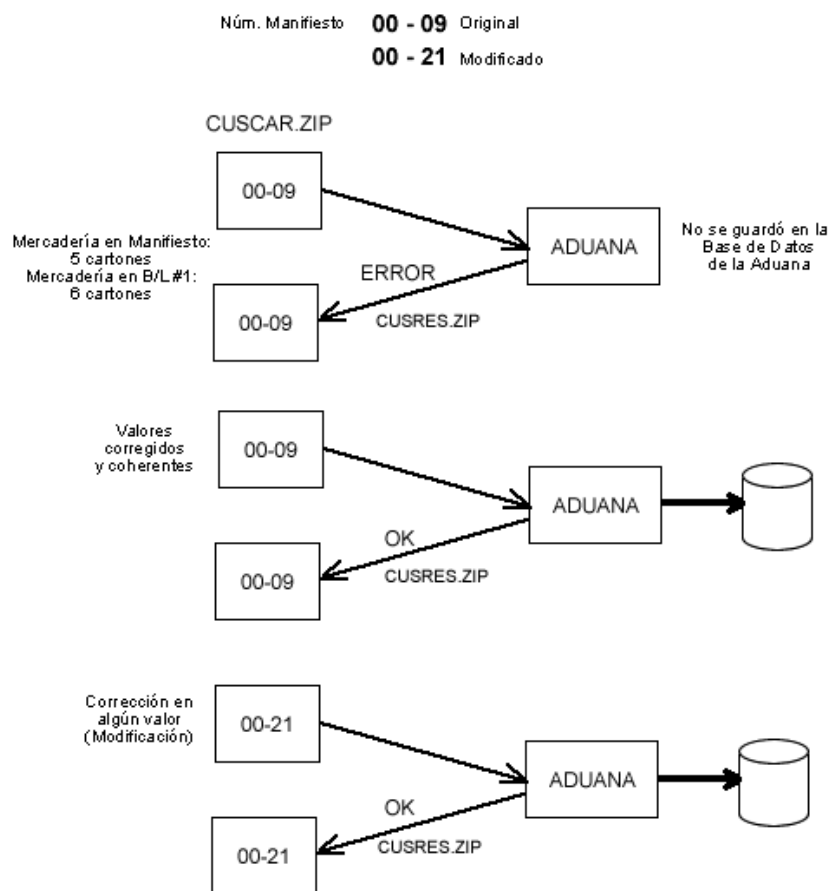


Figura 6 Envío y recepción de Mensajes de Datos y de Respuesta.

La respuesta de la CAE constará a su vez de los siguientes formatos:

RESMENSJ: Datos de detalle de las incidencias detectadas en la validación.

RESACEPT: Datos para la certificación digital del mensaje.

RESCTROL: Datos de control de envío.

Y de los siguientes archivos:

RESMENSJ.TXT

RESACEPT.TXT

RESCTROL.TXT

Estos archivos serán empaquetados con los programas PKZIP o WINZIP y el archivo resultante será el DAURES.ZIP, que será enviado a los Agentes Afianzados de Aduana por la CAE. El anexo 3 describe los campos correspondientes a los archivos que conforman el mensaje de respuesta.

Resumen del Proceso de despacho

La figura siguiente muestra un diagrama completo del proceso de despacho. Comenzando desde el momento en el que el Agente de Aduana envía electrónicamente la información de la mercancía a despachar como un Mensaje de Datos, la cual puede estar completa y coherente o errónea; los sistemas de la CAE validan la información recibida y envían el Mensaje de Respuesta apropiado.

Luego, debido a las condiciones mencionadas oportunamente, se activa o no el mecanismo de selección aleatoria; el cual puede aceptar directamente o requerir un aforo físico, ya sea por personal de aduana o por medio de una verificadora; en cuyo caso se recibe la aceptación y se entrega físicamente la DAU junto con los documentos de

acompañamiento correspondientes en la ventanilla. Estos datos son verificados y se escoge el mecanismo de aforo, el cual puede o no ser satisfactorio.

De no haber alguna discrepancia mayor al 10% del FOB entre la parte documentaria y la parte física se procede a recibir la autorización de pago en el Banco; para finalmente retirar la mercadería del almacén o en el sitio en el que se encuentre dentro de la Zona Primaria.

En el caso de encontrar anomalías como resultado del proceso de aforo, se consigna el caso a la autoridad competente en el Departamento Legal y se dará el trámite pertinente.

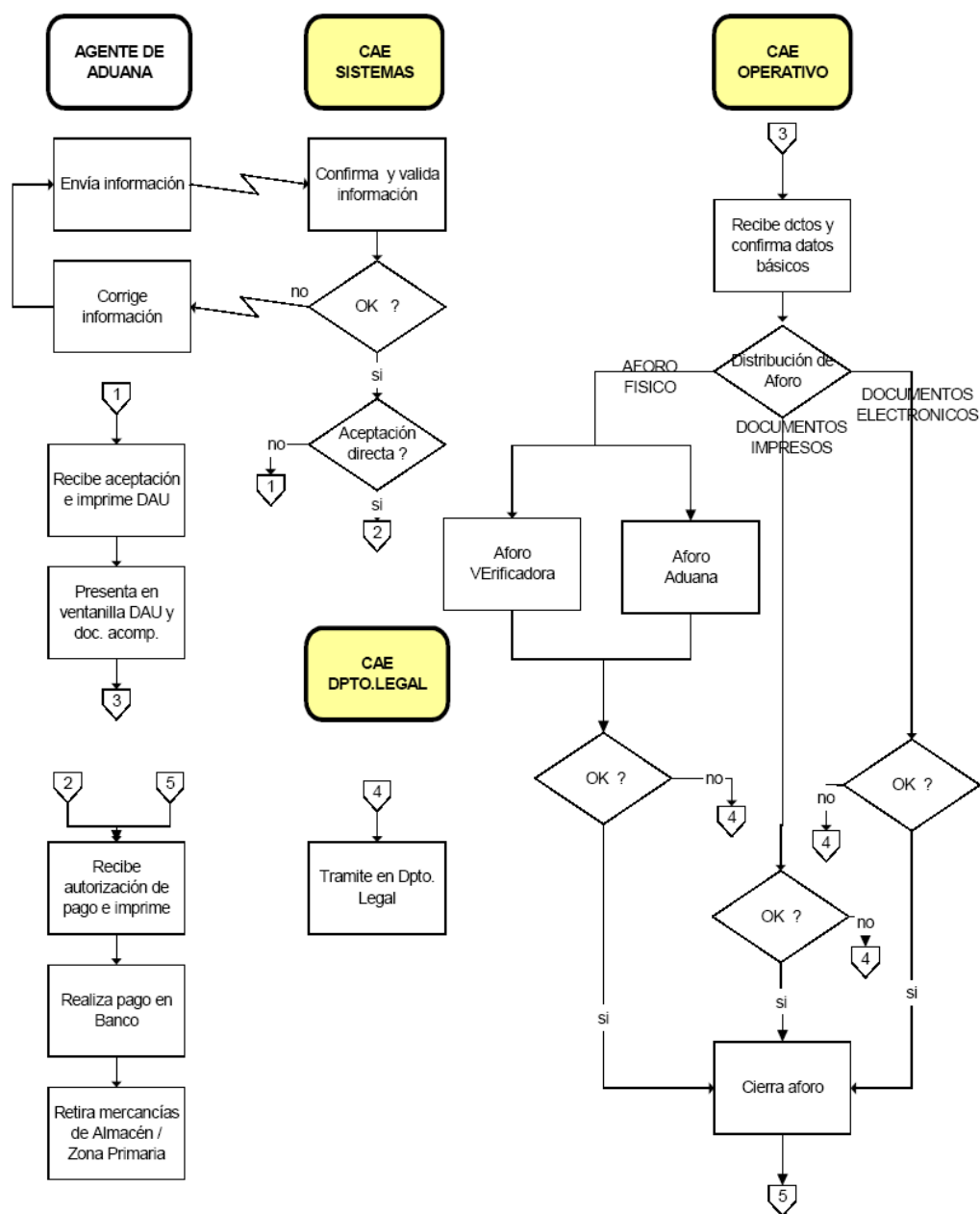


Figura 7 Flujo de trabajo del proceso de nacionalización de mercancías.

