UNIVERSIDAD CATÓLICA SANTO TORIBIO DE MOGROVEJO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

"APLICACIÓN DE LA METODOLOGÍA DE AMÓN Y JIMÉNEZ PARA ASEGURAR LA CALIDAD DE LOS DATOS EN LA CONSTRUCCIÓN DEL ETL DURANTE LA IMPLEMENTACIÓN DE UN DATAMART PARA LA EMPRESA MC EXPRESS DE LA CIUDAD DE CHICLAYO"

TESIS PARA OPTAR EL TÍTULO DE
INGENIERO DE SISTEMAS Y COMPUTACIÓN

LUIS FELIPE RAUL CASTILLO MONTALVAN

Chiclayo, 24 de octubre de 2011

“APLICACIÓN DE LA METODOLOGÍA DE AMÓN Y JIMÉNEZ PARA ASEGURAR LA CALIDAD DE LOS DATOS EN LA CONSTRUCCIÓN DEL ETL DURANTE LA IMPLEMENTACIÓN DE UN DATAMART PARA LA EMPRESA MC EXPRESS DE LA CIUDAD DE CHICLAYO”

POR:

LUIS FELIPE RAUL CASTILLO MONTALVAN

Presentada a la Facultad de Ingeniería de la Universidad Católica Santo Toribio de Mogrovejo para optar el título de

INGENIERO DE SISTEMAS Y COMPUTACIÓN

APROBADA POR EL JURADO INTEGRADO POR

_____

Mgtr. Eduardo Francisco Alonso Pérez
PRESIDENTE

_____ _____

Ing. Marlon Eugenio Vílchez Rivas
SECRETARIO

Ing. Luis Alberto Otake Oyama
ASESOR

DEDICATORIA

Dedico esta Tesis a toda mi familia.

A mis padres Felipe y Julia, por su comprensión y ayuda en todo momento para culminar mis estudios.

A mis abuelos Segundo y Juana, por el ejemplo de perseverancia, a hermanas Dianira y Juliana por el apoyo incondicional que me mostraron siempre.

A mi esposa Rocío, a ella especialmente le dedico esta Tesis; por su paciencia, por su comprensión, por su amor, por ser tal y como es. Es la persona que directamente me ha apoyado en la realización de este trabajo; permitiéndome conseguir un equilibrio para dar el máximo de mí.

AGRADECIMIENTOS

“A mis profesores de la Universidad, por su ejemplo de profesionalidad que siempre tendré presente”.

“Al gerente y administradora de la empresa MC EXPRESS, Sr. César Linares y la Srta. Roxana Chero, por las facilidades otorgadas para realizar esta tesis”.

“A mi asesor, Ing. Luis Otake Oyama, por su guía en la elaboración de esta investigación”.

“A Erick, Vitucho, Jonny y Jesús, que con la afición a la música lograron mantener en mí la calma en los momentos de tensión”.

“Y a todos aquellos que participaron en la presente investigación".

ÍNDICE

ÍNDICE **5**
LISTA DE FIGURAS **7**
RESUMEN 11
ABSTRACT 12

I. INTRODUCCIÓN 13

II. MARCO TEÓRICO
1. Antecedentes 15
2. Bases Teóricas 16
2.1. Business Intelligence 16
2.1.1. Introducción 16
2.1.2. Definición de Business Intelligence 16
2.1.3. Herramientas de Business Intelligence 16
2.1.3.1. Definición de OLAP 17
2.2. Los Data Warehouse 18
2.2.1. Definición de Data Warehouse 18
2.2.2. Elementos del Data Warehouse 19
2.2.3. Arquitectura de un data warehouse 21
2.3. El Data Mart 23
2.3.1. Definición de Data Mart 23
2.3.2. Ventajas y Desventajas de los Data mart 23
2.3.3. Implementación del Data Mart 24
2.4. IBSS BI-Methodology 24
2.4.1. Análisis de Requerimientos 25
2.4.2. Arquitectura Tecnológica y Modelamiento de Datos 26
2.4.3. Extracción Inicial de Datos 28
2.5. Diseñador SSIS de Visual Studio 2005 para implementación de una Data Mart 32
2.5.1. Diseñar un flujo de control de paquetes 33
2.5.2. Uso del diseñador de flujo de control 33
2.5.3. Diseñar un flujo de datos de paquetes 34
2.5.4. Usar el diseñador de flujo de datos 35
2.5.5. Metodología para ETL 36
2.5.5.1. Extraer datos 36
2.5.5.2. Transformar datos 37
2.5.5.3. Cargar datos 37
2.6. Calidad de Datos 39
2.6.1. Definición de Calidad de Datos 39
2.6.2. Dimensiones de la Calidad de Datos 39
2.6.3. Los Problemas de la Calidad de Datos 40
2.6.3.1. Problemas de Origen de Datos Único 40
2.7. Metodología de Amón y Jiménez 43
2.7.1. Guía Metodológica para la selección de técnicas para la detección de duplicados. 44
2.7.2. Guía Metodológica para la Selección de las Técnicas para Valores Faltantes. 45

2.7.3. Guía Metodológica para Selección de Técnicas para Detección de Valores Atípicos 47
2.7.4. Limpieza de Datos 48

III. MATERIALES Y MÉTODOS 51

IV. RESULTADOS
1. Definición del Star Net para el Sistema de Ventas 54
2. Definición de Star Net para el Sistema de Compras 54
3. Modelamiento Dimensional para Data mart Compras 55
4. Modelamiento Dimensional para Data mart Ventas 56
5. Modelo Lógico de Datamart Compras 56
6. Modelo Lógico de Data mart Ventas 58
7. Herramientas y Plataformas 57
8. Proceso ETL y Depuración de Datos en Data Mart Ventas 58
8.1. Flujo de Datos con Herramientas de SQL Server Business Intelligence Development de Visual Studio 2005 58
8.1.1. Flujo de Datos Simple 58
8.1.2. Pasos para el proceso ETL según la metodología de BI Development de Visual Studio 63
8.2. Proceso ETL usando la Metodología propuesta por Jiménez y Amón para la Depuración de Datos 69
8.2.1. Flujo de datos por pasos para asegurar la calidad de los datos durante el proceso ETL 69
8.2.1.1. Pasos Para la Extracción, transformación y Carga 69
8.2.1.2. Pasos para la depuración de datos usando la metodología de Amón y Jiménez. 72
9. Proceso ETL y Depuración de Datos en Data Mart Ventas 88
9.1. Flujo de Datos con Herramientas de SQL Server Business Intelligence Development de Visual Studio 2005 88
9.1.1. Flujo de Datos Simple 88
9.1.2. Flujo de Datos con Herramientas de SQL Server Business Intelligence Development de Visual Studio 2005 90
9.2. Flujo de Datos usando la Metodología propuesta por Jiménez y Amón para la Depuración 97
9.2.1. Desarrollo de la Metodología Paso por Paso 97
9.2.1.1. Carga de Datos 97

V. DISCUSIÓN 108

VI. PROPUESTA 113

VII. CONCLUSIONES 118

VIII. REFERENCIAS BIBLIOGRAFICAS 119

LISTA DE FIGURAS

Figura 1: Herramientas de Business Intelligence .....
Figura 2: Tabla de Hechos y Dimensiones Tipo Estrella 20
Figura 3: Tabla de Hechos y Dimensiones Copo de Nieve 21
Figura 4: Arquitectura Tipo BUS..... 21
Figura 5: Arquitectura Tipo Entreprise o CIF 22
Figura 6: Arquitectura Data Warehouse 2.0 22
Figura 7: Relación entre Hubs, Links y Satellites 23
Figura 8: Metodología de Implementación de BI, IBSS BI-Methodology 25
Figura 9: Entregables de etapas de IBSS BI-Methodology..... 25
Figura 10: Análisis de Requerimientos 26
Figura 11: Star Net..... 26
Figura 12: Arquitectura Tecnológica y Modelamiento de Datos 27
Figura 13: Modelamiento Dimensional (I) 28
Figura 14: Modelamiento Dimensional (II) 32
Figura 15: Superficie de diseño para crear el flujo de control de paquetes..... 34
Figura 16: Diseñador de Flujo de datos (I)..... 35
Figura 17: Diseñador de Flujo de datos (II)..... 36
Figura 18: Diagrama para la carga de datos 38
Figura 19: Calidad de los Datos y deficiencias del diseño 40
Figura 20: Diagrama para Valores Duplicados..... 44
Figura 21: Diagrama para Valores Faltantes 45
Figura 22: Diagrama para Valores Atípicos..... 47
Figura 23: Compras Durante periodo Octubre-Diciembre 2010 52
Figura 24: Ventas Durante periodo Octubre-Diciembre 2010 52
Figura 25: Star Net de Metadatos para Ventas 54
Figura 26: Star Net de Metadatos para Compras 54
Figura 27: Modelamiento Dimensional Data mart Compras..... 55
Figura 28: Modelamiento Dimensional Data mart Ventas 55
Figura 29: Esquema Estrella Data mart Compras..... 56
Figura 30: Esquema Estrella Data mart Ventas 57
Figura 31: Herramientas proporcionadas por Visual Studio BI Development..... 58
Figura 32: Herramientas de la Tarea de Flujo de Datos..... 59
Figura 33: Configuración del Origen de Datos..... 60
Figura 34: Configuración de la Columna de Origen de Datos 60
Figura 35: Configuración de las Tablas Destino 61
Figura 36: Resultado de Flujo de Datos Simple..... 61
Figura 37: Resultado de Flujo de Datos Simple en Tablas Destino 62
Figura 38: Datos de Tablas de Origen..... 62
Figura 39: Tipo de Datos de Tablas Destino 62
Figura 40: Comparación de Tipo de Datos de Tablas Origen y Destino 63
Figura 41: Configuración del Origen de Datos 63
Figura 42: Configuración de la Columna de Origen de Datos 64
Figura 43: Configuración de la Columna de Origen de Datos 64
Figura 44: Ordenamiento de Datos 65
Figura 45: Transformación de Datos..... 65
Figura 46: Ordenamiento de Datos en el Data mart 66
Figura 47: Mezcla de Datos en Data mart Ventas..... 66

Figura 48: Depuración de Datos Usando la Herramienta División Condicional 67
Figura 49: Carga de Datos usando la Herramienta Destino OLE DB..... 67
Figura 50: Asignación de Datos usando la Herramienta Destino OLE DB 68
Figura 51: Diseño del flujo de datos para el proceso ETL en Data mart Ventas 68
Figura 52: Resultados del proceso ETL usando la Metodología BI Development..... 69
Figura 53: Configuración de Origen de Datos..... 70
Figura 54: Configuración Columnas de Origen de Datos..... 70
Figura 55: Comando de Selección de Tablas en Base de Datos Destino 71
Figura 56: Diseño del ETL según Metodología de Jimenez y Amón 71
Figura 57: Resultados del ETL según Metodología de Jimenez y Amón 72
Figura 58: Distribución de Datos en data mart Ventas 72
Figura 59: Comando SQL para la Depuración de Datos en "idArea" 73
Figura 60: Asignación de la Tabla Temp_HechoVentas..... 73
Figura 61: Relación de los Datos de Origen con los Datos Destino..... 74
Figura 62: Resultados de la Depuración de Datos en "idArea", Datamart Ventas 74
Figura 63: Comparación de los datos de IF0000000000000353..... 75
Figura 64: Comando SQL para la Depuración de datos en "subtotal" del Hecho_Ventas 75
Figura 65: Resultados de la Depuración con Hot Deck en "Subtotal" del Hecho_Ventas 76
Figura 66: Comando para de la Depuración con Hot Deck en "igv" del Hecho_Ventas……...76
Figura 67: Comando para de la Depuración con Hot Deck en "igv" del Hecho_Ventas 77
Figura 68: Comando para de la Depuración con Hot Deck en "total" del Hecho_Ventas 77
Figura 69: Resultados de la Depuración con Hot Deck en "total" del Hecho_Ventas 77
Figura 70: Comando para de la Depuración con Hot Deck en "saldo" del Hecho_Ventas 78
Figura 71: Resultados de la Depuración con Hot Deck en "saldo" del Hecho_Ventas 78
Figura 72: : Comparación de datos para "idEstado" en Hecho_Ventas..... 79
Figura 73 Comparación de datos para "idEstado" en Hecho_Ventas..... 80
Figura 74: Comando para de la Depuración con Hot Deck en "idEstado" del Hecho_Ventas 80
Figura 75: Resultados de la Depuración con Hot Deck en "idEstado" del Hecho_Ventas 81
Figura 76: Datos almacenados en el Hecho_Ventas..... 81
Figura 77: Resultados de la Prueba de Dixon en el Hecho_Ventas 82
Figura 78: Comando SQL para la depuración de datos adaptado con la prueba de Dixon..... 83
Figura 79: Resultados de la Depuración en "igv" del Hecho_Ventas..... 83
Figura 80: Esquema final de la Depuración de Datos con la metodología de Jiménez y Amón 83
Figura 81: Resultados de la Depuración de Datos en Hecho_Ventas..... 84

Figura 82: Editor de Origen de Excel 84
Figura 83: Selección de Columnas con el Editor de Origen de Excel 85
Figura 84: Flujo de paquete de datos 85
Figura 85: Comandos SQL para la depuración de datos 86
Figura 86: Conexión Origen y Destino OLE DB 86
Figura 87: Editor de destino OLE DB 87
Figura 88: Resultados de la Depuración de Datos en Hecho_Ventas 87
Figura 89: Esquema final usando la metodología de Jiménez y Amón en dos pasos 88
Figura 90: Selección de Tipo de conexión usando el administrador de Conexión para Excel 88
Figura 91: Selección de Tablas usando el administrador de conexión para Excel 89
Figura 92: Resultados del proceso ETL para el data mart Compras, usando el flujo de datos simple. 90
Figura 93: Selección de Tablas usando el administrador de conexión para Excel 90
Figura 94: Selección de Columnas usando el administrador de conexión para Excel 91
Figura 95: Selección del Origen-Destino usando el administrador de Conexión para Excel 91
Figura 96: Ordenar los datos con Herramienta de Ordenación de Datos de VB 2005 92
Figura 97: Transformación de Datos usando la Herramienta Conversión de Datos 93
Figura 98: Ordenación de datos Origen-Destino usando la herramienta de Ordenar Datos 93
Figura 99: Transformación de Datos usando la herramienta Transformación de Mezcla 94
Figura 100: Imputación de Datos nulos usando la herramienta de División Condicional 95
Figura 101: Selección de Destino de Datos usando el administrador de conexión OLDB 95
Figura 102: Unión de datos de origen con datos destino usando el administrador de conexión OLEDB 96
Figura 103: Esquema final para la ETL en Data mart Compras 96
Figura 104: Resultados del proceso ETL para el data mart Compras, usando el flujo de la metodología de BI Development de Visual Studio 2005 97
Figura 105: Selección del Origen de Datos usando el administrador para Excel 98
Figura 106: Selección de Columnas usando el administrador de conexión para Excel 98
Figura 107: Selección de Columnas usando el Comandos de SQL 99
Figura 108: Resultados del proceso ETL para el data mart Compras, usando el flujo de la metodología de Jiménez y Amón paso por paso 99
Figura 109: Resultados de la prueba de Dixon para la columna idFecha en Data mart Compras 100
Figura 110: Esquema de Conexión Usando la herramienta de Flujo de Datos 100
Figura 111: Comando SQL para la imputación de datos de la columna idFecha

en el data mart Compras 101
Figura 112: Resultados de la imputación Hot Deck: Muestreo aleatorio Simple para la columna idFecha en Data mart Compras.....101
Figura 113: Datos para verificar y realizar la imputación según los vecinos más cercanos 102
Figura 114: Resultados la imputación en la columna "subtotal" del data mart compras 102
Figura 115: Datos para verificar y realizar la imputación según los vecinos más cercanos. 103
Figura 116: Resultados de la prueba de Dixon para la columna "igv" del data mart compras 103
Figura 117: Comando SQL para la imputación de datos en la columna "igv" del data mart compras.....104
Figura 118: Resultado de la Imputación de la columna "igv" del data mart Compras..... 105
Figura 119: Datos usados para la prueba de Dixon en la columna "total" del data mart Compras 105
Figura 120: Resultado de la prueba de Dixon en la columna "total" del data mart Compras 106
Figura 121: Comando SQL para la imputación de la columna "total" del data mart Compras 106
Figura 122: Resultado de la Imputación en la columna "total" del data mart Compras..... 107
Figura 123: Esquema final para el ETL del data mart Compras.....107
Figura 124: Registro Original de datos depurados para el Data mart Ventas 108
Figura 125: Resultados del ETL según Metodología de Amón y Jiménez en data mart Ventas 108
Figura 126: Datos obtenidos después de la depuración para el Data mart Ventas 109
Figura 127: Registro Original de datos depurados para el Data mart Ventas 110
Figura 128: Resultados de la depuración de datos en el Data mart Ventas resumiendo los pasos de la metodología de Jimenez y Amón.....110
Figura 129: Resultados de la búsqueda por idCliente y código de fecha.....110
Figura 130: Datos reales obtenidos de los registros de Compras.....111
Figura 131: Resultados de la depuración de idFecha en data mart Compras..111
Figura 132: Registros reales de Compras.....112
Figura 133: Resultado de la imputación en la columna Subtotal del Data mart Compras.....112
Figura 134: Diagrama para Valores Faltantes según guía metodológica de Amón y Jiménez.....113
Figura 135: Identificación de las adaptaciones a realizar en el algoritmo de valores faltantes.....114
Figura 136: Algoritmo adaptado para la depuración de datos tipo cadena en bases de datos comunes.....114
Figura 137: Diagrama para Valores Atípicos según guía metodológica de Amón y Jiménez.....115
Figura 138: Identificación de las adaptaciones a realizar en el algoritmo de valores incoherentes.....116
Figura 139: Algoritmo adaptado para la depuración de datos tipo cuantitativo en bases de datos comunes.....117

RESUMEN

Errores de digitación, datos inconsistentes, valores ausentes o duplicados, son algunos de los problemas que pueden presentar los datos, deteriorando su calidad; y en consecuencia las decisiones que se tomen en base a ellos.

En la empresa MC EXPRESS existen múltiples base de datos para cada proceso, la información que se muestra muchas veces no es la correcta, dando como resultado informes erróneos que afectan directamente en las decisiones de la alta dirección,

Es así que se optó por asegurar la calidad de los datos cuando se extraigan, transformen y carguen de las bases de datos de compras y ventas, ayudados por la guía metodológica de Amón y Jiménez al momento de elegir la mejor técnica de depuración de datos durante el proceso ETL.

Esta metodología nos ayudó a elegir las mejores técnicas de depuración según su eficacia para cada caso en particular, identificando el tipo de problema en los datos para que de esa forma se asegure la calidad de los datos durante el proceso ETL.

Finalmente, se pudo reducir el alto índice de error en los datos, cometidos por la extracción de datos de múltiples fuentes, que al no ser analizadas como corresponde, produce consultas con errores en la información, contribuyendo así en la malas decisiones por parte de la alta dirección, y que conllevan a la pérdida de tiempo, dinero, oportunidades de nuevos negocios, y hasta la banca rota de las empresas.

PALABRAS CLAVE: Business Intelligence, proceso ETL, calidad de datos, técnicas de business intelligence, data warehouse, data mart.

ABSTRACT

Typing errors, inconsistent data, duplicate or missing values are some of the problems that may present data, impairing their quality, and consequently thedecisions made based on them.

In the company there are multiple MC EXPRESS database for each process, theinformation displayed is often not correct, resulting in erroneous reports that directly affect the decisions of top management.

Thus it was decided to ensure the quality of the data when extracting, transforming, and loading of databases of purchases and sales, helped by Amon methodological guide and Jimenez when choosing the best technique for data cleansing the ETL process.

This methodology helped us choose the best debugging techniques as effective ineach particular case, identifying the type of problem in the data to thereby ensurethe quality of data during the ETL process.

Finally, it could reduce the high rate of data error committed by the extraction of data from multiple sources, not being properly analyzed, produce queries with errors in the information, thus contributing to the bad decisions by the senior management and leading to the loss of time, money, new business opportunities and evenbankrupt companies.

KEYWORDS: Business Intelligence, ETL process, data quality, technical of business intelligence, data warehouse, data mart.

INTRODUCCIÓN

En la década de 1970, los sistemas de apoyo a decisiones fueron las primeras aplicaciones diseñadas para apoyar la toma de decisiones. A principios de la década de 1990, Howard Dressner, un analista del Grupo Gartner, acuñó el término de inteligencia de negocio de largo plazo. BI es ahora ampliamente utilizado, para describir aplicaciones analíticas.

Actualmente existen problemas en la mayoría de los proyectos de BI, como indica Gartner (2007) al decir que la mala calidad de los datos sobre los clientes, lleva a costos importantes, como el sobreestimar el volumen de ventas, el exceso de gastos en los procesos de contacto con los clientes y a la pérdida de oportunidades de ventas. Según las investigaciones realizadas por IDC y SAS (2009), nos señalan que debido a la mala calidad de los datos provocan en las empresas españolas pérdidas del 30% en su productividad, es decir de diez mil millones de dólares que facturaron dejaron de ganar tres millones. También la revista Marketing y Ventas (2006) en un artículo dedicado a la gestión adecuada de una base de datos, nos dice que las pérdidas por una base de datos defectuosa pueden ascender al 20%, estima que entre un 10% y un 15% de la información manejada en las bases de datos de las empresas es incorrecta.

Las técnicas desarrolladas por los investigadores hasta el momento, son variadas. Es así como existen técnicas para tratar el problema de la detección de duplicados, para detección y corrección de valores atípicos, para tratar con los valores faltantes y para cada posible problema que puedan presentar los datos. Amón y Jiménez (2009), nos dicen que "Para la detección de duplicados, se encuentran técnicas como la distancia de edición, distancia de brecha afín, distancia de SmithWaterman, distancia de Jaro, qgrams, Whirl y técnicas fonéticas como soundex, NYSIIS, ONCA, Metaphone y Double Metaphone.

La única metodología que se centra específicamente en el aseguramiento de la calidad de datos es la propuesta por Amón y Jiménez, quienes plantean el análisis profundo de las técnicas a aplicar en un caso específico, haciéndose necesario identificar el tipo de problema de los datos, el problema de valores que pueden ser diferentes pero que debieran ser el mismo y además conocer como es la distribución de los datos, a partir de ello se selecciona las técnicas más adecuadas, esto es, identificar sus fortalezas y debilidades, estableciéndose una serie de criterios de evaluación que permitan comparar las técnicas y elegir una de ellas. Los criterios determinarán la eficacia de la técnica ante diferentes situaciones. Se trata de calificar a cada técnica, según sea eficaz o no, al ser aplicada. La eficacia de una técnica será baja para un criterio, si la similitud entre dos textos decae al presentarse la situación cuando se comparan valores nominales (Amón y Jiménez, 2010).

En la empresa MC EXPRESS existen múltiples base de datos asignadas para cada proceso, en especial las designadas para las funciones administrativas, la cual comprende los sistemas de compra y venta de servicios, en estos sistemas la información que se muestra muchas veces no es la correcta, dando como resultado informes erróneos que afectan directamente en las decisiones que se tomen en la alta dirección, es por ello que se hace necesario depurar y o limpiar

los datos contenidos en las bases de datos para asegurar la calidad de los datos que se extraigan de estos almacenes.
Ante estas adversidades, se buscó la forma de cómo ayudar a asegurar la calidad de los datos de un datamart durante la construcción del proceso ETL. Para ello, se propuso con este trabajo asegurar la calidad de los datos cuando se extraigan, transformen y carguen de las bases de datos de compras y ventas hacia los data mart, ayudados por la guía metodológica de Amón y Jiménez al momento de elegir la mejor técnica de depuración de datos durante el proceso ETL.

Por ello se tuvo que analizar el proceso ETL, luego diseñar este proceso, desarrollarlo y finalmente verificar los resultados del proceso ETL.

La metodología de Amón y Jiménez, pretende reducir el alto índice de error en los datos, cometidos por la extracción de datos de múltiples fuentes, que al no ser analizadas como corresponde, produce consultas con errores en la información, contribuyendo así en la malas decisiones por parte de la alta dirección, y que conllevan a la pérdida de tiempo, dinero, oportunidades de nuevos negocios, y hasta la banca rota de las empresas.

El motivo por el cual solo se construirá dos data mart es porque la metodología que se plantea está basada en el análisis profundo de las técnicas a aplicar en un caso específico, haciéndose necesario identificar el tipo de problema de los datos, además conocer como es la distribución de los datos, que a partir de ello se selecciona las técnicas más adecuadas que en su mayoría hacen uso de métodos matemáticos y estadísticos, y que se tendrían que aplicar para cada uno de los problemas que presenten los datos, dándole así un cierto grado de dificultad para realizarla.

Cabe resaltar que la metodología es aplicable a datos estructurados, como atributos de tablas de una base de datos, campos de una bodega de datos, columnas de una hoja de cálculo o un archivo de texto. No aplica a datos no estructurados como páginas Web o correos electrónicos, también se basa principalmente en la detección de duplicados, detección de valores atípicos incorrectos y datos faltantes o nulos, estos tres tipos de problemas, se seleccionaron ya que, según la literatura, son los más frecuentes en los datos y se dispone de diversas técnicas para tratarlos, también se debe de recordar que la metodología es para seleccionar las técnicas de depuración de datos más no para realizar la limpieza en sí, es por eso que se trabajará con ella durante la etapa más compleja en la implementación de un data mart, nos referimos a l proceso ETL.

II. MARCO TEÓRICO

1. Antecedentes

Múltiples trabajos se han realizado en la temática de calidad de datos. A continuación, se relacionan algunos que son de interés para el propósito de esta tesis. En cuanto a trabajos relacionados con la clasificación y la detección de los problemas, son varios los que han realizado clasificaciones de las anomalías en los datos Rahm et al (2000), Kim et al (2003), Müller et al (2003.).

Rosenthal (2001), realizó un trabajo extendido en los sistemas de bases de datos, haciendo uso de anotaciones para cada atributo de la base de datos, concluyendo con el uso de metadatos para facilitar la depuración de los datos..

Hancong, (2004) expone otro tipo de problemas, el de los valores extremos atípicos, conocidos como Outliers, concluyendo que no necesariamente son errores, estos pueden ser generados por un mecanismo diferente de los datos normales como problemas en los sensores, distorsiones en el proceso, mala calibración de instrumentos y/o errores humanos.

Oliveira et. al. (2005) no sólo realizan una taxonomía con treinta y cinco problemas de calidad de los datos, planteando métodos semiautomáticos para detectarlos, los cuales representan mediante árboles binarios. Los árboles corresponden al razonamiento que se necesita hacer para detectar un problema particular.

Rittman (2006) presenta una metodología para la calidad de datos concluyendo en la elaboración del módulo de Oracle encargado de realizar depuración a los datos (Oracle Warehouse Builder), para realizar este proceso.

Uno de los trabajos más recientes es el resumen de Elmagarmid et. al. (2007) en el cual se exponen las principales técnicas para el problema de la detección de duplicados, concluyendo que para evaluar los registros completos como los campos tipo texto, estos se deben de realizar en forma individual.

En los trabajos realizados, también se encuentran algunos de tipo metodológico. Tierstein (sin fecha), que presenta una metodología que incorpora dos tareas que se interrelacionan y se traslapan: limpiar los datos de un sistema legacy y convertirlos a una base de datos.

Amón y Jiménez (2009) Este artículo pone de manifiesto la necesidad de una guía metodológica que apoye a los analistas de datos en la selección de las técnicas de depuración, considerando los diferentes tipos de errores en los datos y la naturaleza de los mismos.

Amón y Jiménez (2010) Estudian las múltiples técnicas desarrolladas de detección de duplicados, de detección de valores atípicos y datos faltantes, entre otros, concluyendo en una guía metodológica que ayuda a la depuración de los datos.

2. Bases Teóricas

2.1. Business Intelligence

2.1.1. Introducción

Para Méndez (2000) Business Intelligence es el reconocimiento del valor de suministrar hechos e información como soporte a la toma de decisiones, debido a que allá por los años 70 los sistemas para la toma de decisiones (DSS) eran la gran promesa que ayudaría a las organizaciones a obtener esa deseada inteligencia, y que apenas se quedó en promesa.

De la Fuente (2004) nos dice que Business Intelligence es un proceso de convergencia que propuso una nueva filosofía de organizar y explotar los datos, al objeto de obtener información orientada al análisis y la gestión de la empresa. Este concepto luego de haber observado que en los inicios de la década de los 90 las empresas se interesaban cada vez más con las tecnologías de información.

Así mismo, Curto y Conesa (2010) nos dice que la evolución del Business Intelligence se debió gracias a la aparición de la informática personal, en donde el uso de programas informáticos de gestión pasó a ser algo común y estar al alcance de cualquier empresa.

2.1.2. Definición de Business Intelligence

Autores como Méndez (2000) definen a Business Intelligence como un conjunto de herramientas y aplicaciones para la ayuda a la toma de decisiones que ofrecen ventajas, como una plataforma integrada que se añadirá a las inversiones ya realizadas por una organización, que ayudan a conocer el pasado de una organización para controlar y comunicar el presente y predecir el futuro con fiabilidad, interfaces de usuario personalizadas que se adaptan a cada tarea y niveles de experiencia y patrones de uso de los usuarios.

Al igual que Méndez (2000), Horváth & Partners (2007) ambos prestan más atención a la parte técnica de las herramientas que ofrece BI dentro de su plataforma de desarrollo, al decir que BI engloba las herramientas técnicas de información que apoyan la evaluación de los hechos disponibles a nivel de toda la empresa, pero este último incluye al usuario como parte de un BI y lo coloca como el más importante del proceso al decir que BI describe las posibilidades de acceso y de análisis de las personas, que lo utilizarán con respecto a los datos y a la información almacenada en la empresa.

Curto y Conesa (2010) no dice que se entiende como Business Intelligence al conjunto de metodologías, prácticas y capacidades enfocadas a la creación y administración de información que permite tomar mejores decisiones a los usuarios de una organización.

2.1.3. Herramientas de Business Intelligence

BI es un conjunto de conceptos y metodologías que, haciendo uso de acontecimientos (hechos) y sistemas sustentados en los mismos, apoya la toma

de decisiones en los negocios. Para esto ser posible es necesario adquirir los datos, por ejemplo, por medio de un sistema de procesamiento on-line de transacciones (OLTP), almacenarlos en un sistema de base de datos, como un Data warehouse del cual se puede generar aún un subconjunto más específico de datos, Data mart y, finalmente procesar estos datos con una herramienta de análisis que puede ser: una herramienta de procesamiento analítico on-line (OLAP), un sistema de informaciones para ejecutivos (EIS); un sistema de apoyo a la decisión (DSS); o aun, un sistema de descubierta y predicción. (Vieira et al, 2009).

Figura 1: Herramientas de Business Intelligence

Tipo de Herramienta	Cuestión básica	Ejemplo de respuesta
Data Mining	¿Qué es interesante? ¿Qué puede suceder?	Tipos de clientes Predicción de ventas
OLAP	¿Qué sucedió y por qué?	Ventas mensuales versus variaciones de precios de los competidores
EIS/DSS	¿Qué necesito saber ahora ?	Cotizaciones diversas
Estudios y informes	¿Qué sucedió ?	Ventas del último mes

Fuente: Vieira et al. (2009)

2.1.3.1. Definición de OLAP

Codd y Date (1993) presentaron un documento en el que definían OLAP como un proceso de análisis de los datos en línea, ayudando de esta forma a un ejecutivo a manejar los datos de la organización de forma inmediata, como nos dice Parra (1998) que la idea básica de OLAP es permitir que un decisor (gestor, analista, ejecutivo) sea capaz de manejar los datos de su empresa de forma interactiva para comprender los cambios que se producen, algo así como navegar por los datos lo más libremente posible buscando respuestas.

Para Rob y Coronel (2006) OLAP se define como un ambiente de análisis de datos avanzados que soporta actividades de toma de decisiones, modelo del negocio e investigación de operaciones, la palabra clave en este caso es "ambiente", el cual incluye tecnología cliente/servidor.

Para que un sistema pueda llamarse OLAP, según Trujillo et al (2011) debe de cumplir las 12 reglas que Codd propuso, estas propiedades o reglas son:

- Vista Conceptual Multidimensional.- mediante la metáfora de cubo de datos o tabla multidimensional.
- Transparencia.- En el acceso de fuentes de datos heterogéneas y en el proceso de transformación de datos realizado por los procesos ETL.
- Accesibilidad.- En los datos presentados a los usuarios con un esquema lógico sencillo de interpretar.
- Rendimiento de Informes Consistentes.- Con lo que no se debería demorar en exceso conforme el número de dimensiones crece.
- Arquitectura Cliente Servidor.- Para sistemas abiertos y modulares.
- Dimensionalidad General.- No limitada a tres dimensiones (3D) y no particularizado a ninguna dimensión en concreto.

- Gestionar Matrices Vacías Dinámicas.- Que se debería adaptar a la variación de almacenamiento y opciones de consulta de datos.
- Soportar Multiusuario.- Es decir, múltiples usuarios actuando de maneta concurrente.
- Operaciones a través de Dimensiones no Restringidas.- y por ello no limitar las relaciones entre las celdas de datos.
- Manipulación de datos intuitiva.-Para los usuarios.
- Informes Flexibles.- Para que los usuarios sean capaces de imprimir solo aquello que necesitan.
- Dimensiones y niveles de agregación no limitados.- Donde se debería soportar al menos 15 dimensiones y, preferiblemente 20.

2.2. Los Data Warehouse

Para poder hacer un uso adecuado de las herramientas de BI, es necesario contar con una o múltiples base de datos, que serán nuestras principales fuentes de información para el análisis respectivo antes de la toma de decisiones.

2.2.1. Definición de Data Warehouse

Según Calle (1997) haciendo referencia al Gartner Group nos dice que es una plataforma en donde se almacenan y mantienen datos de áreas interfuncionales de una organización. Los datos procedentes de los almacenes operacionales (una vez que son integrados, consolidados y depurados) sirven no solamente para apoyar aplicaciones específicas sino también para aplicaciones de ayuda a la decisión. Otra definición de Data Warehouse es la que nos proporciona Giner (2004), quienes basándose en Bill Inmon y Richard Hackathorn en 1994 dan a conocer que es una colección de datos orientados a temas, integrados, no volátiles y variantes en el tiempo, organizados para soportar necesidades empresariales. A partir de esa definición Giner (2004) infieren en que el Data Warehouse se caracteriza por:

- Ordenar, organizar un conjunto de datos, normalmente procedente de fuentes muy diversas (transaccionales, externas, otras bases de datos…).
- Con fines específicos, temáticos (análisis clientes, productos, mercados…).
- Los datos organizados suelen tener relaciones entre ellos, relaciones causales o de causa efecto (base de datos relacionales). O bien se estructuran por dimensiones temáticas con más jerarquías (cubos multidimensionales).
- Los datos presentan cierta estabilidad en el tiempo.
- No hay volatilidad. Los datos del Data warehouse son copia de los transaccionales o de las fuentes, son éstos los que cambian.
- Permiten remontarse a una cierta historia. Se puede tener una visión temporal de un hecho.
- El enfoque: se organizan así los con el fin de obtener información para la toma de decisiones en la empresa a diferentes niveles. Por lo tanto, se orienta a proporcionar información a los analistas de la empresa y a todo tipo de directivos.

También Curto y Conesa (2010) afirma que el data warehouse frecuentemente está constituido por una base de datos relacional, pero no es la única opción factible, también es posible considerar las bases de datos orientadas a columnas o incluso basadas en lógica asociativa.
Debemos tener en cuenta que existen otros elementos en el contexto de una data warehouse.

- Data Warehousing.- Es el proceso de extraer y filtrar datos de las operaciones comunes de la organización, procedentes de los distintos sistemas de información.
- Data Mart.- Es un subconjunto de los datos del data warehouse cuyo objetivo es responder a una determinado análisis, función o necesidad, con una población de usuarios específica, los datos están estructurados al igual que el data warehouse.
- Operational Data Store.- Es un tipo de almacén de datos que proporciona solo los últimos valores de los datos y no su historial; además, generalmente admite un pequeño desfase o retraso sobre los datos operacionales.
- Stating Area.- Es el sistema que permanece entre las fuentes de datos y el data warehouse con el objetivo de facilitar la extracción de datos desde fuente de origen, con una heterogeneidad y complejidad grande.
- Procesos ETL.- Tecnología de integración de datos basada en la consolidación de datos que se usa tradicionalmente para alimentar al data warehouse.
- Metadatos.- Datos estructurados y codificados que describen características de instancias; aportan informaciones para ayudar a identificar, descubrir, valorar y administrar las instancias descritas.

2.2.2. Elementos del Data Warehouse

A diferencia de las bases de datos normalizadas que comúnmente encontramos en los sistemas transaccionales, los data warehouse tienen como idea principal que la información sea presentada desnormalizada para optimizar las consultas. Es por ello que para realizar una data warehouse se deben de identificar los siguientes elementos:

a) Las tablas de Hechos.- Es una representación de un proceso de negocio. Es un tabla que a nivel de diseño nos permite guardar dos tipos de atributos diferenciados:

- Medidas del proceso, actividad, flujo de trabajo, evento que se pretende modelizar.
- Claves foráneas hacia registros en una tabla de dimensión.

Existen diferentes tipos de tablas de hecho:

- Transaction Fact Table.- representan eventos que suceden en un determinado espacio-tiempo, se caracteriza por permitir analizar los datos con el máximo detalle. Por ejemplo, podemos pensar en una

venta que tiene como resultado métricas como el importe de la misma.

- Factless Fact Tables/Coverage.- Son tablas que no tienen medidas, y tiene sentido dado que representan el hecho de que el evento suceda.
- Periodic Snapshopt Fact Table.- Son tablas de hechos usadas para recoger información de forma periódica a intervalos de tiempo regulares. Dependiendo de la situación medida o de la necesidad de negocio.
- Acumulating Snapshot Fact Table.- Representan el ciclo de vida completo de una actividad o un proceso. Se caracteriza por presentar múltiples dimensiones relacionadas con los eventos presentes en un proceso.

b) Tipos de Dimensiones.- Las dimensiones recogen los puntos de análisis de un hecho. Por ejemplo, una venta se puede analizar en función del día de venta, producto, cliente, vendedor o canal de venta.

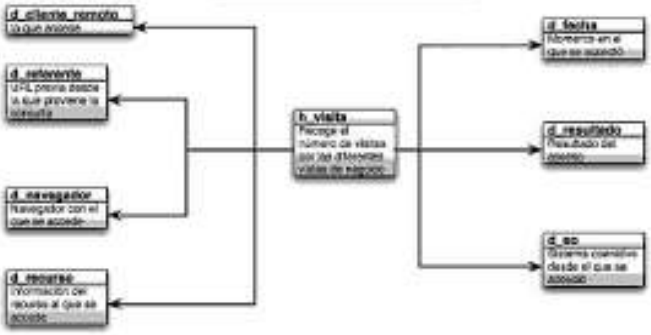
c) Tipo de Métricas.- podemos distinguir diferentes tipos de medidas, basadas en el tipo de información que recopilan así como su funcionalidad asociada. Pueden ser:

- Métricas de Realización de Actividad.- miden la realización de una actividad. Por ejemplo, la participación de una persona en un evento.
- Métricas de resultado de una actividad.- Recogen los resultados de una actividad. Por ejemplo, la cantidad de puntos de un jugador.
- Indicadores Clave.- Valores correspondientes que hay que alcanzar y que suponen el grado de asunción de los objetivos, estas medidas proporcionan información sobre el rendimiento de una actividad o sobre la consecución de una meta.

d) Esquemas para Estructurar la Información.- Existen principalmente dos tipos de esquemas para estructurar los datos en un almacén de datos:

- Esquema Estrella.- Consiste estructurar la información en procesos, vistas y métricas recordando a una estrella. A nivel de diseño consiste en una tabla de hechos en el centro para el hecho objeto de análisis y una o varias tablas de dimensión por cada punto de vista de análisis.

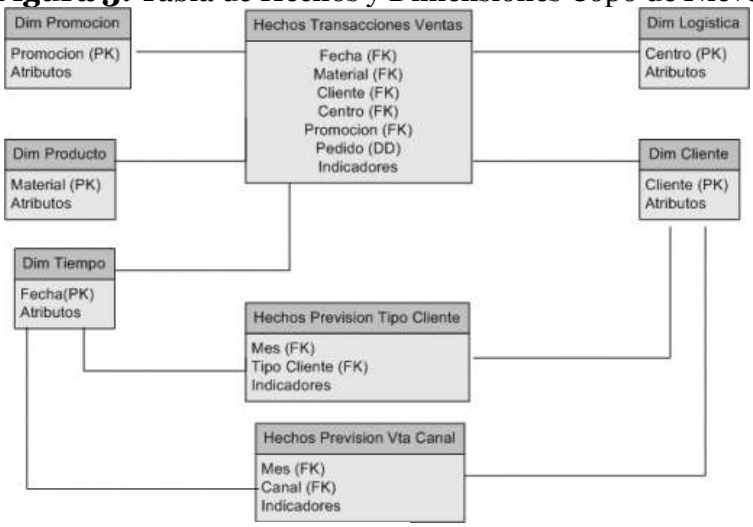
Figura 2: Tabla de Hechos y Dimensiones Tipo Estrella



Fuente: Curto y Conesa (2010)

- Esquema Copo de Nieve.- Es un esquema de representación derivado del esquema en estrella, en el que las tablas de dimensión se normalizan en múltiples tablas. Por esta razón, la tabla de hechos deja de ser la única tabla del esquema que se relaciona con otras tablas.