Anexo 1: CÓDIGOS DE RÉGIMENES ADUANEROS

Código	Descripción de Régimen	Tipo Régimen	Tipo Producto Entrada
10	Importación a consumo	Consumo	--
11	Importación a consumo para Reposición en Franquicia Arancelaria	Consumo	IN
12	Importación a consumo bajo Reposición en Franquicia Arancelaria	Especial	IN
15	Orden de Embarque	Consumo	--
20	Importación Temporal para Reexportación en el Mismo Estado	Especial	PF
21	Importación Temporal para Perfeccionamiento Activo	Especial	IN
24	Importación Ferias Internacionales	Especial	PF
31	Reimportación de Mercancías Exportadas Temporalmente	Especial	--
32	Reimportación de Mercancías Exportadas Temporalmente sin Pago de Impuesto	Especial	--
39	Solicitud Previa a Régimen de Depósito	Especial	--
40	Exportación a consumo	Consumo	--
50	Exportación Temporal para Reimportación en el Mismo Estado	Especial	--
51	Exportación Temporal para Perfeccionamiento Pasivo	Especial	PF
53	Exportación con Devolución condicionada de Tributos (Drawback)	Especial	--
60	Reexportación de Mercancías en el Mismo Estado	Consumo	--
61	Reexportación de Mercancías de Perfeccionamiento Activo	Consumo	--
70	Importación Depósito Comercial Público	Especial	PF
71	Importación Depósito Comercial Privado	Especial	PF
72	Importación Depósito Industrial	Especial	IN
73	Importación Almacén Libre (In Bond o Duty Free)	Especial	PF
74	Importación Maquila	Especial	IN
80	Transito Aduanero	Especial	--

81	Transbordo	Especial	--
82	Guía de Movilización	Especial	--
83	Reembarque	Consumo	--
84	Transferencia de Insumos/No sujetos a Cambio de Estado; Régimen Especial	Especial	--
85	Transferencia de Productos Compensadores; Régimen Especial	Especial	--
86	Destrucción de Sobrantes; Régimen Especial	Especial	--
87	Regularización por pérdida o destrucción; Régimen Especial	Especial	--
88	Venta en Almacenes Libres y Especiales	Especial	--
90	Importación Zona Franca	Especial	IN
91	Tráfico Postal Internacional y Courier	Particular	--
92	Tráfico Fronterizo	Consumo	--
93	Zona de Libre Comercio	Especial	--

IN = Insumos. Corresponde esta denominación a las materias primas, bienes intermedios y bienes semielaborados, y productos terminados, que serán objeto de transformación, elaboración, reparación o incorporación en un producto final.

PF = Productos Finales. Denominación utilizada para los productos no sujetos a cambio de estado, es decir, que serán objeto de conservación, custodia, venta o exhibición en un recinto autorizado, sin sufrir transformación o modificación alguna.

Anexo 2: FORMATOS DE DATOS DE ENVÍO DEL AGENTE**AFIANZADO A LA CAE**

a) DATOS GENERALES DE LA DECLARACIÓN ADUANERA – DAUHDR01

Nº	CAMPO	TIPO	TAMAÑO	DESCRIPCIÓN	DUI	OBS.
1	COD_ADUAN	CARACTER	3	CÓDIGO DE LA ADUANA DE PRESENTACIÓN	A 2	M
2	AÑO_ORDEN	CARACTER	4	AÑO DE LA ORDEN INTERNA DEL AGENTE	A 1	M
3	NUME_ORDEN	CARACTER	6	NÚMERO DE LA ORDEN INTERNA DEL AGENTE	A 1	M
4	CODI_REGI	CARACTER	2	CÓDIGO DE RÉGIMEN (10)	A 3	M
5	TIPO_DESPA	CARACTER	1	TIPO DE DESPACHO 0: NORMAL 1: DESPACHO ANTICIPADO 2: DESPACHO DE ENVÍO DE URGENCIA 3: DESPACHO DE ENVÍO DE SOCORRO	---	M
6	BAN_CIUADAD	CARACTER	3	CÓDIGO DE CIUDAD DEL BANCO QUE AUTORIZA	A 4	M
7	CODI_BANCO	CARACTER	3	CÓDIGO DEL BANCO QUE AUTORIZA	A 5	M
8	UBI_BANCO	CARACTER	15	UBICACIÓN DE LA OFICINA DEL BANCO QUE AUTORIZA	A 6	M
9	FECH_BANCO	NUMÉRICO	8	FECHA DE APROBACIÓN DE IMPORTACIÓN DEL BANCO	A 7	M
10	NOMB_IMPORTOR	CARACTER	80	NOMBRE DEL IMPORTADOR	B 8	M
11	DIRE_IMPORTOR	CARACTER	80	DIRECCIÓN DEL IMPORTADOR	B 11	M
12	TIPO_DOCUM	CARACTER	1	CÓDIGO DE DOCUMENTO DE IDENTIFICACIÓN DEL IMPORTADOR	B 12	M
13	NUME_DOCUM	CARACTER	13	NÚMERO DEL DOCUMENTO DE	B	M

				IDENTIFICACIÓN DEL IMPORTADOR	12	
14	SECTOR	CARACTER	2	CÓDIGO AL QUE PERTENECE EL IMPORTADOR O CONSIGNATARIO	B 13	M
15	COD_CIIU	CARACTER	4	CÓDIGO DE CLASIFICACIÓN INDUSTRIAL, INTERNACIONAL UNIFORME	B 16	M
16	FORM_PAGOX	CARACTER	2	CÓDIGO DE LA FORMA DE PAGO DE LA TRANSACCIÓN	B 18	M
17	COD_MONED	CARACTER	3	CÓDIGO DE LA MONEDA DE TRANSACCIÓN	B 19	M
18	NOMB_REMIT	CARACTER	20	NOMBRE DEL REMITENTE	D 20	M
19	PUER_EMBAR	CARACTER	5	CÓDIGO DEL PUERTO DE EMBARQUE	D 21	M
20	NUME_FACTU	CARACTER	20	NÚMERO DE FACTURA COMERCIAL	---	M
21	FECH_FACTU	NUMÉRICO	8	FECHA DE FACTURA COMERCIAL	D 22	M
22	FECH_CARCR	NUMÉRICO	8	FECHA DE LA CARTA DE CRÉDITO	D 23	C
23	DIRE_REMIT	CARACTER	20	DIRECCIÓN DEL REMITENTE	D 24	M
24	PAÍS_PROCE	CARACTER	2	PAÍS DE PROCEDENCIA DE LA MERCANCÍA	D 25	M
25	BENEF_GIRO	CARACTER	20	NOMBRE DEL BENEFICIARIO DEL GIRO EN EL EXTERIOR	D 26	C
26	NUME_CERTI	CARACTER	18	NÚMERO DE CERTIFICADO DE INSPECCIÓN	E 27	C
27	CEXO_CERT	CARACTER	2	CÓDIGO DE EXONERACIÓN DEL CERTIFICADO DE INSPECCIÓN	E 28	C
28	CERT_ORIGE	CARACTER	10	NÚMERO DE CERTIFICADO DE ORIGEN	E 29	C
29	FECH_ORIGE	NUMÉRICO	8.0	FECHA DEL CERTIFICADO DE ORIGEN	E 29	C
30	CERT_OTRO	CARACTER	10	NÚMERO DE OTRO CERTIFICADO	E 30	C
31	NUME_DEPAR	CARACTER	6	NÚMERO DE DESPACHO PARCIAL	E 31	C

32	VÍA_TRANSP	NUMÉRICO	1.0	CÓDIGO DE LA VÍA DE TRANSPORTE	F 32	M
33	FECH_EMBAR	NUMÉRICO	8.0	FECHA DE EMBARQUE DE LA MERCANCÍA	F 33	M
34	FECH_LLEGA	NUMÉRICO	8.0	FECHA DE LLEGADA DE LA MERCANCÍA	F 34	M
35	TIPO_CARGA	NUMÉRICO	2.0	TIPO DE CARGA	F 35	M
36	BAND_NAVES	CARACTER	2.0	CÓDIGO DEL PAÍS AL QUE PERTENECE EL MEDIO DE TRANSPORTE	F 36	M
37	CADU_MANIF	CARACTER	3	CÓDIGO DE LA ADUANA DEL MANIFIESTO	---	M
38	AÑO_MANIF	CARACTER	4	AÑO DEL MANIFIESTO	F 37	M
39	TIPO_MANIF	CARACTER	2	01: MANIFIESTO MARÍTIMO 02: MANIFIESTO AÉREO 03: MANIFIESTO TERRESTRE	---	M
40	NUME_MANIF	NUMÉRICO	6.0	NÚMERO DEL MANIFIESTO	F 37	M
41	EMPR_TRANS	CARACTER	4	CÓDIGO DE LA EMPRESA DE TRANSPORTE	F 38	M
42	AGEN_TRANS	CARACTER	4	CÓDIGO DE LA AGENCIA DE TRANSPORTE	F 39	M
43	CONO_EMBAR	CARACTER	25	NÚMERO DEL DOCUMENTO DE TRANSPORTE	F 41	M
44	CLIN_TRANS	CARACTER	4	CÓDIGO DE EMPRESA DE TRANSPORTE DE SALIDA	G 42	N/A
45	DVEH_TRANS	CARACTER	25	NOMBRE DE MATRÍCULA DE TRANSPORTE DE SALIDA	G 43	N/A
46	CODI_ALMA	CARACTER	4	CÓDIGO DE TERMINAL DE ALMACENAMIENTO	G 47	M
47	CODI_DEPO	CARACTER	4	CÓDIGO DEL DEPÓSITO ADUANERO AUTORIZADO	G 48	C
48	CADUTRASAL	CARACTER	3	CÓDIGO DE ADUANA DE SALIDA	G 49	N/A
49	CADUDEST	CARACTER	3	CÓDIGO DE ADUANA DE DESTINO	G 50	N/A
50	VÍA_TRADES	NUMÉRICO	1	VÍA DE TRANSPORTE DE SALIDA	---	N/A
				TIPO DE DESTINO		

51	TDESTINO	CARACTER	1	1: EN EL PAÍS 2: EN EL EXTRANJERO	---	N/A
52	CPAÍS_DEST	CARACTER	2.0	PAÍS DE DESTINO	G 51	N/A
53	TFOB_DOLPO	NUMÉRICO	15.2	VALOR FOB	H 53	M
54	CODI_MOFLE	NUMÉRICO	3.0	CÓDIGO DE MONEDA DEL FLETE	---	M
55	TFLE_MONOR	NUMÉRICO	15.2	MONTO DEL FLETE EN MONEDA ORIGINAL	---	M
56	TFLE_DÓLAR	NUMÉRICO	15.2	VALOR DEL FLETE EN DÓLARES	H 55	M
57	CANT_SERIE	NUMÉRICO	4.0	CANTIDAD DE SERIES	H 56	M
58	TSEG_DÓLAR	NUMÉRICO	15.2	VALOR DEL SEGURO	H 57	M
59	TPESO_NETO	NUMÉRICO	14.2	TOTAL PESO NETO	H 58	M
60	TAJUSTES	NUMÉRICO	14.2	TOTAL AJUSTES	---	C
61	VAL_ADUANA	NUMÉRICO	15.2	TOTAL VALOR ADUANA	H 59	M
62	TPESO_BRUT	NUMÉRICO	14.2	TOTAL PESO BRUTO	H 60	M
63	TCANT_BULT	NUMÉRICO	8.0	TOTAL CANTIDAD DE BULTOS	H 61	M
64	TQUNIFIS	NUMÉRICO	15.2	TOTAL CANTIDAD DE UNIDADES FÍSICAS	H 62	M
65	TGARANTÍA	CARACTER	1	TIPO DE GARANTÍA	I 63	N/A
66	NGARANTÍA	CARACTER	10	NÚMERO DE GARANTÍA PRESENTADA	I 64	N/A
67	FVENGARAN	NUMÉRICO	8.0	FECHA DE VENCIMIENTO DE LA GARANTÍA	I 65	N/A
68	MONEGARAN	CARACTER	3	CÓDIGO DE LA MONEDA DE GARANTÍA	I 66	N/A
69	PLAZO_SOL	NUMÉRICO	3.0	PLAZO SOLICITADO EN DÍAS	I 67	N/A
70	CGARANTE	CARACTER	3	CÓDIGO DE LA ENTIDAD QUE GARANTIZA	I 68	N/A