Figura 3: Tabla de Hechos y Dimensiones Copo de Nieve



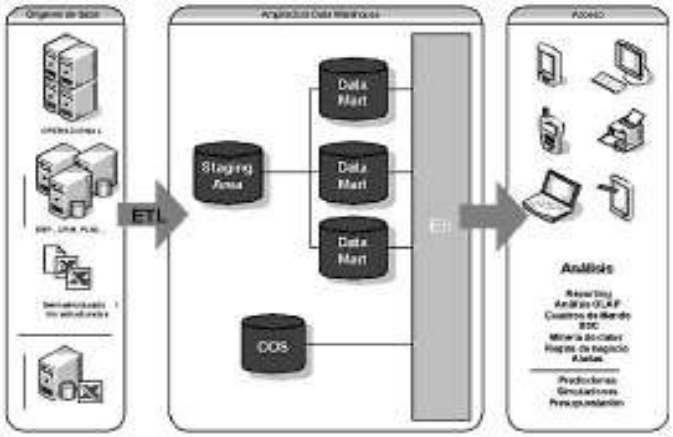
Fuente: Respinozamilla (2009)

2.2.3. Arquitectura de un data warehouse

Para Curto y Conesa (2010), existen tres enfoques en la arquitectura corporativa de un data warehouse.

a) Enterprice Bus Architecture (Data warehouse Virtual/Federado). - También conocido como MD (Multidimensional Arquitecture), consiste en una arquitectura basada en data mart independientes federados que pueden hacer uso de una stating area en el caso de ser necesario. Federados quiere decir que se hace uso de una herramienta EII (Enterprise Information Integration) para realizar las consultas como si se tratara de un único data warehouse.

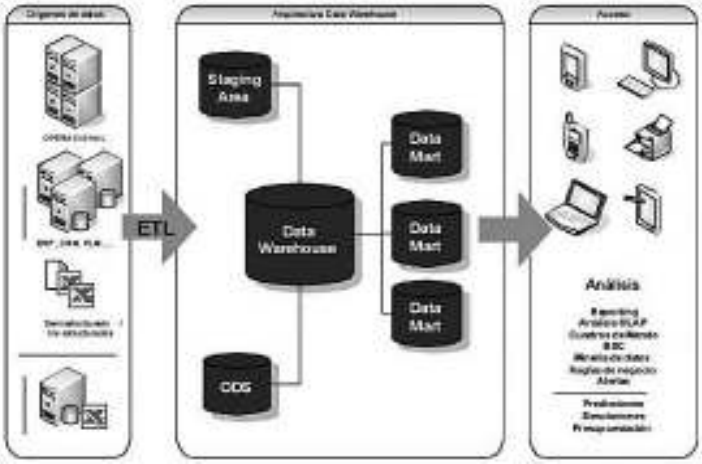
Figura 4: Arquitectura Tipo BUS



Fuente: Curto y Conesa (2010)

b) Corporate Information Factory (Enterprise Data Warehouse). - Consiste en una arquitectura en la que existe un data warehouse corporativo y unos data mart (o incluso cubos OLAP) dependientes del mismo. El acceso a datos se realiza a los data mart o a la ODS en caso de existir, pero nunca al propio data warehouse. Puede existir en el caso de ser necesaria una stating area.

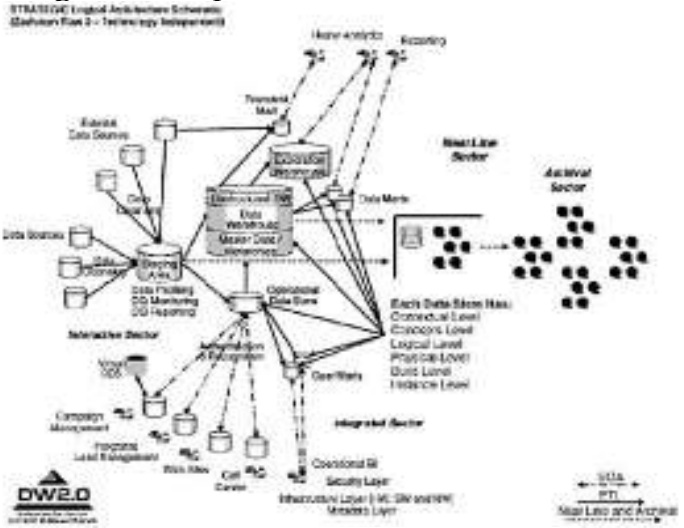
Figura 5: Arquitectura Tipo Entreprise o CIF



Fuente: Curto y Conesa (2010)

c) Enterprise Data Warehouse 2.0.- Consiste en la revisión de la metodología Bill Inmon para incluir toda la experiencia de los últimos veinte años. El punto diferencial es que se separa la información por la edad de la misma y se clasifica por su uso. Se caracteriza por completar tanto la inclusión de información estructurada como no estructurada y por focalizarse en el objetivo de responder a todas las necesidades actuales de negocio. El siguiente gráfico representa una arquitectura completa.

Figura 6: Arquitectura Data Warehouse 2.0



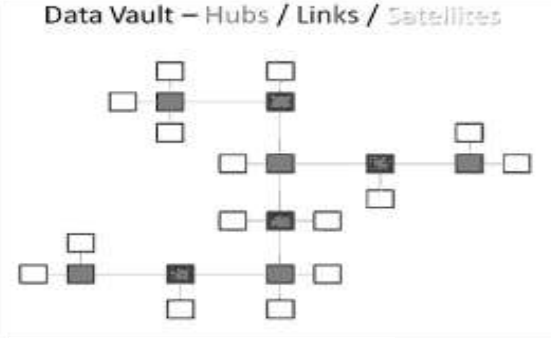
Fuente: Curto y Conesa (2010)

Esta metodología está aún en proceso de despliegue en el contexto empresarial. Se combina, frecuentemente, con Data Vault, un modelo de datos basado en tres tipos de entidades:

- Hubs.- Contiene los indicadores claves de negocio.
- Links.- Contiene las relaciones.
- Satellites.- Contiene las descripciones.

Este tipo de diseño busca la máxima flexibilidad del data warehouse con el objetivo que éste sea adaptable a cualquier evolución del modelo de negocio de la organización.

Figura 7: Relación entre Hubs, Links y Satellites



Fuente: Curto y Conesa (2010)

2.3. El Data Mart

Para nuestro trabajo de investigación nos vamos a centrar en los que son los data mart, se va a conocer sus elementos, su estructura y su implementación.

2.3.1. Definición de Data Mart

Para Date (2001) data mart es un almacén de datos especializado, orientado a un tema, integrado, volátil y variante en el tiempo para apoyar a un subconjunto específico de decisiones de administración, la principal diferencia entre un data mart y un data warehouse es que el data mart es especializado y volátil, es decir solamente contiene datos para apoyar a un área específica de análisis del negocio, volátil porque los usuarios pueden actualizar los datos e incluso, posiblemente, crear nuevos datos, es decir nuevas tablas, para algún propósito.

Para Moliner (2005) es una aplicación de data warehouse, construida rápidamente para soportar una línea de negocios simple, los data mart, tienen las mismas características de integración, no volatilidad, orientación temática que el data warehouse, representan una estrategia de "divide y vencerás" para ámbitos muy genéricos de un data warehouse.

2.3.2. Ventajas y Desventajas de los Data mart

Stair y Reynolds (2000) nos dice que la principal ventaja de los data mart es su facilidad de implementación, a comparación de los data warehouse que resultaron ser difíciles de desarrollar y administrar, y casi imposible de usar por algunos administradores, además, ayuda a procesar con rapidez transacciones

para operaciones rutinarias que a menudo con los data warehouse se contraponía con las necesidades de los gerentes de manipular y resumir datos que les ayudaran a tomar decisiones basadas en información.

Stair y Reynolds (2000) identificaron que con los data mart se puede ganar en rapidez de procesamiento de información, no se requiere de una gran infraestructura para su implementación, pero también identificaron que se perdía en efectividad al momento de intercambiar información, capacidad de integración entre estas bases de datos, y que los data mart resultan muy pequeños para manejar las solicitudes de procesamiento e información que en comparación con los data warehouse se pueden obtener.

Así mismo Kroenke (2003) añade más ventajas de los data mart, al decir que permite que los metadatos son más fáciles de identificar y mantener, porque el data mart se puede restringir a un tipo particular de datos, a determinada función de negocios, a una unidad de negocio específica, o a un área geográfica. También permiten que la administración del data warehouse sea más sencilla

2.3.3. Implementación del Data Mart

Para implementar un data mart se sigue los mismos pasos de implementación de un data warehouse, pero con la diferencia que el data warehouse puede estar constituidos de varios data mart.

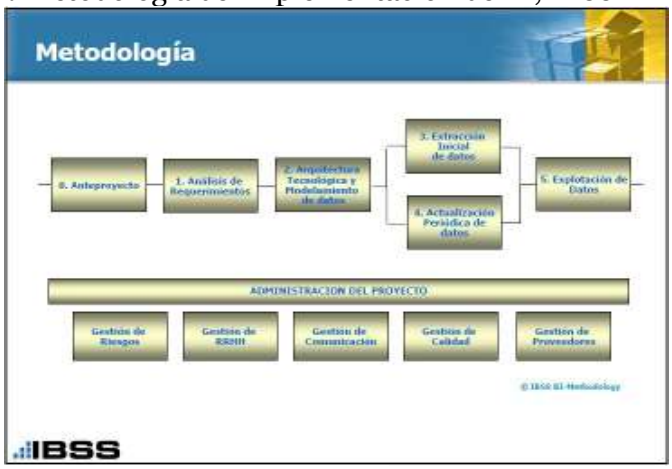
Existe una metodología que es la más usada para la implementación de herramientas de BI, denominada IBSS BI-Methodology, la cual consiste en seguir unos pasos de acuerdo como se va avanzando dentro del proyecto.

2.4. IBSS BI-Methodology para la implementación de data warehouse y data mart

Según Bermúdez (Sin fecha), nos dicen que la mejor forma de implementar un BI en una organización es aplicando la metodología de IBSS, que consiste en realizar los siguientes pasos:

1. Análisis de Requerimientos
2. Arquitectura Tecnológica y Modelamiento de Datos.
3. Extracción Inicial de Datos.
4. Actualización Periódica de Datos.
5. Explotación de Datos.

Figura 8: Metodología de Implementación de BI, IBSS BI-Methodology

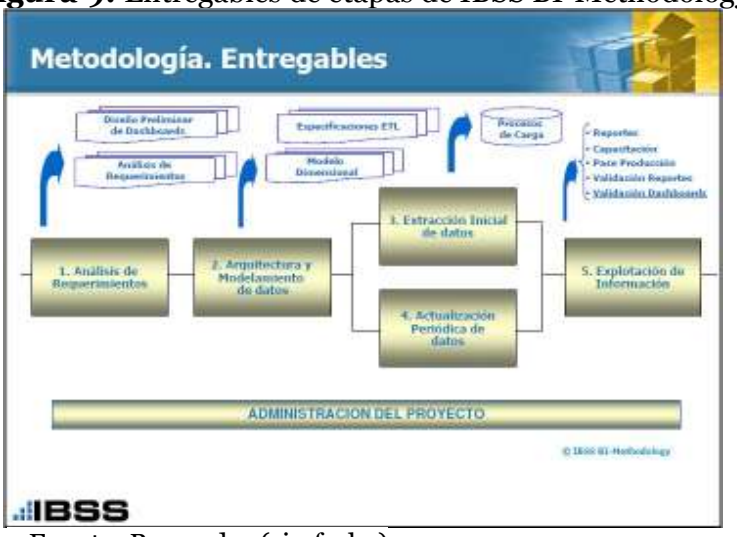


Fuente: Bermúdez (sin fecha)

En la cual cada una de estas etapas nos proporciona unos entregables que nos ayudará a implementar adecuadamente un BI. Estos entregables son:

- Diseño Preliminar de Dashboard, Análisis de Requerimientos.
- Especificaciones ETL, Modelo Dimensional
- Proceso de Carga
- Reportes, capacitación, pase Producción, Validación de Reportes, Validación de Dashboard

Figura 9: Entregables de etapas de IBSS BI-Methodology

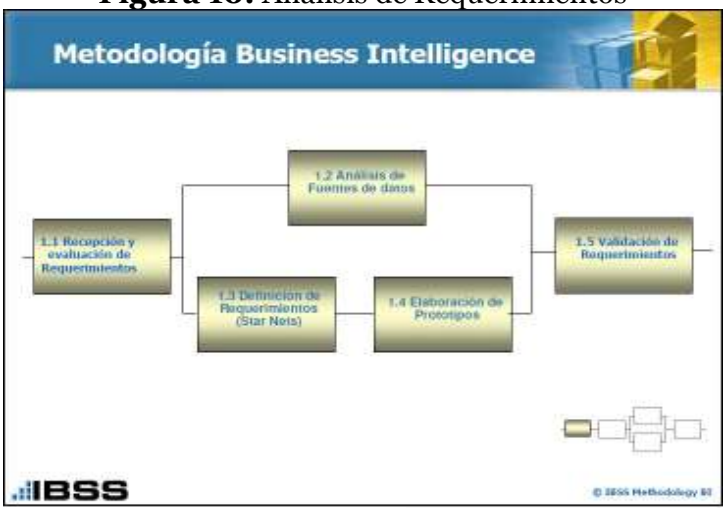


Fuente: Bermudez (sin fecha)

2.4.1. Análisis de Requerimientos

Es la parte fundamental de de todo proyecto de implementación de BI, en esta etapa recepcionamos y evaluamos los requerimientos, analizaremos las fuentes de datos, definiremos los requerimientos, elaboramos los prototipos y validaremos los requerimientos.

Figura 10: Análisis de Requerimientos



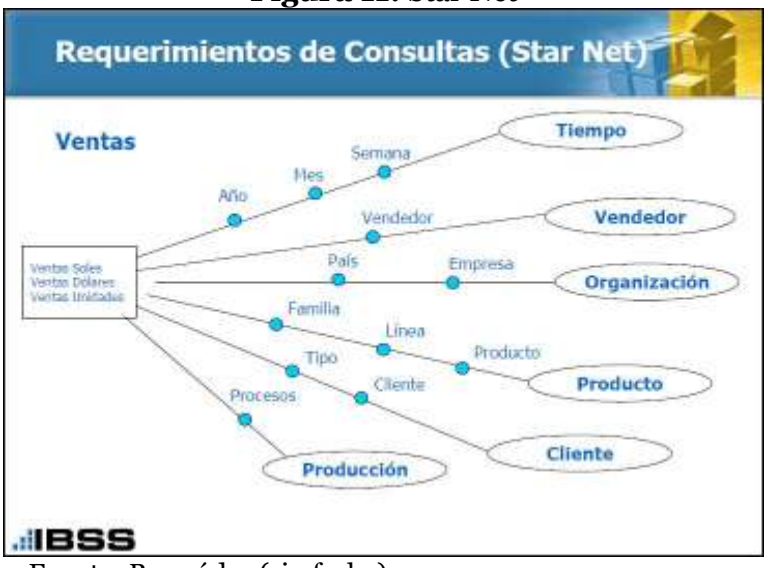
Fuente: Bermúdez (sin fecha)

Las actividades que se deben realizar en esta etapa son las siguientes:

1. Definir los usuarios responsables
2. Establecer el plan de entrevistas
3. Identificar riesgos
4. Entrevistas a usuarios responsables
5. Validación de requerimientos
6. Formalizar "Alcance de Requerimientos"

Luego de haber realizado estas actividades se deben plasmar los requerimientos en una red de metadatos o Star Net, la cual según Kimball (2008), define como el contenido del almacén de datos que se describe en términos más fáciles de usar, los metadatos del negocio dicen lo que contienen los datos, de donde vienen, lo que significan y cuál es su relación con otros datos en data warehouse. Como se muestra en la siguiente figura.

Figura 11: Star Net



Fuente: Bermúdez (sin fecha)

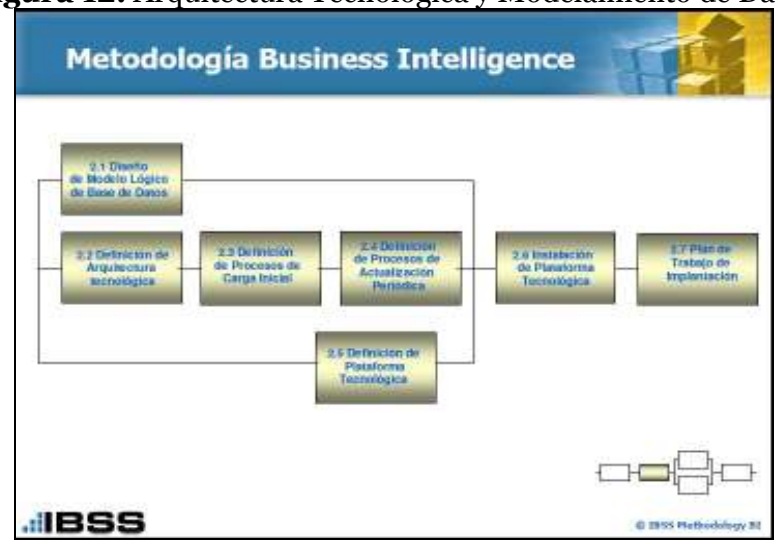
Las consideraciones que se deben tener en esta etapa son las siguientes:

1. Formalizar requerimientos
2. Negociar requerimientos no prioritarios
3. Comprometer a todo el equipo técnico y funcional
4. Fomentar en las entrevistas la participación y compromiso de los usuarios
5. Contar con alternativas de prototipo

2.4.2. Arquitectura Tecnológica y Modelamiento de Datos

En esta etapa nos encargaremos de diseñar y definir el modelo lógico, la arquitectura, el proceso de carga inicial, el proceso de actualización periódica, la plataforma tecnológica; además de la instalación de la plataforma tecnológica y el plan de trabajo de implantación.

Figura 12: Arquitectura Tecnológica y Modelamiento de Datos

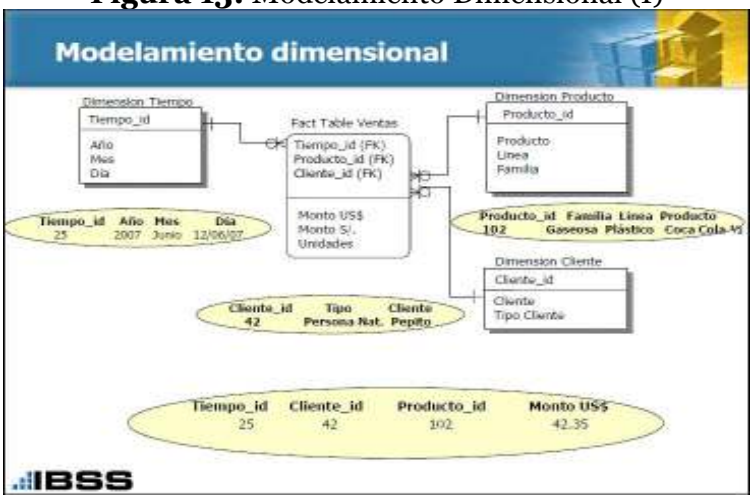


Fuente: Bermúdez (sin fecha)

El modelamiento dimensional es una técnica de diseño lógico comúnmente utilizada para data warehouse, que busca presentar los datos en una arquitectura estándar y permita una alta performance de acceso a los usuarios finales. El modelamiento se basa en esquemas estrella, conformado por tablas de hechos y tablas dimensionales (Medina y Bermúdez).

Para poder realizar el modelamiento de datos dimensional es necesario contar con la Star Net, a partir de ello se deberá modelar según los metadatos establecidos, como se muestra en la siguiente figura.

Figura 13: Modelamiento Dimensional (I)



Fuente: Bermúdez (sin fecha)

Durante esta etapa se deben tener las siguientes consideraciones:

1. En lo posible evitar crear muchas dimensiones.
2. Diseñar teniendo en cuenta una perspectiva global
3. Considerar las particularidades de las herramientas de análisis en el diseño
4. Validar modelo frente a requerimientos de usuarios
5. Prepararse para comenzar nuevamente
6. Documentar el modelo final
7. El equipo técnico debe quedar conociendo a plenitud el modelo final.

2.4.3. Extracción Inicial de Datos

Para poder realizar la extracción, transformación y carga de los datos, primero debemos de realizar la construcción de la base de datos destino, la cual debe coincidir con los metadatos que hemos tomado en cuenta, además de haber determinado las tablas de hechos y las dimensiones a llenar.

A) Definición de ETL.- Según Conesa y Curto (2010), ETL es una tecnología que permite extraer datos del entorno de origen, transformarlos según nuestras necesidades de negocio para integración de datos y cargar estos datos en los entornos destino. Los entornos de origen y destino son usualmente bases de datos y/o ficheros, pero en ocasiones también pueden ser colas de mensajes de un determinado mddleware, así como ficheros u otras fuentes estructuradas, semi estructurada o no estructurada.

B) Tipos de ETL.- Existen diferentes actividades que se pueden realizar con la herramienta ETL:

- ETL de Generación de código.- Constan de un entorno gráfico donde se diseñan y especifican los datos de origen, sus transformaciones y los entornos destino- El resultado generado es un programa de tercera generación que permite realizar las transformaciones de datos.

- ETL Basados en Motor.- Permite crear flujos de trabajo en tiempo de ejecución definidos mediante herramientas gráficas. El entorno gráfico permite hacer un mapping de los entornos de datos de origen y destino, las transformaciones de datos necesarios, el flujo de procesos y los procesos por lotes necesarios.
  - Motor de Extracción.- Utiliza como adaptadores a ODBC, JDBC, JNDI, SQL Nativo, adaptadores en modo pull planificado, típicamente soportando técnicas de consolidación en procesos por lotes, o mediante modo push, típicamente utilizando técnicas de programación en procesos de tipo en línea.
  - Motor de Transformación.- Proporciona una librería de objetos que permite a los desarrolladores transformar los datos de origen para adaptarse a las estructuras de datos destino.
  - Motor de Carga.- Utiliza adaptadores a los datos de destino, como SQL Nativo, o cargadores masivos de datos para insertar o actualizar los datos en las bases de datos o ficheros de destino.
  - Servicios de Operación y Administración.- Permite la planificación, ejecución y monitorización de los procesos ETL, así como la visualización de eventos y la recepción y resolución de errores en los procesos.
- ETL Integrado en las Bases de Datos.- Algunos fabricantes incluyen capacidades ETL, dentro del motor de la base de datos (al igual que lo hacen con otro tipo de características como soporte OLAP y minería de datos). En general, presentan menos funcionalidades y complejidad, y son una solución menos completa que los ETL comerciales basados en motor o de generación de código.
  - ETL Cooperativos.- Con ellos los productos comerciales pueden usar funciones avanzadas del gestor de base de datos para mejorar los procesos de ETL, por ejemplo los ETL cooperativos son aquellos que pueden utilizar procedimientos almacenados y SQL complejo para realizar las transformaciones de los datos de origen.
  - ETL Complementarios.- Cuando los ETL de bases de datos ofrecen funcionalidades complementarias a los ETL comerciales, por ejemplo hay gestores de bases de datos que ofrecen soporte MQT (Materializad Query Tables) o vistas de sumarización pre calculadas, mantenidas y almacenadas por el gestor que pueden usarse para evitar transformaciones de datos realizadas por el ETL comercial.
  - ETL Competitivos.- Algunos gestores ofrecen herramientas gráficas integradas que explotan sus capacidades ETL en lo que claramente es competencia con los ETL Comerciales.
- EII.- El objetivo de la tecnología EII es permitir a las aplicaciones el acceso a datos dispersos (desde un data mart hasta ficheros de texto o incluso web services) como si estuviesen todos residiendo en una base de datos común. Por lo tanto se basa en la federación.

C) Diseño de Programas de Carga.- Bermúdez (sin fecha) nos sugiere tomar en cuenta las siguientes consideraciones:

a) A nivel de Herramientas.- siempre es bueno preguntarse por los siguiente:

- ¿Utilizaremos alguna herramienta de ETL o escribiremos los programas a mano?
- ¿Las herramientas de ETL se puede conectar directamente a las plataformas de nuestras fuentes de datos?
- ¿La performance de la herramienta de ETL es óptima?

b) A nivel de Ventana de Tiempo.- tener en cuenta lo siguiente:
- ¿De cuánto tiempo (horas) disponemos para hacer la carga de datos? ¿Es suficiente?
- ¿Cuántas fuentes de información serán utilizadas en la carga de datos?
- ¿Cuántos modelos deberán de ser cargados?

c) A Nivel de flujo de Carga.- Considerar los siguiente:
- ¿Cómo debo organizar los programas de carga?
- ¿Cuántos programas podré correr en paralelo para reducir el tiempo de carga?
- ¿Cuánto tiempo tomará la carga inicial e histórica? ¿Cuántos años de información son necesarios?
- ¿Cuánto tiempo toma la carga incremental?

d) A Nivel de Performance.- Se debe tener en consideración:
- ¿Debo desactivar los índices antes de correr la carga? ¿Cómo afecta la performance?
- ¿Debo desactivar la integridad referencial antes de correr la carga? ¿Cómo afecta la performance?
- ¿Nos conectamos directamente a la fuente o usamos archivos de texto?
- ¿La infraestructura de comunicaciones es la adecuada?

e) A Nivel de Validaciones.- Se debe de preguntar
- ¿Qué tipo de validaciones debo de utilizar en la carga?
- ¿Dónde debo considerar las validaciones (bitácoras de error)?
- ¿Qué debe hacer el programa al detectar errores?

f) Las fuentes de orígenes de datos provienen de diferentes plataformas de bases de datos y sistemas operativos. EL objeto del proceso de ETL es consolidar los datos de estas fuentes heterogéneas en una plataforma estándar y en un formato estándar.

g) Actividades en el Diseño del ETL.- Las actividades son las siguientes:
- Elaboración del documento de Mapeo.
- Testeo de la herramienta de ETL.
- Diseño de los flujos de carga.
- Diseño de los programas de carga.
- Estrategia de Stating Area.

D) Desarrollo de Programas de Carga.- Medina y Bermúdez nos sugieren tomar en cuenta las siguientes consideraciones:

a) A Nivel de Fuentes de Orígenes de Datos.- Se debe de tener en cuenta:
- ¿Quién desarrollará los programas de carga? ¿Los programadores asignados tienen experiencia en ETL? ¿Entienden el proceso de ETL?
- ¿Existen versiones anteriores que puedan utilizarse?

- ¿Existe documentación de las fuentes o contamos con los programadores de esos sistemas?
- ¿Qué necesitamos saber de los sistemas fuentes antes de iniciar los ETL?

b) A Nivel de Herramienta de ETL.- se debe de considerar:

- ¿Hemos trabajado con esta herramienta antes o es nueva para nosotros?
- ¿Hemos recibido el entrenamiento adecuado para el ETL?
- ¿La herramienta soportará todos los tipos de transformaciones de datos y conexiones a base de datos o debemos escribir las rutinas extras? ¿de ser así en que lenguaje?

c) A Nivel de Flujos y Procesos.- Considerar los siguiente:

- ¿Existen dependencias entre los programas de carga?
- ¿Qué módulos pueden correr en paralelo? ¿Necesitamos algún artificio antes?
- ¿Cuántas tablas podemos levantar en paralelo? ¿Los recursos de hardware serán suficientes?

d) A Nivel de Pruebas.- se debe de tener en cuenta:

- ¿Quién realizará las pruebas?
- ¿Qué tan reales serán los volúmenes de información para las pruebas?
- ¿Tenemos una estrategia de pruebas definida?
- ¿Bajo qué parámetros confirmaremos que las pruebas son Satisfactorias o no?

e) A Nivel Técnico.- Considerar lo siguiente:

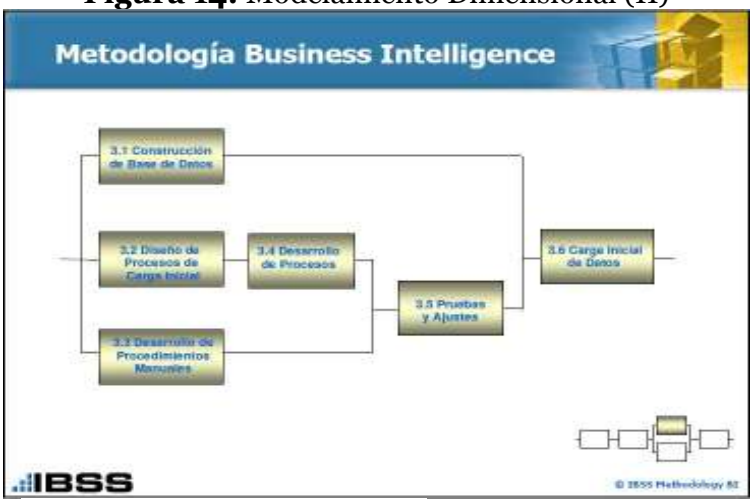
- ¿Qué características debo tener en cuenta para la plataforma de ETL?
- ¿La estructura de base de datos y estrategia de almacenamiento es la adecuada?
- ¿La plataforma está configurada para soluciones de BI?

f) Muchas veces una herramienta de ETL no es toda la solución. Se deben de considerar temas de integración de la herramienta de ETL con la plataforma de base de datos. Así mismo el uso de un ETL no debería demandar más hardware que el dimensionado para el adecuado manejo del repositorio de datos.

g) Actividades en el desarrollo del ETL.- Las actividades que se deben de realizar durante el desarrollo del ETL.

- Desarrollo unitario de cada uno de los programas de carga.
- Prueba unitaria de la secuencia del flujo de carga.
- Integración de los programas desarrollados.
- Prueba integral del Flujo.
- Optimización de procesos de carga.
- Prueba integral final.

Figura 14: Modelamiento Dimensional (II)



Fuente: Bermúdez (sin fecha)

2.5. Diseñador SSIS de Visual Studio 2005 para implementación de una Data Mart.

El Diseñador SSIS es una herramienta gráfica para crear paquetes que incluyen superficies de diseño con fichas independientes para generar el flujo de control, el flujo de datos y los controladores de eventos en paquetes, sus principales atributos son:

a) Ficha Flujo de control.- En la ficha Flujo de control, se organizan y configuran las tareas, incluida la tarea Flujo de datos, que proporciona funcionalidad en paquetes, los contenedores que proporcionan la estructura de los paquetes y servicio a las tareas y las restricciones de precedencia que conectan contenedores y tareas en un flujo de control. El menú contextual disponible en la superficie de diseño de Flujo de control permite agregar anotaciones de texto, establecer puntos de interrupción para la depuración y acercar o alejar el diseño del paquete. El menú contextual disponible en tareas individuales permite ejecutar las propias tareas, sin ejecutar todo el paquete.

b) Ficha Flujo de datos.- En la ficha Flujo de datos, se combinan en unos orígenes de flujo de datos que extraen datos, transformaciones que modifican y agregan datos, destinos que cargan datos y rutas de acceso que conectan las entradas y las salidas de los componentes de flujo de datos. El menú contextual disponible en la superficie de diseño de Flujo de datos también permite agregar anotaciones de texto. El menú contextual disponible en las rutas de acceso que combinan componentes de flujo de datos permite configurar visores de datos para examinar los datos según pasan por el flujo de datos.

c) Ficha Controladores de eventos.- En la ficha Controladores de eventos, se configuran flujos de trabajo para responder a eventos de paquetes. Por ejemplo, puede crear un controlador de eventos que envíe un mensaje de correo electrónico cuando se origine un error en una tarea.

d) Ficha Explorador de paquetes.- La ficha Explorador de paquetes proporciona una cómoda vista de explorador del paquete, con el paquete como un contenedor en la parte superior de la jerarquía y, debajo, las

conexiones, ejecutables, controladores de eventos, proveedores de registro, restricciones de precedencia y variables que ha configurado en el paquete.

e) Ficha Progreso. La ficha Progreso muestra información sobre la ejecución del paquete cuando se ejecuta en Business Intelligence Development Studio.

f) Área Administradores de conexiones. Integration Services utiliza administradores de conexiones para encapsular las conexiones en un origen de datos. Estos administradores de conexión se comparten en el paquete por los componentes del flujo de control, los componentes de flujo de datos y los proveedores de registro, y se muestran en un área especial del diseñador en la parte inferior de las fichas Flujo de control, Flujo de datos y Controladores de eventos.

El diseñador también proporciona acceso a los cuadros de diálogo, ventanas y asistentes que se utilizan para agregar funcionalidad y características avanzadas a paquetes y para solucionar problemas de paquetes.

2.5.1. Diseñar un flujo de control de paquetes

El flujo de control de un paquete de Integration Services se crea con diferentes tipos de elementos de flujo de control: los contenedores que proporcionan la estructura de los paquetes y servicios a las tareas que proporcionan la funcionalidad de los paquetes y las restricciones de precedencia que conectan los contenedores y las tareas en un flujo de control.
El flujo de control en un paquete se crea mediante el diseñador de flujo de control, la superficie de diseño en la ficha Flujo de control en el Diseñador SSIS.
Crear un flujo de control incluye las siguientes tareas:

- Agregar contenedores que implementan flujos de trabajo repetidos en un paquete o dividen un flujo de control en subconjuntos.
- Agregar tareas que admiten flujo de datos, preparan datos, realizan flujo de trabajo y funciones de inteligencia empresarial e implementan script.
- Conectar contenedores y tareas en un flujo de control ordenado mediante restricciones de precedencia.
- Si el flujo de control incluye tareas y contenedores que se conectan a orígenes de datos, también debe agregar administradores de conexión al paquete. Puede agregar administradores de conexión al trabajar en el diseñador de flujo de control, pero también puede agregarlos cuando las fichas Flujo de datos o Controladores de eventos están activas.

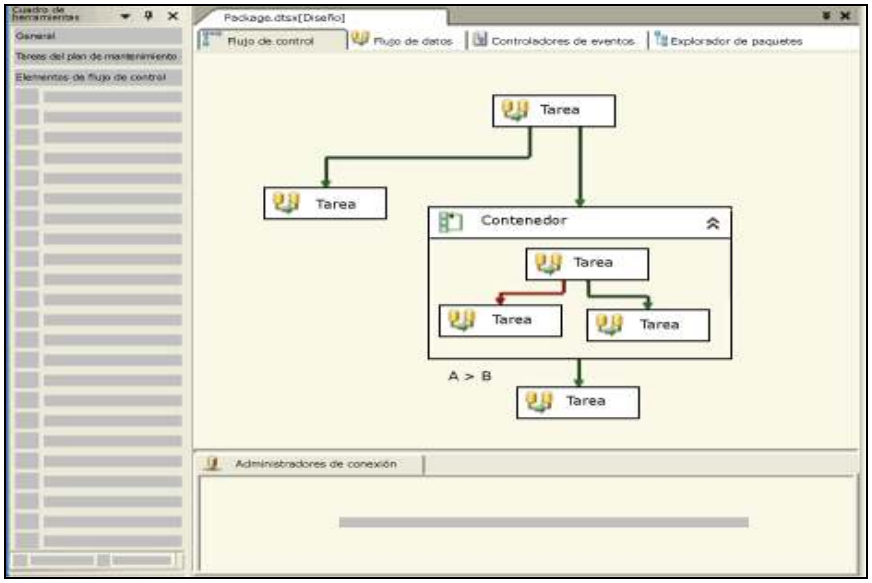
El Diseñador SSIS también incluye muchas características de tiempo de diseño que se pueden usar para administrar la superficie de diseño y hacer que el flujo de control se auto documente.

2.5.2. Uso del diseñador de flujo de control

Cuando la ficha Flujo de control está activa, el Diseñador SSIS muestra la superficie de diseño para crear el flujo de control en un paquete, el área Administradores de conexión le permite agregar o modificar los administradores de conexión que usa el paquete y el cuadro de herramientas

enumera los Elementos de flujo de control y las Tareas del plan de mantenimiento. El nodo Elementos de flujo de control del cuadro de herramientas enumera varios tipos de tareas y contenedores, mientras que el nodo Tareas del plan de mantenimiento enumera solamente tareas necesarias para mantener las bases de datos y trabajos de SQL Server.

Figura 15: Superficie de diseño para crear el flujo de control de paquetes



Fuente: Visual Studio BI Development 2005

En la figura 15 se muestra el flujo de control de un paquete simple en el diseñador de flujo de control. El flujo de control que se muestra se compone de tres tareas de nivel de paquete y un contenedor de nivel de paquete que contiene tres tareas. Las tareas y el contenedor se conectan mediante restricciones de precedencia.

2.5.3. Diseñar un flujo de datos de paquetes

Para generar el flujo de datos de un paquete de Integration Services se utilizan diferentes tipos de elementos de flujo de datos: orígenes que extraen datos, transformaciones que los modifican y agregan, destinos que los cargan y rutas de acceso que conectan las salidas y entradas de los componentes de flujo de datos en un flujo de datos.
Antes de empezar a generar un flujo de datos, un paquete debe incluir por lo menos una tarea Flujo de datos.
Se crea un flujo de datos en un paquete mediante el diseñador de flujo de datos, la superficie de diseño en la ficha Flujo de datos en el Diseñador SSIS.
Crear un flujo de datos incluye las siguientes tareas:

- Agregar uno o más orígenes para extraer datos de los archivos y bases de datos.
- Agregar las transformaciones que satisfacen los requisitos empresariales del paquete. No es obligatorio que un flujo de datos incluya transformaciones.

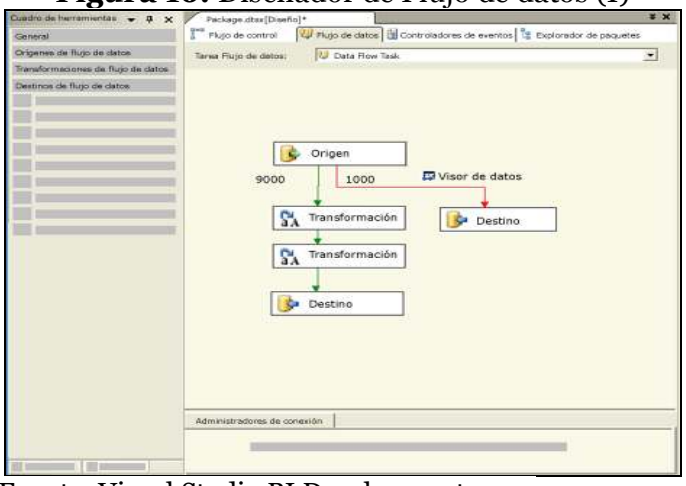
- Conectar componentes de flujo de datos conectando la salida de orígenes y transformaciones con la entrada de transformaciones y destinos.
- Agregar uno o más destinos para cargar datos en almacenes de datos tales como archivos y bases de datos.
- Configurar salidas de error en componentes para administrar problemas tales como errores o valores de datos truncados.
- Si el flujo de datos incluye componentes que se conectan a orígenes de datos, también debe agregar administradores de conexión al paquete. Puede agregar administradores de conexión mientras trabaja en el diseñador de flujo de datos, pero también puede agregarlos cuando las fichas Flujo de control o Controladores de eventos están activas.

Al crear un paquete nuevo, también puede utilizar un asistente para ayudarle a configurar correctamente los administradores de conexiones, los orígenes y los destinos.
El Diseñador SSIS también incluye anotaciones que se pueden usar para que el flujo de datos se auto documente.

2.5.4. Usar el diseñador de flujo de datos

Cuando la ficha Flujo de datos está activa, el Diseñador SSIS muestra la superficie de diseño para crear el flujo de datos en un paquete, y el área Administradores de conexión para agregar los administradores de conexión que usa el paquete. Mientras tanto, el cuadro de herramientas cambia para contener nodos para orígenes de flujo de datos, transformaciones de flujo de datos y destinos de flujo de datos.
El siguiente diagrama muestra el flujo de datos de un paquete simple en el diseñador de flujo de datos. El flujo de datos que figura en el diagrama se compone de un origen con una salida normal y una salida de error, dos transformaciones y dos destinos.