71	MGARANTÍA	NUMÉRICO	15.2	MONTO DE LA GARANTÍA	I 69	N/A
72	AÑO_PRESE	CARACTER	4	AÑO DE PRESENTACIÓN DE LA ORDEN	---	M
73	TIPO_TRAT	CARACTER	1	TIPO DE TRATAMIENTO 1: NORMAL 2: DIPLOMÁTICO 3: DONACIÓN	---	M
74	CDURGE	CARACTER	2	CÓDIGO DE PRODUCTO ACOGIDO A DESPACHO URGENTE	---	C
75	TSOLAFO	CARACTER	1	CÓDIGO DE SOLICITUD DE AFORO	---	C
76	MAUTOLIQUI	NUMÉRICO	14.2	TOTAL AUTOLIQUIDACIÓN (SUMA CAMPOS DEL 79 AL 90)	---	M
77	MADV_DÓLAR	NUMÉRICO	14.2	TOTAL AD-VALOREM AUTOLIQUIDADO	---	C
78	MDES_DÓLAR	NUMÉRICO	14.2	TOTAL DERECHOS ESPECÍFICOS AUTO LIQUIDADO	---	C
79	MICE_DÓLAR	NUMÉRICO	14.2	TOTAL ICE AUTOLIQUIDADO	---	C
80	MIVA_DÓLAR	NUMÉRICO	14.2	TOTAL IVA AUTOLIQUIDADO	---	C
81	MDER_ADUMP	NUMÉRICO	14.2	TOTAL DERECHO ANTIDUMPING AUTOLIQUIDADO	---	C
82	MMU_DÓLAR	NUMÉRICO	14.2	TOTAL MULTAS AUTOLIQUIDADO	---	C
83	MTAS_MODER	NUMÉRICO	14.2	TOTAL TASA MODERNIZACIÓN AUTOLIQUIDADO	---	C
84	MTAS_CONTR	NUMÉRICO	14.2	TOTAL TASA CONTROL AUTOLIQUIDADO	---	C
85	MTAS_ALMA	NUMÉRICO	14.2	TOTAL TASA ALMACENAJE AUTOLIQUIDADO	---	C
86	MFOD_DÓLAR	NUMÉRICO	14.2	TOTAL FODINFA AUTOLIQUIDADO	---	C
87	MCOR_DÓLAR	NUMÉRICO	14.2	TOTAL CORPEI AUTOLIQUIDADO	---	C
88	MOTR_DÓLAR	NUMÉRICO	14.2	TOTAL OTROS AUTOLIQUIDADO	---	C
89	TIPO_ENVIO	CARACTER	2	TIPO DE ENVÍO DE LA TRANSACCIÓN 09: ENVÍO ORIGINAL 53: PRUEBA	---	M
90	TFOB_DIVFA	NUMÉRICO	15.2	TOTAL FOB EN MONEDA DE TRANSACCIÓN	---	C

91	NUME_FORME	NUMÉRICO	9	NÚMERO DE FORMULARIO DEL DAU	---	M
92	METD_VALOR	NUMÉRICO	1	MÉTODO DE VALORACIÓN OMC	---	C

b) DATOS DE SERIES O SUBPARTIDAS NACIONALES – DAUDET01

Nº	CAMPO	TIPO	TAMAÑO	DESCRIPCIÓN	DUI	OBS
1	CODI_ADUAN	CARACTER	3	CÓDIGO DE LA ADUANA DE PRESENTACIÓN	---	M
2	AÑO_ORDEN	CARACTER	4	AÑO DE LA ORDEN INTERNA DEL AGENTE	---	M
3	NUME_ORDEN	CARACTER	6	NÚMERO DE LA ORDEN INTERNA DEL AGENTE	---	M
4	CODI_REGI	CARACTER	2	CÓDIGO DE RÉGIMEN (10)	---	M
5	NUME_SERIE	NUMÉRICO	4.0	NÚMERO DE LA SERIE	J 71	M
6	ESTA_MERCA	CARACTER	1	ESTADO DE LA MERCANCÍA	J 72	M
7	PAÍS_ORIGE	CARACTER	2	CÓDIGO DEL PAÍS DE ORIGEN	J 73	M
8	PAÍS_ADQUI	CARACTER	2	CÓDIGO DEL PAÍS DE ADQUISICIÓN	J 74	M
9	CONV_INTER	NUMÉRICO	4.0	TRATAMIENTO PREFERENCIAL POR CONVENIO INTERNACIONAL	J 75	C
10	TRAT_PREFE	NUMÉRICO	4.0	TRATAMIENTO PREFERENCIAL POR NORMA GENERAL	J 76	C
11	CODI_LIBER	NUMÉRICO	4.0	TRATAMIENTO PREFERENCIAL POR NORMA ESPECIAL	J 77	C
12	CANT_BULTO	NUMÉRICO	11.2	CANTIDAD DE BULTOS	J 78	M
13	CLASE	CARACTER	3	CLASE DE BULTOS	J 79	M
14	UNID_FIQTY	NUMÉRICO	11.2	CANTIDAD DE UNIDADES FÍSICAS	J 80	M
15	UNID_FIDES	CARACTER	3	TIPO DE UNIDAD FÍSICA	J 81	M
16	PESO_NETO	NUMÉRICO	14.2	TOTAL PESO NETO	J 82	M
17	PESO_BRUTO	NUMÉRICO	14.2	TOTAL PESO BRUTO	J 83	M
18	FOB_DIVFAC	NUMÉRICO	14.2	FOB MONEDA DE TRANSACCIÓN	J 84	C
19	PART_NANDI	NUMÉRICO	10.0	SUBPARTIDA NACIONAL	J 85	M

20	DVER_PARNA	CARACTER	1	DIGITO VERIFICADOR SUBPARTIDA NACIONAL	---	C
21	PART_NALAD	NUMÉRICO	8.0	SUBPARTIDA NALADISA	J 86	C
22	TIPO_MARGE	CARACTER	1	TIPO DE MARGEN	J 86	C
23	AJUSTE	NUMÉRICO	14.2	AJUSTE	---	C
24	VAL_ADUANA	NUMÉRICO	14.2	VALOR ADUANAS	J 87	M
25	FOB_DOLPOL	NUMÉRICO	14.2	VALOR FOB	---	M
26	FLETE	NUMÉRICO	14.2	VALOR FLETE	---	M
27	SEGURO	NUMÉRICO	14.2	VALOR SEGURO	---	M
28	DESC_COMER	CARACTER	100	DESCRIPCIÓN MERCANCÍAS	J 89	M
29	MARC_NUMER	CARACTER	100	MARCAS Y NÚMEROS	J 90	C
30	UBIGEO	CARACTER	6	CÓDIGO DE UBICACIÓN GEOGRÁFICA DE CONSUMO / PRODUCCIÓN DE MERCANCÍAS	---	M

c) DATOS DE DESCRIPCIONES MÍNIMAS DE VEHÍCULOS – DAUDET02

Nº	CAMPO	TIPO	TAMAÑO	DESCRIPCIÓN	DUI	OBS
1	CODI_ADUAN	CARACTER	3	CÓDIGO DE LA ADUANA DE PRESTACIÓN	---	M
2	AÑO_ORDEN	CARACTER	4	AÑO DE LA ORDEN INTERNA DEL AGENTE	---	M
3	NUME_ORDEN	CARACTER	6	NÚMERO DE LA ORDEN INTERNA DEL AGENTE	---	M
4	CODI_REGI	CARACTER	2	CÓDIGO DE RÉGIMEN	---	M
5	NUME_SERIE	NUMÉRICO	4	NÚMERO DE LA SERIE	J 71	M
6	CODI_MARCA	CARACTER	3	CÓDIGO DE MARCA COMERCIAL	---	M
7	DESC_MARCA	CARACTER	20	DESCRIPCIÓN DE LA MARCA SOLO EN CASO CODI_MARCA = 'ZZZ'	---	C
8	NOMB_MODEL	CARACTER	60	MODELO	---	M
9	PAIS_ORIGE	CARACTER	2	PAÍS DE ORIGEN	---	M
10	NUME_VIN	CARACTER	25	NÚMERO DE VIN	---	M
11	NUME_MOTOR	CARACTER	25	NÚMERO DE MOTOR	---	M
12	CODI_COLOR	CARACTER	3	CÓDIGO DE COLOR	---	M

13	DESC_COLOR	CARACTER	20	DESCRIPCIÓN DE COLOR SOLO EN CASO DE CODI_COLOR = 'ZZZ'	---	C
14	CODI_CLASE	CARACTER	2	CÓDIGO DE CLASE	---	M
15	CODI_CARRO	CARACTER	2	CÓDIGO DE CARROCERÍA	---	M
16	CODI_COMBU	CARACTER	3	CÓDIGO DE COMBUSTIBLE	---	M
17	DESC_COMBU	CARACTER	20	DESCRIPCIÓN DE COMBUSTIBLE SOLO EN CASO DE CODI_COMBU = 'ZZZ'	---	C
18	NUME_CILIN	NUMBER	5	NÚMERO DE CILINDRAJE	---	M
19	NUME_CAPAC	NUMBER	5	CAPACIDAD	---	M
20	NUME_TONE	NUMBER	9.2	TONELAJE	---	M
21	PART_MANDI	CARACTER	10	PARTIDA NANDINA	---	M
22	CODI_TRANS	CARACTER	3	CÓDIGO DE TRANSMISIÓN	---	M
23	CODI_TRACC	CARACTER	3	CÓDIGO DE TRACCIÓN	---	M
24	ANNO_FABR	NUMÉRICO	4	AÑO DE FABRICACIÓN	---	M

D) DATOS DE DOCUMENTOS SUSTENTATORIOS O QUE AMPARAN LA
DAU – DAUDOCAS

Nº	CAMPO	TIPO	TAMAÑO	DESCRIPCIÓN	DUI	OBS
1	CODI_ADUAN	CARACTER	3	CÓDIGO DE ADUANA	---	M
2	AÑO_ORDEN	CARACTER	4	AÑO DE LA ORDEN	---	M
3	NUME_ORDEN	CARACTER	6	NÚMERO DE LA ORDEN	---	M
4	COD_REGI	CARACTER	2	CÓDIGO DE RÉGIMEN (10)	---	M
5	NUME_SERIE	NUMÉRICO	4.0	NÚMERO DE LA SERIE	---	C
6	TIPO_PROCE	CARACTER	1	TIPO OPERACIÓN / PROCESO DOCUMENTO AUTORIZACIÓN DE DESPACHO DOCUMENTO DE DONACIÓN DOCUMENTO AUTORIZACIÓN DESPACHO	---	M

				URGENTE		
7	TIPO_DOCAS	CARACTER	2	TIPO DE DOCUMENTO ASOCIADOS 1: EXPEDIENTE 2: RESOLUCIÓN 3: PÓLIZA DE SEGURO 4: CERTIFICADO DE INSPECCIÓN 5: CERTIFICADO DE ORIGEN	---	M
8	AÑO_DOCAS	CARACTER	4	AÑO DEL DOCUMENTO ASOCIADO	---	M
9	NUME_DOCAS	CARACTER	18	NÚMERO DEL DOCUMENTO ASOCIADO	---	M
10	FECH_DOCAS	NUMÉRICO	8.0	FECHA DEL DOCUMENTO ASOCIADO	---	M
11	FECH_VENCI	NUMÉRICO	8.0	FECHA DE VENCIMIENTO DE PLAZO	---	C
12	ENTI_EMITO	CARACTER	40	NOMBRE DE LA ENTIDAD EMISORA	---	M

e) DATOS DE OBSERVACIONES SOBRE LOS DATOS GENERALES, SERIES, ETC. – DAUOBSER

Nº	CAMPO	TIPO	TAMAÑO	DESCRIPCIÓN	DUI	OBS
1	CODI_ADUAN	CARACTER	3	CÓDIGO DE LA ADUANA DE PRESENTACIÓN	---	M
2	AÑO_ORDEN	CARACTER	4	AÑO DE LA ORDEN INTERNA DEL AGENTE	---	M
3	NUME_ORDEN	CARACTER	6	NÚMERO DE LA ORDEN INTERNA DEL AGENTE	---	M
4	CODI_REGI	CARACTER	2	CÓDIGO DE RÉGIMEN (10)	---	M
5	SECUENCIA	CARACTER	4	NÚMERO DE SECUENCIA DEL REGISTRO	---	M
6	TIPO_OBS	CARACTER	1	PO DE OBSERVACIÓN: OBSERVACIÓN GENERAL LUGAR DE TRASLADO PARA DESPACHO ANTICIPADO	---	M
7	OBSERVA	CARACTER	250	CONTENIDO DE LA OBSERVACIÓN SEGÚN EL TIPO		M

f) DATOS DE DECLARACIONES PRECEDENTES RELACIONADAS CON EL DAU
– DAUREGAP

Nº	CAMPO	TIPO	TAMAÑO	DESCRIPCIÓN	DU	OBS
1	COD_ADUAN	CARACTER	3	CÓDIGO DE ADUANA	---	M
2	AÑO_ORDEN	CARACTER	4	AÑO DE LA ORDEN	---	M
3	NUME_ORDEN	CARACTER	6	NÚMERO DE LA ORDEN	---	M
4	CODI_REGI	CARACTER	2	CÓDIGO DE RÉGIMEN (10)	---	M
5	NUME_SERIE	NUMÉRICO	4.0	NÚMERO DE SERIE	---	M
6	REGI_PROCE	CARACTER	2	RÉGIMEN DE PROCEDENCIA	G 44	M
7	CADUREGP	CARACTER	3	ADUANA DEL RÉGIMEN PRECEDENTE	G 45	M
8	FAÑOREGPRE	CARACTER	4	AÑO DEL RÉGIMEN PRECEDENTE	G 45	M
9	NDCLREGPRE	CARACTER	6	NÚMERO DEL RÉGIMEN PRECEDENTE	G 45	M
10	FINIREGP	NUMÉRICO	8.0	FECHA DE INICIO DEL RÉGIMEN DE PRECEDENCIA	---	C
11	FTERREGPRE	NUMÉRICO	8.0	FECHA DE VENCIMIENTO DEL RÉGIMEN PRECEDENTE	G 46	M
12	NUME_SERPR	NUMÉRICO	4.0	NÚMERO DE SERIE DEL RÉGIMEN PRECEDENTE	---	M

g) DATOS DE CONTENEDORES RELACIONADOS CON EL DAU -
DAUCONTE

Nº	CAMPO	TIPO	TAMAÑO	DESCRIPCIÓN	DUI	OBS
1	COD_ADUAN	CARACTER	3	CÓDIGO DE ADUANA	---	M
2	AÑO_ORDEN	CARACTER	4	AÑO DE LA ORDEN	---	M
3	NUME_ORDEN	CARACTER	6	NÚMERO DE LA ORDEN	---	M