Figura 16: Diseñador de Flujo de datos (I)



Fuente: Visual Studio BI Development 2005

2.5.5 Metodología para ETL

2.5.5.1 Extraer datos

Integration Services proporciona diferentes orígenes para extraer datos de diferentes tipos de orígenes de datos. Un origen de Integration Services le permite extraer datos de archivos planos, archivos XML, libros de Microsoft Excel y archivos que contienen datos sin formato. También puede extraer datos obteniendo acceso a las tablas y vistas en las bases de datos y ejecutando consultas.
Una vez que un paquete de Integration Services contiene una tarea Flujo de datos, puede empezar a crear el flujo de datos del paquete. Integration Services incluye tres tipos de componentes de flujo de datos: orígenes, transformaciones y destinos. En la figura 17 se muestra un flujo de datos simple con un origen, dos transformaciones y un destino.

Figura 17: Diseñador de Flujo de datos (II)

Fuente: Visual Studio BI Development 2005

Para extraer datos de los archivos y bases de datos se usan los orígenes. Un flujo de datos puede incluir un solo origen o varios orígenes. Las salidas de orígenes pueden conectarse a transformaciones o destinos, también puede escribir orígenes personalizados. Para obtener más información, vea Desarrollar un componente de flujo de datos personalizado y Desarrollar tipos específicos de componentes de flujo de datos.

Después de agregar el origen al diseñador de flujo de datos y configurar el origen, puede conectar la salida del origen a la entrada de otro componente en el flujo de datos. Para agregar y configurar un origen al crear un paquete utilice el Asistente para importación y exportación de SQL Server o el Asistente para proyectos de conexiones de Integration Services.

Además de crear y configurar un origen, estos asistentes también ayudan a crear y configurar los destinos y los administradores de conexiones que los orígenes y los destinos usan. Los orígenes usan administradores de conexión para conectarse a los orígenes de datos. Puede agregar y configurar un administrador de conexión cuando configura el origen, o puede agregar los administradores de conexión necesarios al paquete antes de empezar a generar el flujo de datos.

2.5.5.2 Transformar datos

Integration Services proporciona una gama de transformaciones para modificar datos, realizar operaciones de inteligencia empresarial, así como para dividir, copiar y combinar datos. Mediante una transformación de Integration Services, puede modificar valores en columnas, buscar valores en tablas, limpiar datos y agregar valores de columna.
Algunas transformaciones usan administradores de conexión. Por ejemplo, la transformación Búsqueda usa un administrador de conexión para conectarse a la base de datos que contiene los datos de búsqueda. Puede agregar y configurar un administrador de conexión cuando configura la transformación, o puede agregar los administradores de conexión necesarios al paquete antes de empezar a generar el flujo de datos.
Las transformaciones de Integration Services ofrecen la siguiente funcionalidad:

- Dividir, copiar y combinar conjuntos de filas y realizar operaciones de búsqueda.
- Actualizar valores de columnas y crear nuevas columnas aplicando transformaciones tales como cambio de minúsculas por mayúsculas.
- Operaciones de inteligencia empresarial tales como limpiar datos, realizar minería de texto y ejecutar consultas de predicción de minería de datos.
- Crear nuevos conjuntos de filas que se componen de valores agregados u ordenados, datos de muestra o datos dinamizados y de anulación de dinamización.
- Realizar tareas tales como exportar e importar datos, proporcionar información de auditoría y trabajar con dimensiones de variación lenta, también puede escribir transformaciones personalizadas.

Después de agregar la transformación al diseñador de flujo de datos, pero antes de configurar la transformación, debe conectar la transformación al flujo de datos conectando la salida de otra transformación u origen del flujo de datos a la entrada de esta transformación. El conector entre dos componentes de flujo de datos se denomina ruta.

2.5.5.3 Cargar datos

Integration Services proporciona diferentes destinos para cargar datos en diferentes tipos de almacenes de datos. Mediante un destino de Integration Services, puede cargar datos en archivos planos, procesar objetos analíticos y proporcionar datos a otros procesos. También puede cargar datos obteniendo acceso a las tablas y vistas en las bases de datos y ejecutando consultas.
Los destinos usan administradores de conexión para conectarse a los orígenes de datos. Puede agregar y configurar un administrador de conexión cuando configura el destino, o puede agregar los administradores de conexión necesarios al paquete antes de empezar a generar el flujo de datos.

a) Conectar componentes.- Los componentes de flujo de datos se conectan conectando la salida de orígenes y destinos con la entrada de transformaciones y destinos. Al generar un flujo de datos normalmente

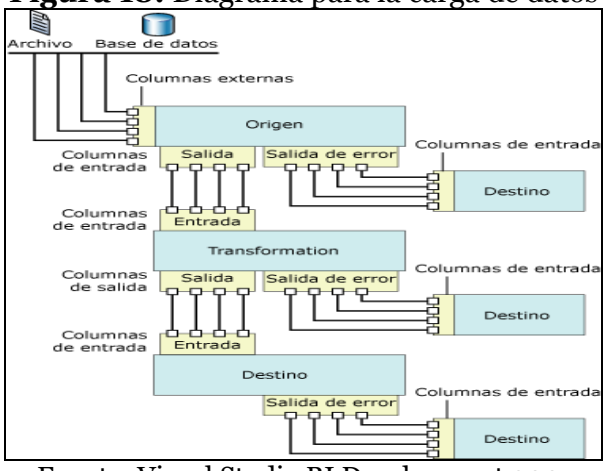
se conecta el segundo componente y los componentes subsiguientes a medida que se agregan al flujo de datos. Después de conectar el componente, las columnas de entrada están disponibles para su uso en la configuración del componente. Cuando no hay columnas de entrada disponibles, tiene que completar la configuración del componente después de conectarse al flujo de datos. Para obtener más información, vea Rutas de Integration Services y Conectar componentes con rutas de acceso.

b) Controlar errores de datos.- Muchos orígenes, transformaciones y destinos de Integration Services admiten salidas de error. Si el componente de flujo de datos admite una salida de error, se pueden especificar los efectos de los truncamientos o errores de cada columna sobre el comportamiento de tiempo de ejecución del componente. La salida de error se puede conectar a las transformaciones que aplican transformaciones adicionales o datos directos a un destino diferente. Para obtener más información, vea Control de errores en el flujo de datos.

c) Agregar administradores de conexión.- Muchos componentes de flujo de datos se conectan a orígenes de datos, y se deben agregar al paquete los administradores de conexión que necesitan los componentes antes de que el componente se pueda configurar correctamente. Puede agregar los administradores de conexión a medida que genera el flujo de datos o antes de empezar a generar el flujo de datos. Para obtener más información, vea Conexiones de Integration Services y Agregar administradores de conexión.

d) Anotaciones.- El Diseñador SSIS incluye anotaciones que se pueden agregar a un flujo de datos. Agregar anotaciones a la superficie de diseño ayuda a lograr que los paquetes se auto documenten. Para obtener más información.

Figura 18: Diagrama para la carga de datos



Fuente: Visual Studio BI Development 2005

2.6. Calidad de Datos

Amón y Jiménez (2009) nos dicen que los errores de digitación, datos inconsistentes, valores ausentes o duplicados, son algunos de los problemas que pueden presentar los datos almacenados en las bases y bodegas de datos, deteriorando su calidad y en consecuencia, la calidad de las decisiones que se tomen con base en el nuevo conocimiento obtenido a partir de ellos.

2.6.1 Definición de Calidad de Datos

Según Lozano (2000) nos dice que la definición de calidad de datos es relativa, está en los ojos del observador por lo que podemos considerar la calidad como un concepto multidimensional, sujeta a restricciones y ligada a compromisos aceptables.

Para Méndez (2006) una definición más amplia sobre la calidad de los datos es que la esta se consigue cuando se utilizan datos que son completos, consistentes, relevantes y oportunos, Si solo consideramos que la calidad de datos tiene que ver con malos datos, nos habremos contentado con una falsa seguridad cuando, de hecho, pronto descubrimos que el trabajo realizado y todos nuestros esfuerzos han sido en vano.

2.6.2. Dimensiones de la Calidad de Datos.

Lozano (2000), nos dice que se debe considerar tres tipos de categorías en las que pueden agrupar las dimensiones de la calidad (que son auto explicativas): dimensiones de calidad de la vista de los datos, dimensiones de calidad de la representación de los datos y dimensiones de calidad de los valores de datos.

English (1999), destaca dos tipos de cuestiones relativas a la calidad de los propios datos:

a) Calidad Inherente.- Es decir la precisión de los datos, el grado en que los datos reflejan exactamente los objetos del mundo real que se representan, que abarcaría: conformidad con la definición, compleción de valores, validez o conformidad con las reglas de negocio, precisión respecto a la fuente, precisión respecto a la realidad, no duplicación, accesibilidad.

b) Calidad Pragmática.- el grado en que los datos permiten a los "Trabajadores del conocimiento" satisfacer los objetivos de la empresa de forma eficaz y eficiente: oportunidad, claridad contextual, integridad de derivación, usabilidad, corrección o compleción de hechos.

También se analizan, desde principios ontológicos, algunas de las causas de la mala calidad de los datos debida a deficiencias en el diseño, identificando cuatro dimensiones de calidad, como se muestra en la siguiente tabla

Figura 19: Calidad de los Datos y deficiencias del diseño

Dimensión Calidad de Datos	Naturaleza de la deficiencia	Fuente de la deficiencia
Compleción	Representación impropia: Estados del SI[9] ausentes	Fallo en el diseño
No ambigüedad	Representación impropia: Varios estados del MR[10] mapeados al mismo estado SI	Fallo en el diseño
Significación	Estados SI sin sentido y confusión: mapeo a estado sin sentido	Fallo en el diseño y fallo en la operación
Corrección	Confusión: mapeo a un estado incorrecto	Fallo en la operación

Fuente: Lozano (1999)

Como se señala existe una relación unívoca que al operar con los datos no se obtengan los valores esperados y que se produzca una deficiencia de datos. Estas deficiencias como lo muestran los autores se dan en su mayoría por:

a) Deficiencias de Diseño.- Se produce una anomalía de este tipo cuando hay estados del mundo real que no se corresponden con un único estado correcto del sistema de información o viceversa. Nos encontramos con los siguientes tipos:

- Representación incompleta.- cuando hay estados del mundo real, que perteneciendo a la semántica de nuestro problema, se quedan sin representación en el SI.
- Representación Ambigua.- Se produce cuando dos o más estados del mundo real son representados por el mismo estado del SI.
- Estados Sin Sentido.- Ocurre cuando aparecen en el SI estados, que no están asociados a ningún estado del mundo real.

2.6.3. Los problemas de la calidad de datos

Kedad (2002), Cuando nos referimos a los problemas de DQ entre dos o más fuentes, sólo estamos pensando en los problemas que se producen en el plano extensional, es decir, problemas que están relacionados con los casos de los datos. También hay problemas que se producen en el nivel intencional, es decir, problemas que están relacionados con la estructura de los datos. Estos también son conocidos como los problemas entre los esquemas de bases de datos.

2.6.3.1. Problemas de origen de datos único

a) **Relación Individual.-** Los problemas que surgen dentro DQ una sola relación se organizan en cuatro grupos. DQ problemas que implican: (i) un valor de atributo de una sola tupla, (ii) los valores de un solo atributo, (iii) los valores de atributo de una sola tupla, y (iv) los valores de los atributos de tuplas varias.

i. **un valor de atributo de una sola tupla.-** El grupo más básico de los problemas encontrados DQ se compone de los problemas detectados mediante el análisis de sólo un valor de atributo de una sola tupla En este grupo, se identifican los siguientes problemas de DQ:

• Falta de valor - La falta de cumplimentación de un atributo necesario (por ejemplo, ausencia de valor en el atributo obligatorio Nombre o Denominación de un cliente).
• Violación de sintaxis - Existe una discrepancia entre la sintaxis valor del atributo y la sintaxis establecida para dicho atributo (por ejemplo, el atributo Order_ Fecha contiene el valor 13/12/2004, en lugar de 2004/12/13).
• superado el valor - El valor del atributo no se actualiza y no corresponde a la situación real (por ejemplo, las tiendas Dirección atribuir un valor que no es la dirección de los clientes actualizada).
• Intervalo de violación - No es una violación al intervalo de valores válidos de un atributo cuyo tipo es numérico (por ejemplo, el atributo Ordered_Quantity contiene un valor negativo).
• violación Set - En un atributo cuyo tipo es enumerado, hay una violación al conjunto de valores válidos (por ejemplo, el atributo Customer_Zone contiene un nombre de una ciudad, cuando los valores aceptables son sólo: Norte, Centro y Sur).
• Error de errores ortográficos - Este problema es consecuencia de un error de falta de ortografía, accidental o no, en un atributo cuyo tipo es textual (por ejemplo, el atributo almacena el valor Address_Place Gamdra, en lugar de Gandra).
• El valor inadecuada para el contexto atributo - Un valor de un atributo de texto no se ajusta en su contexto, pero encaja en el contexto de otro atributo textual (por ejemplo, el valor del atributo de dirección es el nombre del cliente).
• Los objetos de valor más allá del contexto atributo - Múltiples artículos introducidos todos juntos, como un valor de un atributo cuyo tipo es textual. Algunos de estos artículos van más allá del contexto de atributo (por ejemplo, en el valor del atributo de dirección también está el código postal, cuando un atributo de la debida para ello).
• El valor sin sentido - En esta situación, ni el valor ni ningún subconjunto de ella, de un ajuste de atributos de texto en su contexto, o en el contexto de otro atributo existente textuales (por ejemplo, el atributo de direcciones contiene el valor XYZ).
• Relación de significado impreciso o dudoso - Es una consecuencia de la utilización de abreviaturas o acrónimos en el valor de un atributo de texto. Pueden estar sujetos a diferentes interpretaciones con el paso del tiempo o de persona a persona (por ejemplo, la hormiga valor. Customer_Contact en atributo puede representar Anthony, Antonia, etc.)
• Violación de restricción de dominio - Violación de apremio relacionados con el atributo, inherente al dominio (por ejemplo, el nombre del atributo / Denominación de un cliente debe estar formado, al menos, por dos palabras).

ii. Los valores de un atributo individual.- Este grupo de problemas de DQ se compone de los problemas detectados mediante el análisis del conjunto de valores almacenados en un solo atributo (una columna), en este grupo, los siguientes problemas de DQ se identificaron:

• Violación valor Único - Dos o más tuplas que representan entidades diferentes tienen el mismo valor en un atributo que se suponía que era de un valor único (por ejemplo, dos clientes distintos poseen el mismo número de identificación fiscal).
• Existencia Sinónimos - El uso arbitrario de valores sintácticamente diferentes con el mismo significado semántico (por ejemplo, la relación Profesiones simultáneamente contiene dos tuplas, una con el profesor y la otra con la maestra).
• Violación de restricción de dominio - Violación de apremio relacionados con el atributo de participación de los valores que puede asumir en varias tuplas. La limitación es inherente a la secuencialmente, ordenados por orden ascendente).

iii. Los valores de los atributos de una sola tupla.- Este grupo de problemas de DQ comprende los problemas detectados mediante el análisis del conjunto formado por los valores de atributo de una sola tupla (fila), se identificaron los siguientes problemas de DQ en este grupo:
• Semi-tupla vacía - En esta situación, un gran número de atributos de tuplas no se cumplen. Si un determinado umbral (definido por el usuario) se supera la tupla se clasifica como semi-vacío (por ejemplo, si 60% o más de los atributos de tuplas están vacías la tupla se considera semi-vacío).
• La inconsistencia entre los valores de los atributos - No es una violación a una dependencia existente entre los valores de los atributos tupla (por ejemplo, en una tupla dada de la relación Sales_Details, la dependencia entre los siguientes valores de los atributos se viola: Total_Product = Quantity_Product Sell_Price_Product *).
• Violación de restricción de dominio - Violación de la restricción de la participación, dos atributos tupla, por lo menos. La restricción es inherente al dominio (por ejemplo, cuando el envío del producto (s) se hace por el cliente, los gastos de envío no se cobran).

iv. Los valores de los atributos de tuplas Varios.-Este grupo de problemas de DQ incluye los problemas detectados mediante el análisis del conjunto de valores de los atributos almacenados en varias tuplas, este grupo está compuesto por los siguientes problemas de DQ:
• Redundancia de una entidad - La misma entidad está representada por una igual o equivalente representación en más de una tupla (por ejemplo, el cliente tupla (10, 'Smith Barney', 'Flores de la calle ", 123, 502 899 106) es equivalente al Cliente tupla (72 'S. Barney', 'Flores St., 123, 502 899 106)).
• La inconsistencia de una entidad - Hay inconsistencias o contradicciones entre los valores de uno o más atributos de una misma entidad, representada en más de una tupla (por ejemplo, el cliente tupla (10, 'Smith Barney', 'Flores de la calle, 123, 502 899 106) es incompatible con el Cliente tupla (72, 'Smith Barney "," Sun Street., 321, 502 899 106)).
• dominio violación limitación - Violación de apremio relacionada con la relación en su conjunto.La limitación es intrínseca al

dominio (por ejemplo, en relación Product_Families el número máximo de familias de productos autorizados (tuplas) es 10).

b) **Las relaciones entre las múltiples relaciones.-** En esta sección se presenta todos los problemas DQ en un conjunto de relaciones, cuando las relaciones entre ellos existen.
• Violación de integridad referencial.- En un atributo tupla que es clave externa no es un valor que no existe como clave principal en la relación relacionados (por ejemplo, el Customer_Zip_Code atributo de la relación del cliente contiene el valor 4415-206, que no existe en el relación zip_code).
• Referencia superado - A pesar de la integridad referencial se respete, el valor de clave externa de una tupla no se actualiza y no corresponde a la situación real (por ejemplo, el Customer_Zip_Code atributo de la relación del cliente contiene un valor que no es el código postal del cliente actualizado).
• Inconsistencia de sintaxis - En función de la relación, hay sintaxis diferentes de representación entre los atributos cuyo tipo es el mismo (por ejemplo, en relación con el atributo órdenes Order_Date tiene la sintaxis dd / mm / aaaa, mientras que en relación con las facturas, el atributo Invoice_Date tiene la sintaxis aaaa / mm / dd).
• La inconsistencia entre los valores de los atributos relacionados - Existen inconsistencias entre los valores de los atributos de las relaciones donde existe una relación entre ellos (por ejemplo, en relación con las facturas el atributo de una tupla Invoice_Total contiene el valor 100, mientras que la suma de valores de atributos Product_Value, en Invoices_Details relación, porcada uno de los productos que pertenecen a esa factura sólo es igual a 90).
• La circularidad entre tuplas en una relación consigo mismo - Corresponde a las situaciones de ciclo entre los dos (circularidad directa) o más (circularidad indirectos) relacionados con tuplas en un auto / reflexiva-relación (por ejemplo, supongamos que un producto puede ser sub-producto en otro producto y que esta información se almacena en el atributo Sub-product_Cod del producto; pero no se logra registrar en la dependencia del producto) .

2.7. Metodología de Amón y Jiménez

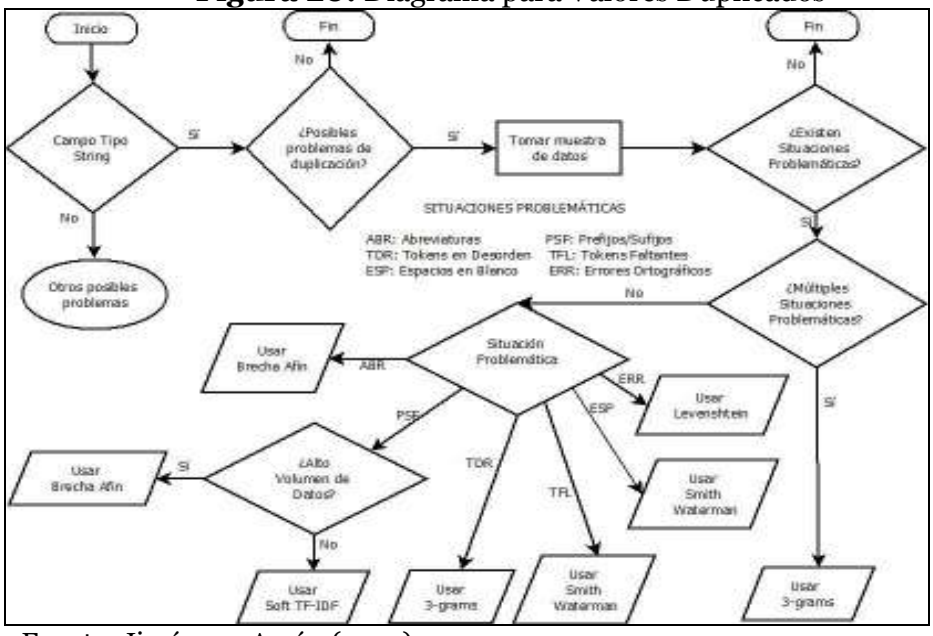
Es una realidad que parte de los datos almacenados por las organizaciones, contienen errores y estos pueden conducir a tomar decisiones erróneas, ocasionando pérdidas de tiempo, dinero y credibilidad. Esta situación ha capturado la atención de los investigadores, llevando al desarrollo de múltiples técnicas para detectar y corregir los problemas en los datos, pero no es trivial decidir cuáles técnicas deben aplicarse a un conjunto de datos particular de la vida real, La guía metodológica construida en este trabajo, orienta la selección de técnicas para tres de los posibles problemas que pueden presentar los datos: detección de duplicados, valores atípicos incorrectos y valores faltantes. (Amón y Jiménez, 2010).

Esta metodología se basa principalmente en resolver 3 tipos de problema en la calidad de los datos, como a continuación se muestra:

2.7.1. Guía Metodológica para la selección de técnicas para la detección de duplicados.

El principal aporte de esta metodología, es proveer guías que orienten a los analistas de datos en la selección de las técnicas más apropiadas para la situación particular que pueda presentar un cierto conjunto de datos. La figura 20 presenta el algoritmo guía para el problema de la detección de duplicados, mediante un diagrama de flujo de datos. El diagrama comienza indagando si el tipo de datos de una columna dada a la cual se desee hacer limpieza, es de tipo String. Aunque la duplicación no es un problema exclusivo de este tipo de datos, las técnicas están diseñadas para ser aplicadas a cadenas de texto. Otros tipos de datos a los cuales se desee hacer análisis de duplicación, requerirán ser convertidos previamente y por tanto se vuelve a la condición inicial. Para campos que tengan otros tipos de datos, se requeriría buscar otros tipos de problemas (por ejemplo: valores atípicos para campos numéricos). Para campos tipo texto, se interroga sobre la existencia de posibles problemas de duplicación. Aunque aparentemente es una pregunta difícil de responder por parte de un usuario, realmente no lo es tanto. Para un usuario conocedor de los datos y que entienda el concepto de detección de duplicados, no es difícil prever si sus datos son susceptibles a esta situación. Recuérdese que se habla de detección de duplicados cuando el contenido de un campo o de un registro completo, aparece dos o más veces (duplicado) con diferencias textuales en sus valores y no se tiene un identificador único. Si por ejemplo, en una tabla se almacenan datos de proyectos de investigación, en un proyecto pueden participar varios investigadores y no existe un identificador único como un código que se le asigne previamente a cada proyecto, es fácil que cada investigador entre el título del proyecto con alguna variación en el texto, haciendo que pueda considerarse como un proyecto distinto.

Figura 20: Diagrama para Valores Duplicados



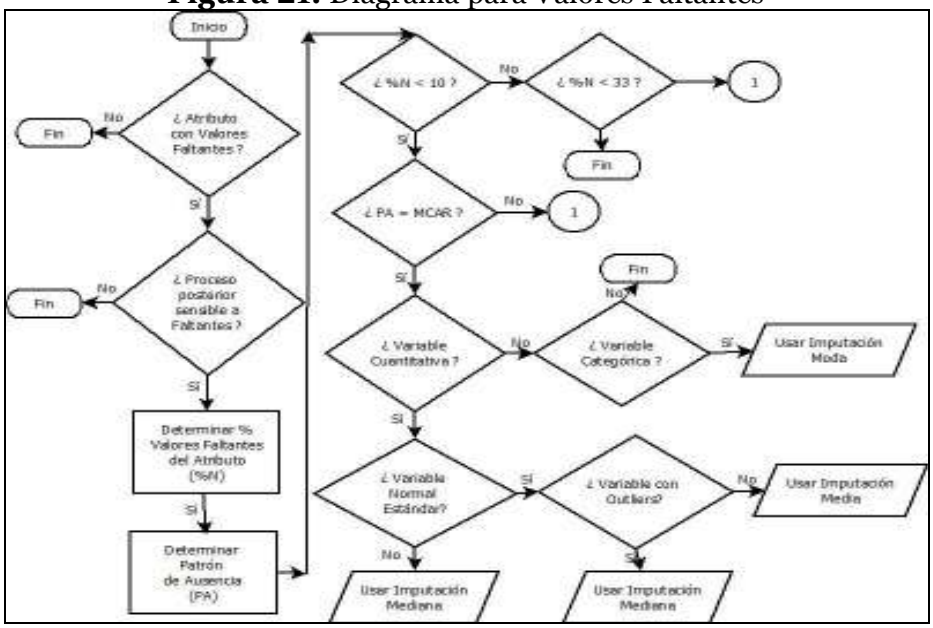
Fuente: Jiménez y Amón (2010)

En caso de existir posibles problemas de duplicados, se sugiere tomar una muestra de datos para tomar decisiones con base en ella. Esto es necesario si se tiene un alto volumen de datos, pues de lo contrario se analiza todo el conjunto de datos. Además, la muestra debe ser representativa de la totalidad de los datos. Sobre la muestra de datos, debe examinarse si se detecta alguna(s) de las seis situaciones problemáticas, es decir, si en los datos se visualizan errores ortográficos, abreviaturas, tokens faltantes o en desorden, presencia de prefijos y sufijos o espacios en blanco adicionales o suprimidos. En caso de que detecte una o más de estas situaciones problemáticas, debe establecerse si se perciben varias de ellas simultáneamente o si hay un alto predominio de sólo una de ellas. En caso de predominar una sola situación problemática, se recomienda la técnica de mayor eficacia excepto para el caso de PSF en el que la función recomendada dependerá del volumen de datos (la técnica Soft TF-Idf es muy pesada computacionalmente y por tanto se recomienda sólo para un volumen bajo de datos). En caso de observarse en los datos varias situaciones problemáticas simultáneamente, se recomienda la técnica tri-grams ya que es de bajo costo computacional.

2.7.2. Guía Metodológica para la Selección de las Técnicas para Valores Faltantes.

Al igual que para la detección de duplicados, la guía tiene la forma de un diagrama de flujo de datos. Siguiente figura corresponde al diagrama guía para la selección de las técnicas para Valores Faltantes.

Figura 21: Diagrama para Valores Faltantes



Fuente: Amón y Jiménez (2010)

El proceso descrito en la Figura 21 comienza indagando si el atributo a limpiar (la guía debe seguirse por cada atributo que requiera limpiarse), contiene valores faltantes. Acá debe tenerse presente que no todo valor nulo es un

faltante pues de acuerdo con la naturaleza del dato, es posible que éste no sea obligatorio y por tanto no sea un error el que no tenga valor. Luego, se pregunta si el proceso a realizar con los datos es sensible a los datos faltantes. Se debe tener claro lo que se pretende hacer con los datos. Si por ejemplo, se piensa realizar un proceso de minería utilizando árboles de decisión, estos pueden trabajar con un cierto nivel de ruido y datos faltantes, lo que no sucede con las técnicas de agrupamiento.

Después de realizar lo anterior, se debe determinar el porcentaje de valores faltantes que contiene el atributo que se está examinando. Esta labor se puede realizar utilizando herramientas de perfilamiento de datos (Data Profiling) disponibles tanto en forma comercial como libre. De igual forma, determinar el patrón de ausencia de los datos puede realizarse con la ayuda de herramientas de software o con un análisis detallado de los datos previo entendimiento de los patrones de ausencia MCAR, MAR y MNAR explicados en este documento.

La imputación usando la media se recomienda sólo cuando el porcentaje de faltantes es bajo (menor a 10%), el patrón de ausencia es MCAR, se trata de una variable numérica, la variable tiene una distribución normal estándar (simétrica y mesocúrtica) y no hay presencia de valores atípicos (outliers). Determinar si una distribución es normal estándar, puede hacerse con la ayuda de programas estadísticos. Si el porcentaje de faltantes es bajo (menor a 10%), el patrón de ausencia es MCAR, se trata de una variable numérica pero no se está en presencia de una distribución normal estándar y/o hay presencia de valores atípicos, la técnica recomendada es la imputación usando la mediana. Para variables categóricas como género o estado civil, se recomienda la imputación usando la moda. Hot Deck, es la técnica de imputación que la guía recomienda para mejorar la calidad de los datos si el patrón de ausencia es MAR, se cuenta con un volumen alto de datos por grupo y existen otras variables correlacionadas. Un volumen alto de datos por grupo se refiere a que existan suficientes datos completos con las mismas características (por ejemplo igual género, años de estudio y estado civil) para servir como donantes.

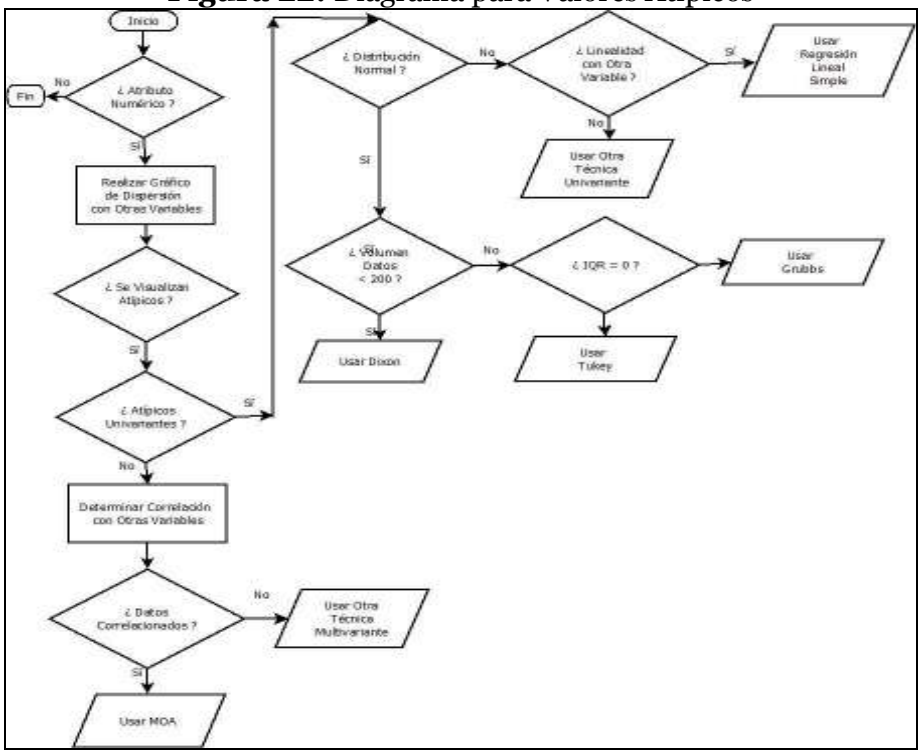
Para determinar la correlación entre variables también se puede utilizar un paquete estadístico. Si no se tiene un volumen alto de datos por grupo y se trata de una variable cuantitativa se recomienda imputación por regresión. Si la variable no es cuantitativa deberá acudirse a otra técnica no examinada en este trabajo. Por último, para porcentajes de valores faltantes entre 10% y 33%, se descartan las técnicas de imputación de la media, mediana y moda recomendables sólo para bajos niveles de faltantes. En esta situación, se consideran sólo las técnicas Hot Deck y Regresión bajo las mismas consideraciones anteriores. Para porcentajes de valores faltantes superiores a 33%, no se recomienda hacer imputación ya que se “fabricarán” demasiados valores. Este límite se establece de acuerdo con lo expresado por Laaksonen quien considera como tasa de no-respuesta elevada cuando dicha tasa supera un tercio del total.

2.7.3. Guía Metodológica para Selección de Técnicas para Detección de Valores Atípicos

La siguiente figura presenta la guía para el problema de la detección de valores atípicos, mediante un diagrama de flujo de datos. El diagrama comienza indagando si el tipo de datos de un atributo al cual se desee hacer limpieza, es de tipo numérico, ya que las técnicas para detección de atípicos operan sobre este tipo de datos. Para atributos numéricos, la guía solicita realizar gráficos de dispersión con otras variables. Esta tarea, fácilmente realizable en Excel o en algún paquete estadístico, permite detectar visualmente valores alejados de los demás, en cuyo caso la respuesta a la pregunta ¿Se visualizan atípicos? sería verdadera en caso contrario, el proceso termina.

A continuación, se indaga si los datos atípicos son univariantes. Para dar respuesta a esta pregunta el usuario debe entender el concepto de atípico univariante y multivariante. Recuérdese que un valor atípico univariante es un punto de datos que es muy diferente al resto para una sola variable. Un valor atípico multivariante, es un caso que es un valor extremo para una combinación de variables. Por tanto, el usuario debe evaluar si los datos son atípicos vistos individualmente o si lo son porque no son valores normales en relación con los valores que toman otras variables. Por ejemplo un salario de dos millones de pesos colombianos puede no ser considerado atípico, pero seguramente lo es si corresponde a una persona de 15 años de edad, en cuyo caso se estaría en presencia de un atípico multivariante.

Figura 22: Diagrama para Valores Atípicos



Fuente: Amón y Jiménez (2010)

La única técnica disponible, entre las evaluadas, que es capaz de tratar atípicos multivariantes, es MOA (Mahalanobis Outlier Analysis). Esta técnica sólo debe aplicarse si la variable bajo análisis está correlacionada con otras variables ya que la distancia de Mahalanobis toma en cuenta dicha correlación, por tanto esta es la técnica recomendada para esa situación. Para atípicos multivariantes, pero sin correlación con otras variables, debe aplicarse alguna otra técnica no estudiada en este trabajo.

Para la situación de atípicos univariantes, si se distribuyen normalmente y son pocos datos (menos de 200), se recomienda la prueba de Dixon. Para una cantidad de datos superior, debe verificarse si el IQR o rango intercuantil es cero, ya que en ese caso todos los valores diferentes de cero serán catalogados como atípicos. Si el IQR es diferente de cero, puede usarse la prueba de Tukey; si es igual a cero puede usarse la prueba de Grubbs. Para atípicos univariantes que no se distribuyen normalmente, si existe linealidad con otra variable, se recomienda regresión lineal simple. Si no existe linealidad, debe aplicarse alguna otra técnica no estudiada en este trabajo.

2.7.4. Limpieza de Datos

Los datos faltantes hacen referencia a la ausencia de un valor para un atributo requerido, esto es, sólo puede hablarse de datos faltantes cuando se trata de la ausencia de un valor en un atributo obligatorio y por tanto no todo campo vacío es realmente un problema de calidad en los datos (Oliveira et. al., 2005).

Aunque la imputación tiene sus detractores por considerar que se están "inventando datos", es ampliamente reconocida entre la comunidad estadística como un método, que bien aplicado, puede mejorar la calidad de los datos. En la literatura se han reportado diferentes técnicas de imputación, se debe evitar la no respuesta en la medida posible, para usar imputación sólo cuando sea absolutamente necesario, pues nunca unos datos imputados serán mejores que unos datos reales (Useche y Mesa, 2006) citados por Amón y Jiménez (2010).

Autores como Medina y Galván (2007) y Cañizares et. al. (2004) citados por Amón y Jiménez (2010), Coinciden en que antes de analizar la técnica de imputación a aplicar sobre un conjunto de datos, es necesario identificar primero el mecanismo o patrón que describe la distribución de los datos faltantes y para ello utilizan la clasificación hecha por Little y Rubin (1987) citado por Amón y Jiménez (2010) la cual se basa en la aleatoriedad con que se distribuyen los valores faltantes. Estos autores definen tres tipos de patrones: datos ausentes completamente al azar (Missing Completely At Random, MCAR), datos ausentes al azar (Missing At Random, MAR) y datos ausentes no al azar (Missing Not At Random, MNAR), Medina y Galván (2007) citado por Amón y Jiménez (2010), a continuación explican en detalle los tres patrones:

Los datos siguen un patrón MCAR cuando los objetos tuplas o registros, para el caso de una tabla en una base de datos relacional- con los datos completos son similares a los de los datos incompletos; es decir, los objetos con datos incompletos constituyen una muestra aleatoria simple de todos los sujetos que conforman la muestra. Pensando los datos como una gran matriz, los valores ausentes están distribuidos aleatoriamente a través de la matriz. Sirva de ejemplo, una encuesta nacional en la cual se necesitan estudios costosos, como

los electrocardiogramas; podría entonces seleccionarse una sub muestra mediante muestreo aleatorio simple de los encuestados, para que se aplique este examen (Medina y Galván, 2007) citado por Amón y Jiménez (2010).
El patrón MAR se presenta cuando los objetos con información completa difieren del resto. Los patrones de los datos faltantes se pueden predecir a partir de la información contenida en otras variables –atributos o campos- y no de la variable que está incompleta. Un ejemplo de MAR lo presentan Useche y Mesa (2006) citado por Amón y Jiménez (2010): En un estudio de depresión maternal, 10% o más de las madres puede negarse a responder preguntas acerca de su nivel de depresión.

Supóngase que el estudio incluye el estado de pobreza, el cual toma los valores 1 para pobreza y 0 para no pobreza. El puntaje de las madres en cuanto a la depresión es MAR, si los valores faltantes de la depresión no dependen de su nivel de depresión. Si la probabilidad de negarse a responder está relacionada con el estado de pobreza mas no con la depresión dentro década nivel del estado de pobreza, entonces los valores ausentes son MAR. El asunto no es si el estado de pobreza puede predecir la depresión maternal, sino si el estado de pobreza es un mecanismo para explicar si una madre reportará o no su nivel de depresión (patrón de ausencia). Bajo este patrón la distribución depende de los datos pero no depende de los datos ausentes por sí mismos y es asumida por la mayoría de los métodos existentes para imputación de datos ausentes [Little y Rubin, 1987] citado por Amón y Jiménez (2010). En el caso de MCAR, la suposición es que las distribuciones de los datos ausentes y completos son las mismas, mientras que para MAR ellas son diferentes y los datos ausentes pueden predecirse usando los datos completos (Shafer, 1997) citado por Amón y Jiménez (2010).

En el caso MNAR, el patrón de los datos ausentes no es aleatorio y no se puede predecir a partir de la información contenida en otras variables. Bajo este patrón, contrario al MAR, el proceso de ausencia de los datos sólo se explica por los datos que están ausentes (p. ej., un ensayo sobre la pérdida de peso en que un participante abandona el estudio debido a preocupaciones por su pérdida de peso). La distribución depende de los datos ausentes y es raramente usada en la práctica.

Para los casos de datos faltantes o nulos existen técnicas de imputación que ayudarán a su limpieza, estas son: Imputación usando la Media, Imputación usando la Mediana, Imputación Hot Deck, Imputación por Regresión simple.

a) **Imputación Media (IM).-** Es la imputación usando la media, la media aritmética de los valores de una variable que contiene datos faltantes es usada para sustituir los valores faltantes (Farhangfar et. al., 2007) citado por Amón y Jiménez (2010).
   Según Myrtveit et. al. citado por Amón y Jiménez (2010), la imputación de la media (IM) es probablemente la técnica más ampliamente usada y la motivación para seleccionarla es su rapidez computacional. Anderson et. al. Afirman que en el caso de una distribución normal, la media muestral provee un estimado óptimo del valor más probable. En el mismo sentido, Mcknight et. al. Afirman "la media es el valor más probable de las observaciones cuando los datos se distribuyen normalmente y por lo tanto sirve como estimación para los valores

faltantes.Esta técnica consiste en en modificar los datos faltantes usando la media aritmética de los valores que se depositan en la columna donde se encuentra el problema. La fórmula de la media aritmética es la siguiente:

$$\bar{X} = \frac{X_1 + X_2 + X_3 + \ldots + X_n}{N}$$

b) **Imputación Hot Deck.-** Es un procedimiento no paramétrico basado en la suposición de que los individuos cercanos en un mismo espacio tienen características similares. Reemplaza los valores ausentes en una observación por aquellos de otra(s) observación(es) de alguna forma cercana a ella, de acuerdo con una idea predefinida de cercanía en el espacio de las variables comunes X. Un elemento importante a considerar es el número de vecinos. Si el número de vecinos es pequeño, la estimación se hará sobre una muestra pequeña y por lo tanto el efecto será una mayor varianza en la estimación. Por otro lado, si la imputación se hace a partir de un número grande de vecinos, el efecto puede ser la introducción de sesgo en la estimación por información de individuos alejados.

c) **Prueba de Dixon.-** La prueba de Dixon consiste en saber que registros contienen valores atípicos o incoherentes o también llamados outlier's. El método define la relación entre la diferencia del mínimo/máximo valor y su vecino más cercano y la diferencia entre el máximo y el mínimo valor aplicado (Li y Edwards, 2001) citado por Amón y Jiménez (2010). Esto quiere decir que se va a comparar la forma en que se obtiene el valor del registro con valores atípicos con sus vecinos.