4	CODI_REGI	CARACTER	2	CÓDIGO DE RÉGIMEN (10)	---	M
5	NUME_CONT	CARACTER	15	NÚMERO DE CONTENEDOR	---	M
6	NUME_PRECI	CARACTER	15	NÚMERO DE PRECINTO	---	M
7	NUME-MARCA	CARACTER	60	MARCAS	---	M

h) DATOS GENERALES DE LA DECLARACIÓN ANDINA DEL VALOR –
DAVHDR01

Nº	CAMPO	TIPO	TAMAÑO	DESCRIPCIÓN	DAV	OBS
1	NUME_SECUP	NUMÉRICO	4.0	NÚMERO SECUENCIAL DE PROVEEDOR POR DAU	---	M
2	COD_ADUAN	CARACTER	3	CÓDIGO DE ADUANA	---	M
3	AÑO_ORDEN	CARACTER	4	AÑO DE LA ORDEN	---	M
4	NUME_ORDEN	CARACTER	6	NÚMERO DE LA ORDEN	---	M
5	CODI_REGI	CARACTER	2	CÓDIGO DE RÉGIMEN (10)	---	M
6	CNIV_COIMP	CARACTER	1	NIVEL COMERCIAL DEL IMPORTADOR	1.4	M
7	CNIV_OTROS	CARACTER	40	OTRO TIPO DE NIVEL COMERCIAL	1.5	M
8	NOMB_PROVE	CARACTER	50	NOMBRE DEL PROVEEDOR	3.1	M
9	COND_PROVE	CARACTER	1	CONDICIÓN DEL PROVEEDOR	3.2	M
10	DIRE_PROVE	CARACTER	55	DIRECCIÓN DEL PROVEEDOR	3.3	M
11	CIUD_PROVE	CARACTER	20	CIUDAD DEL PROVEEDOR	3.4	M
12	CODI_PAIS	CARACTER	2	CÓDIGO DE PAÍS DEL PROVEEDOR	3.5	M
13	FAX_PROVE	CARACTER	18	NÚMERO DE FAX DEL PROVEEDOR	3.6	C
14	TELEF_PROVE	CARACTER	18	NÚMERO TELEFÓNICO DEL PROVEEDOR	3.7	M
15	PAG_WEBPR	CARACTER	60	PÁGINA WEB DEL PROVEEDOR	---	C
16	EMAIL_PR	CARACTER	40	E-MAIL DEL PROVEEDOR	3.8	C
17	INTE_COMVE	CARACTER	1	INTERMEDIARIO ENTRE COMPRADOR Y VENDEDOR	6.1	M
18	NOMB_INTER	CARACTER	50	NOMBRE DEL INTERMEDIARIO	6.2	C
19	DOMI_INTER	CARACTER	50	DOMICILIO DEL INTERMEDIARIO	6.3	C

20	CIUD_INTER	CARACTER	15	CIUDAD DEL INTERMEDIARIO	6.4	C
21	PAIS_INTER	CARACTER	2	CÓDIGO DE PAÍS DEL INTERMEDIARIO	6.5	C
22	TIPO_INTER	CARACTER	1	TIPO DE INTERMEDIARIO	6.6	C
23	VINC_PROEX	CARACTER	1	VINCULACIÓN CON PROVEEDOR EXTRANJERO	7.1	M
24	INFL_PRECI	CARACTER	1	INFLUENCIA DE LA VINCULACIÓN EN PRECIO	7.2	M
25	SPAG_INDIR	CARACTER	1	PAGOS INDIRECTOS DE LAS MERCANCÍAS	7.3	M
26	SDER_LICEN	CARACTER	1	CÁNONES Y DERECHO DE LICENCIA DE LAS MERCANCÍAS	7.4	M
27	VENT_CONDI	CARACTER	1	VENTA CONDICIONADA POR UN ACUERDO	7.5	M
28	PRECE1	CARACTER	1	RESTRICCIONES CESIÓN UTILIZACIÓN ART. 1 OMC	7.6	M
29	PCONMCI A	CARACTER	1	DEPENDE VENTA O PRECIO CON MERCANCÍAS A VALORAR	7.7	M
30	PDECOND	CARACTER	1	PUEDA DETERMINARSE VALOR CONDICIONES	7.8	M
31	CANT_FACTU	NUMÉRICO	4.0	TOTAL DE FACTURAS	---	M
32	PREC_FACTU	NUMÉRICO	15.2	PRECIO SEGÚN FACTURA	8.1.1	M
33	PAGO_INDIR	NUMÉRICO	15.2	PAGOS INDIRECTOS, DESCUENTOS RETROACTIVOS	8.1.2	C
34	PREC_REPAG	NUMÉRICO	15.2	TOTAL BASE DE CÁLCULO 8.1	8.1	M
35	VCOM_CORRE	NUMÉRICO	15.2	COMISIONES CORRETAJES	8.2.1	C
36	VENV_EMBAL	NUMÉRICO	15.2	ENVASES Y EMBALAJES	8.2.2	C
37	PRES_BISER	NUMÉRICO	15.2	BIENES Y SERVICIOS SUMINISTRADOS	8.2.3	C
38	VDER_LICEN	NUMÉRICO	15.2	CÁNONES Y DERECHOS DE LICENCIA	8.2.4	C
39	PROD_REVEN	NUMÉRICO	15.2	PRODUCTO DE REVENTA CESIÓN O UTILIZACIÓN	8.2.5	C
40	VGAS_TRIMP	NUMÉRICO	15.2	GASTOS DE ENTREGA HASTA EL LUGAR DE IMPORTACIÓN	8.2.6	C
41	VGAS_TREMB	NUMÉRICO	15.2	GASTOS DE TRANSPORTE HASTA EL	8.2.7	C

				EMBARQUE		
42	VGAS_TREIM	NUMÉRICO	15.2	GASTO DE TRANSPORTE DESDE EL EMBARQUE AL LUGAR DE IMPORTACIÓN	8.2.8	C
43	VGAS_CARDE	NUMÉRICO	15.2	GASTOS DE CARGA, DESCARGA O MANIPULACIÓN	8.2.9	C
44	VGAS_SEGUR	NUMÉRICO	15.2	GASTOS DE SEGURO	8.2.10	C
45	TOTA_ADICI	NUMÉRICO	15.2	TOTAL ADICIONES 8.2	8.2	N/A
46	VGAS_EPOSI	NUMÉRICO	15.2	GASTOS DE ENTREGA POSTERIORES A IMPORTACIÓN	8.3.1	C
47	VINT_DEVEN	NUMÉRICO	15.2	INTERESES	8.3.2	C
48	VASI_TECNI	NUMÉRICO	15.2	ASISTENCIA TÉCNICA	8.3.3	C
49	VDER_ADUAN	NUMÉRICO	15.2	DERECHOS DE ADUANA	8.3.4	C
50	VOTR_GASTO	NUMÉRICO	15.2	OTROS GASTOS	8.3.5	C
51	VTOT_DEDUC	NUMÉRICO	15.2	TOTAL DEDUCCIONES 8.3	8.3	C
52	VAL_ADUANA	NUMÉRICO	15.2	TOTAL 8.1 + 8.2 – 8.3	8.4	M
53	VAL_PROVIN	CARACTER	1	TIENE VALOR PROVISIONAL 8.2.4 Y 8.2.5	8.5	C
54	VAL_FOBDOL	NUMÉRICO	15.2	VALOR FOB	9.1	M
55	VAL_FLETE	NUMÉRICO	15.2	VALOR FLETE	9.2	M
56	VAL_SEGURO	NUMÉRICO	15.2	VALOR SEGURO	9.3	M
57	VAL_AJUST	NUMÉRICO	15.2	VALOR OTROS AJUSTES	9.4	C
58	NOMB_DECLA	CARACTER	40	NOMBRE DEL DECLARANTE	---	M
59	CARG_DECLA	CARACTER	30	CARGO DEL DECLARANTE	---	M
60	TIPDOCDECL	CARACTER	1	TIPO DOCUMENTO DECLARANTE	---	M
61	NUMDOCDECL	CARACTER	13	DOCUMENTO DE IDENTIFICACIÓN DECLARANTE	---	M

i) DATOS DE SERIES O SUBPARTIDAS NACIONALES – DAVDET01

N°	CAMPO	TIPO	TAMAÑO	DESCRIPCIÓN	DAV	OBS
1	NUME_SECUF	NUMÉRICO	4.0	NÚMERO SECUENCIAL DEL PROVEEDOR	---	M
2	NUME_SECUF	NUMÉRICO	4.0	NÚMERO SECUENCIAL DE FACTURA	---	M
3	CODI_ADUAN	CARACTER	3	CÓDIGO DE ADUANA	---	M
4	AÑO_ORDEN	CARACTER	4	AÑO DE LA ORDEN	---	M
5	NUME_ORDEN	CARACTER	6	NÚMERO DE LA ORDEN	---	M
6	CODI_REGI	CARACTER	2	CÓDIGO DE RÉGIMEN (10)	---	M
7	CANT_ITEM	NUMÉRICO	4.0	TOTAL DE ÍTEMS POR FACTURA EN EL EJEMPLAR B	---	M
8	NATU_TRANS	CARACTER	2	NATURALEZA DE LA TRANSACCIÓN	4.1	M
9	TERM_TRANS	CARACTER	3	TÉRMINO DE LA TRANSACCIÓN (INCOTERM)	4.2	M
10	LUGA_TRANS	CARACTER	40	LUGAR DE LA TRANSACCIÓN	4.3	M
11	NUME_RESOL	CARACTER	15	NÚMERO DE RESOLUCIÓN DE ADUANAS	4.3	C
12	FECH_RESOL	NUMÉRICO	8.0	FECHA DE RESOLUCIÓN DE ADUANAS	4.4	C
13	NUME_FACTU	CARACTER	40	NÚMERO DE FACTURA	4.5	M
14	FECH_FACTU	NUMÉRICO	8.0	FECHA DE FACTURA	4.6	M
15	NUME_CONTR	CARACTER	40	NÚMERO DE CONTRATO	4.7	C
16	FECH_CONTR	NUMÉRICO	8.0	FECHA DE CONTRATO	4.8	C
17	TIPO_CAMBI	NUMÉRICO	10.4	TIPO DE CAMBIO	4.9	C
18	FECH_CAMBI	NUMÉRICO	8.0	FECHA DE TIPO DE CAMBIO	4.10	C
19	CODI_MONED	CARACTER	3	CÓDIGO DE MONEDA	4.11	M
20	PAIS_ORIGE	CARACTER	2	CÓDIGO PAÍS DE ORIGEN	4.12	M
21	PAIS_PROCE	CARACTER	2	CÓDIGO PAÍS DE PROCEDENCIA	4.13	M
22	FORM_ENVIO	CARACTER	1	FORMA DE ENVÍO	4.14	M
23	NUME_ENVIO	NUMÉRICO	8.0	NÚMERO DE ENVÍO	4.15	M
24	MODO_TRANS	NUMÉRICO	1	MODO DE TRANSPORTE	4.16	M
25	PUER_EMBAR	CARACTER	5	PUERTO DE EMBARQUE	4.17	M
26	PUER_DESC	CARACTER	5	PUERTO DE DESCARGA	4.18	M
27	FORM_PAGO	CARACTER	2	FORMA DE PAGO	4.19	M
28	VFOB_FACTU	NUMÉRICO	15.2	VALOR TOTAL FOB FACTURA	---	M

j) DATOS DE SERIES O SUBPARTIDAS NACIONALES – DAVDET02

Nº	CAMPO	TIPO	TAMAÑO	DESCRIPCIÓN	DAV	OBS
1	NUME_SECUF	NUMÉRICO	4.0	NÚMERO SECUENCIAL DEL PROVEEDOR	---	M
2	NUME_SECUF	NUMÉRICO	4.0	NÚMERO SECUENCIAL DE FACTURA	---	M
3	CODI_ADUAN	CARACTER	3	CÓDIGO DE ADUANA	---	M
4	AÑO_ORDEN	CARACTER	4	AÑO DE LA ORDEN	---	M
5	NUME_ORDEN	CARACTER	6	NÚMERO DE LA ORDEN	---	M
6	CODI_REGI	CARACTER	2	CÓDIGO DE RÉGIMEN (10)	---	M
7	NUME_ITEM	NUMÉRICO	4.0	NÚMERO DE ÍTEM POR FACTURA	---	M
8	PART_NANDI	NUMÉRICO	10.0	PARTIDA NANDINA	5.1	M
9	NOMB_COMER	CARACTER	50	NOMBRE COMERCIAL	5.2	M
10	CARA_TIPO	CARACTER	50	CARACTERÍSTICAS Y TIPOS	5.3	M
11	PAÍS_ORIGE	CARACTER	2	PAÍS DE ORIGEN	5.4	M
12	MARC_COMER	CARACTER	40	MARCA COMERCIAL	5.5	M
13	MODE_MERCD	CARACTER	40	MODELO	5.6	M
14	ANNO	CARACTER	4	AÑO	5.7	C
15	ESTA_MERCD	CARACTER	2	ESTADO DE LA MERCANCÍA	5.8	M
16	CANT_MERCD	NUMÉRICO	14.2	CANTIDAD DE LA MERCANCÍA	5.9	M
17	UNID_MERCD	CARACTER	3	UNIDAD DE LA MERCANCÍA	5.10	M
18	VFOB_UNITA	NUMÉRICO	15.2	FOB UNITARIO US\$	5.11	M
19	TOTAJUSTE	NUMÉRICO	15.2	TOTAL AJUSTE UNITARIO	---	C
20	VFOB_ITEM	NUMÉRICO	15.2	VALOR FOB DEL ÍTEM	---	M

k) DATOS DE CONTROL DEL ENVÍO - ENVCTROL

ORDEN	CAMPO	LONGITUD	DESCRIPCIÓN
			Tipo de Operador de Comercio 01: Agente Afianzado de Aduana 02: Línea Naviera o Transportista – COA

1	C_TIP_OPER	Varchar 2(2)	03: Línea Aérea o Transportista 04: Transportista Terrestre 05: Almacén Temporal 06: Depósito Comercial 07: Agente de Carga Internacional 08: Empresa de mensajería internacional 09: Consolidadora de Carga 10: Autoridad Portuaria
2	C_OPERADOR	Varchar 2(4)	Código de Operador
3	N_REG_DGEN	Number (6)	Número de Registros en Archivo de Cabecera
4	N_REG_DDAV	Number (6)	Número de Registros en Cabecera del Formato DAV
5	N_REG_DDET	Number (6)	Número de Registros en Series
6	N_FOB_TOT	Number (15.2)	Total FOB de las Series
7	N_TOT_PESO	Number (15.2)	Total Peso Bruto de las Series
8	T_CLAV_ELE	Varchar 2(4)	Clave Electrónica
9	N_REG_DITE	Number (6)	Número de Registros de ítems del DAV
10	N_DOC_RUC	Varchar 2(13)	NA
11	C_EMP_SOFT	Varchar 2(4)	Código del Empleado de la Agencia