III. MATERIALES Y MÉTODOS

El presente trabajo consistió en aplicar una nueva metodología para asegurar la calidad de los datos durante el proceso ETL, esta metodología propuesta por el PHD. Claudia Jiménez y el Mgtr. Ivan Amón nos orienta sobre que técnicas utilizar frente a diversos tipos de errores en los datos que vamos a cargar en una base de datos destino.

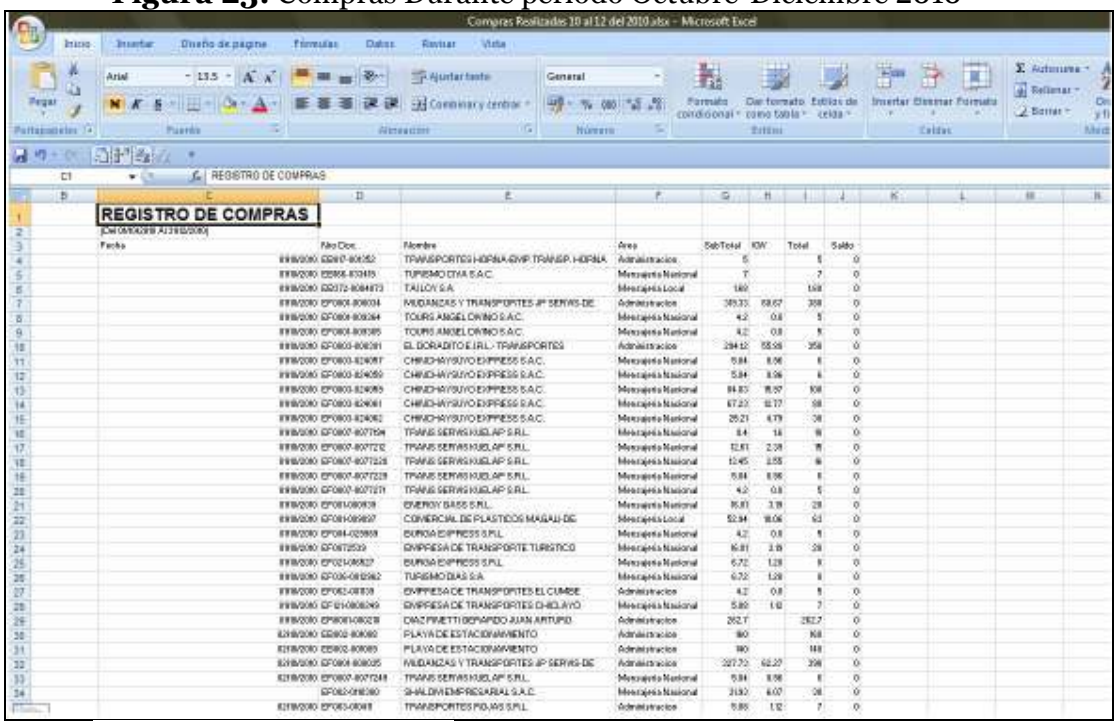
Los autores, en su tesis maestra proponen analizar cada uno de los registros que se crean que los datos estén sucios, existan datos faltantes o haya incoherencia entre ellos, porque no todos los datos que se crean estén sucios o incoherentes lo deben de estar, por ejemplo si encontramos en una base de datos que un trabajador tiene como sueldo S/.15,000 soles, no necesariamente tiene que ser un dato incoherente, porque puede haber gerentes que tengan ese sueldo, pero se vuelve incoherente cuando este sueldo le corresponde a un trabajador que realiza las tareas de limpieza y que tiene un horario de trabajo de medio tiempo.

Los autores de esta guía metodológica plantean que una vez analizado cada uno de los registros que cuenten con alguna anomalía, estos deben pasar por distintas técnicas de depuración, ellos nos proponen que se utilice una técnica para cada tipo de anomalía que se encuentren en los datos, por ejemplo si existen datos faltantes estos se pueden mejorar usando una técnica de imputación que puede ser la moda, la mediana o la que se crea conveniente, lo cual va a ser muy distinto para un dato incoherente, debido a que cada registro tiene su particularidad y responden a distintos valores dentro de una misma base de datos.

El desarrollo de este trabajo consistió en comparar la metodología y técnicas de depuración de datos, que nos proporciona Visual Studio 2005 BI Development y la propuesta por Jiménez y Amón, para ello desarrollamos primero el proceso ETL bajo las recomendaciones proporcionadas por ambos y luego comparamos los resultados, en los cuales vamos a evaluar el tiempo que toma desarrollar cada uno de los pasos, el número de datos erróneos que se encuentren y la capacidad de depuración y limpieza de los datos de ambas metodologías.

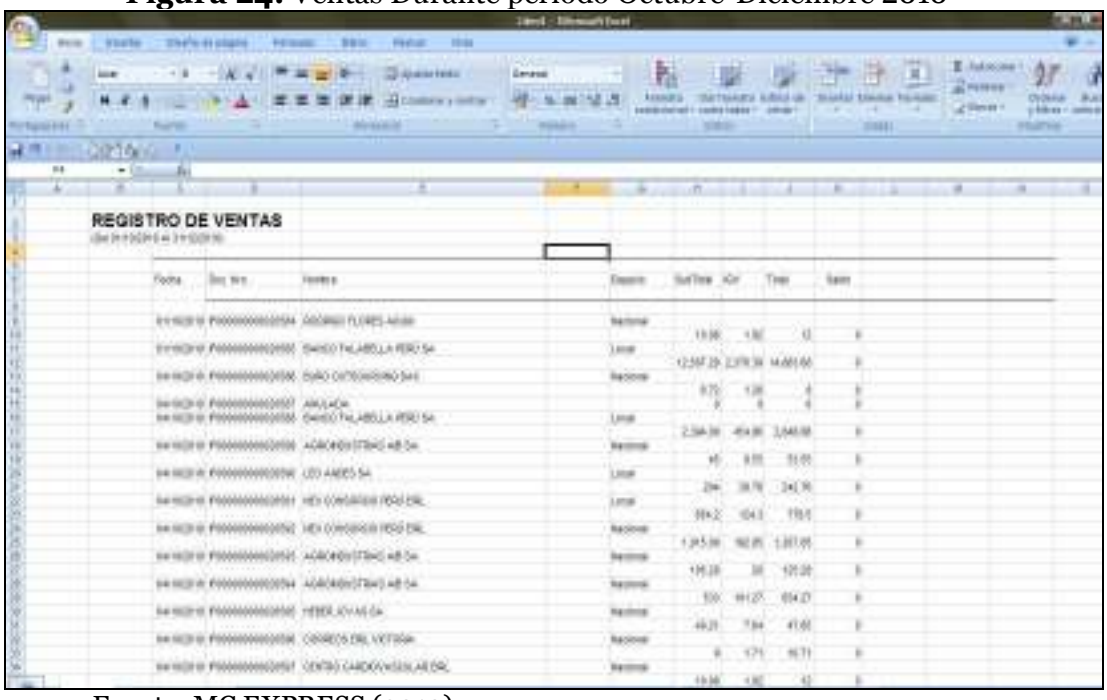
Para lograr nuestros objetivos hemos trabajado directamente en la tabla de hechos de los data mart, cabe resaltar que los datos de origen provienen de dos sistemas, el sistema de Ventas y el sistema de Compras de la empresa MC EXPRESS, los cuales, por motivo de normas de seguridad de la empresa, solo nos proporcionó los datos de los sistemas en libros de EXCEL.

Figura 23: Compras Durante periodo Octubre-Diciembre 2010



Fuente: MC EXPRESS (2010)

Figura 24: Ventas Durante periodo Octubre-Diciembre 2010



Fuente: MC EXPRESS (2010)

Los datos que nos pudieron proporcionar pertenecen al sistema denominado Sistema Administrativo de la empresa, en donde distribuyen su carga en un subsistema de compras y otro de ventas.

Estos datos se recogieron a través de múltiples coordinaciones con la administradora de la sucursal Chiclayo de la empresa, quien amablemente nos concedió la entrevista con el encargado de dar mantenimiento a los sistemas anteriormente descritos.

Fueron varias las entrevistas que se llevaron a cabo con el encargado de los sistemas de la empresa, quien como ya hemos mencionado anteriormente nos brindó la data registrada en la base de datos, la cual por cuestiones de seguridad y fidelidad planteadas por la empresa solo pudimos acceder a ella a través de reportes exportados en registros Excel, la cual contenía data de los 3 últimos meses del año 2010.

Los datos con lo cual se ha trabajado corresponden a los meses octubre, noviembre y diciembre de 2010. Una vez que obtuvimos los datos necesarios para esta tesis, la forma que se trabajó fue siguiente:

- Primero se analizó y diseñó el proceso ETL para los data mart Ventas y Compras.
- El ETL se desarrolló bajo 2 metodologías de desarrollo, bajo el enfoque de Visual Studio, la cual nos proporciona herramientas y técnicas de depuración de datos. Primero analizamos esta metodología haciendo la carga de datos de manera simple y luego usando las herramientas que nos proporciona dicha metodología.
- Luego usando la misma plataforma de Visual Studio aplicamos el enfoque de Jiménez y Amón, para asegurar la calidad de datos. Primero lo realizamos paso por paso como se recomienda y luego lo desarrollamos en un solo paso, para lo cual vemos las mejoras en el tiempo.
- Por último comparamos los resultados de las metodologías, en cuadros resúmenes.

Para llegar hasta el objetivo de comparar los resultados primero se trabajó en los siguientes puntos:

- Definir el Star Net para el Sistema de Ventas.
- Definir el Star Net para el Sistema de Compras.
- Realizar el modelamiento dimensional para data mart Compras.
- Realizar el modelamiento dimensional para data mart Ventas.
- Diseñar el modelo lógico del Data mart Compras.
- Diseñar el modelo lógico del Data mart Ventas.
- Definir las herramientas y plataformas para ejecutar el proceso ETL.
- Ejecutar el proceso ETL y Depuración de Datos en Data Mart Ventas.
- Ejecutar el proceso ETL y Depuración de Datos en Data Mart Compras.

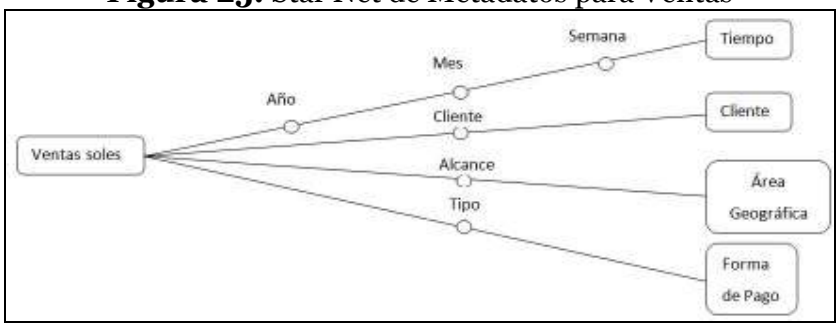
Para lograr la depuración de los datos tal y como lo propone la metodología de Jiménez y Amón, se tuvo que adaptar las técnicas de depuración que ellos proponen para cada uno de los registros anómalos que se encuentren, los cuales en su mayoría usan algoritmos propios de sistemas estadísticos y matemáticos, se tuvo que estudiar en profundidad el comportamiento de dichas técnicas para luego ser adaptadas en comandos SQL, que luego fueron ejecutados para limpiar cada uno de los registros anómalos que se encontraron.

IV. RESULTADOS

1. Definición del Star Net para el Sistema de Ventas

Luego de obtener los datos en las tablas de Excel, pasamos a establecer la estructura de los metadatos, con la finalidad de establecer la tabla de hechos y las dimensiones que va a tener el datamart de ventas. Después de haber evaluado los metadatos se diseño la forma en que se van a ser consultados, se definió tal como lo muestra la siguiente figura.

Figura 25: Star Net de Metadatos para Ventas

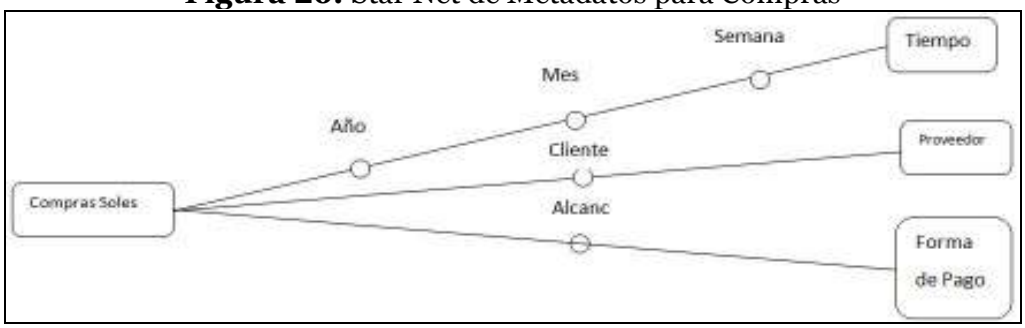


En donde los principales metadatos encontrados fueron tiempo, el cual es importante para medir los ingresos según el año, el mes y la semana, otro metadato de importancia fue el cliente, en donde se puede medir cual es el promedio de compra de cada cliente según un determinado periodo. El área geográfica fue otro metadato destacable, debido a que a la empresa le interesa cual agencia es la que más ingresos produce, de esa forma puede evaluar la producción de cada una de ellas y tomar la mejor decisión. La forma de pago es otro aspecto a evaluar, debido a que les interesa saber en qué periodos se trabaja más a crédito y que otros se trabaja al contado, con la finalidad de establecer acuerdos con sus clientes para dichos periodos.

2. Definición de Star Net para el Sistema de Compras

Al igual como lo hicimos para las ventas, en este punto tomamos los datos proporcionados por la empresa, luego de la evaluación respectiva establecimos los metadatos tal y como se muestran en la siguiente figura.

Figura 26: Star Net de Metadatos para Compras



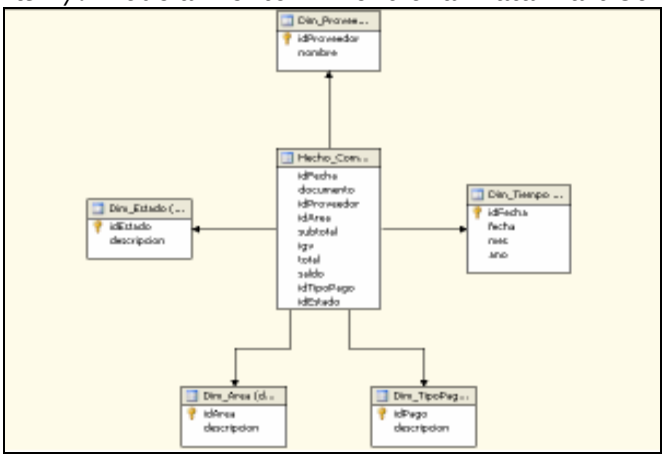
Al igual que para ventas, el tiempo aparece como un metadato muy importante, debido a que les servirá para evaluar sus gastos según el año, el mes y la semana, y de esa forma realizar un plan de adquisiciones y pagos a proveedores.

Así mismo otro metadato de importancia fue el proveedor, que junto con el tiempo se puede establecer al proveedor que más se le compra, de esa forma se puede evaluar algún trato comercial con ellos. La forma de pago ayudará a la empresa a establecer su plan de pagos, porque a través de este metadato se sabrá a quienes se les compra a crédito, contado según los periodos del año.

3. Modelamiento Dimensional para Data mart Compras

En este punto se estableció la tabla de hechos y sus dimensiones, según los resultados del Modelamiento de metadatos (Satar Net), tal y como se muestra en la figura 27.

Figura 27: Modelamiento Dimensional Data mart Compras

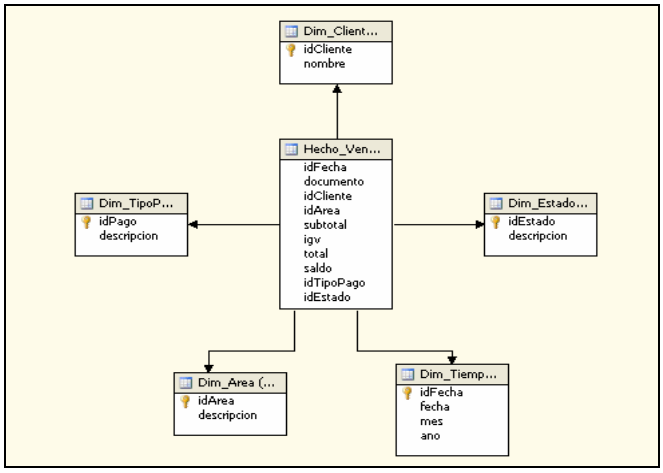


Como se puede apreciar en la figura, la tabla de hechos va a contener cada una de las claves de las dimensiones, y las dimensiones, como se muestra, son los metadatos que se establecieron en el Modelamiento anterior.

4. Modelamiento Dimensional para Data mart Ventas

Al igual que en el Modelamiento dimensional para el data mart de compras, en este punto se tomo como referencia el Modelamiento de metadatos (Star Net) realizado anteriormente, los resultados se muestran en la figura 28.

Figura 28: Modelamiento Dimensional Data mart Ventas

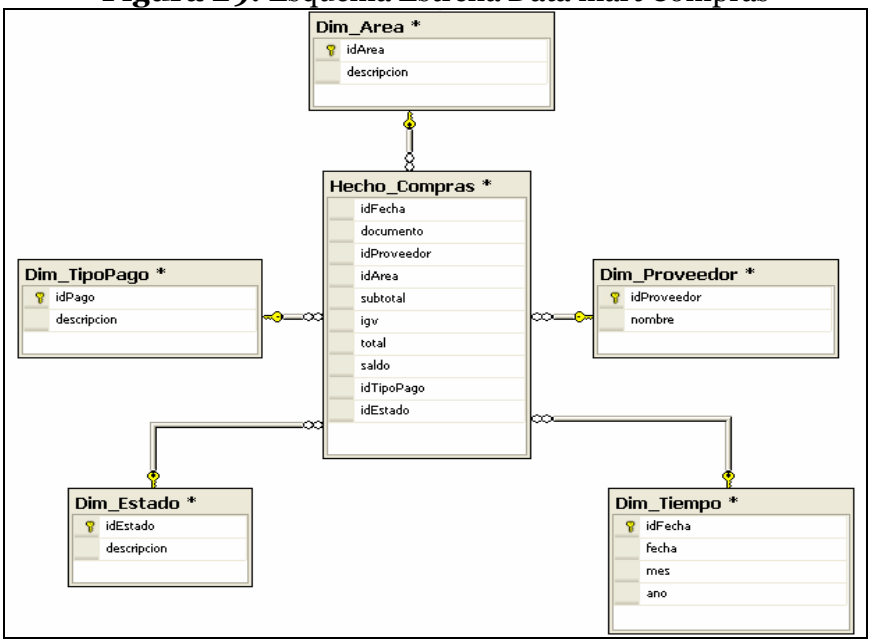


Como se puede apreciar en la figura, la tabla de hechos va a contener cada una de las claves de las dimensiones, y las dimensiones, como se muestra, son los metadatos que se establecieron en el Modelamiento anterior.

5. Modelo Lógico de Data mart Compras

Luego de tener el modelo dimensional, con la tabla de hechos y sus dimensiones, procedimos a plasmarlo en una base de datos lógica, para ello usamos el gestor de base de datos SQL Server 2005, en donde principalmente se ve el tipo de relación entre la tabla de hechos y sus dimensiones, como también el esquema usado para este data mart, los resultados se muestran en la siguiente figura.

Figura 29: Esquema Estrella Data mart Compras

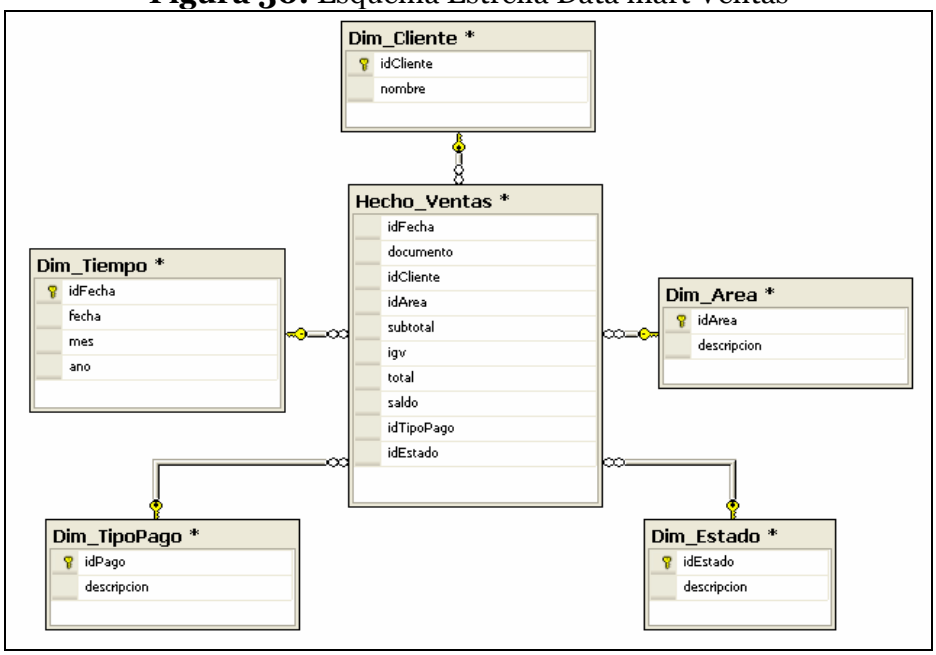


Como se muestra en la figura el esquema de la base de datos para el data mart compras es tipo estrella, con la tabla de hechos que contiene las claves de todas las dimensiones.

6. Modelo Lógico de Data mart Ventas

Luego de realizar la base de datos para el data mart compras, se procedió a hacer lo mismo para el data mart ventas, en donde, la base de datos tendría también un esquema tipo estrella, conformada por su tabla de hechos y sus dimensiones como se muestra en la figura 30.

Figura 30: Esquema Estrella Data mart Ventas



Como se muestra en la figura 30 el esquema de la base de datos para el data mart ventas es tipo estrella, con la tabla de hechos que contiene las claves de todas las dimensiones.

7. Herramientas y Plataformas

Para realizar nuestro trabajo, el cual se basa en el ETL y depuración de los datos, para asegurar la calidad de los datos, se escogieron las siguientes herramientas.

a) Plataforma SQL Server Business Intelligence Development.- Es una plataforma que nos proporciona Visual Studio 2005, en la cual te brinda herramientas para el diseño del flujo de datos, herramientas para la extracción, transformación y carga de datos, además también brinda herramientas para la depuración de datos. También te brinda facilidades para conectarte a diferentes base de datos entre, entre ellas tablas de Excel con las que se trabajó en este estudio, SQL Server con las que se diseñaron las tablas lógicas para el destino de los datos. Otro de los puntos por lo que se trabajó con esta plataforma es la facilidad de su uso, facilidad para adquirirla debido a que es una herramienta muy comercial, y sobre todo porque ya se tenía conocimiento del lenguaje que usa Visual Studio para su desarrollo.

b) Gestor de Base de Datos SQL Server 2005.- Las razone por la cual se optó por usar este gestor fue su facilidad de uso, además que ya se tenía conocimiento en el lenguaje que usa para su implementación, su facilidad de adquisición, la compatibilidad con el sistema operativo Win XP con el que contaba la computadora de desarrollo.

8. Proceso ETL y Depuración de Datos en Data Mart Ventas

En esta primera parte nos hemos centrado en evaluar el proceso ETL y la depuración de datos en el data mart para el sistema de ventas

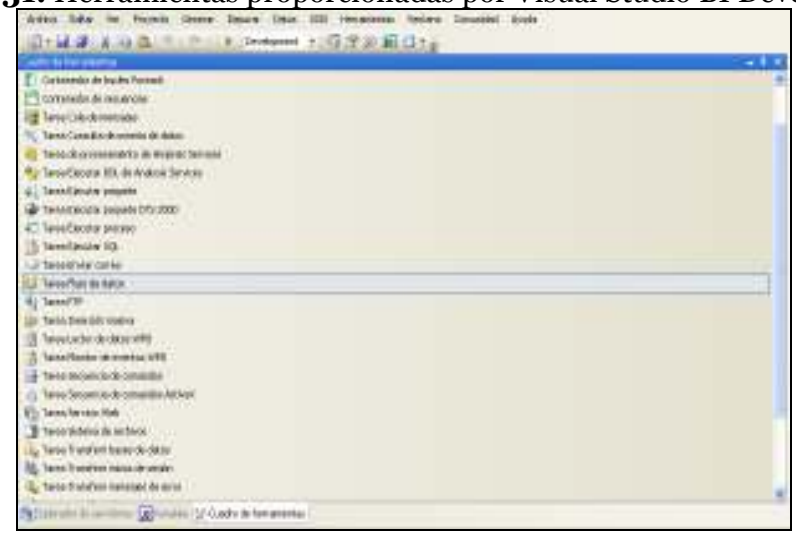
8.1. Flujo de Datos con Herramientas de SQL Server Business Intelligence Development de Visual Studio 2005

Para este caso Visual Studio proporciona diversas herramientas para el proceso ETL como:

8.1.1. Flujo de Datos Simple

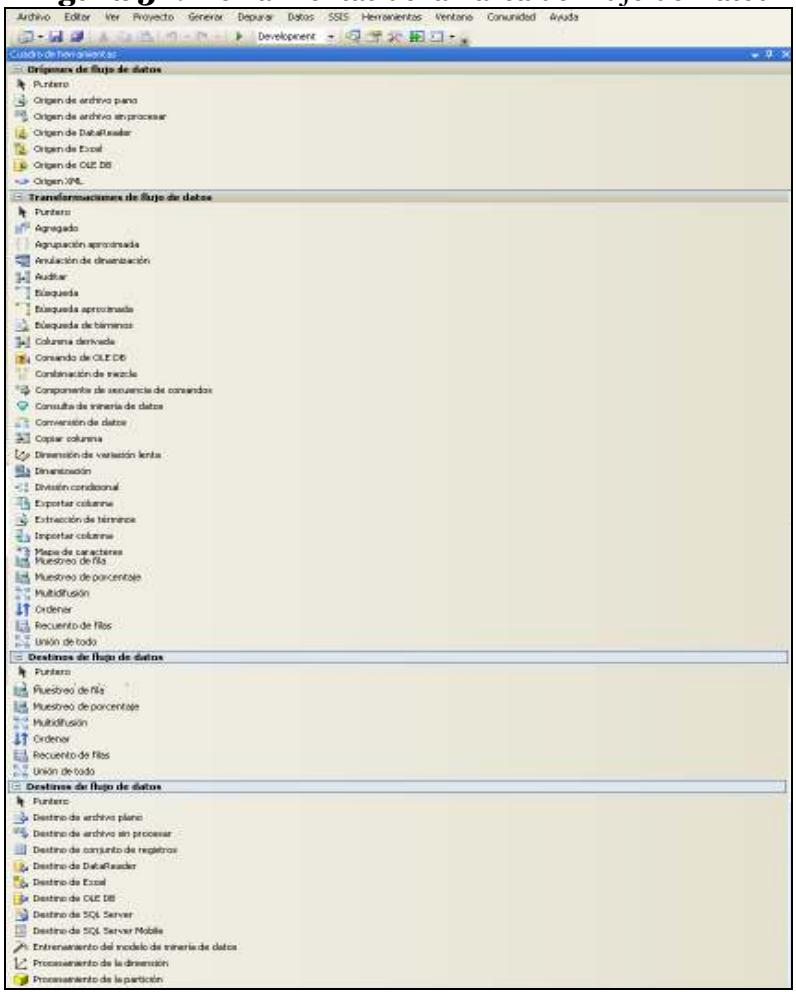
Para poder realizar el proceso ETL la plataforma nos proporciona herramientas diseñadas para tal fin, la más recomendada para el proceso ETL es el paquete de herramientas del Flujo de Datos, el cual contiene herramientas para elegir el origen de datos, la transformación de datos y el destino de los datos, como se muestran en la figura 31.

Figura 31: Herramientas proporcionadas por Visual Studio BI Development



Fuente: Visual Studio Business Development (2005)

Figura 32: Herramientas de la Tarea de Flujo de Datos

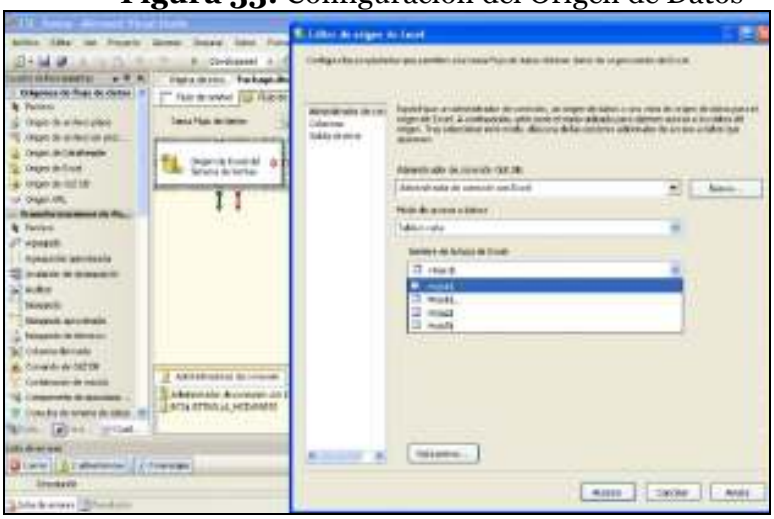


Fuente: Visual Studio Business Development (2005)

Como se aprecia en la figura 32 las herramientas a utilizar son diversas, todas ellas nos ayudaron a realizar el proceso de extracción transformación y carga, desde una base de datos Excel a otra en SQL Server. A continuación describimos los pasos para ejecutar el proceso ETL para el data mart Ventas.

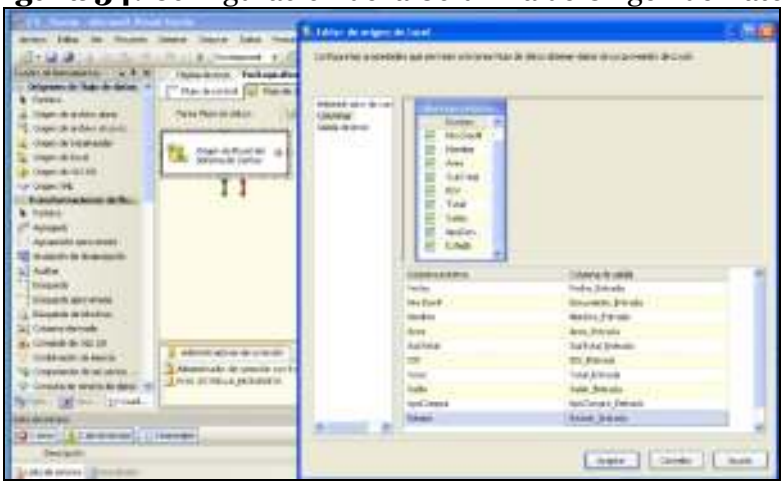
a) Establecer el Origen de datos.- Primero se estableció el origen de datos, para ello se utilizo la herramienta Origen de Excel, debido a que nuestro origen de datos son tablas de Excel.

Figura 33: Configuración del Origen de Datos



En esta parte del trabajo se configuró el origen de los datos que se van a usar para llenar nuestro data mart Ventas, primero se escogió del cuadro de herramientas el tipo de origen de datos que en este caso es Origen Excel, luego se estableció la conexión usando el administrador de conexión con Excel, se escoge la hoja con la que se va a trabajar.

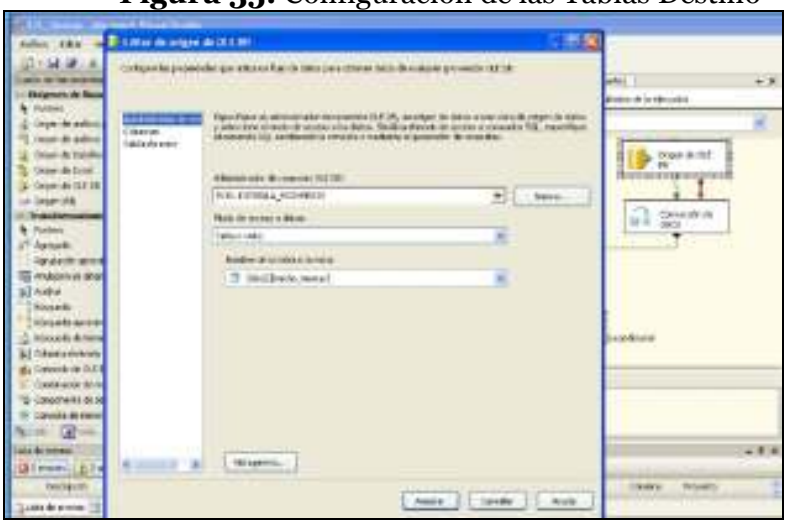
Figura 34: Configuración de la Columna de Origen de Datos



Luego escogemos las columnas que vamos a extraer y se le coloca un alias a cada columna, con la finalidad de no confundirnos durante el resto del proceso. Luego de realizar las configuraciones se da clic en aceptar y pasamos a la siguiente fase.

b) Escoger el Origen-Destino de los datos a cargar.- En este paso se establece el destino usando la herramienta "Destino de OLE DB", luego se establece la conexión con la base de datos en SQL Server con la cual se va a trabajar, para ello usamos el administrador de conexión "OLE DB", se escoge la tabla con la que se va a trabajar.

Figura 35: Configuración de las Tablas Destino



Al igual como sucedió con el origen, aquí se configuró el destino de los datos que se van a usar para llenar nuestro data mart Ventas, primero se escogió del cuadro de herramientas el tipo de origen de datos que en este caso es Destino OLE DB, luego se estableció la conexión usando el administrador de conexión OLE DB y se escoge las columnas con la que se va a trabajar. Para un flujo simple de datos estos dos pasos son suficientes.

A) Resultados del Flujo Simple de Datos

Al hacer clic en ejecutar nos encontramos que no se puede efectuar la operación debido a que los metadatos de origen no son iguales a los del destino, como se muestra en la figura de la tarea de flujo de datos.

Figura 36: Resultado de Flujo de Datos Simple

Los datos de origen si logran ser extraídos pero no son cargados en el repositorio destino como se muestra en la siguiente figura.

Figura 37: Resultado de Flujo de Datos Simple en Tablas Destino

Revisando los metadatos podemos comparar que el tipo de datos Excel no son iguales que los tipo de datos de SQL Server 2005, aunque a simple vista parecieran que fuesen iguales como se muestra en la siguiente figura.

Figura 38: Datos de Tablas de Origen

Si apreciamos, a simple vista los tipos de datos de Excel parecieran ser compatibles con los de la base de datos del data mart, que se muestra a continuación.

Figura 39: Tipo de Datos de Tablas Destino

Excel usa datos muy diferentes que SQL Server 2005, como podemos apreciar en la figura 40.

Figura 40: Comparación de Tipo de Datos de Tablas Origen y Destino

Metadatos de ruta:

Nombre	Tipo de datos	Precisión	Escala	Longi...	Página ...	Posició...	Indicadores d...	Componente de origen
Fecha	DT_R8	0	0	0	0	0		Origen de Excel Sistema de Ventas
Nro Docst	DT_WSTR	0	0	255	0	0		Origen de Excel Sistema de Ventas
Nombre	DT_R8	0	0	0	0	0		Origen de Excel Sistema de Ventas
Area	DT_R8	0	0	0	0	0		Origen de Excel Sistema de Ventas
SubTotal	DT_R8	0	0	0	0	0		Origen de Excel Sistema de Ventas
IGV	DT_R8	0	0	0	0	0		Origen de Excel Sistema de Ventas
Total	DT_R8	0	0	0	0	0		Origen de Excel Sistema de Ventas
Saldo	DT_R8	0	0	0	0	0		Origen de Excel Sistema de Ventas
tipoCompra	DT_R8	0	0	0	0	0		Origen de Excel Sistema de Ventas
Estado	DT_R8	0	0	0	0	0		Origen de Excel Sistema de Ventas

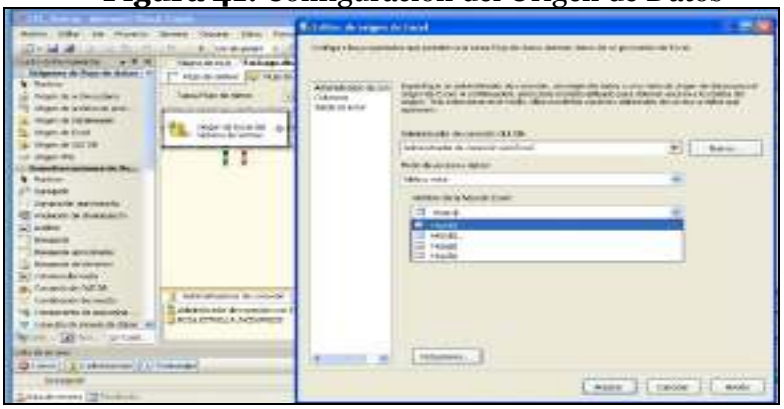
El tipo de dato DT_R8, hace referencia al tipo de dato float de las tablas Excel, es incompatible con los tipos de datos int y smallmoney de SQL Server 2005; el tipo de dato DT_WSTR (255), es el único compatible con los tipo de datos char, varchar y nvarchar de SQL Server 2005, lo único que varia es su longitud, DT_WSTR por defecto tiene una longitud de 255 caracteres. Para poder extraer y cargar datos de Base de Datos en Excel hacia Base de Datos en SQL Server 2005, primero se deben de transformar los datos de origen al tipo de datos de destino o viceversa. Mientras no suceda esto no se podrán extraer los datos que se desean.

8.1.2. Pasos para el proceso ETL según la metodología de BI Development de Visual Studio.

Para poder extraer, transformar y cargar los datos de una BD en Excel hacia una BD en SQL Server 2005, según lo que propone la metodología de Visual Studio para el desarrollo de Business Intelligence, se debe de hacer uso de varias herramientas las cuales servirán en cada uno de los pasos que a continuación se muestran.

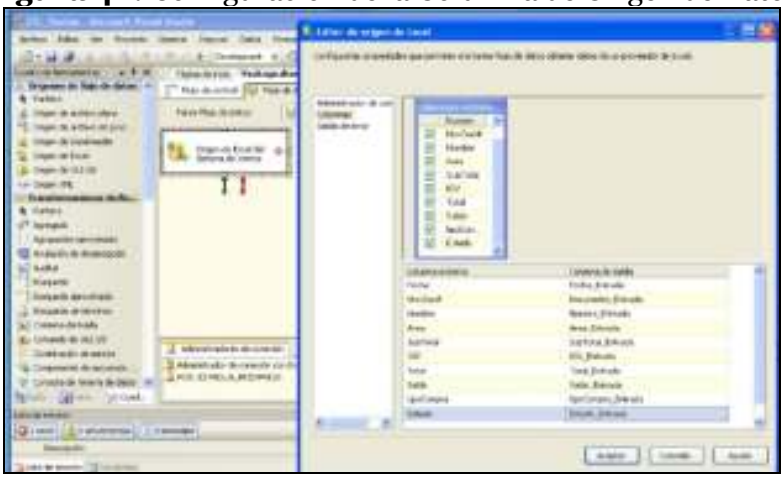
a) Elección de Tablas de la base de datos de origen.- Se procedió a realizar el mismo trabajo que se hizo durante el flujo de datos simple, se escogió del cuadro de herramientas el tipo de origen de datos "Origen Excel", luego se establece la conexión usando el administrador de conexión con Excel y se escoge la hoja con la que se va a trabajar.

Figura 41: Configuración del Origen de Datos



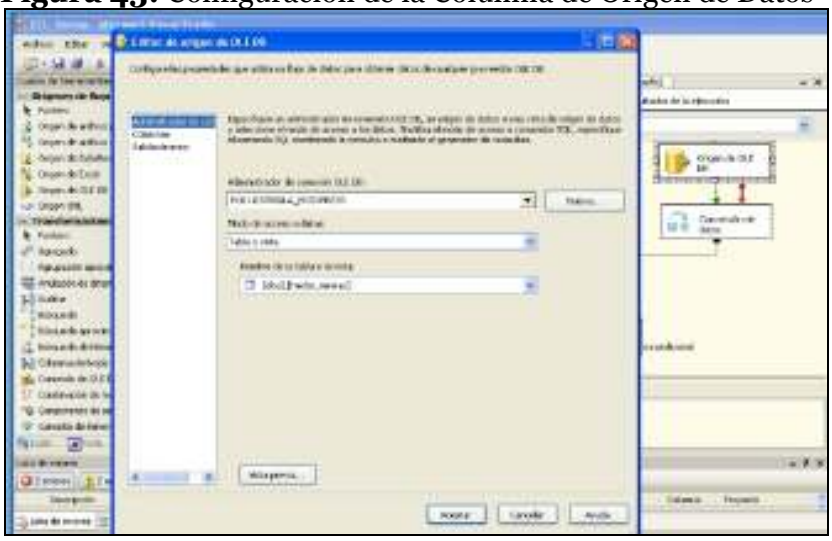
Luego escogemos las columnas que vamos a extraer y se le coloca un alias a cada columna, con la finalidad de no confundirnos durante el resto del proceso.