**Anexo 3: FORMATO DE DATOS DE RESPUESTA DE LA CAE AL
AGENTE AFIANZADO DE ADUANA**

a) DATOS DE CONTROL DEL ENVÍO DE RESPUESTA - RESCTROL

ORDEN	CAMPO	LONGITUD	DESCRIPCIÓN
			Tipo de Operador de Comercio 01: Agente Afianzado de Aduana 02: Línea Naviera o Transportista – COA

1	C_TIP_OPER	Varchar 2(2)	03: Línea Aérea o Transportista 04: Transportista Terrestre 05: Almacén Temporal 06: Depósito Comercial 07: Agente de Carga Internacional 08: Empresa de mensajería internacional 09: Consolidadora de Carga 10: Autoridad Portuaria
2	C_OPERADOR	Varchar 2(4)	Código de Operador
3	F_INI_PROC	Number (8)	Fecha de Inicio de Proceso
4	H_INI_PROC	Varchar 2(8)	Hora de Inicio de Proceso
5	F_FIN_PROC	Number (8)	Fecha de Fin de Proceso
6	H_FIN_PROC	Varchar 2(8)	Hora de Fin de Proceso
7	C_TIP_PROC	Varchar 2(2)	Código del Tipo de Procesamiento 1 Mensaje Aceptado Totalmente 14 Mensaje Con Error
8	N_DOC_OK	Number (6)	Total de Órdenes Aceptados
9	N_DOC_BAD	Number (6)	Total de Órdenes con Errores

b) DATOS DE DETALLE DE LAS INCIDENCIAS DETECTADAS EN LA VALIDACIÓN – RESMENSJ.

ORDEN	CAMPO	LONGITUD	DESCRIPCIÓN
1	C_ADU_INGR	Varchar 2(3)	Código de Aduana del DUI
2	C_AÑO_DOC	Varchar 2(4)	Año de Orden Interna del Agente
3	C_NUM_DOC	Varchar 2(6)	Número de la Orden Interna del Agente
4	C_REG_ADUA	Varchar 2(2)	Tipo de Régimen
5	C_PUER_EMB	Varchar 2(5)	NA
6	C_EMP_RESP	Varchar 2(4)	NA
7	C_TIP_ERR	Varchar 2(1)	NA

8	C_TIP_GUIA	Varchar 2(3)	NA
9	N_SER_DEC	Varchar 2(4)	Número de Serie del DUI
10	N_ITE_DEC	Varchar 2(4)	Número de ítem de la DAV
11	N_CON_EMB	Varchar 2(25)	NA
12	C_ERR_PRO	Varchar 2(4)	Código de Error del Proceso
13	T_DES_ERR	Varchar 2(200)	Descripción del Error del Proceso
14	C_TIP_PROC	Varchar 2 (1)	NA

c) DATOS PARA LA CERTIFICACIÓN DIGITAL DEL MENSAJE – RESACEPT

ORDEN	CAMPO	LONGITUD	DESCRIPCIÓN
1	C_ADU_INGR	Varchar 2(3)	Código de Aduana
2	C_ANO_DOC	Varchar 2(4)	Año de Numeración de la DAU (Aduana)
3	C_REG_ADUA	Varchar 2(2)	Tipo de Régimen
4	C_NUM_DOC	Varchar 2(6)	Número de Declaración Aceptada DAU
5	C_TIP_TRAN	Varchar 2(2)	Tipo de Documento 01 : DAU (*): Se van a implementar otras transacciones (liquidación, etc)
6	N_DIG_VERI	Number (1)	Dígito Verificador
7	C_SEC_TRAN	Varchar 2(3)	Secuencia de Documento (N/A)
8	N_ANO_ORD	Varchar 2(4)	Año de Orden Interna del Operador
9	N_NUM_ORD	Varchar 2(6)	Número de Orden Interna del Operador
10	T_FUN_RESU	Varchar 2(40)	Función Resumen de Verificación
11	C_TIP_AFOR	Varchar 2(1)	Código de Selección Aleatoria V: Canal Verde N: Canal Naranja R: Canal Rojo
12	T_MSG_NOTI	Varchar 2(200)	Mensaje de Notificación
13	ULTD_PAGO	Number (8)	ÚLTIMO DÍA DE PAGO

14	TOPT_DOLAR	Number (14.2)	TOTAL A PAGAR
15	TADV_DOLAR	Number (14.2)	TOTAL ADV
16	TDES_DOLAR	Number (14.2)	TOTAL DERECHO ESPECÍFICO
17	TICE_DOLAR	Number (14.2)	TOTAL ICE
18	TIVA_DOLAR	Number (14.2)	TOTAL IVA
19	TDER_ADUP	Number (14.2)	TOTAL DERECHO ANTIDUMPING
20	TMUL_DOLAR	Number (14.2)	TOTAL MULTAS
21	TTAS_MODER	Number (14.2)	TOTAL TASA MODERNIZACIÓN
22	TTAS_CONTR	Number (14.2)	TOTAL TASA CONTROL
23	TTAS_ALMA	Number (14.2)	TOTAL TASA ALMACENAJE
24	TFOD_DOLAR	Number (14.2)	TOTAL FODINFA
25	TCOR_DOLAR	Number (14.2)	TOTAL CORPEI
26	TOTR_DOLAR	Number (14.2)	TOTAL OTROS

BIBLIOGRAFÍA

- Rational Software Corporation, The Unified Modeling Language (UML), 1997, <http://www.rational.com/uml>
- Use Cases: the Pros and cons, 2004, The Dangers of Misusing Use Cases, <http://www.ksc.com/article7.htm>
- J.A. Zachman, 1999, A framework for information systems architecture
- Management Information Systems Research Group, 2000, Manual del Uso de MERMAID.
- Snoeck Monique, 1995, On a process Algebra Approach to the construction and analysis of MERODE based conceptual models
- Monique Snoeck, Guido Dedene, Object – Oriented Enterprise Modelling with MERODE
- Management Information Systems Research Group, <http://www.econ.kuleuven.ac.be/tew/academic/infosys/research/merode.htm>
- Richard Paige, Jonathan Ostroff, 2002, The single Model Principle, <http://www.jot.fm>
- Monique Snoeck, Cindy Michiels, Guido Dedene, Consistency by Construction: the case of MERODE.
- CAE, 1998, Ley Organica de Aduanas L.99
- CAE, 2000, Manual de Procedimientos Aduaneros

- CAE, 2000, Reglamento General a la Ley Orgánica de Aduanas
- CAE, 2001, Procedimiento de Control de Operaciones Aduaneras Marítimas
- CAE, 2000, Procedimientos para Importar, <http://www.aduana.gov.ec>
- CAE, 2000, Procedimientos para Exportar, <http://www.aduana.gov.ec>
- CAE, 2000, Procedimientos para Importación de Regímenes Especiales, <http://www.aduana.gov.ec>
- CAE, 2000, Proceso Operativo Aduanero, <http://www.aduana.gov.ec>