Figura 42: Configuración de la Columna de Origen de Datos



b) Escoger el Origen-Destino de los datos a cargar.- se escogió del cuadro de herramientas el tipo de Origen-Destino de datos que en este caso es Origen OLEDB, luego se estableció la conexión con la base de datos en SQL Server con la cual se va a trabajar, para ello usamos el administrador de conexión OLE DB, se escogió la tabla con la que se va a trabajar.

Figura 43: Configuración de la Columna de Origen de Datos

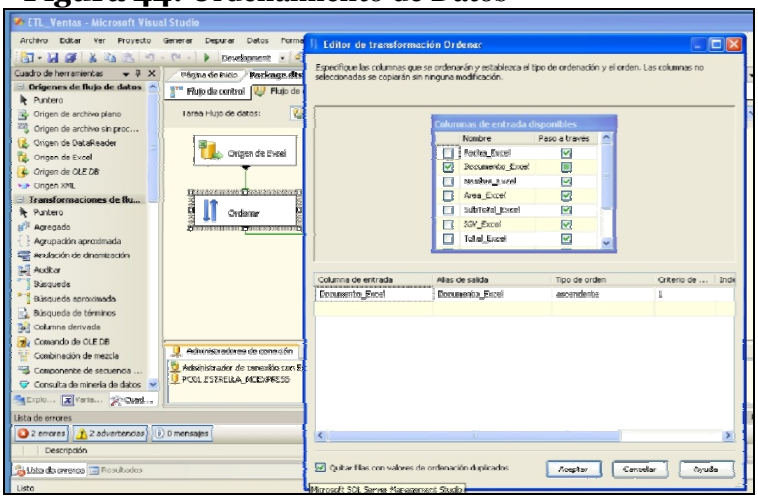


c) Luego hacemos clic en "columnas" se escogen las columnas que vamos a extraer y se le coloca un alias a cada columna, con la finalidad de no confundirnos durante el resto del proceso.

d) Ordenar Datos Origen.- Para poder mezclar y posteriormente cargar los datos en la base de datos del data mart en SQL Server, primero debemos ordenar los datos de la tabla que se está usando, se escoge la herramienta "Ordenar" que se encuentra en el cuadro de herramientas, se hace doble clic y se procede a ordenar los datos, para ello debemos tomar la columna con datos que no se repitan, como por ejemplo una clave primaria, pero esta herramienta también nos permite quitar los datos duplicados, solo debemos hacer un check en la opción "quitar filas con valores de

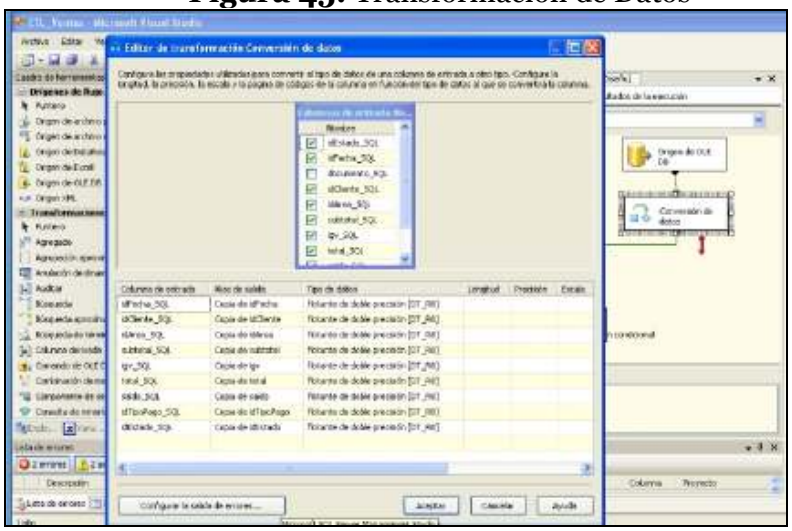
ordenación duplicados", como se muestra en la siguiente figura

Figura 44: Ordenamiento de Datos



e) Conversión de Datos Origen-Destino OLE DB.- Para no caer en el error de datos no compatibles, hemos transformamos los datos que se han extraido, para lo cual usamos la herramienta "Transformación de Datos", que se encuentra en el cuadro de herramientas, se hizo doble clic y procedemos a elegir los valores a transformar, luego se escogió el tipo de dato que se requiere para que sean compatibles, como se muestra en la figura

Figura 45: Transformación de Datos



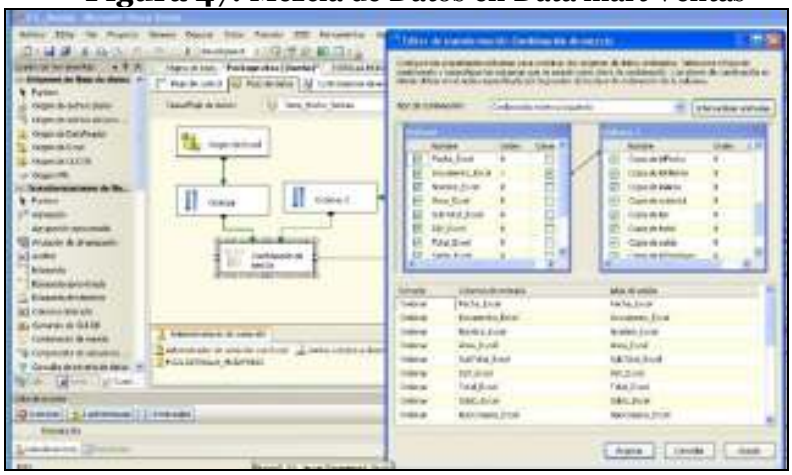
f) Ordenar Datos Origen-Destino OLE DB.- Se escogió la herramienta "Ordenar" que se encuentra en el cuadro de herramientas, en esta ocasión se realizó para poder ordenar los datos en la base destino (Data mart), se hace doble clic y se procede a ordenar los datos, para ello debemos tomar la columna con datos que no se repitan, como por ejemplo una clave primaria, pero esta herramienta también nos permite quitar los datos duplicados, solo debemos hacer un check en la opción

"quitar filas con valores de ordenación duplicados", como se muestra en la siguiente figura

Figura 46: Ordenamiento de Datos en el Data mart

g) Transformación de Mezcla.- Después de tener los datos origen y destino ordenados procedimos a mezclarlos para poder cargarlos en la base de datos final. Se seleccionaron todos los ítems de origen y los ítems que se transformaron para lograr la compatibilidad de los tipos de datos, se escoge el tipo de combinación que para este caso fue la "combinación externa izquierda", una vez realizado esto estuvimos listos para depurar los datos. Nos quedó como se muestra en la siguiente figura.

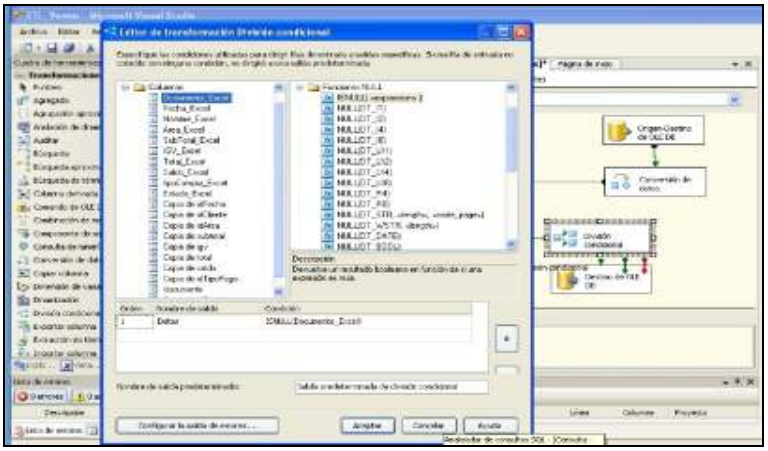
Figura 47: Mezcla de Datos en Data mart Ventas



h) División condicional.- Esta herramienta nos fue útil para la depuración de datos, tiene diversas funciones que nos ayudan a la depuración de datos como son: funciones matemáticas, funciones de cadena, funciones de fecha y hora, función de valores null, conversiones de tipo de datos y operadores lógicos. hicimos uso de la función NULL para evitar cargar datos nulos en nuestra BD dimensional, escogimos la función ISNULL y la arrastramos hacia el ítem de condición que se encuentra en la parte inferior del cuadro, arrastramos la variable que escogimos para ordenar

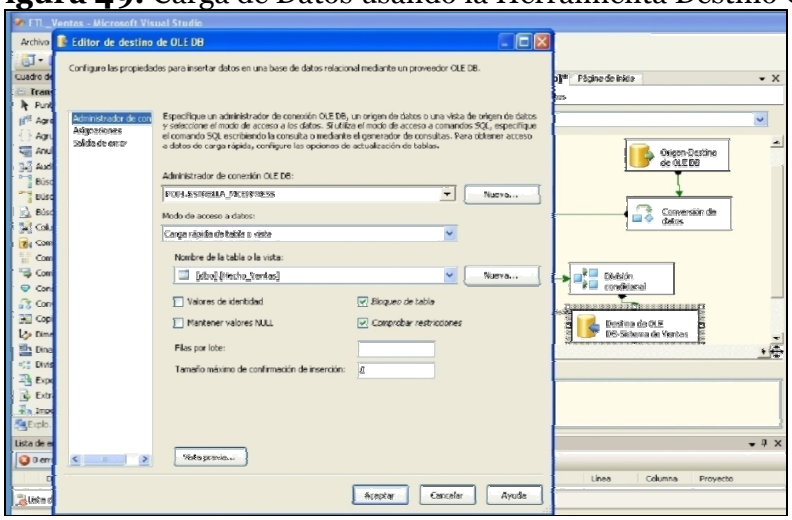
los datos de origen, y le colocamos un alias como nombre de salida llamado “Datos”.

Figura 48: Depuración de Datos Usando la Herramienta División Condicional



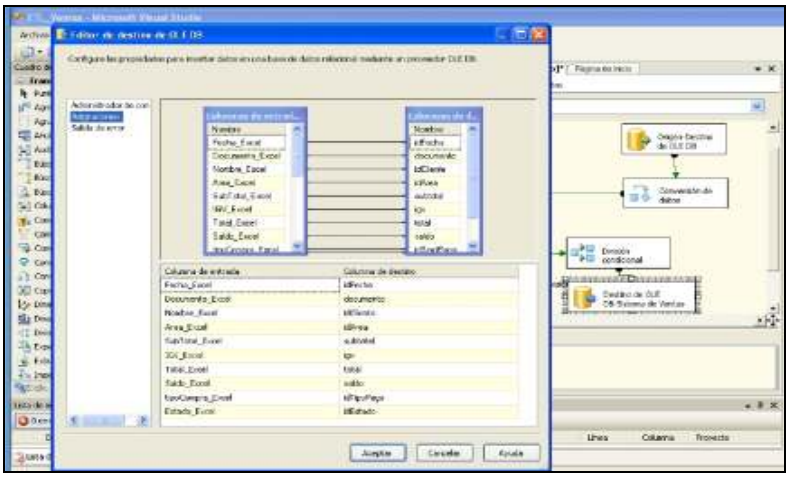
i) Destino OLEDB Sistema de Ventas.- En el último paso para cargar los datos se hizo uso de la herramienta “Destino OLE DB”, primero se escogió la conexión OLE DB a través del administrador de conexión de OLE DB de visual estudio, se escogió el modo de acceso a datos, que para este caso es una carga rápida de tabla o vista, luego se escogió el nombre de la tabla o vista de la BD destino, que para este caso es el “Hecho_Ventas”, nos debe queda de la siguiente forma.

Figura 49: Carga de Datos usando la Herramienta Destino OLE DB



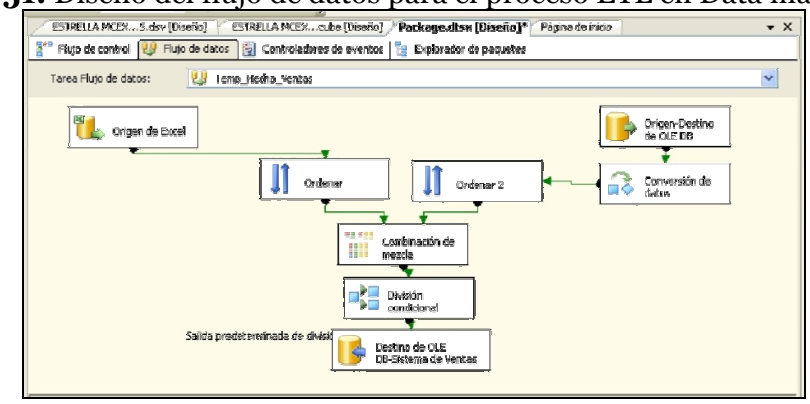
Luego asignamos y relacionamos los datos de origen con los de destino, como se muestra en la figura 50.

Figura 50: Asignación de Datos usando la Herramienta Destino OLE DB



Al final de todo el proceso nos debe quedar el siguiente diseño con el cual vamos a lograr extraer, transformar y cargar los datos de una base de datos de origen hacia una base de datos destino.

Figura 51: Diseño del flujo de datos para el proceso ETL en Data mart Ventas



j) Iniciar Depuración.- luego de realizar los pasos que nos propone la metodología de BI Development, para lograr el proceso ETL, nos quedó iniciar depuración dando clic en el símbolo de "Iniciar Depuración" y verificar la calidad de los datos luego de la ejecución.

Resultado del ETL según la metodología de BI Development de Visual Studio

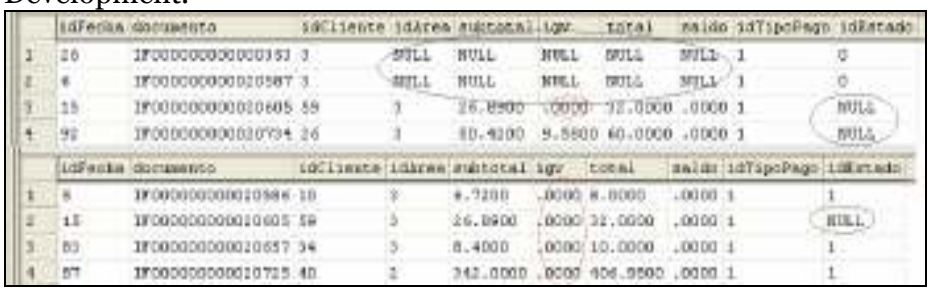
Luego de haber iniciado la depuración del proceso utilizando la metodología y herramientas que nos brinda la plataforma de visual Studio, los resultados fueron los siguientes.

Número de Pasos en el proceso ETL: 9
Número de Datos Faltantes o Nulos: 13
Número de Datos Incoherentes: 5

Como nos muestra la siguiente figura.

Figura 52: Resultados del proceso ETL usando la Metodología BI

Development.



	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF0000000000000351	3	NULL	NULL	NULL	NULL	NULL	1	0
2	6	IF0000000000010587	3	NULL	NULL	NULL	NULL	NULL	1	0
3	15	IF0000000000010605	59	3	26.8900	.0000	32.0000	.0000	1	NULL
4	92	IF0000000000010734	26	3	80.4200	9.5800	60.0000	.0000	1	NULL

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	6	IF0000000000010586	10	3	6.7200	.0000	8.0000	.0000	1	1
2	15	IF0000000000010605	59	3	26.8900	.0000	32.0000	.0000	1	NULL
3	83	IF0000000000010657	34	3	8.4000	.0000	10.0000	.0000	1	1
4	87	IF0000000000010715	40	1	342.0000	.0000	404.9500	.0000	1	1

Como se puede observar los datos faltantes o nulos están encerrados en un círculo negro, con respectos a los datos incoherentes (círculos rojos) si observamos pertenecen al campo IGV de la tabla, si verificamos el subtotal no es igual al total, esto quiere decir que no es una boleta sino una factura, y en toda factura debe ir indicado el monto del IGV, que en el año 2010 es del 19%.

Observaciones.- Se procedió a realizar la depuración de datos con las herramientas que nos brinda la plataforma de Visual Studio, que para estos casos es la de "División Condicional" y tras varios intentos no se pudo limpiar los datos faltantes y los incoherentes.
Las herramientas de depuración de datos de la plataforma de SQL BI Development de Visual Studio, en especial la que se propone (División Condicional) y todas sus opciones, no son muy efectivas para el aseguramiento de la calidad de los datos

8.2. Proceso ETL usando la Metodología propuesta por Jiménez y Amón para la Depuración de Datos.

Esta metodología nos propone el uso de técnicas plasmadas en algoritmos, para poder extraer, transformar y cargar los datos de una base de datos Origen a una Destino, los algoritmos en su totalidad están diseñados para sistemas expertos y/o base de datos netamente estadística, para este caso, luego de haber estudiado su funcionamiento, hemos adaptado estos algoritmos en comandos SQL.

8.2.1. Flujo de datos por pasos para asegurar la calidad de los datos durante el proceso ETL

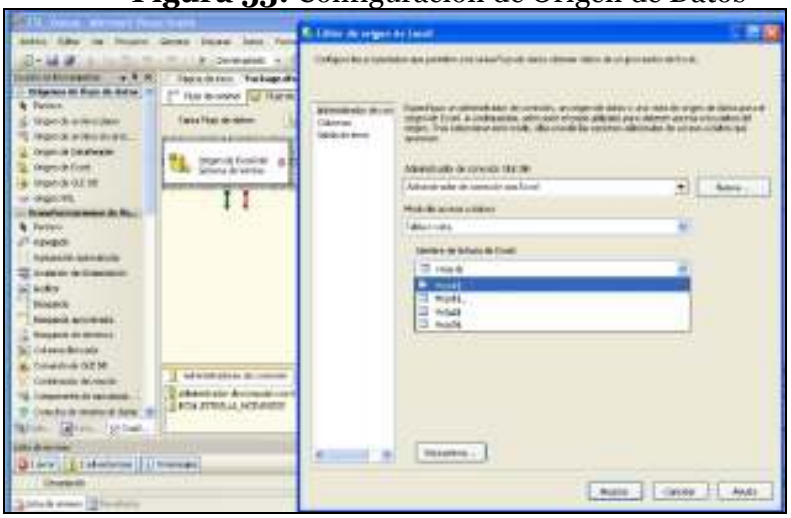
En este punto se tomó la recomendación que nos hacen los autores, que para lograr una buena depuración se debe de ejecutar la metodología por cada uno de los datos anómalos que se encuentren o se sospeche. Por ello el número de pasos puede variar según la cantidad de datos anómalos, a continuación se describe la metodología paso por paso. Como no es una metodología con plataforma propia, se uso la plataforma de Visual Studio para ejecutar todo lo que en ella se explica.

8.2.1.1. Pasos Para la Extracción, transformación y Carga

a) Escoger los datos de una Base de Datos Origen.- Primero se escogió del cuadro de herramientas el tipo de origen de datos que en este caso es

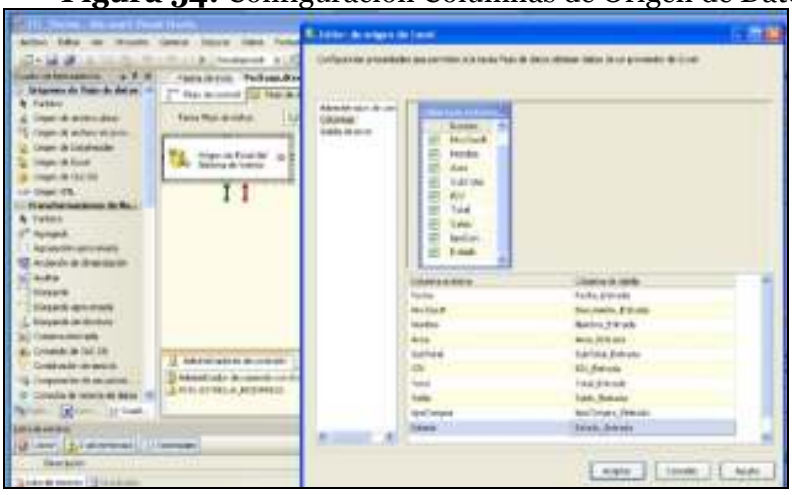
"Origen Excel", luego se establece la conexión usando el administrador de conexión con Excel, se escoge la hoja con la que se va a trabajar, tal como se muestra en la figura 53.

Figura 53: Configuración de Origen de Datos



Luego hicimos clic en "columnas" se escogieron las columnas que vamos a extraer y se le colocó un alias a cada columna, con la finalidad de no confundirnos durante el resto del proceso, como se puede apreciar en la figura 54.

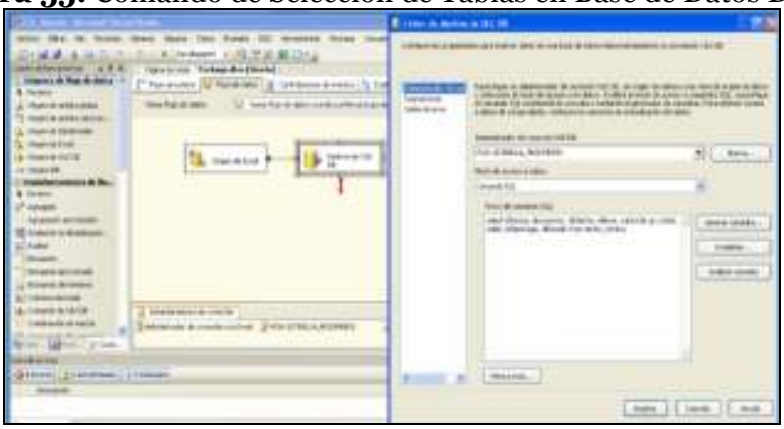
Figura 54: Configuración Columnas de Origen de Datos



b) Escoger el Repositorio destino en una Base de Datos.- Del cuadro de herramientas se escoge la herramienta "Destino de OLE DB", utilizando el administrador de conexión OLE DB se procede a establecer con la base de datos que para este caso es "ESTRELLA_MCEXPRESS", luego usamos el modo de acceso a datos "Comando SQL" y procedemos a seleccionar las columnas de la tabla destino "Hecho_Ventas", con la siguiente consulta *"select idFecha, documento, idCliente, idArea, subtotal, igv, total, saldo, idTipoPago, idEstado from Hecho_Ventas",* todo

este comando se coloca en la casilla en blanco como se muestra en la figura 55.

Figura 55: Comando de Selección de Tablas en Base de Datos Destino



c) Iniciar Depuración.- Luego de colocar el comando SQL nos debe quedar de la siguiente forma como se muestra en la figura 56, Lo único que debemos hacer es iniciar la depuración.

Figura 56: Diseño del ETL según Metodología de Jimenez y Amón

d) Resultados del Flujo de datos por pasos de Jiménez y Amón para asegurar la calidad de los datos durante el proceso ETL

Luego de haber iniciado la depuración del proceso utilizando la metodología y herramientas que nos brinda la plataforma de visual Studio, los resultados fueron los siguientes.

Número de Pasos en el proceso ETL: 3
Número de Datos Faltantes o Nulos: 12
Número de Datos Incoherentes: 4

Figura 57: Resultados del ETL según Metodología de Jimenez y Amón

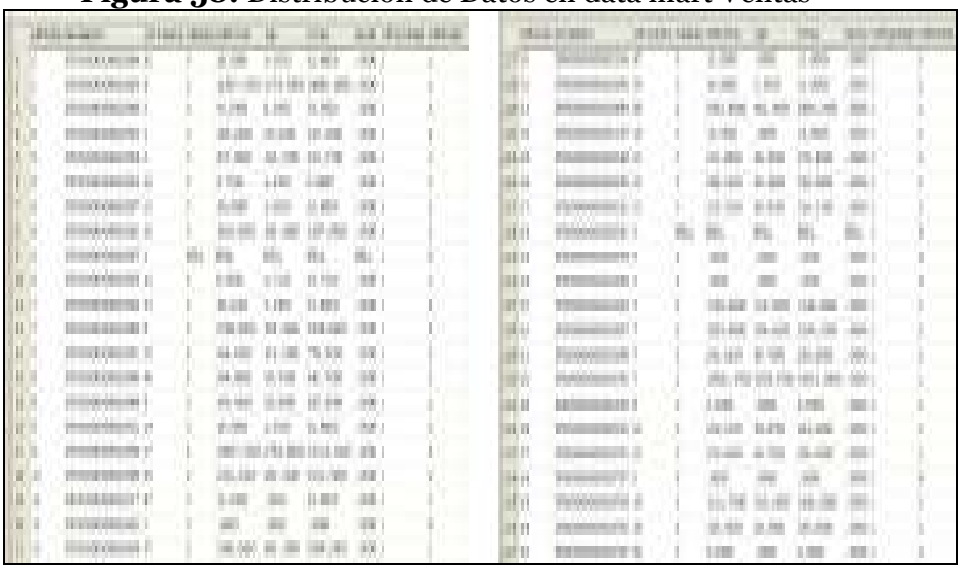
	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF0000000000000353	3	NULL	NULL	NULL	NULL	NULL	1	0
2	6	IF0000000000010587	3	NULL	NULL	NULL	NULL	NULL	1	0
3	15	IF0000000000010605	59	3	26.6900	.0000	32.0000	.0000	1	NULL
4	92	IF0000000000010734	26	3	50.4200	9.5800	60.0000	.0000	1	NULL

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	66	IF0000000000020586	36	3	67.2300	.0000	80.0000	.0000	1	1
2	85	IF0000000000010605	8	2	2433.7500	.0000	2896.1600	.0000	1	1
3	33	IF0000000000010657	47	3	12.0000	.0000	12.0000	.0000	1	1
4	45	IF0000000000010725	8	2	957.8100	.0000	1139.7900	.0000	1	1

8.2.1.2. Pasos para la depuración de datos usando la metodología de Amón y Jiménez.

a) Depuración de la Columna "idArea" usando Imputación media.- Para el caso de los valores faltantes de la columna "igv" de la tabla "Hecho_Ventas", los datos están distribuidos normalmente como se aprecia en la siguiente figura:

Figura 58: Distribución de Datos en data mart Ventas



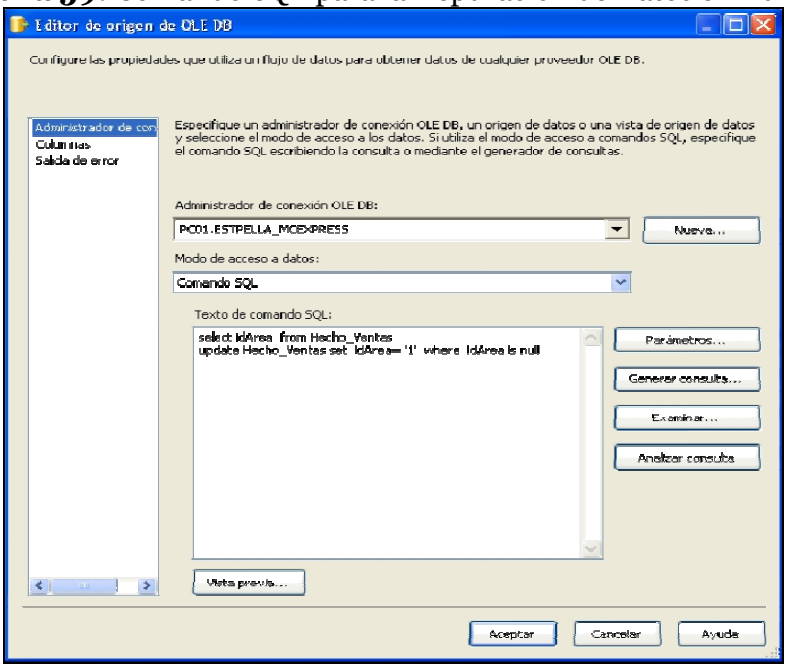
Esta técnica consiste en modificar los datos faltantes usando la media aritmética de los valores que se depositan en la columna donde se encuentra el problema. La fórmula de la media aritmética es la siguiente:

$$\bar{X} = \frac{X_1 + X_2 + X_3 + \ldots + X_n}{N}$$

Que en resumen es la sumatoria de todos los valores dividido entre el número de valores por ejemplo: x= (1+4+7+9)/4

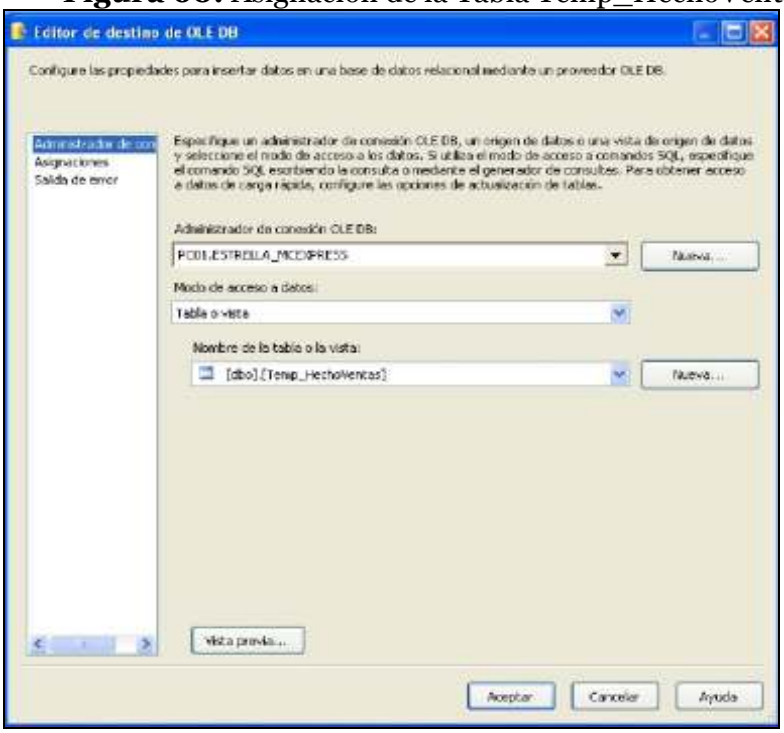
En el caso que nos encontramos los valores van de 1 a 3 esto nos daría el siguiente comando SQL para la depuración de estos datos: *"update Hecho_Ventas set idArea=(1+2+3)/3 where idArea is null"*, este comando SQL se coloca dentro del editor de origen de OLE DB nos debe quedar de la siguiente forma:

Figura 59: Comando SQL para la Depuración de Datos en "idArea"



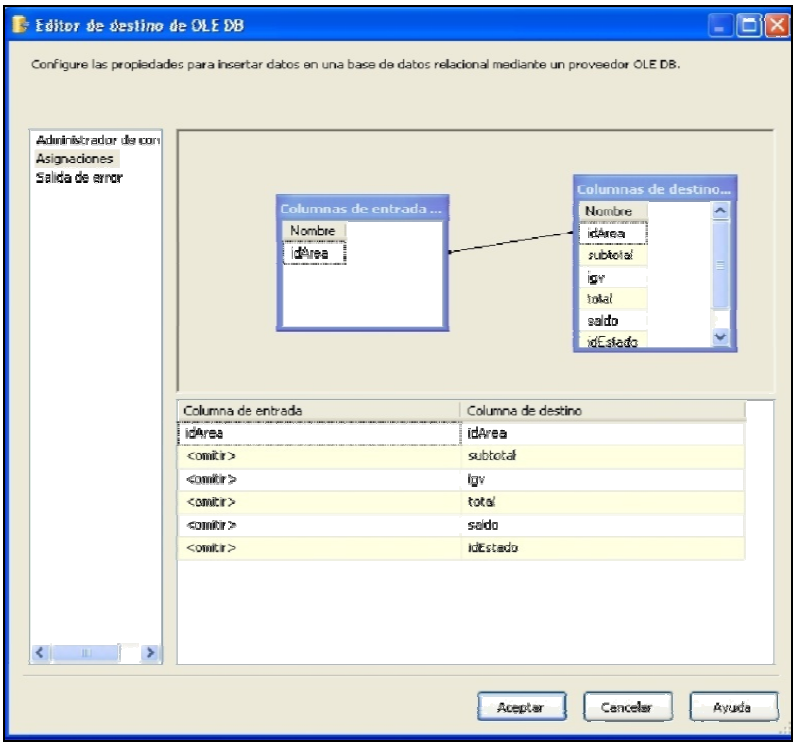
Luego asignamos las columnas de origen con las columnas destino, para ello usamos una tabla temporal a la que hemos llamado "Temp_HechoVentas", seleccionamos dicha tabla.

Figura 60: Asignación de la Tabla Temp_HechoVentas



Luego relacionamos los datos origen que deseamos depurar con los datos destino, nos debe quedar de la siguiente forma.

Figura 61: Relación de los Datos de Origen con los Datos Destino



Solo nos queda iniciar la depuración y nos debería haber imputado los datos nulos de la columna igv de la tabla Hecho_Ventas como se muestra a continuación:

Figura 62: Resultados de la Depuración de Datos en "idArea", Datamart Ventas

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF0000000000000353	3	NULL	NULL	NULL	NULL	NULL	1	0
2	6	IF0000000000020587	3	NULL	NULL	NULL	NULL	NULL	1	0
3	15	IF0000000000020605	59	3	26.8900	.0000	32.0000	.0000	1	NULL
4	92	IF0000000000020734	26	3	50.4200	9.5800	60.0000	.0000	1	NULL

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF0000000000000353	3	2	NULL	NULL	NULL	NULL	1	0
2	92	IF0000000000020734	26	3	50.4200	9.5800	60.0000	.0000	1	NULL
3	6	IF0000000000020587	3	2	NULL	NULL	NULL	NULL	1	0
4	15	IF0000000000020605	59	3	26.0900	5.1100	32.0000	.0000	1	NULL

Como se muestra en la figura en los documentos IF0000000000000353 y IF0000000000020587, los campos de "idArea" han sido depurados con números producto de la técnica de la imputación media.

b) Depuración de la Columna Subtotal usando Imputación Hot Deck (Vecino más cercano).- En este caso observamos a los vecinos de la columna "subtotal" en donde se encuentra el valor nulo, ver figura 63.

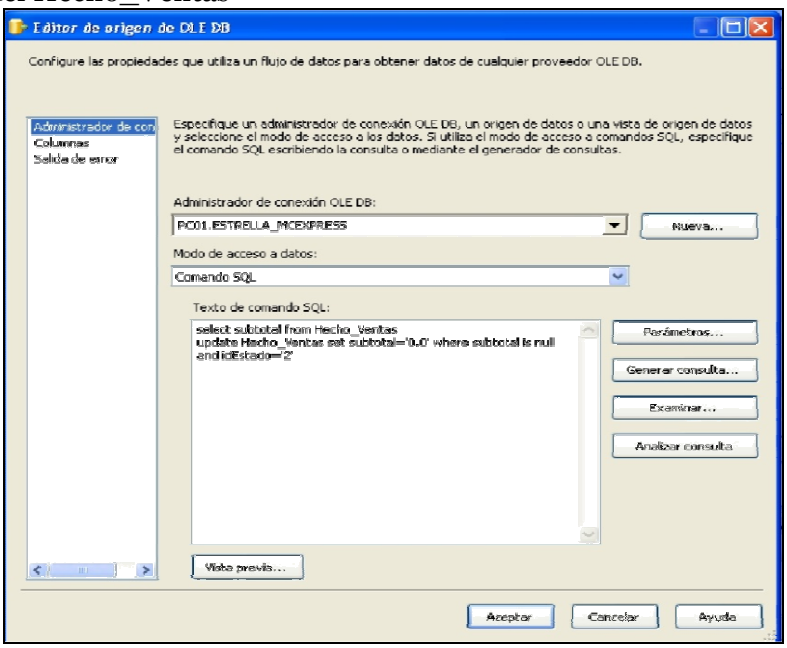
Figura 63: Comparación de los datos de IF0000000000000353

idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
31	IP0000000000000338	3	3	.0000	.0000	.0000	.0000	1	2
32	IP0000000000000339	3	3	.0000	.0000	.0000	.0000	1	2
33	IP0000000000000340	23	3	327.5200	62.2300	389.7500	.0000	1	1
34	IP0000000000000341	49	3	2016.5000	383.1400	2399.6400	.0000	1	1
35	IP0000000000000342	68	3	21.0100	3.9900	25.0000	.0000	1	1
36	IP0000000000000343	23	3	294.7900	56.0100	350.8000	.0000	1	1
37	IP0000000000000344	23	3	335.7100	63.7900	399.5000	.0000	1	1
38	IP0000000000000345	16	3	8.0000	.0000	8.0000	.0000	1	1
39	IP0000000000000346	23	3	323.5300	61.4700	385.0000	.0000	1	1
40	IP0000000000000347	46	3	11.0000	.0000	11.0000	.0000	1	1
41	IP0000000000000348	4	3	33.6100	6.3900	40.0000	.0000	1	1
42	IP0000000000000349	23	3	326.4400	62.0200	388.4600	.0000	1	1
43	IP0000000000000350	49	3	57.0000	10.8300	67.8300	.0000	1	1
44	IP0000000000000351	49	3	3134.0000	595.4600	3729.4600	.0000	1	1
45	IP0000000000000352	49	3	405.0000	76.9500	481.9500	.0000	1	1
46	IP0000000000000353	3	3	NULL	NULL	NULL	NULL	1	0
47	IP0000000000000354	7	3	1150.0000	218.5000	1368.5000	.0000	1	1
48	IP0000000000000355	7	3	16386.1500	3113.3700	19499.5200	.0000	1	1

Si observamos a las columnas vecinas del registro nulo, que para este caso es el campo "subtotal", las cuatro primeras no nos brinda ningún tipo de información, debido a que la columna subtotal hace referencia a un tipo de dato moneda y las anteriores hacen referencia a claves foráneas de tipo entero, las otras cinco columnas que le siguen tampoco nos brindan mucha información. Ahora pasamos a comparar las filas vecinas, observando a las filas vecinas nos damos cuenta que las dos primeras son registros anulados y el valor de subtotal es igual a cero, en cambio las filas siguientes son valores no anulados y el valor de subtotal es mayor a cero, por lo tanto lo más probable que al registro en la columna subtotal le corresponde el valor de cero por ser un registro anulado.

Por lo tanto el comando SQL quedaría de la siguiente forma *"update Hecho_Ventas set subtotal='0.0'where subtotal is null",* lo colocamos dentro del editor de origen "Ole DB" como se aprecia en la figura 64.

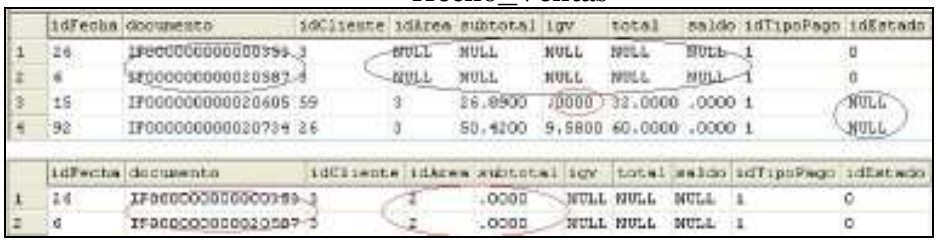
Figura 64: Comando SQL para la Depuración de datos en "subtotal" del Hecho_Ventas



Iniciamos la Depuración y obtuvimos los siguientes resultados, lo que

nos sirve para comparar resultados

Figura 65: Resultados de la Depuración con Hot Deck en "Subtotal" del Hecho_Ventas



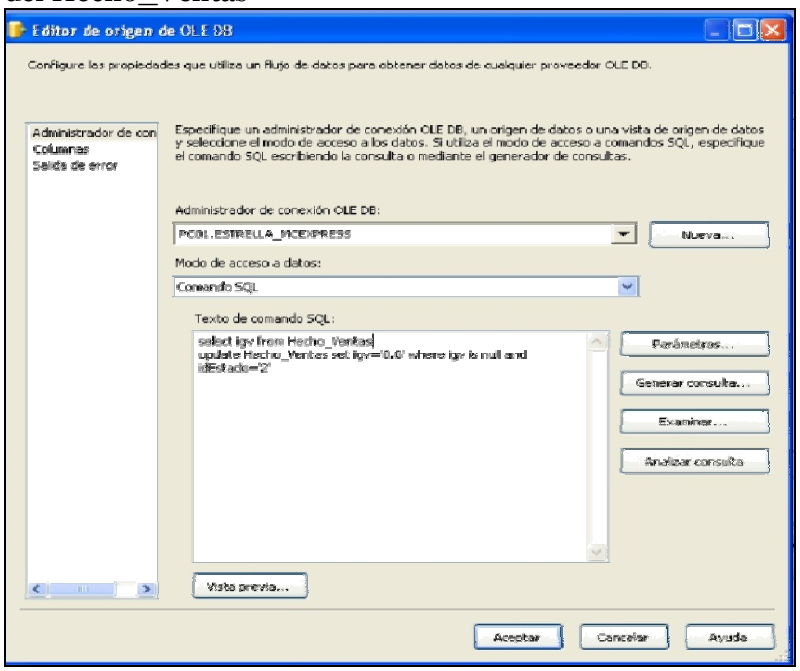
	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF0000000000000399	3		NULL	NULL	NULL	NULL	1	0
2	6	IF0000000000020387	9		NULL	NULL	NULL	NULL	1	0
3	15	IF0000000000020405	59	3	26.8900	.0000	32.0000	.0000	1	NULL
4	92	IF0000000000020734	26	3	50.4200	9.5800	60.0000	.0000	1	NULL

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF0000000000000399	3	2	.0000	NULL	NULL	NULL	1	0
2	6	IF0000000000020387	9	2	.0000	NULL	NULL	NULL	1	0

Como podemos apreciar en la figura, el campo "subtotal" donde antes existía valores nulos, ahora han tomado un valor referencial según a los valores de sus vecinos más cercanos.

c) Depuración de la Columna IGV usando Imputación Hot Deck (Vecino más cercano).- Nos encontramos en la misma situación que con la columna subtotal, ya tenemos la referencia de la columna "idEstado" que hace referencia a los registros anulados y los sin anular, lo único que nos queda hacer es aplicar el mismo comando SQL que aplicamos para imputar la columna "subtotal", pero para este caso cambiamos por la columna "igv", nos quedaría de la siguiente forma *"update Hecho_Ventas set igv='0.0'where igv is null"*

Figura 66: Comando para de la Depuración con Hot Deck en "igv" del Hecho_Ventas



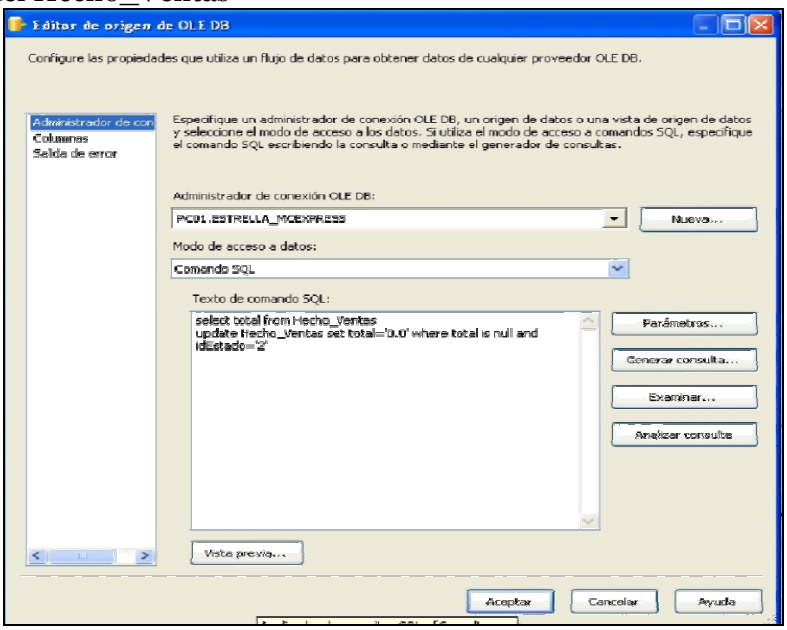
Después de depurar nos ha imputado los datos faltantes en la columna "igv".

Figura 67: Resultados de la Depuración con Hot Deck en "igv" del Hecho_Ventas

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF0000000000000353	3	2	.0000	.0000	NULL	NULL	1	0
2	6	IF0000000000020587	3	2	.0000	.0000	NULL	NULL	1	0

d) Depuración de la Columna "total" usando Imputación Hot Deck: Vecino más cercano.- Nos encontramos en la misma situación que con la columna subtotal, ya tenemos la referencia de la columna "idEstado", que hace referencia a los registros anulados y los sin anular, lo único que nos queda hacer es aplicar el mismo comando SQL que aplicamos para imputar la columna "subtotal", pero para este caso cambiamos por la columna "total", nos quedaría de la siguiente forma: *"update Hecho_Ventas set total='0.0'where total is null"*

Figura 68: Comando para de la Depuración con Hot Deck en "total" del Hecho_Ventas



Depuramos y los datos faltantes fueron imputados en la columna "total".

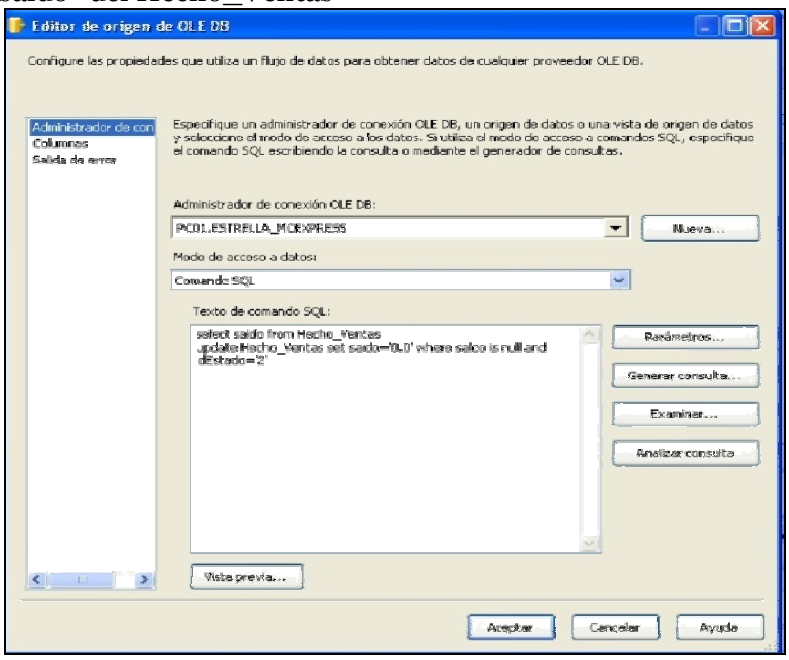
Figura 69: Resultados de la Depuración con Hot Deck en "total" del Hecho_Ventas

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF0000000000000353	3	2	.0000	.0000	.0000	NULL	1	0
2	6	IF0000000000020587	3	2	.0000	.0000	.0000	NULL	1	0

Como apreciamos en la figura 69 los datos nulos que se apreciaban en la columna "total" fueron imputados con el valor de referencia 0, tras aplicar el comando de depuración para datos nulos.

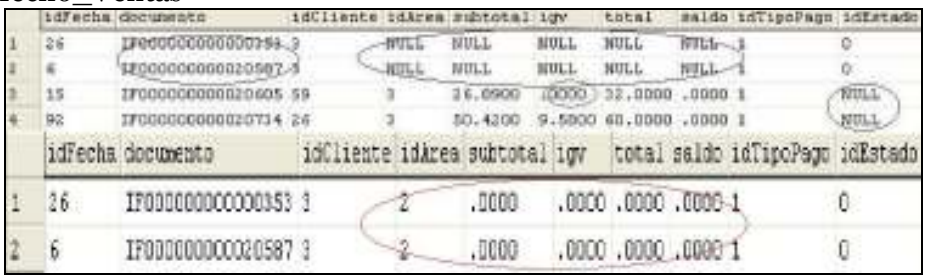
e) Depuración de la Columna "saldo" usando Imputación Hot Deck: Vecino más cercano.- Nos encontramos en la misma situación que con la columna subtotal, ya tenemos la referencia de la columna "idEstado", que hace referencia a los registros anulados y los sin anular, lo único que nos queda hacer es aplicar el mismo comando SQL que aplicamos para imputar la columna "subtotal", pero para este caso cambiamos por la columna "saldo", nos quedaría de la siguiente forma "*update Hecho_Ventas set saldo='0.0'where saldo is null* "

Figura 70: Comando para de la Depuración con Hot Deck en "saldo" del Hecho_Ventas



Solo hacemos click en "iniciar depuración" y nos debería haber imputado los datos faltantes en la columna "saldo".

Figura 71: Resultados de la Depuración con Hot Deck en "saldo" del Hecho_Ventas



	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF0000000000000353	3	2	.0000	.0000	.0000	.0000	1	0
2	6	IF0000000000020587	3	2	.0000	.0000	.0000	.0000	1	0

Como vemos en la figura 71 con este comando de depuración se ha colocado un valor referencial a la columna "saldo", y como también podemos apreciar luego haber depurado todas las columnas nulas o vacías, los registros "IF0000000000000353" y "IF0000000000020587" han quedado limpios de datos nulos.

f) Depuración de la Columna "idEstado" usando Imputación Hot Deck: Vecino más cercano.- Aquí nos encontramos en una situación parecida a la que experimentamos en la columna subtotal, primero procedemos a analizar la columna "idEstado" con la columna vecina más cercana que nos de algún tipo de información sobre el valor de la variable, para ello analizamos la columna "idTipoPago", la cual no nos da ningún tipo de información debido a que es un tipo de variable entera que hace referencia a un registro externo a la tabla, pasamos a analizar la columna saldo, en ella encontramos información como vemos en la figura 72.

Figura 72: Comparación de datos para "idEstado" en Hecho_Ventas

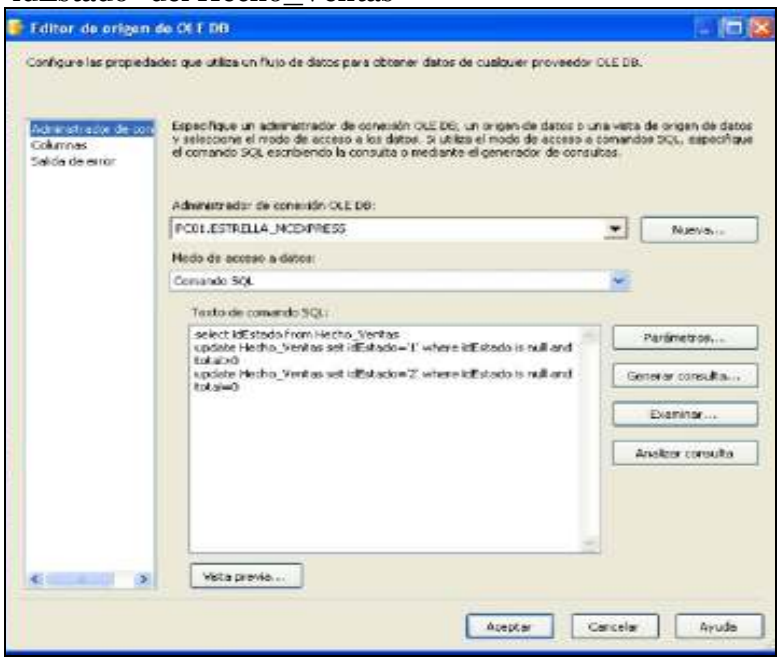
Los vecinos más cercanos antes del registro con valor faltante en la columna "idEstado", solo nos muestra que cuando el saldo es cero el tipo de Pago es 1, que quiere decir esto, que es un registro que se a pagado en efectivo, y sus vecino posteriores nos muestra que cuando el saldo es mayo a cero el tipo de pago es 2, quiere decir que ha sido pagado a través de un crédito.

Seguimos analizando para ver si la columna "total" nos da más información, podemos apreciar en sus vecinos anteriores que cuando la columna "total" es cero y la columna "idEstado" tiene el valor 2, esto quiere decir que es un registro anulado, y cuando el valor de la columna "total" es mayor a cero, la columna "idEstado" tiene el valor 1, lo cual nos da a entender que es un registro no anulado, como se puede apreciar en la figura 73.

Figura 73: Comparación de datos para "idEstado" en Hecho_Ventas

Luego de haber obtenido esta información procedemos a adaptar la técnica de imputación en un comando SQL, la cual nos quedaría de la siguiente forma *"update Hecho_Ventas set idEstado='1' where idEstado is null and total > 0 ", "update Hecho_Ventas set idEstado='2' where idEstado is null and total ='0' "*.
Este comando lo colocamos dentro del editor de origen de OLE DB de la siguiente manera:

Figura 74: Comando para la Depuración con Hot Deck en "idEstado" del Hecho_Ventas



Solo nos queda hacer clic en "iniciar depuración" y nos debería haber imputado los datos faltantes en la columna "idEstado".

Figura 75: Resultados de la Depuración con Hot Deck en "idEstado"

del Hecho_Ventas

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF0000000000000794	3	NULL	NULL	NULL	NULL	NULL	1	0
2	4	IF0000000000020507	3	NULL	NULL	NULL	NULL	NULL	1	0
3	15	IF0000000000020605	59	3	26.0900	.0000	32.0000	.0000	1	NULL
4	92	IF0000000000020734	26	3	50.4200	9.5800	60.0000	.0000	1	NULL

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	15	IF0000000000020605	59	2	26.8900	.0000	32.0000	.0000	1	1
2	92	IF0000000000020734	26	3	50.4200	9.5800	60.0000	.0000	1	1

g) Depuración de Datos Outlier's o Incoherentes en la Columna "igv".- Cuando se tiene una gran cantidad de datos numéricos y se piensa que entre ellos existen datos incoherente u outlier's, antes de su depuración primero debemos saber cuáles son, y para ello se deben de hacer algunas pruebas con técnicas de detección de datos atípicos, para ello existen varias técnicas como: Prueba de Grubbs, Prueba de Dixon, Prueba de Tukey, MOA, Regresión Lineal Simple.

Para este caso usamos la prueba de Dixon, que consiste en saber que registros contienen valores atípicos o incoherentes o también llamados outlier's, esto quiere decir que se va a comparar la forma en que se obtiene el valor del registro con valores atípicos con sus vecinos, analicemos la siguiente figura.

Figura 76: Datos almacenados en el Hecho_Ventas

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	2	IB0000000000002237	52	3	10.0800	1.9200	12.0000	.0000	1	1
2	2	IB0000000000002238	8	2	12507.2900	2376.3900	14883.6800	.0000	1	1
3	5	IB0000000000002239	2	3	45.0000	8.5500	53.5500	.0000	1	1
4	5	IB0000000000002240	2	3	105.2800	20.0000	125.2800	.0000	1	1
5	5	IB0000000000002241	2	3	533.0000	101.2700	634.2700	.0000	1	1
6	5	IB0000000000002242	10	3	6.7200	1.2800	8.0000	.0000	1	1
7	6	IB0000000000002243	13	3	10.0800	1.9200	12.0000	.0000	1	1
8	6	IB0000000000002244	20	3	1015.0000	192.8500	1207.8500	.0000	1	1
9	6	IB0000000000002245	3	2	.0000	.0000	.0000	.0000	1	2
10	6	IB0000000000002246	22	3	9.0000	1.7100	10.7100	.0000	1	1
11	7	IB0000000000002247	33	3	40.2100	7.6400	47.8500	.0000	1	1
12	7	IB0000000000002248	8	2	2394.0000	454.8600	2848.8600	.0000	1	1
13	7	IB0000000000002249	20	2	654.2000	124.3000	778.5000	.0000	1	1
14	8	IB0000000000002250	40	2	204.0000	38.7600	242.7600	.0000	1	1
15	8	IB0000000000002251	5	3	155.4600	29.5400	185.0000	.0000	1	1
16	8	IB0000000000002252	14	3	10.0800	1.9200	12.0000	.0000	1	1
17	9	IB0000000000002253	27	2	14467.3900	2748.8000	17216.1900	.0000	1	1
18	10	IB0000000000002254	55	2	2532.2500	481.1300	3013.3800	.0000	1	1
19	11	IB0000000000002255	47	3	12.0000	.0000	12.0000	.0000	1	1
20	12	IB0000000000002256	3	1	.0000	.0000	.0000	.0000	1	2
21	13	IB0000000000002257	57	2	2596.0400	493.2500	3089.2900	.0000	1	1

Tomamos un registro de la figura 76 para analizar, en este caso tomamos el primer registro y vemos que el valor de la columna igv se calcula de la diferencia de su vecino máximo que es total con su vecino mínimo que es subtotal, cabe resaltar que no todos los valores de igv que sean cero son valores incoherentes, debido a que algunas ventas son boletas y otras facturas, por ejemplo el valor del igv en el registro 19 de la figura es una boleta, porque la diferencia entre el total y el subtotal es cero.
Después de analizar cómo se obtienen los valores para la columna igv

procedemos hacer la prueba de Dixon, para saber que registros contienen valores atípicos en la columna igv que es la sospechosa. Para ello hemos adaptado el funcionamiento del algoritmo de Dixon en un comando SQL, *"select * from Hecho_Ventas where igv<>abs(total-subtotal)"*, arrojándonos los siguientes resultados

Figura 77: Resultados de la Prueba de Dixon en el Hecho_Ventas

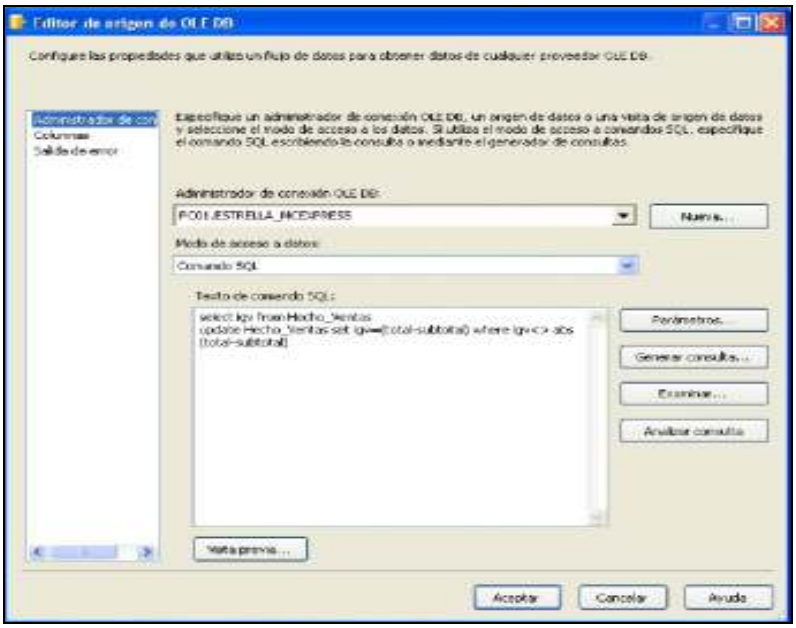
```
select * from Hecho_Ventas where igv<>abs(total-subtotal)
```

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	46	IF0000000000000353	49	3	809.5000	153.8100	963.3000	.0000	1	1
2	66	IF0000000000020586	36	3	67.2300	.0000	80.0000	.0000	1	1
3	85	IF0000000000020605	8	2	2433.7500	.0000	2896.1600	.0000	1	1
4	45	IF0000000000020725	8	2	957.8100	.0000	1139.7900	.0000	1	1

La prueba de Dixon es muy fácil de utilizar, pero el resultado depende fuertemente de escoger correctamente la ubicación de todos las columnas sospechosas, por esto y ser una prueba muy susceptible al ocultamiento o enmascaramiento, se recomienda utilizar la prueba de Dixon sólo para pequeñas muestras cuando sólo uno o dos valores son considerados como atípicos. Luego de haber resuelto como vamos a identificar los valores atípicos procedemos a depurar los datos, se puede usar el algoritmo de Hot Deck Vecino más Cercano para su depuración, para ello hemos adaptado los comandos SQL de la imputación Hot Deck *""update nombre_tabla set variable='valor_numerico' where variable is null and variable > valor_número "*, con el comando SQL de la prueba de Dixon *"select * from nombre_tabla where variable<>abs(valor_máximo-valor_mínimo)",* nos quedaría de la siguiente forma *"update nombre_tabla set variable=(valor_máximo-valor_mínimo) where variable <> abs (valor_máximo-valor_mínimo)"*

Para este caso específico el comando SQL que introduciremos para la depuración es le siguiente *"update Hecho_Ventas set igv=(total-subtotal) where igv<> abs (total-subtotal)"*. Colocamos el comando SQL en el editor de origen de OLE DB, como se muestra en la siguiente figura:

Figura 78: Comando SQL para la depuración de datos adaptado con la prueba de Dixon



Depuramos arrojandonos el siguiente resultado:

Figura 79: Resultados de la Depuración en "igv" del Hecho_Ventas

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	46	IF0000000000000353	49	3	809.5000	153.8100	963.3000	.0000	1	1
2	66	IF0000000000020586	36	3	67.2300	.0000	80.0000	.0000	1	1
3	85	IF0000000000010605	8	2	2433.7500	.0000	2896.1600	.0000	1	1
4	45	IF0000000000010715	6	3	957.8100	.0000	1139.7900	.0000	1	1

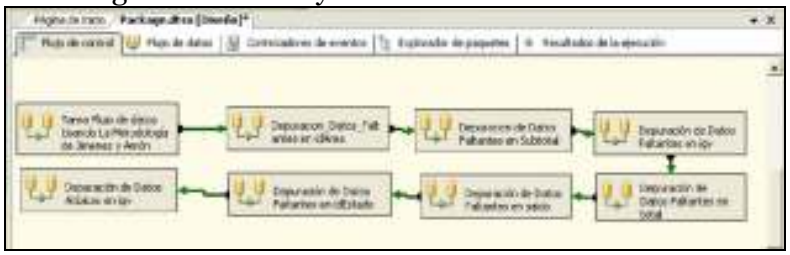
	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	46	IF0000000000000353	49	3	809.5000	153.8100	963.3000	.0000	1	1
2	66	IF0000000000020586	36	3	67.2300	12.7700	80.0000	.0000	1	1
3	85	IF0000000000010605	8	2	2433.7500	462.4100	2896.1600	.0000	1	1
4	45	IF0000000000010715	6	3	957.8100	181.9800	1139.7900	.0000	1	1

Como podemos apreciar los datos se han imputado y han dado coherencia a los resultados que arroja la columna igv de la tabla de hechos venta.

En resumen los pasos a Seguir en la Metodología de Jiménez y Amón son 10, el número de Datos de Faltantes o Nulos son 0 y los números de Datos Atípicos o incoherentes son 0.

Esquema Final

Figura 80: Esquema final de la Depuración de Datos con la metodología de Jiménez y Amón



Comparado con la metodología de Business Intelligence Development Studio 2005, que en 9 pasos realizaba el proceso ETL con resultados de

datos sucios, con 12 datos faltantes y 4 datos atípicos, la metodología de Jiménez y Amón en 10 pasos limpia el 100% de los datos sucios obtenidos con el anterior metodología.

Figura 81: Resultados de la Depuración de Datos en Hecho_Ventas

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF0000000000000353	3	NULL	NULL	NULL	NULL	NULL	1	0
2	6	IF0000000000020587	3	NULL	NULL	NULL	NULL	NULL	1	0
3	15	IF0000000000020605	59	3	26.8900	.0000	32.0000	.0000	1	NULL
4	92	IF0000000000020734	26	3	50.4200	9.5800	60.0000	.0000	1	NULL

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF0000000000000353	3	2	NULL	NULL	NULL	NULL	1	0
2	92	IF0000000000020734	26	3	50.4200	9.5800	60.0000	.0000	1	NULL
3	6	IF0000000000020587	3	2	NULL	NULL	NULL	NULL	1	0
4	15	IF0000000000020605	59	3	26.8900	5.1100	32.0000	.0000	1	NULL

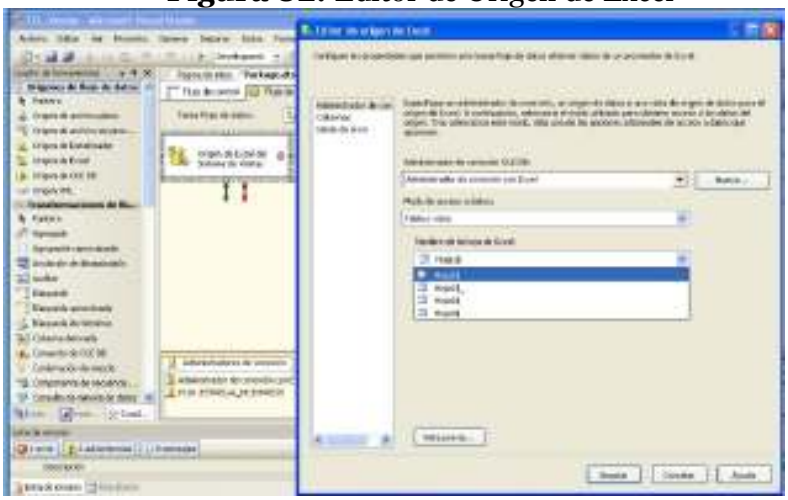
	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF0000000000000353	3	2	.0000	.0000	.0000	.0000	1	2
2	66	IF0000000000020586	36	3	67.2300	12.7700	80.0000	.0000	1	1
3	6	IF0000000000020587	3	2	.0000	.0000	.0000	.0000	1	2
4	15	IF0000000000020605	59	2	26.8900	.0000	32.0000	.0000	1	1
5	33	IF0000000000020657	47	3	12.0000	.0000	12.0000	.0000	1	1
6	45	IF0000000000020725	8	2	957.8100	181.9800	1139.7900	.0000	1	1
7	92	IF0000000000020734	26	3	50.4200	9.5800	60.0000	.0000	1	1

Resultados del Flujo de datos con los pasos resumidos de Jiménez y Amón para asegurar la calidad de los datos durante el proceso ETL

Para los programadores que tienen experiencia, conocen a profundidad los registros de las bases de datos y sobre todo tienen facilidad para detectar posibles anomalías en los datos a cargar, esta metodología te permite realizar la depuración de datos en un solo paso, luego de haber extraído los datos de las BD origen como apreciaremos más adelante.

a) **Extracción de Datos de las BD Origen.-** Primero se escoge del cuadro de herramientas el tipo de origen de datos que en este caso es Origen Excel, luego se establece la conexión usando el administrador de conexión con Excel, se escoge la hoja con la que se va a trabajar.

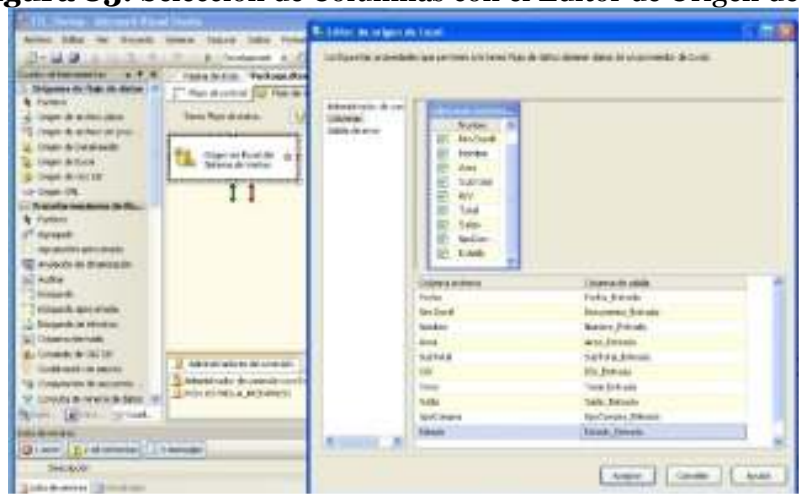
Figura 82: Editor de Origen de Excel



Se escogió las columnas a extraer y se le colocó un alias a cada columna, con la finalidad de no confundirnos durante el resto del proceso, como se

aprecia en la figura 83.

Figura 83: Selección de Columnas con el Editor de Origen de Excel



b) **Depuración de Datos.-** Del cuadro de herramientas se escoge la herramienta flujo de datos la cual llamaremos "Depuración de Datos", se conecta con el flujo de datos al que hemos llamado "ETL Usando la Metodología de Amón y Jiménez Resumida", nos debe quedar de la siguiente forma:

Figura 84: Flujo de paquete de datos

Hacemos doble clic en el flujo de datos "Depuración de Datos", escogemos la herramienta Origen "OLE DB", hacemos doble clic en ella y colocamos los comandos SQL que hemos usado anteriormente para depuración de datos siguiendo la metodología paso a paso:
"*select idarea, subtotal, igev, total, saldo, idEstado from Hecho_Ventas*
Update Hecho_Ventas set idArea=(1+2+3)/3
Update Hecho_Ventas set subtotal =0.0 where subtotal is null and idEstado=2
Update Hecho_Ventas set igv=0.0 where igv is null and idEstado=2
Update Hecho_Ventas set total=0.0 where total is null and idEstado=2
Update Hecho_Ventas set saldo=0.0 where saldo is null and idEstado=2
Update Hecho_Ventas set idEstado=1 where idEstado is null and idEstado >0
Update Hecho_Ventas set idEstado=2 where idEstado is null and idEstado =0

Update Hecho_Ventas set igv=(total-subtotal) where igv <> abs (total - subtotal)".
Nos debe quedar como se aprecia en la figura 85.

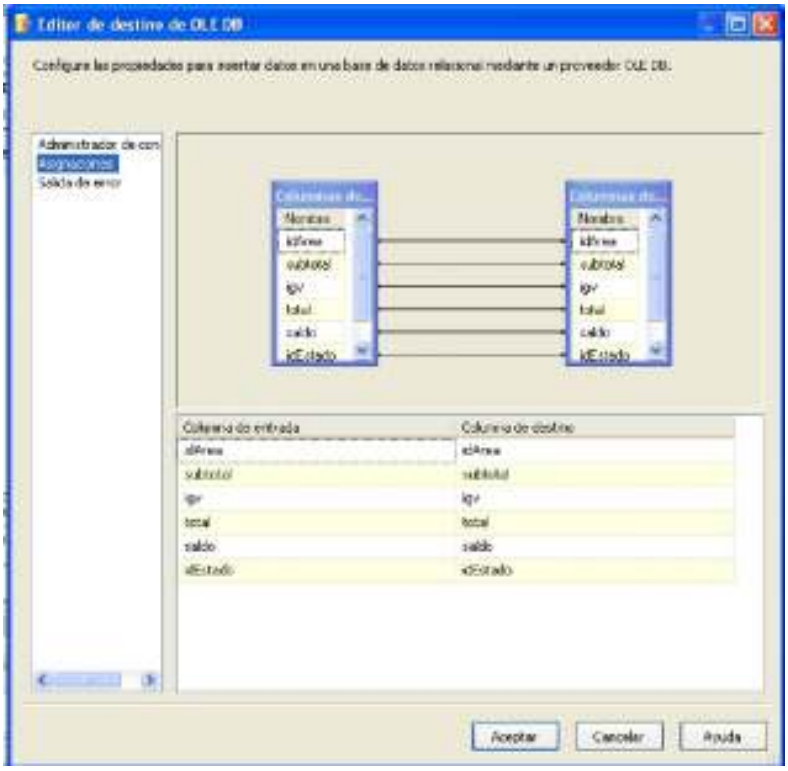
Figura 85: Comandos SQL para la depuración de datos

Damos clic en aceptar y procedemos a escoger del cuadro de herramientas el Destino OLE DB, conectamos con la herramienta de Origen OLE DB, nos debería quedar como en la figura 86.

Figura 86: Conexión Origen y Destino OLE DB

Damos doble clic en Destino OLE DB y en ella escogemos la tabla Temp_HechoVentas, allí asignamos cada una de los ítems de entrada con los de salida, nos debería quedar como en la figura 87.

Figura 87: Editor de destino OLE DB



Por último damos clic en aceptar e iniciamos la depuración, de esta forma los datos han sido depurados en un solo paso, ahorrando tiempo, y mostrando gran mejoría en la calidad de los datos.

Figura 88: Resultados de la Depuración de Datos en Hecho_Ventas

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IFO0000000000000353	3	NULL	NULL	NULL	NULL	NULL	1	0
2	6	IFO0000000000020587	3	NULL	NULL	NULL	NULL	NULL	1	0
3	15	IFO0000000000020605	59	3	26.8900	.0000	32.0000	.0000	1	NULL
4	92	IFO0000000000020734	26	3	50.4200	9.5800	60.0000	.0000	1	NULL

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IFO0000000000000353	3	2	NULL	NULL	NULL	NULL	1	0
2	92	IFO0000000000020734	26	3	50.4200	9.5800	60.0000	.0000	1	NULL
3	6	IFO0000000000020587	3	2	NULL	NULL	NULL	NULL	1	0
4	15	IFO0000000000020605	59	3	26.8900	5.1100	32.0000	.0000	1	NULL

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IFO0000000000000353	3	2	.0000	.0000	.0000	.0000	1	2
2	66	IFO0000000000020586	36	3	67.2300	12.7700	80.0000	.0000	1	1
3	6	IFO0000000000020587	3	2	.0000	.0000	.0000	.0000	1	2
4	15	IFO0000000000020605	59	2	26.8900	.0000	32.0000	.0000	1	1
5	33	IFO0000000000020657	47	3	12.0000	.0000	12.0000	.0000	1	1
6	45	IFO0000000000020725	8	2	957.8100	181.9800	1139.7900	.0000	1	1
7	92	IFO0000000000020734	26	3	50.4200	9.5800	60.0000	.0000	1	1

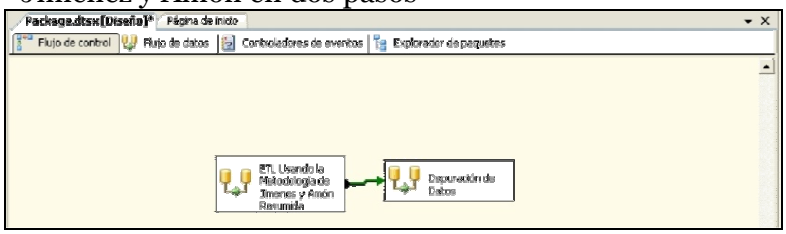
c) Comparado con la metodología de Business Intelligence Development Studio 2005, que en 9 pasos realizaba el proceso ETL con resultados de datos sucios, con 12 datos faltantes y 4 datos atípicos, la metodología de Jiménez y Amón en 2 pasos limpia el 100% de los datos sucios obtenidos con la anterior metodología.

Pasos a Seguir en la Metodología de Jiménez y Amón resumida: 2
Número de Datos de Faltantes o Nulos: 0

Número de Datos Atípicos o incoherentes: 0

Esquema Final

Figura 89: Esquema final usando la metodología de Jiménez y Amón en dos pasos



9. Proceso ETL y Depuración de Datos en Data Mart Compras

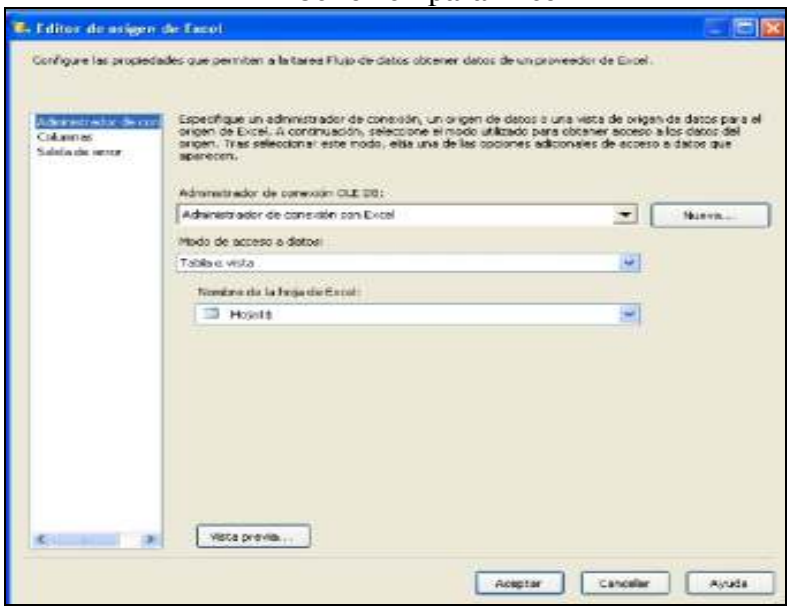
En este parte del trabajo se realizó el proceso ETL en el data mart Compras, para ello se usó la misma plataforma y herramientas que en el data mart Ventas.

9.1. Flujo de Datos con Herramientas de SQL Server Business Intelligence Development de Visual Studio 2005

9.1.1. Flujo de Datos Simple

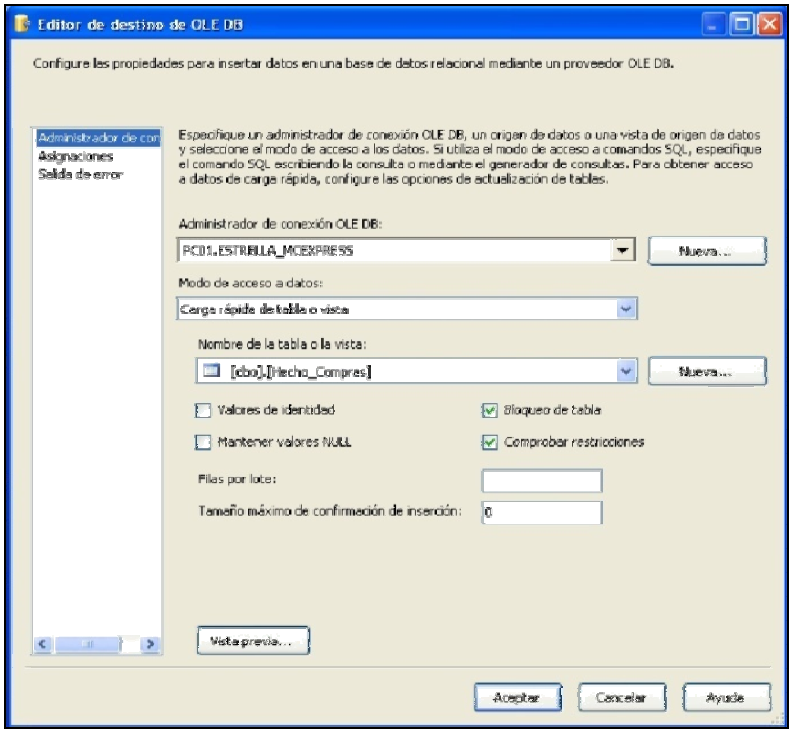
Para demostrar la extracción, transformación y cargar de los datos de una manera simple solo escogió del cuadro de herramientas el control de flujo de datos; el cual contiene en su cuadro de herramientas el tipo de origen con el que estamos tratando, que para este caso es "Origen Excel", se estableció el tipo de conexión a usar, escogimos el modo de acceso a datos y por último escogimos la hoja en donde se encuentran los datos a extraer.

Figura 90: Selección de Tipo de conexión usando el administrador de Conexión para Excel



Luego se escoge el destino en donde se van a cargar los datos, del cuadro de herramientas se escoge el tipo de destino, para este caso es el destino “OLE DB”, para establecer la conexión con la base de datos usamos el administrador de conexión de OLE DB, luego escogemos la tabla de la base de datos en la que vamos a depositar los datos y por último no debemos de olvidar conectar el origen con el destino, debería quedarnos de esta forma.

Figura 91: Selección de Tablas usando el administrador de conexión para Excel



Resultados del Flujo de Datos Simple

Al término del proceso los resultados fueron los siguientes:

Datos Faltantes o Nulos: 1
Datos Atípicos o incoherentes: 17

Tal y como se pueden apreciar en la siguiente tabla.

Figura 92: Resultados del proceso ETL para el data mart Compras, usando el flujo de datos simple.

	idFecha	documento	idProveedor	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	4	EB001-000062	81	1	788.3000	NULL	788.3000	.0000	1	1
2	3	EFC05-02585500	172	1	40.4200	7.6800	48.0900	.0000	1	1
3	32	EFC05-02585501	172	1	39.9700	7.6000	47.5600	.0000	1	1
4	77	EFC05-02586310	172	1	60.0700	11.4100	71.4600	.0000	1	1
5	33	EFC05-02588802	172	1	54.3800	10.3300	64.6800	.0000	1	1
6	37	EFC05-02853356	172	1	44.3700	8.4300	52.3000	.0000	1	1
7	89	EB017-001386	180	1	.0000	.0000	5.0000	.0000	1	1
8	6	EF021-017110	14	3	.0000	1.1200	7.0000	.0000	1	1
9	79	EF0003-025707	21	3	8.4000	1.6000	.0000	.0000	1	1
10	14	EF0003-025843	21	3	5.0400	.9600	.0000	.0000	1	1
11	80	EF054-004227	30	3	6.7200	1.2800	.0000	.0000	1	1
12	44	EF001-001087	34	3	8.4000	1.6000	.0000	.0000	1	1
13	85	EF0010499935	68	3	16.8100	.0000	20.0000	.0000	1	1
14	29	EF001-21548749	120	3	80.1400	17.1600	408.8100	.0000	1	1
15	44	EF0003-001083	142	3	.0000	11.1800	70.0000	.0000	1	1
16	31	EF708-0004934	169	3	.0000	7.1200	44.5700	.0000	1	1
17	-6	EF127-0020167	179	3	5.0400	.9600	6.0000	.0000	1	1
18	-24	EF127-0020502	179	3	5.0400	.9600	6.0000	.0000	1	1

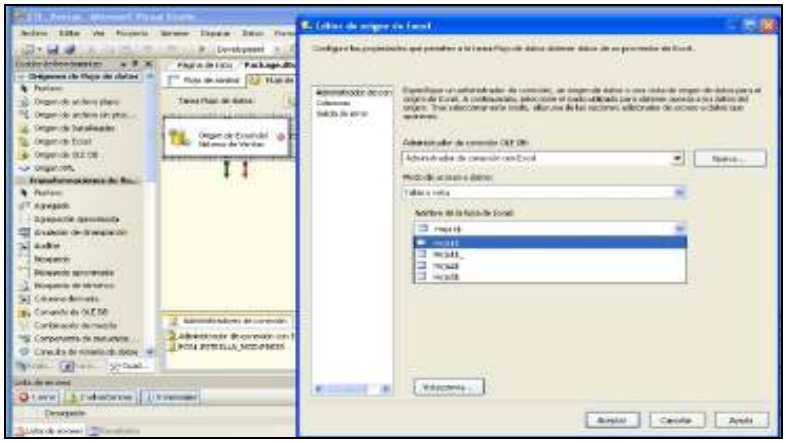
Luego de intentar depurar los datos que nos arroja el flujo de datos simple, ninguna herramienta proporcionada por la plataforma que estamos usando pudo depurar.

9.1.2. Flujo de Datos con Herramientas de SQL Server Business Intelligence Development de Visual Studio 2005

Para poder extraer, transformar y cargar los datos de una BD en Excel hacia una BD en SQL Server 2005, según lo que propone la metodología de Visual Studio para el desarrollo de Business Intelligence, se debe de hacer uso de varias herramientas las cuales servirán en cada uno de los pasos de extracción transformación y carga que a continuación se muestran.

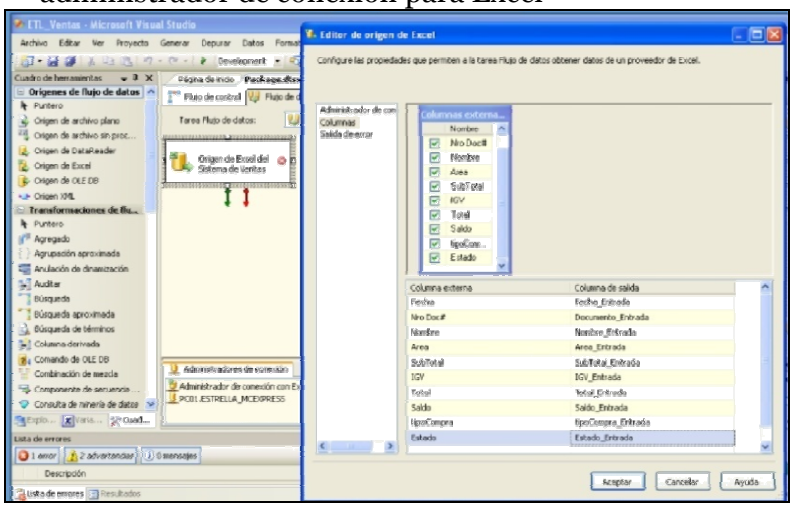
a) Elección de Tablas de la base de datos de origen.- Primero se escoge del cuadro de herramientas el control flujo de datos, hacemos doble clic en él y procedemos a escoger del mismo cuadro de herramientas el tipo de origen de datos que en este caso es “Origen Excel”, luego se establece la conexión usando el administrador de conexión con Excel, se escoge la hoja con la que se va a trabajar.

Figura 93: Selección de Tablas usando el administrador de conexión para Excel



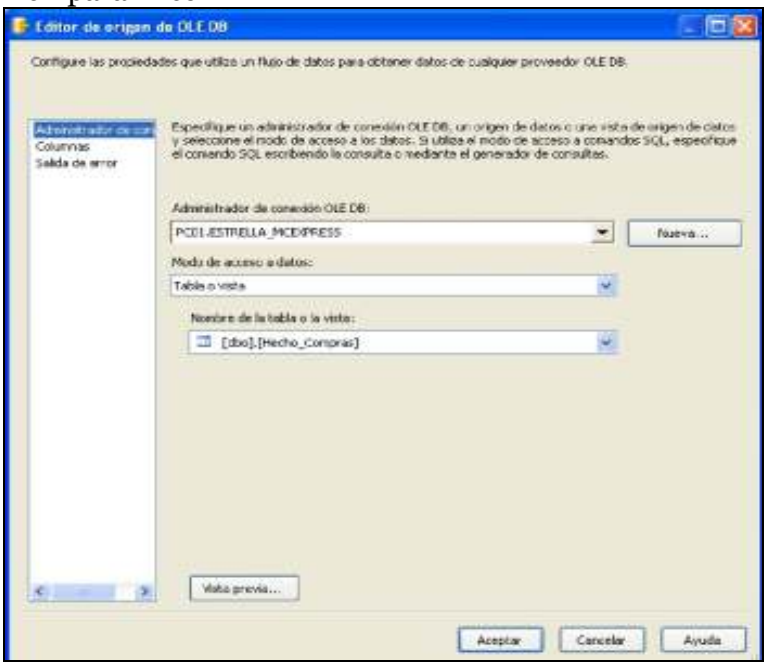
Luego hacemos click en “columnas” se escogen las columnas que vamos a extraer y se le coloca un alias a cada columna, con la finalidad de no confundirnos durante el resto del proceso.

Figura 94: Selección de Columnas usando el administrador de conexión para Excel



b) Escoger el Origen-Destino de los datos a cargar.- La metodología de visual Studio nos indica que para poder cargar los datos a una base de datos destino se tienen que mezclar los datos de entrada como de salida, para ello se usó la tabla de origen-destino; del cuadro de herramientas usamos el tipo de origen “Origen OLE DB”, luego establecimos conexión con la base de datos y escogimos la tabla con la que se va a trabajar.

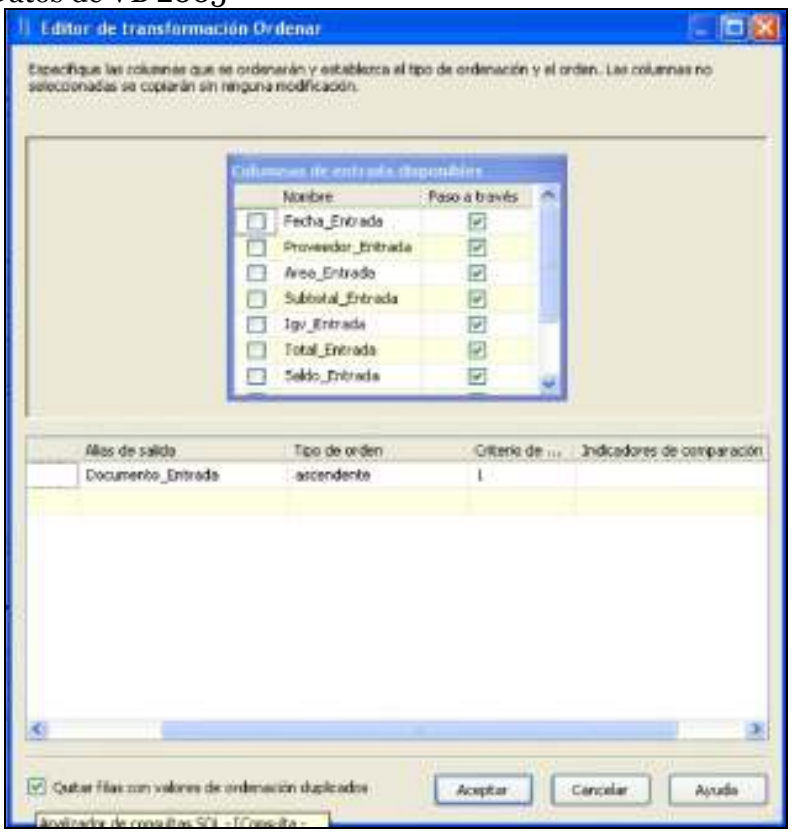
Figura 95: Selección del Origen-Destino usando el administrador de conexión para Excel



Escogimos las columnas que vamos a cargar y se le coloca un alias a cada columna, con la finalidad de no confundirnos durante el resto del proceso.

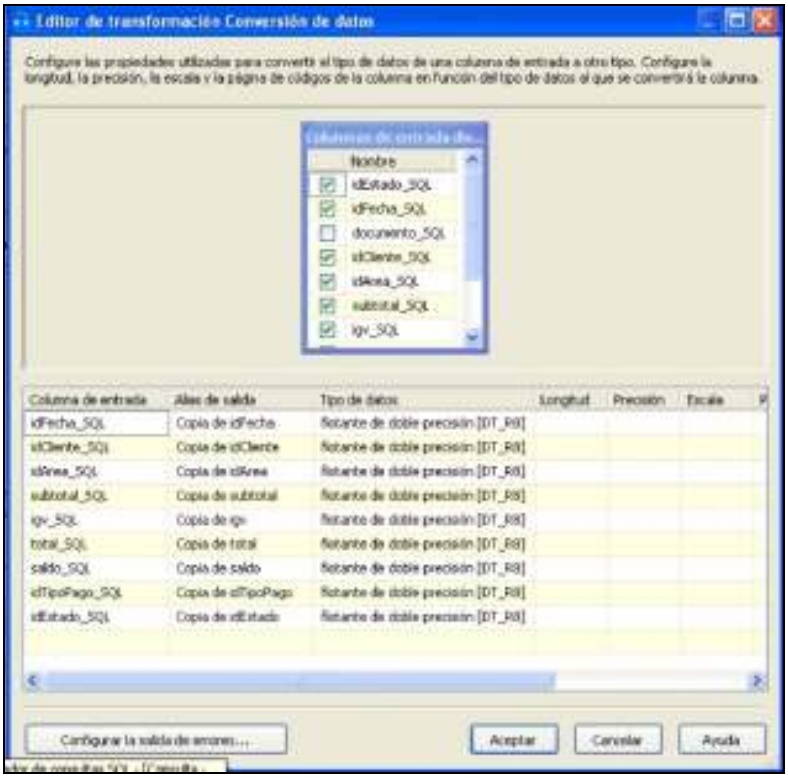
c) Ordenar Datos Origen Excel.- Para poder mezclar y posteriormente cargar los datos en la BD en SQL Server, primero debemos ordenar los datos de la tabla que se está usando, se escoge la herramienta "Ordenar" que se encuentra en el cuadro de herramientas, se hace doble clic y se procede a ordenar los datos, para ello debemos tomar la columna con datos que no se repitan, como por ejemplo una clave primaria, pero esta herramienta también nos permite quitar los datos duplicados, solo debemos hacer un check en la opción "quitar filas con valores de ordenación duplicados", como se muestra en la siguiente figura.

Figura 96: Ordenar los datos con Herramienta de Ordenación de Datos de VB 2005



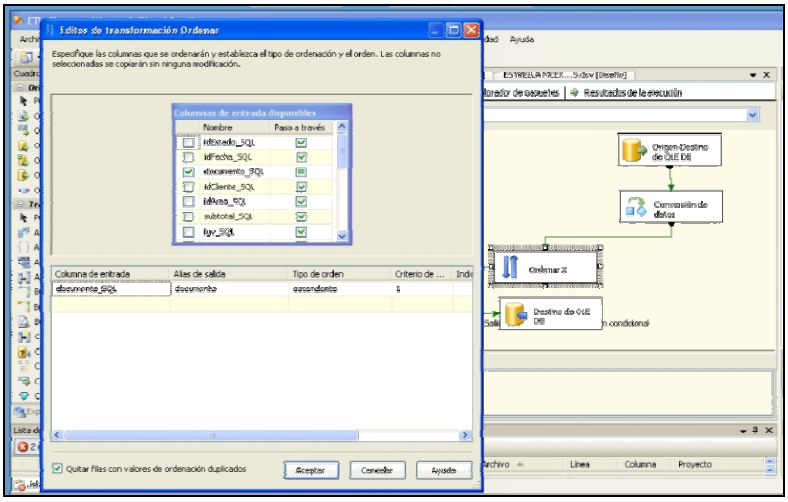
d) Conversión de Datos Origen-Destino OLE DB.- Para no caer en el error de datos no compatibles, se debe de transformar los datos que se están extrayendo, en este caso vamos a transformar los datos en donde se van a depositar, para lo cual usamos la herramienta "Transformación de Datos", que se encuentra en el cuadro de herramientas, hacemos doble clic y procedemos a elegir los valores que se van a transformar, luego se escoge el tipo de dato que se requiere para que sean compatibles, como se muestra en la figura 97.

Figura 97: Transformación de Datos usando la Herramienta Conversión de Datos



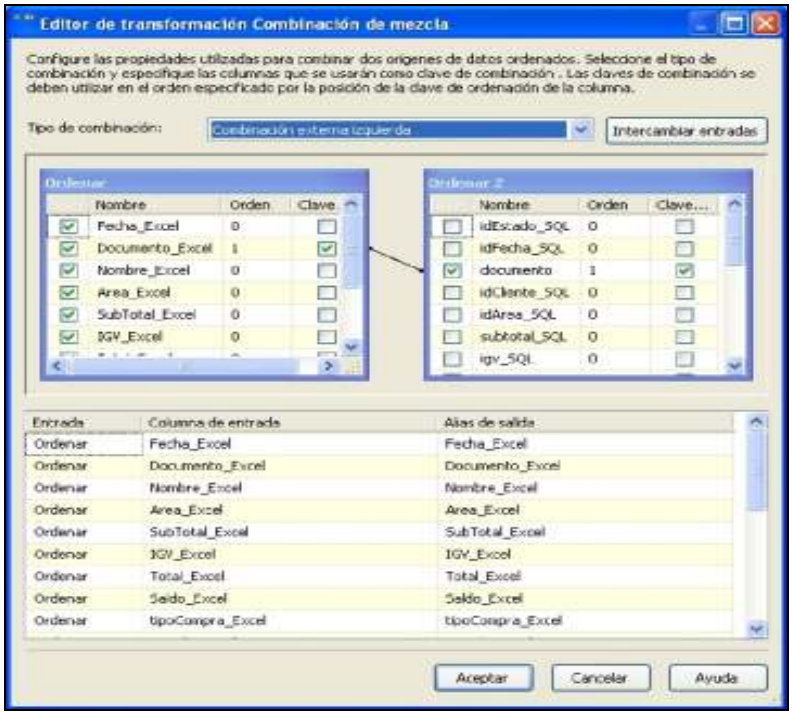
e) Ordenar Datos Origen-Destino OLE DB.- Se escoge la herramienta "Ordenar" que se encuentra en el cuadro de herramientas, se hace doble clic y se procede a ordenar los datos, para ello debemos tomar la columna con datos que no se repitan, como por ejemplo una clave primaria, pero esta herramienta también nos permite quitar los datos duplicados, solo debemos hacer un check en la opción "quitar filas con valores de ordenación duplicados", como se muestra en la figura 98.

Figura 98: Ordenación de datos Origen-Destino usando la herramienta de Ordenar Datos



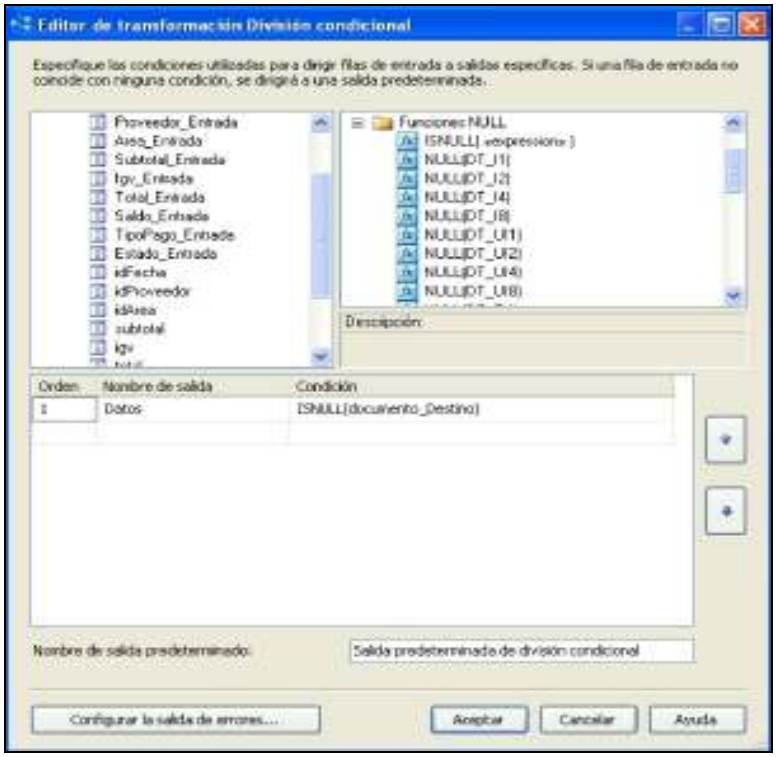
f) Transformación de Mezcla.- Después tener los datos origen y destino ordenados procedemos a mezclarlos para poder cargarlos en la base de datos final. Seleccionamos todos los ítems de origen y los ítems que se transformaron para lograr la compatibilidad de los tipos de datos, se escoge el tipo de combinación que para este caso es la "combinación externa izquierda", y se procedió a depurar los datos.

Figura 99: Transformación de Datos usando la herramienta Transformación de Mezcla



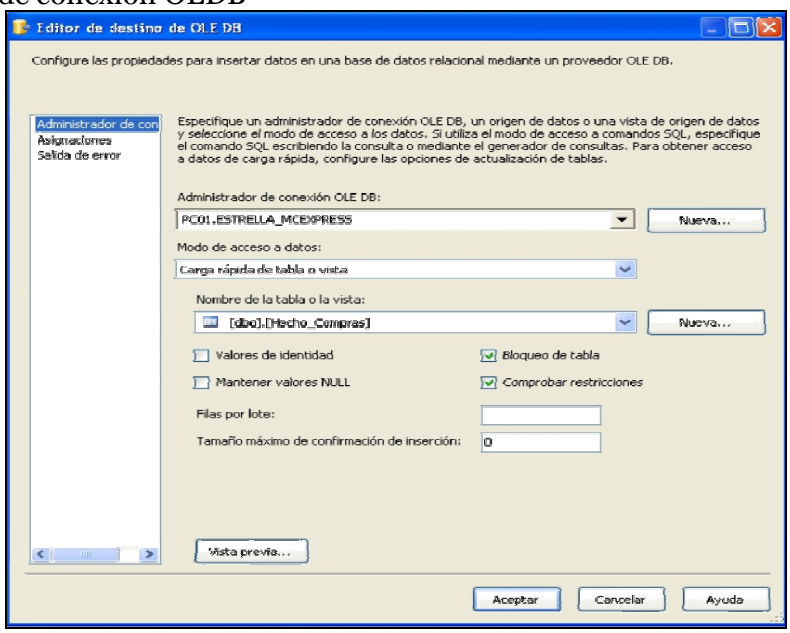
g) División condicional.- Esta herramienta nos es útil para la depuración de datos, tiene diversas funciones que nos ayudan a la depuración de datos como son: funciones matemáticas, funciones de cadena, funciones de fecha y hora, función de valores null, etc. A continuación hacemos uso de la función NULL para evitar cargar datos nulos en nuestra BD dimensional, escogimos la función ISNULL y la arrastramos hacia el ítem de condición que se encuentra en la parte inferior del cuadro.

Figura 100: Imputación de Datos nulos usando la herramienta de División Condicional



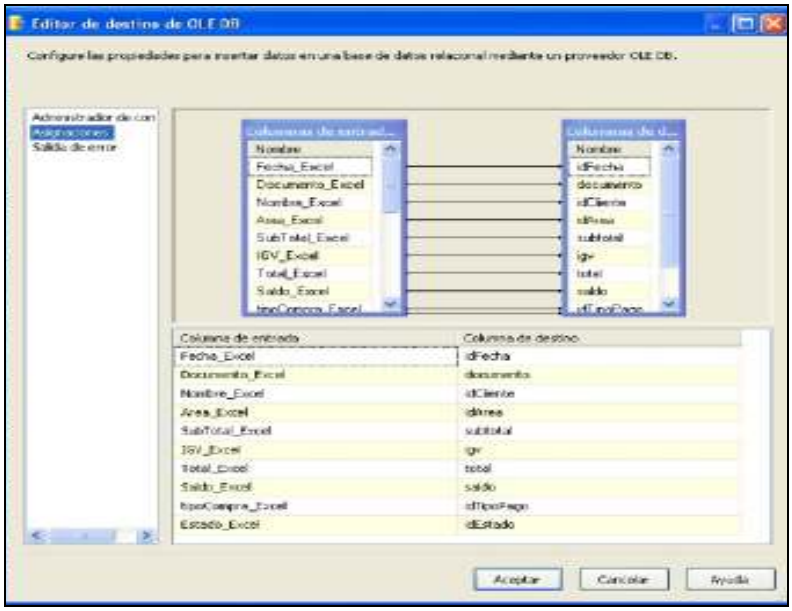
h) Destino OLEDB data mart Compras.- En el último paso para cargar los datos se hizo uso de la herramienta "Destino OLE DB", escogiendo primero la conexión OLE DB a través del administrador de conexión de OLE DB, se hizo una carga rápida de tabla o vista, se escogió la tabla o vista de la BD destino que para este caso es el "Hecho_Compras".

Figura 101: Selección de Destino de Datos usando el administrador de conexión OLDB



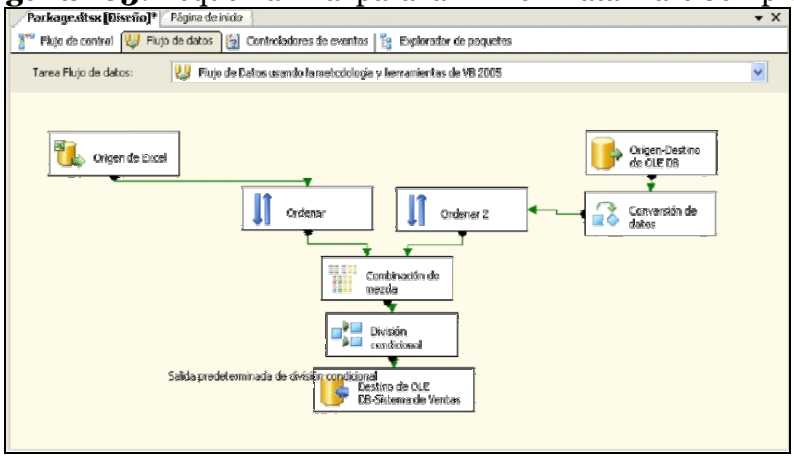
Por último asignamos y relacionamos los datos de origen con los de destino.

Figura 102: Unión de datos de origen con datos destino usando el administrador de conexión OLEDB



Al final de todo el proceso el diseño del ETL es como el que se aprecia en la figura 103.

Figura 103: Esquema final para la ETL en Data mart Compras



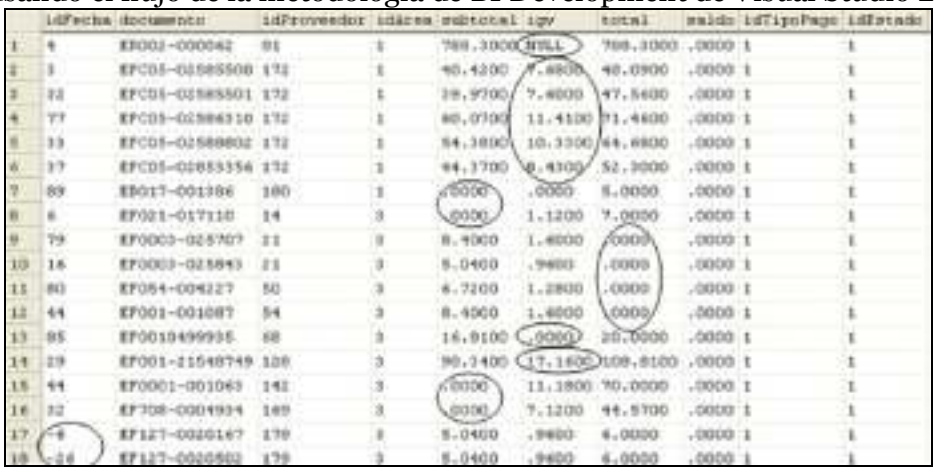
Resultados del Flujo de Datos con Herramientas de SQL Server Business Intelligence Development de Visual Studio 2005

Luego de haber iniciado la depuración del proceso utilizando la metodología y herramientas que nos brinda la plataforma de visual Studio, los resultados fueron los siguientes.

Número de Pasos en el proceso ETL: 9
Número de Datos Faltantes o Nulos: 1
Número de Datos Incoherentes: 17

Tal como se muestra en la figura 104:

Figura 104: Resultados del proceso ETL para el data mart Compras, usando el flujo de la metodología de BI Development de Visual Studio 2005



	idFecha	documento	idProveedor	idAlmacen	subtotal	igv	total	saldo	idTipoPago	idEstado
1	4	EB002-000042	91	1	788.3000	NULL	788.3000	.0000	1	1
2	3	EFC05-01585508	172	1	40.4200	7.6800	48.0900	.0000	1	1
3	22	EFC05-01585501	172	1	39.9700	7.6000	47.5600	.0000	1	1
4	77	EFC05-02586310	172	1	60.0700	11.4100	71.4600	.0000	1	1
5	33	EFC05-02588802	172	1	54.3800	10.3300	64.6800	.0000	1	1
6	17	EFC05-01853356	172	1	44.1700	8.4300	52.3000	.0000	1	1
7	89	EB017-001386	180	1	.0000	.0000	5.0000	.0000	1	1
8	6	EF021-017110	14	3	.0000	1.1200	7.0000	.0000	1	1
9	79	EF0003-025707	21	3	8.4000	1.4000	.0000	.0000	1	1
10	16	EF0003-025893	21	3	5.0400	.9600	.0000	.0000	1	1
11	80	EF054-004227	50	3	6.7200	1.2800	.0000	.0000	1	1
12	44	EF001-001087	54	3	8.4000	1.4000	.0000	.0000	1	1
13	85	EF0019499935	68	3	16.8100	.0000	20.0000	.0000	1	1
14	29	EF001-21548749	120	3	90.3400	17.1600	108.8100	.0000	1	1
15	44	EF0001-001063	142	3	.0000	11.1800	70.0000	.0000	1	1
16	32	EF708-0004934	169	3	.0000	7.1200	44.5700	.0000	1	1
17	-8	EF127-0020167	170	3	5.0400	.9600	6.0000	.0000	1	1
18	-24	EF127-0020502	170	3	5.0400	.9600	6.0000	.0000	1	1

Se procedió a realizar la depuración de datos con las herramientas que nos brinda la plataforma de Visual Studio, que para estos casos es la de "División Condicional" y tras varios intentos no se pudo limpiar los datos faltantes y los incoherentes.

9.2. Flujo de Datos Usando la Metodología propuesta por Jiménez y Amón para la Depuración en data mart Compras

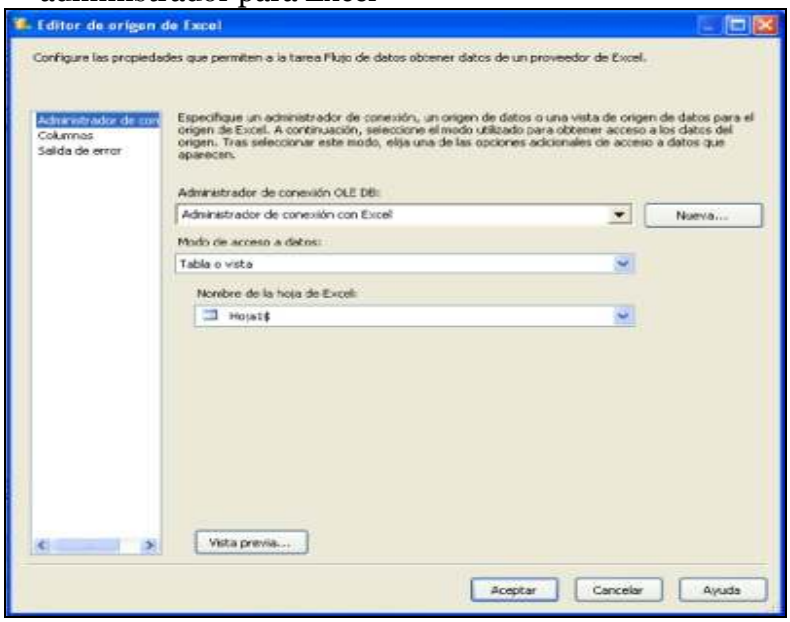
9.2.1. Desarrollo de la Metodología Paso por Paso

Esta metodología nos propone el uso de técnicas plasmadas en algoritmos, para poder extraer, transformar y cargar los datos de una base de datos Origen a una Destino, los algoritmos en su totalidad están diseñado para sistemas expertos y/o base de datos netamente estadísticas, debido al fuerte uso de herramientas estadísticas que hacen, para este caso, luego de haber estudiado su funcionamiento, hemos adaptado estos algoritmos en comandos SQL.

9.2.1.1. Carga de Datos

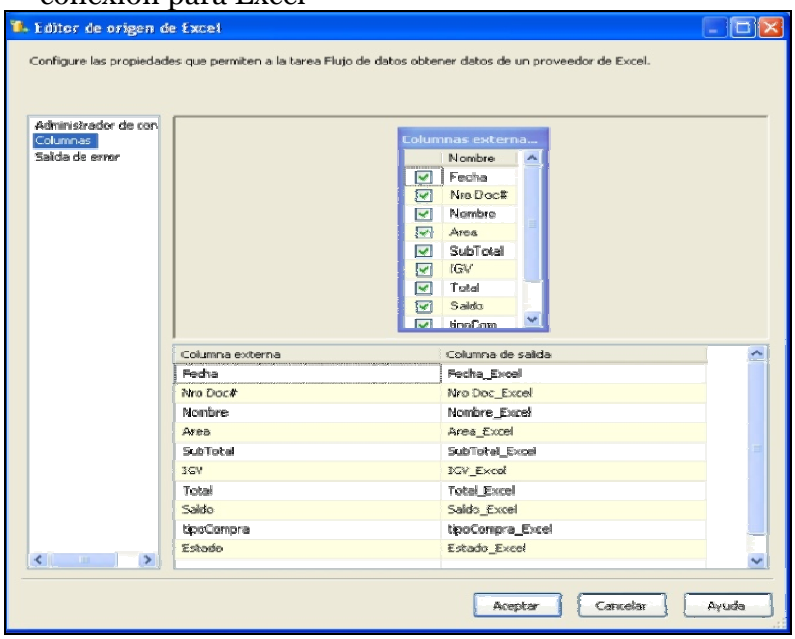
a) Escoger los datos de una Base de Datos Origen en Excel.- Primero se escoge del cuadro de herramientas el control de "flujo de datos", hacemos doble clic y escogemos la herramienta "Origen Excel", luego se establece la conexión usando el administrador de conexión con Excel, se escoge la hoja con la que se va a trabajar.

Figura 105: Selección del Origen de Datos usando el administrador para Excel



Luego hacemos clic en "columnas" se escogen las columnas que vamos a extraer y se le coloca un alias a cada columna, con la finalidad de no confundirnos durante el resto del proceso.

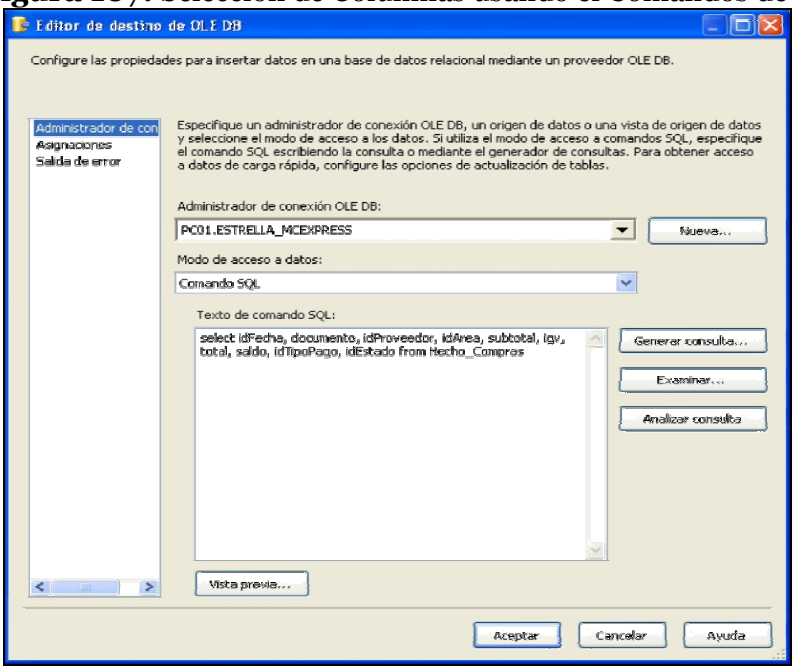
Figura 106: Selección de Columnas usando el administrador de conexión para Excel



b) Escoger el Repositorio destino en una Base de Datos en SQL Server.- Del cuadro de herramientas se escoge la herramienta "Destino de OLE DB", utilizando el administrador de conexión OLE DB se procede a establecer con la base de datos que para este caso es "ESTRELLA_MCEXPRESS", luego usamos el modo de acceso a datos "Comando SQL" y procedemos a

seleccionar las columnas de la tabla destino, para este caso es la tabla "Hecho_Compras", con la siguiente consulta "select idFecha, documento, idCliente, idArea, subtotal, igv, total, saldo, idTipoPago, idEstado from Hecho_Compras", todo este comando se coloca en la casilla en blanco como se muestra en la figura 107.

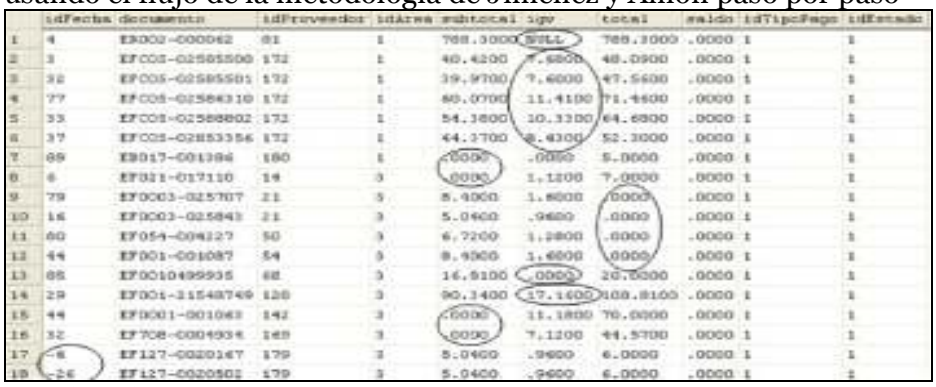
Figura 107: Selección de Columnas usando el Comandos de SQL



c) Resultados del Flujo de Datos Usando la Metodología propuesta por Jiménez y Amón Paso por Paso.- Luego de haber iniciado la depuración de la primera parte del proceso utilizando la metodología de Jiménez y Amón, los resultados fueron los siguientes.

Número de Pasos en el proceso ETL: 9
Número de Datos Faltantes o Nulos: 1
Número de Datos Incoherentes: 17

Figura 108: Resultados del proceso ETL para el data mart Compras, usando el flujo de la metodología de Jiménez y Amón paso por paso



	idFecha	documento	idProveedor	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	4	EB002-000062	81	1	788.3000	NULL	788.3000	.0000	1	1
2	3	EF003-02585500	172	1	40.4200	7.6800	48.0900	.0000	1	1
3	32	EF003-02585501	172	1	39.9700	7.6000	47.5600	.0000	1	1
4	77	EF003-02584310	172	1	60.0700	11.4100	71.4800	.0000	1	1
5	33	EF003-02588802	172	1	54.3800	10.3300	64.6800	.0000	1	1
6	37	EF003-02853356	172	1	44.3700	8.4300	52.3000	.0000	1	1
7	89	EB017-001384	180	1	.0000	.0000	5.0000	.0000	1	1
8	6	EF021-017110	14	3	.0000	1.1200	7.0000	.0000	1	1
9	79	EF003-025707	21	3	5.4000	1.0800	.0000	.0000	1	1
10	16	EF003-025843	21	3	5.0400	.9600	.0000	.0000	1	1
11	60	EF054-006127	50	3	6.7200	1.2800	.0000	.0000	1	1
12	44	EF001-001087	54	3	8.4000	1.6000	.0000	.0000	1	1
13	85	EF0010499935	68	3	16.8100	.0000	20.0000	.0000	1	1
14	29	EF001-31548749	120	3	90.3400	17.1600	108.8100	.0000	1	1
15	44	EF0001-001063	142	3	.0000	11.1800	70.0000	.0000	1	1
16	32	EF708-0004934	169	3	.0000	7.1200	44.5700	.0000	1	1
17	-6	EF127-0020167	179	3	5.0400	.9600	6.0000	.0000	1	1
18	-26	EF127-0020501	179	3	5.0400	.9600	6.0000	.0000	1	1

9.1.1.2. Limpieza de Datos.

a) Depuración de la Columna id Fecha usando la Prueba de Dixón e Imputación Hot Deck aleatorio.- Primero usamos la prueba de Dixón para verificar si existen valores atípicos en nuestros registros de la columna idFEcha, debido a que existen sospechas sobre la calidad de los datos, recordemos que la prueba de Dixón consiste en verificar mediante la diferencia de un valor mínimo y un valor máximo, la atipicidad de un registro. Usando como punto de partida la forma cómo actúa esta técnica se adaptó un comando SQL para usar en esta ocasión *"select * from nombre_Tabla where variable<numero_mínimo or variable> numero_máximo",* resultando en la siguiente consulta *"select * from Hecho_Compras where idFecha<'2' or idFEcha>'92'"*, obteniendo los siguientes resultados para la columna idFecha.

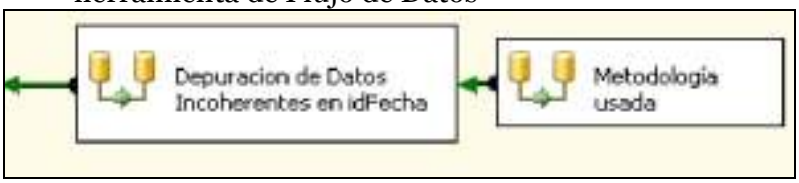
Figura 109: Resultados de la prueba de Dixon para la columna idFecha en Data mart Compras

	idFecha	documento	idProveedor	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	-6	EF127-0020167	179	3	5.0400	.9600	6.0000	.0000	1	1
2	-26	EF127-0020502	179	3	5.0400	.9600	6.0000	.0000	1	1

Como resultado tenemos dos valores atípicos en la columna "idFecha", para limpiar esos datos usaremos una técnica de imputación, descartamos el uso de las técnicas de imputación por medio de la media y mediana por haber muchos registros lo que dificulta sacarles el promedio. Verificamos si se puede usar imputación por Hot deck y vemos que el que más se adapta al tipo de datos es la Imputación Hot Deck: Muestreo aleatorio simple, en donde los donantes se extraen de manera aleatoria. Dado un esquema de muestreo equiprobable, la media se puede estimar como la media de los receptores y los donantes (Juárez, 2003).
Entonces si sabemos cómo funciona esta técnica podemos adaptar un comando SQL para depurar estos datos, el comando quedaría de la siguiente forma *"Update nombre_tabla set variable= abs(cast(newid() as binary (numero_caracteres) % numero_máximo) + 1 where variable<numero_mínimo or variable>numero_máximo"*, lo que resultaría en la siguiente consulta *"UPDATE Hecho_Compras Set idFecha = ABS(CAST(NEWID() as binary(2)) % 92) + 1 where idFecha<2 or idfecha>92".*

Antes de aplicarlo se debe escoger la herramienta "tarea de flujo de datos", y lo unimos al flujo de datos anterior como vemos en la siguiente figura.

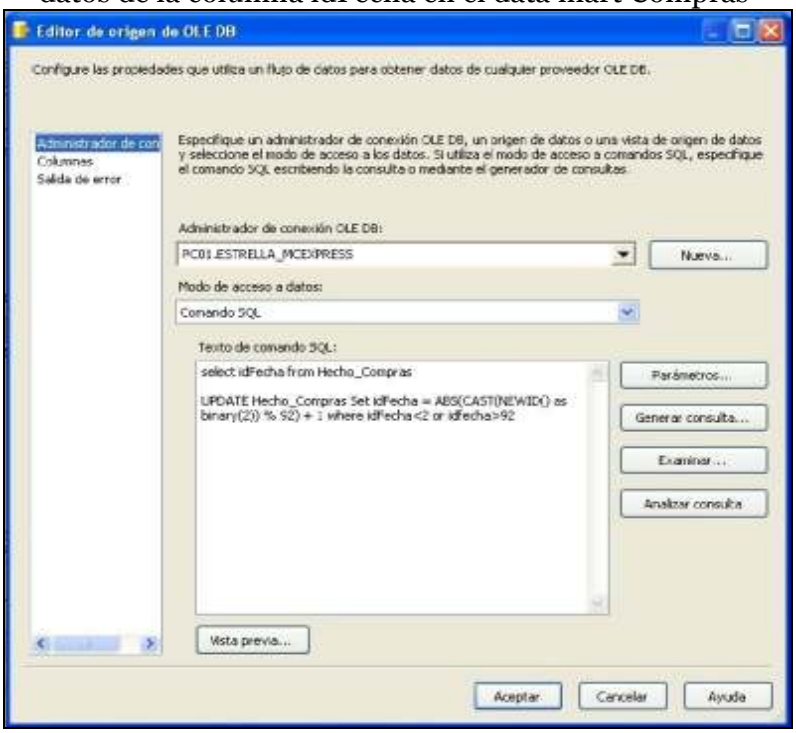
Figura 110: Esquema de Conexión Usando la herramienta de Flujo de Datos



Procedimos a depurar seleccionando "depuración de Datos incoherentes en idFecha", escogemos la herramienta origen de datos "OLE DB",

hacemos doble clic y en el modo de acceso a datos escogemos “comando SQL”, allí colocamos el comando adaptado de la técnica de depuración de datos que vamos a usar y hacemos clic en aceptar.

Figura 111: Comando SQL para la imputación de datos de la columna idFecha en el data mart Compras



Luego de ejecutar el comando nos dio los siguientes resultados, como se muestra en la figura 112.

Figura 112: Resultados de la imputación Hot Deck: Muestreo aleatorio Simple para la columna idFecha en Data mart Compras

	idFecha	documento	idProveedor	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	68	EF127-0020167	179	3	5.0400	.9600	6.0000	.0000	1	1
2	17	EF127-0020503	179	3	5.0400	.9600	6.0000	.0000	1	1

b) Depuración de la Columna Subtotal usando Prueba de Dixon e Imputación Hot Deck: Vecino más cercano.- Para determinar si existen datos atípicos realizamos la prueba de Dixon *“select * from nombre_tabla where variable<>abs (numero_máximo-numero_mínimo)”*, lo que nos da *“select * from Hecho_Compras where subtotal<>abs (total-igv)”*, en este caso observamos la columna “subtotal” en donde se encuentra el valor cero o un valor incoherente.

Figura 113: Datos para verificar y realizar la imputación según los vecinos más cercanos

	idFecha	documento	idProveedor	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	3	EFC05-02585500	172	1	40.4200	7.6800	48.0900	.0000	1	1
2	32	EFC05-02585501	172	1	39.9700	7.6000	47.5600	.0000	1	1
3	77	EFC05-02586310	172	1	60.0700	11.4100	71.4600	.0000	1	1
4	33	EFC05-02588802	172	1	54.3800	10.3300	64.6800	.0000	1	1
5	37	EFC05-02853356	172	1	44.3700	8.4300	52.3000	.0000	1	1
6	69	EB017-001386	180	1	.0000	.0000	5.0000	.0000	1	1
7	6	EF021-017110	14	3	.0000	1.1200	7.0000	.0000	1	1
8	79	EF0003-025707	21	3	8.4000	1.6000	.0000	.0000	1	1
9	16	EF0003-025843	21	3	5.0400	.9600	.0000	.0000	1	1
10	80	EF054-004227	50	3	6.7200	1.2800	.0000	.0000	1	1
11	44	EF001-001087	54	3	8.4000	1.6000	.0000	.0000	1	1
12	85	EF0010499935	66	3	16.8100	.0000	20.0000	.0000	1	1
13	29	EF001-21540749	120	3	90.3400	17.1600	108.0100	.0000	1	1
14	44	EF0001-001063	142	3	.0000	11.1800	70.0000	.0000	1	1
15	32	EF708-0004934	169	3	.0000	7.1200	44.5700	.0000	1	1

Para imputarlos usamos imputación Hot Deck, si observamos a las columnas vecinas las cuatro primeras no nos brinda ningún tipo de información, debido a que la columna subtotal hace referencia a un tipo de dato moneda y las anteriores hacen referencia a claves foráneas de tipo entero, las únicas columnas que nos dan algún tipo de información es igv, total e idEstado, en las dos primeras podemos apreciar que el subtotal se halla mediante la resta de total con el igv, y la última nos dice que son registros no anulado por lo tanto su valor es mayor a cero.

Por lo tanto, el comando SQL quedaría de la siguiente forma *"update nombre_tabla set variable='vecino_máximo-vecino_mínimo'where variable_apoyo>0"*, lo que nos daría la siguiente consulta *SQL "UPDATE Hecho_Compras set subtotal=total-igv where subtotal<>abs (total-igv) and total>0"* veamos los resultados y comparemos su calidad en la figura 114.

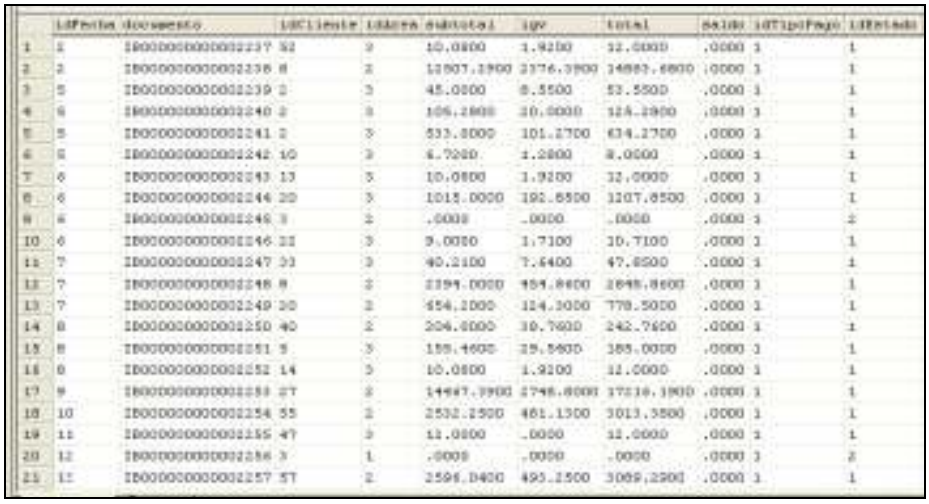
Figura 114: Resultados la imputación en la columna "subtotal" del data mart compras

Si apreciamos las figuras en los registros EB017-001386, EF021-017110 Y EF0010499935 el subtotal antes de la depuración era de incoherente, después de la depuración tiene sentido, esto quiere decir que limpia la totalidad de los datos.

c) Depuración de Datos Outlier's o Incoherentes en la Columna igv usando Dixon e Imputación Hot Deck Vecino más Cercano.- Para este caso usamos la prueba de Dixon, que consiste en saber que

registros contienen valores atípicos o incoherentes o también llamados outlier's. Esto quiere decir que se va a comparar la forma en que se obtiene el valor del registro con valores atípicos con sus vecinos, analicemos la figura 115.

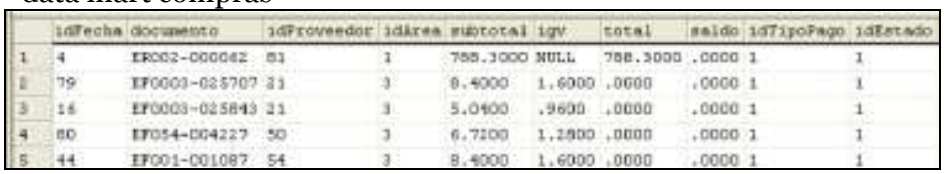
Figura 115: Datos para verificar y realizar la imputación según los vecinos más cercanos.



	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	1	IB0000000000002237	52	3	10.0800	1.9200	12.0000	.0000	1	1
2	2	IB0000000000002238	8	2	12007.1900	2376.3900	14383.6800	.0000	1	1
3	5	IB0000000000002239	2	3	45.0000	8.5500	53.5500	.0000	1	1
4	6	IB0000000000002240	2	3	106.2800	20.0000	126.2800	.0000	1	1
5	5	IB0000000000002241	2	3	533.0000	101.2700	634.2700	.0000	1	1
6	6	IB0000000000002242	10	3	6.7200	1.2800	8.0000	.0000	1	1
7	6	IB0000000000002243	13	3	10.0800	1.9200	12.0000	.0000	1	1
8	6	IB0000000000002244	20	3	1015.0000	192.8500	1207.8500	.0000	1	1
9	6	IB0000000000002245	3	2	.0000	.0000	.0000	.0000	1	2
10	6	IB0000000000002246	22	3	9.0000	1.7100	10.7100	.0000	1	1
11	7	IB0000000000002247	33	3	40.2100	7.6400	47.8500	.0000	1	1
12	7	IB0000000000002248	8	2	2394.0000	454.8600	2848.8600	.0000	1	1
13	7	IB0000000000002249	30	2	654.2000	124.3000	778.5000	.0000	1	1
14	8	IB0000000000002250	40	2	204.0000	38.7600	242.7600	.0000	1	1
15	8	IB0000000000002251	5	3	155.4600	29.5600	185.0000	.0000	1	1
16	8	IB0000000000002252	14	3	10.0800	1.9200	12.0000	.0000	1	1
17	9	IB0000000000002253	27	2	14967.1900	2746.8000	17713.1900	.0000	1	1
18	10	IB0000000000002254	55	2	2532.2500	481.1300	3013.3500	.0000	1	1
19	11	IB0000000000002255	47	3	12.0000	.0000	12.0000	.0000	1	1
20	12	IB0000000000002256	3	1	.0000	.0000	.0000	.0000	1	2
21	12	IB0000000000002257	57	2	2596.0400	493.2500	3089.2900	.0000	1	1

Tomamos un registro de la figura para analizar, en este caso tomamos el primer registro y vemos que el valor de la columna igv se calcula de la diferencia de su vecino máximo que es "total" con su vecino mínimo que es "subtotal", cabe resaltar que no todos los valores de igv que sean cero son valores incoherentes, debido a que algunas ventas son boletas y otras facturas, por ejemplo el valor del igv en el registro IB0000000000002255 de la figura es una boleta, porque la diferencia entre el total y el subtotal es cero. Después de analizar cómo se obtienen los valores para la columna igv procedemos hacer la prueba de Dixon, para saber que registros contienen valores atípicos en la columna igv que es la sospechosa. Para ello hemos adaptado el funcionamiento del algoritmo de Dixon en un comando SQL, *"select * from Hecho_Compras where igv<>abs(total-subtotal)"*, arrojándonos los siguientes resultados.

Figura 116: Resultados de la prueba de Dixon para la columna "igv" del data mart compras



	idFecha	documento	idProveedor	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	4	ER002-000062	81	1	788.3000	NULL	788.3000	.0000	1	1
2	79	EF0003-025707	21	3	8.4000	1.6000	.0000	.0000	1	1
3	16	EF0003-025843	21	3	5.0400	.9600	.0000	.0000	1	1
4	80	EF054-004227	50	3	6.7200	1.2800	.0000	.0000	1	1
5	44	EF001-001087	54	3	8.4000	1.6000	.0000	.0000	1	1

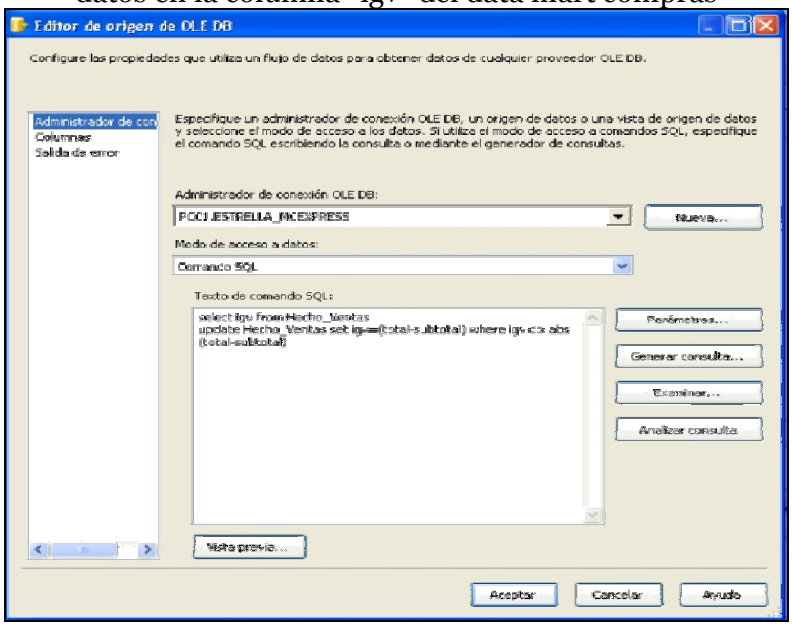
La prueba de Dixon es muy fácil de utilizar, pero el resultado depende fuertemente de escoger correctamente la ubicación de todos las columnas sospechosas, por esto y ser una prueba muy susceptible al ocultamiento o enmascaramiento, se recomienda utilizar la prueba de Dixon sólo para pequeñas muestras cuando sólo uno o dos valores son considerados como atípicos (Iglewicz y Hoaglin, 1993).

Luego de haber resuelto como vamos a identificar los valores atípicos procedemos a depurar los datos, se puede usar el algoritmo de Hot Deck Vecino más Cercano para su depuración, para ello hemos adaptado los comandos SQL de la imputación Hot Deck *"update nombre_tabla set variable='valor_numerico' where variable is null and variable > valor_número"*, con el comando SQL de la prueba de Dixon *"select * from nombre_tabla where variable<>abs(valor_máximo-valor_ mínimo)"*, nos quedaría de la siguiente forma *"update nombre_tabla set variable=(valor_máximo-valor_mínimo) where variable <> abs (valor_máximo-valor_mínimo)"*.

Para este caso específico el comando SQL que introduciremos para la depuración es la *siguiente "update Hecho_Ventas set igv=(total-subtotal) where igv<> abs (total-subtotal) and total>0"*.

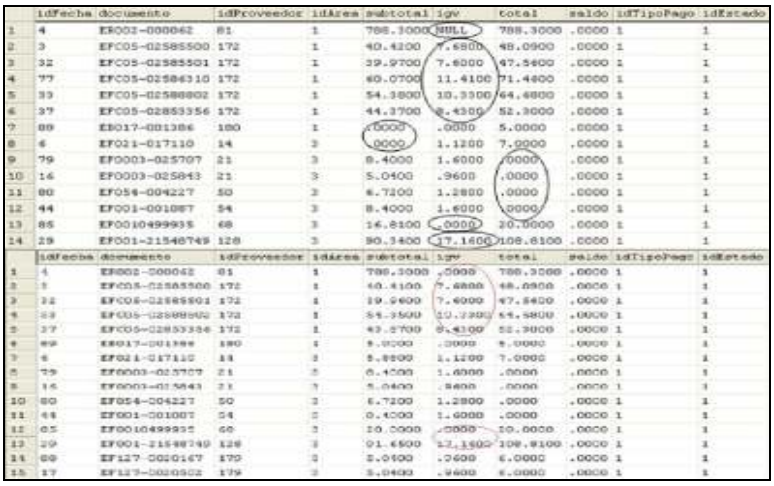
Colocamos el comando SQL en el editor de origen de OLE DB, como se muestra en la figura 117.

Figura 117: Comando SQL para la imputación de datos en la columna "igv" del data mart compras



Hacemos clic en "iniciar depuración" y nos debe de arrojar el siguiente resultado:

Figura 118: Resultado de la Imputación de la columna "igv" del data mart Compras



	idFecha	documento	idProveedor	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	4	EB002-000062	81	1	788.3000	NULL	788.3000	.0000	1	1
2	3	EFC05-02585500	172	1	40.4100	7.6800	48.0900	.0000	1	1
3	32	EFC05-02585501	172	1	39.9700	7.6000	47.5600	.0000	1	1
4	77	EFC05-02586310	172	1	60.0700	11.4100	71.4800	.0000	1	1
5	33	EFC05-02588802	172	1	54.3800	10.3300	64.6800	.0000	1	1
6	37	EFC05-02853356	172	1	44.3700	8.4300	52.3000	.0000	1	1
7	89	EB017-001386	180	1	.0000	.0000	5.0000	.0000	1	1
8	6	EF021-017110	14	3	.0000	1.1200	7.0000	.0000	1	1
9	79	EF0003-025707	21	3	8.4000	1.6000	.0000	.0000	1	1
10	16	EF0003-025843	21	3	5.0400	.9600	.0000	.0000	1	1
11	80	EF054-004227	50	3	6.7200	1.2800	.0000	.0000	1	1
12	44	EF001-001087	54	3	8.4000	1.6000	.0000	.0000	1	1
13	85	EP0010499935	68	3	16.8100	.0000	20.0000	.0000	1	1
14	29	EF001-21548749	128	3	90.3400	17.1600	108.8100	.0000	1	1

	idFecha	documento	idProveedor	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	4	EB002-000062	81	1	788.3000	.0000	788.3000	.0000	1	1
2	3	EFC05-02585500	172	1	40.4100	7.6800	48.0900	.0000	1	1
3	32	EFC05-02585501	172	1	39.9600	7.6000	47.5600	.0000	1	1
4	77	EFC05-02586310	172	1	60.0700	11.4100	71.4800	.0000	1	1
5	33	EFC05-02588802	172	1	54.3800	10.3300	64.6800	.0000	1	1
6	37	EFC05-02853356	172	1	44.3700	8.4300	52.3000	.0000	1	1
7	89	EB017-001386	180	1	5.0000	.0000	5.0000	.0000	1	1
8	6	EF021-017110	14	3	5.8800	1.1200	7.0000	.0000	1	1
9	79	EF0003-025707	21	3	6.4000	1.6000	.0000	.0000	1	1
10	16	EF0003-025843	21	3	5.0400	.9600	.0000	.0000	1	1
11	80	EF054-004227	50	3	6.7200	1.2800	.0000	.0000	1	1
12	44	EF001-001087	54	3	8.4000	1.6000	.0000	.0000	1	1
13	85	EP0010499935	68	3	16.8100	.0000	20.0000	.0000	1	1
14	29	EF001-21548749	128	3	90.3400	17.1600	108.8100	.0000	1	1

Como podemos apreciar en la figura 117 los datos de la columna "igv" que antes no tenían coherencia, luego de haber realizado la imputación nos muestra valores que van de acorde con lo que se quiere expresar.

d) Depuración de Datos Incoherentes en la Columna Total usando Dixon e Imputación Hot Deck Vecino más Cercano.- Para este caso usamos la prueba de Dixon, que consiste en saber que registros contienen valores atípicos o incoherentes o también llamados outlier's. Esto quiere decir que se va a comparar la forma en que se obtiene el valor del registro con valores atípicos con sus vecinos, analicemos la siguiente figura.

Figura 119: Datos usados para la prueba de Dixon en la columna "total" del data mart Compras

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	2	IB0000000000002237	52	3	10.0800	1.9200	12.0000	.0000	1	1
2	2	IB0000000000002238	8	2	12507.2900	2376.3900	14883.6800	.0000	1	1
3	5	IB0000000000002239	2	3	45.0000	8.5500	53.5500	.0000	1	1
4	5	IB0000000000002240	2	3	105.2800	20.0000	125.2800	.0000	1	1
5	5	IB0000000000002241	2	3	533.0000	101.2700	634.2700	.0000	1	1
6	5	IB0000000000002242	10	3	6.7200	1.2800	8.0000	.0000	1	1
7	6	IB0000000000002243	13	3	10.0800	1.9200	12.0000	.0000	1	1
8	6	IB0000000000002244	20	3	1015.0000	192.8500	1207.8500	.0000	1	1
9	6	IB0000000000002245	3	2	.0000	.0000	.0000	.0000	1	2
10	6	IB0000000000002246	22	3	9.0000	1.7100	10.7100	.0000	1	1
11	7	IB0000000000002247	33	3	40.2100	7.6400	47.8500	.0000	1	1
12	7	IB0000000000002248	8	2	2394.0000	454.8600	2848.8600	.0000	1	1
13	7	IB0000000000002249	20	2	654.2000	124.3000	778.5000	.0000	1	1
14	8	IB0000000000002250	40	2	204.0000	38.7600	242.7600	.0000	1	1
15	8	IB0000000000002251	5	3	155.4600	29.5400	185.0000	.0000	1	1
16	8	IB0000000000002252	14	3	10.0800	1.9200	12.0000	.0000	1	1
17	9	IB0000000000002253	27	2	14467.3900	2748.8000	17216.1900	.0000	1	1
18	10	IB0000000000002254	55	2	2532.2500	481.1300	3013.3800	.0000	1	1
19	11	IB0000000000002255	47	3	12.0000	.0000	12.0000	.0000	1	1
20	12	IB0000000000002256	3	1	.0000	.0000	.0000	.0000	1	2
21	13	IB0000000000002257	57	2	2596.0400	493.2500	3089.2900	.0000	1	1

Tomamos un registro de la figura para analizar, en este caso tomamos el primer registro y vemos que el valor de la columna total se calcula de la suma de su vecino máximo que es subtotal con su vecino mínimo que es igv, cabe resaltar que no todos los valores de total que sean cero son valores incoherentes, debido a que algunas compras han sido anuladas, después de analizar cómo se obtienen los valores para la columna total

procedemos hacer la prueba de Dixon, para saber que registros contienen valores atípicos en la columna total que es la sospechosa. Para ello hemos adaptado el funcionamiento del algoritmo de Dixon en un comando *SQL, "select * from Hecho_Compras where total<>abs(subtotal+igv)"*, arrojándonos los siguientes resultados.

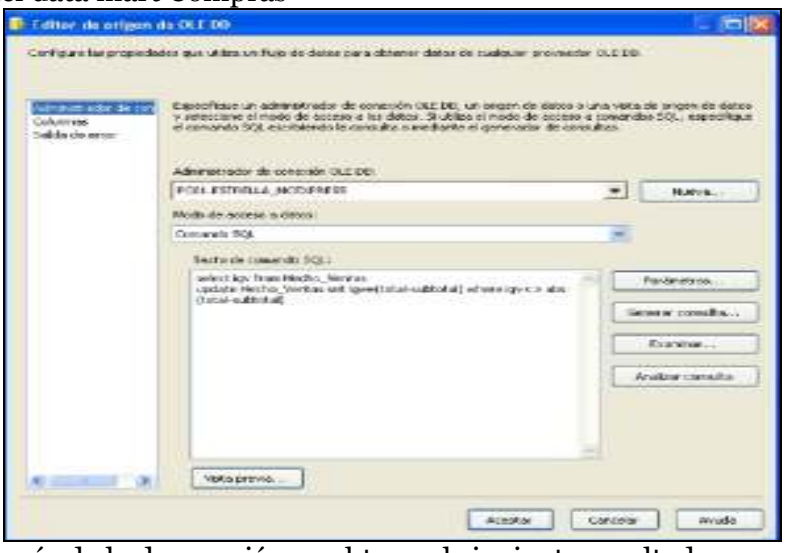
Figura 120: Resultado de la prueba de Dixon en la columna "total" del data mart Compras

	idFecha	documento	idProveedor	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	79	EF0003-025707	21	3	8.4000	1.6000	.0000	.0000	1	1
2	16	EF0003-025843	21	3	5.0400	.9600	.0000	.0000	1	1
3	80	EF054-004217	50	3	6.7200	1.2800	.0000	.0000	1	1
4	44	EF001-001087	54	3	8.4000	1.6000	.0000	.0000	1	1

Luego de haber resuelto como vamos a identificar los valores atípicos procedemos a depurar los datos, se puede usar el algoritmo de Hot Deck Vecino más Cercano para su depuración, para ello hemos adaptado los comandos SQL de la imputación Hot Deck *""update nombre_tabla set variable='valor_numerico' where variable is null and variable > valor_número"*, con el comando SQL de la prueba de Dixon *"select * from nombre_tabla where variable<>abs(valor_máximo-valor_ mínimo)"*, nos quedaría de la siguiente forma *"update nombre_tabla set variable=(valor_máximo+valor_mínimo) where variable <> abs (valor_máximo-valor_mínimo)"*. Para este caso específico el comando SQL que introduciremos para la depuración es el siguiente *"update Hecho_Ventas set total= (subtotal+igv) where total<> abs (subtotal +igv)"*.

Colocamos el comando SQL en el editor de origen de OLE DB, como se muestra en la figura 121.

Figura 121: Comando SQL para la imputación de la columna "total" del data mart Compras



Después de la depuración se obtuvo el siguiente resultado.

Figura 122: Resultado de la Imputación en la columna "total" del data

mart Compras

	idFecha	documento	idProveedor	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	79	EF0003-025707	21	3	8.4000	1.6000	.0000	.0000	1	1
2	16	EF0003-025843	21	3	5.0400	.9600	.0000	.0000	1	1
3	80	EF059-004227	50	3	6.7200	1.2800	.0000	.0000	1	1
4	44	EF001-001087	54	3	8.4000	1.6000	.0000	.0000	1	1

	idFecha	documento	idProveedor	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	79	EF0003-025707	21	3	8.4000	1.6000	10.0000	.0000	1	1
2	16	EF0003-025843	21	3	5.0400	.9600	6.0000	.0000	1	1
3	80	EF059-004227	50	3	6.7200	1.2800	8.0000	.0000	1	1
4	44	EF001-001087	54	3	8.4000	1.6000	10.0000	.0000	1	1

Resultado:

Pasos a Seguir en la Metodología de Jimenez y Amón: 6
Número de Datos de Faltantes o Nulos: 0
Número de Datos Atípicos o incoherentes: 0

Esquema Final

Figura 123: Esquema final para el ETL del data mart Compras

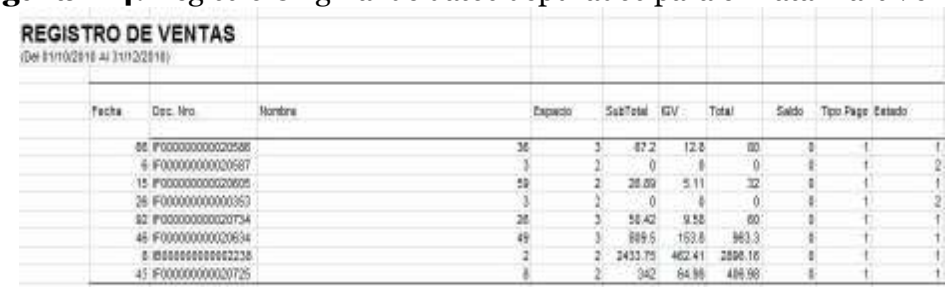
V. DISCUSIÓN

Para poder comparar los datos originales con los datos depurados con la ayuda de la guía metodológica de Amón y Jiménez nos podemos dar cuenta que, si la aplicamos por cada columna o variable de la base de datos, nos ayudan a depurar la información, le da coherencia a datos sucios, pero en ocasiones alteran el contenido de los mismos variando sus datos, como lo podemos apreciar en los siguientes casos escogidos al azar:

- **Caso 1**

  A continuación compararemos los datos iníciales con los datos obtenidos luego del ETL y los datos obtenidos luego de la depuración de datos.

Figura 124: Registro Original de datos depurados para el Data mart Ventas



REGISTRO DE VENTAS

(Del 01/10/2010 al 31/12/2010)

	Fecha	Doc. Nro	Nombre	Espacio	SubTotal	IGV	Total	Saldo	Tipo Pago	Estado
66		F000000000020586		36	3	67.2	12.8	80	0	1	1
6		F000000000020587		1	2	0	0	0	0	1	2
15		F000000000020605		59	2	26.89	5.11	32	0	1	1
26		F000000000000353		1	2	0	0	0	0	1	2
92		F000000000020734		26	3	50.42	9.58	60	0	1	1
46		F000000000020634		49	3	889.5	153.8	963.3	0	1	1
0		00000000000002238		2	2	2433.75	462.41	2896.16	0	1	1
45		F000000000020725		8	2	342	64.98	406.98	0	1	1

Fuente: MC EXPRESS 2010

En la figura 124 tenemos los datos reales alcanzados del modulo de ventas de la empresa MC EXPRESS.

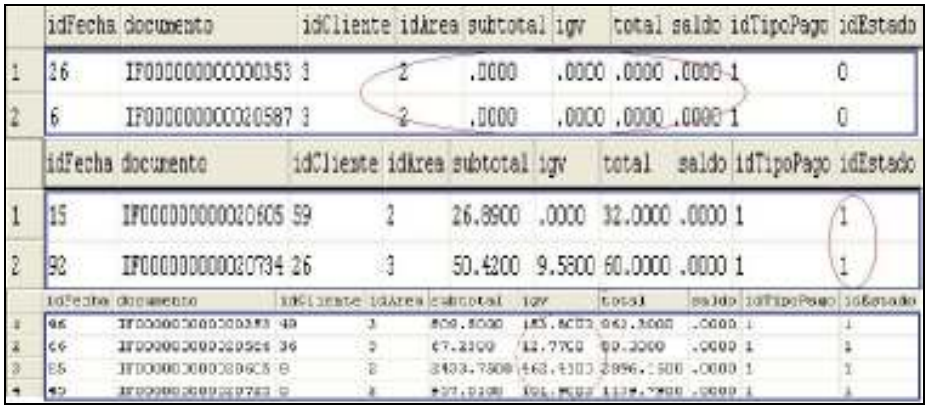
Figura 125: Resultados del ETL según Metodología de Amón y Jiménez en data mart Ventas

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF000000000000353	3	NULL	NULL	NULL	NULL	NULL	1	0
2	6	IF000000000020587	3	NULL	NULL	NULL	NULL	NULL	1	0
3	15	IF000000000020605	59	3	26.8900	.0000	32.0000	.0000	1	NULL
4	92	IF000000000020734	26	3	50.4200	9.5800	60.0000	.0000	1	NULL

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	66	IF000000000020586	36	3	67.2300	.0000	80.0000	.0000	1	1
2	85	IF000000000020605	8	2	2433.7500	.0000	2896.1400	.0000	1	1
3	33	IF000000000020657	47	3	12.0000	.0000	12.0000	.0000	1	1
4	45	IF000000000020725	8	2	957.8100	.0000	1139.7900	.0000	1	1

Comparando los resultados del ETL con los datos reales podemos apreciar que los registros IF000000000000353 y IF000000000020587, la mayor parte de sus atributos contienen datos nulos, los registros IF000000000020605 y IF000000000020734 presentan problemas de datos incoherentes y valores nulos, los registros IF000000000020586, IF000000000020605 y IF000000000020725 presentan datos incoherentes.

Figura 126: Datos obtenidos después de la depuración para el Data mart Ventas



	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	16	IF0000000000000353	3	2	.0000	.0000	.0000	.0000	1	0
2	6	IF0000000000020587	3	2	.0000	.0000	.0000	.0000	1	0

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	15	IF0000000000020605	59	2	26.8900	.0000	32.0000	.0000	1	1
2	92	IF0000000000020734	26	3	50.4200	9.5800	60.0000	.0000	1	1

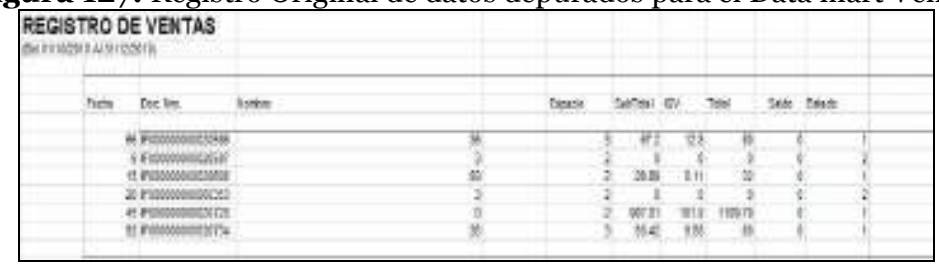
	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	96	IF0000000000000353	49	3	809.8000	185.8000	962.3000	.0000	1	1
2	66	IF0000000000020586	36	3	67.2100	12.7900	80.2000	.0000	1	1
3	85	IF0000000000020605	0	3	2433.7800	462.4300	2896.2100	.0000	1	1
4	42	IF0000000000020725	0	3	937.0100	191.8900	1128.9000	.0000	1	1

En la figura 126 podemos observar los registros luego de ser depurados con la ayuda de la guía de Amón y Jiménez, el registro que nos llama la atención es el IF0000000000000353 , como podemos apreciar en el archivo original los datos del registros "IF0000000000000353" pertenecen a un registro que ha sido anulado, porque sus columnas subtotal, igv, total y saldo contienen valor cero, pero luego de haber depurado los registros observamos que en nuestro data mart existen dos registros con el mismo código uno con valores de cero para sus columnas subtotal, igv total y saldo y otro con valores distintos a los del registro original, esto nos da a entender que por una parte nos ayudó a darle sentido a las columnas con datos "NULL" acertando los valores que le corresponden, por otro lado no nos ayudó a depurar registros duplicados.
En el caso del registro "IF0000000000020605" luego de haber sido imputados para sus columnas idcliente, subtotal, igv, total, saldo e idEstado, los cuales luego del ETL tenían un valor nulo, fueron imputados correctamente acertando con los valores originales.
En los registros IF0000000000020605 y IF0000000000020734 presentan problemas de datos incoherentes y valores nulos, los registros IF0000000000020586, IF0000000000020605 y IF0000000000020725, luego de imputar la columna "igv" que es la que presentaba problemas de coherencia, nos dio como resultado que si bien es cierto le da coherencia a dichos valores, no son exactamente los valores reales.

Amón y jiménez (2010) nos dice que las causas para este caso pueden ser muchas, estos errores pueden ser causados por restricciones de formato, de longitud y/o en el conjunto de caracteres permitidos, errores humanos al capturar los datos, errores que surgen integrando bases de datos diferentes o haciendo migración entre sistemas, modelos de datos mal diseñados, entre otras causas. Porque si apreciamos bien ambos registros las fechas, los clientes, el área, el subtotal, el igv, el total y el resto de columnas son diferentes, solo el código del documento es igual, por lo tanto es posible que este error se halla originado durante el registro de los datos a través de los sistemas de información que usan y no durante el ETL y depuración de los datos en el data mart Ventas.
Por otra parte la guía metodológica nos ayuda a depurar datos duplicados más no registros completos que se han duplicado como observamos con el registro IF0000000000000353.

- **Caso 2**
  Cuando se procedió a ejecutar todos los comandos de depuración de datos, usados en cada una de las columnas de la base de datos, unidos en un solo comando, los resultados fueron diferentes a los que se obtuvieron cuando se ejecutaron los comandos siguiendo las indicaciones de la metodología, los resultados fueron los siguientes:

Figura 127: Registro Original de datos depurados para el Data mart Ventas



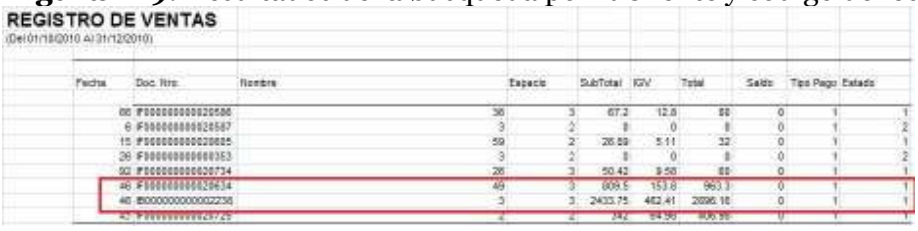
Fuente: MC EXPRESS 2010.

Figura 128: Resultados de la depuración de datos en el Data mart Ventas resumiendo los pasos de la metodología de Jimenez y Amón

	idFecha	documento	idCliente	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	26	IF0000000000000353	3	2	.0000	.0000	.0000	.0000	1	2
2	66	IF0000000000020586	36	3	67.2300	12.7700	80.0000	.0000	1	1
3	6	IF0000000000020587	3	2	.0000	.0000	.0000	.0000	1	2
4	15	IF0000000000020605	59	2	26.8900	.0000	32.0000	.0000	1	1
5	33	IF0000000000020657	47	3	12.0000	.0000	12.0000	.0000	1	1
6	45	IF0000000000020725	8	2	957.8100	181.9800	1139.7900	.0000	1	1
7	92	IF0000000000020734	26	3	50.4200	9.5800	60.0000	.0000	1	1

Como podemos apreciar en la figura donde se muestran los resultados de la depuración de datos en el Data mart Ventas resumiendo los pasos de la metodología de Jiménez y Amón, los datos comparados con los originales son correctos, la única diferencia con respecto a la depuración de los datos paso por paso es que los registros "IF0000000000000353" y "IF0000000000020605" no aparecen como duplicados. Haciendo una búsqueda con el código de cliente y el código de la fecha, encontramos que dichos registros que aparecen como duplicados en la depuración anterior realmente les puede pertenecer a los registros con código "IF0000000000020634" y "IB0000000000002238" respectivamente como lo podemos apreciar en las siguientes figuras, no se asegura porque un cliente puede haber hecho varias transacciones el mismo día.

Figura 129: Resultados de la búsqueda por idCliente y código de fecha



Fuente: MC EXPRESS 2010.

La respuesta para esta variabilidad en los datos puede ser que hubo un error durante el proceso ETL para este data mart, tal como nos dice

Chandola et. al. (2007). Aunque no necesariamente son errores, pueden ser generados por un mecanismo diferente de los datos normales como problemas en los sensores, distorsiones en el proceso, mala calibración de instrumentos y/o errores humanos. Esto quiere decir que el error pudo venir de la herramienta de ETL y no de errores humanos como lo habíamos planteado en un inicio.
Pero queremos destacar, que cuando se realizó la depuración resumiendo los pasos de la metodología en un solo comando, los resultados fueron un poco distintos, se mejoró en la depuración de registros duplicados, una explicación a este acontecimiento, es que se mejora en la depuración de duplicados porque se depura todo el registro al mismo tiempo y no columna por columna.

- **Caso 3**
  En el data mart Compras los resultados fueron parecidos a los que nos arrojó el proceso ETL del data mart Ventas

Figura 130: Datos reales obtenidos de los registros de Compras

REGISTRO DE COMPRAS

(Del 01/10/2010 Al 31/12/2010)

idFecha	Nro Doc.	idProveedor	idArea	SubTotal	IGV	Total	Saldo
81	EF127-0020167	179	3	5.04	0.96	6	0
12	EF127-0020502	179	3	5.04	0.96	6	0

Fuente: MC EXPRESS 2010.

Figura 131: Resultados de la depuración de idFecha en data mart Compras.

	idFecha	documento	idProveedor	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
1	88	EF127-0020167	179	3	5.0400	.9600	6.0000	.0000	1	1
2	17	EF127-0020502	179	3	5.0400	.9600	6.0000	.0000	1	1

Como podemos apreciar, luego de haber depurado la columna idFecha en el data mart Compras, los resultados en comparación con los registros reales de la empresa no son exactos, es decir solo se acercan a los valores reales, por ejemplo en el registro "EF127-0020167" y "EF127-0020502" el código de la columna idFecha luego de haber realizado la depuración es de 88 y 17 respectivamente, pero en el registro real es de 81 y 12, esto nos quiere decir que si bien es cierto la técnicas de depuración que hemos usado nos da coherencia a los datos, no siempre van a ser iguales a los datos reales, tal y como nos afirma Amón y Jiménez (2010) al decirnos que los procedimientos de imputación mejoran la calidad de los datos, debido a que si se tiene muchos datos sucios estos procedimientos le dan coherencia y una cercanía a sus verdaderos valores.

- **Caso 4**
  Si bien es cierto las imputaciones le dan coherencia a los datos sucios reemplazándolos por valores que representan información válida para las organizaciones, estas muchas veces pueden variar y generar inconvenientes, como nos dice Goicoechea (2002) sobre que estos métodos tienen algunas desventajas, ya que distorsionan la relación con el resto de las variables, carecen de un mecanismo de probabilidad y requieren tomar decisiones subjetivas que afectan a la calidad de los

datos, lo que imposibilita calcular su confianza. Como podemos ver en las siguientes figuras.

Figura 132: Registros reales de Compras

REGISTRO DE COMPRAS

(Del 01/10/2010 Al 31/12/2010)

idFecha	Nro Doc.	idProve	idArea	SubTotal	IGV	Total	Saldo
89	EB017-001386	180	1	5.00	0.00	5.00	0
6	EF021-017110	14	3	5.88	1.12	7.00	0
85	EF0010499935	68	3	16.81	3.19	20.00	0

Fuente: MC EXPRESS.

Figura 133: Resultado de la imputación en la columna Subtotal del Data mart Compras

idFecha	documento	idProveedor	idArea	subtotal	igv	total	saldo	idTipoPago	idEstado
89	EB017-001386	180	1	5.0000	.0000	5.0000	.0000	1	1
6	EF021-017110	14	3	5.8800	1.1200	7.0000	.0000	1	1
85	EF0010499935	68	3	20.0000	.0000	20.0000	.0000	1	1

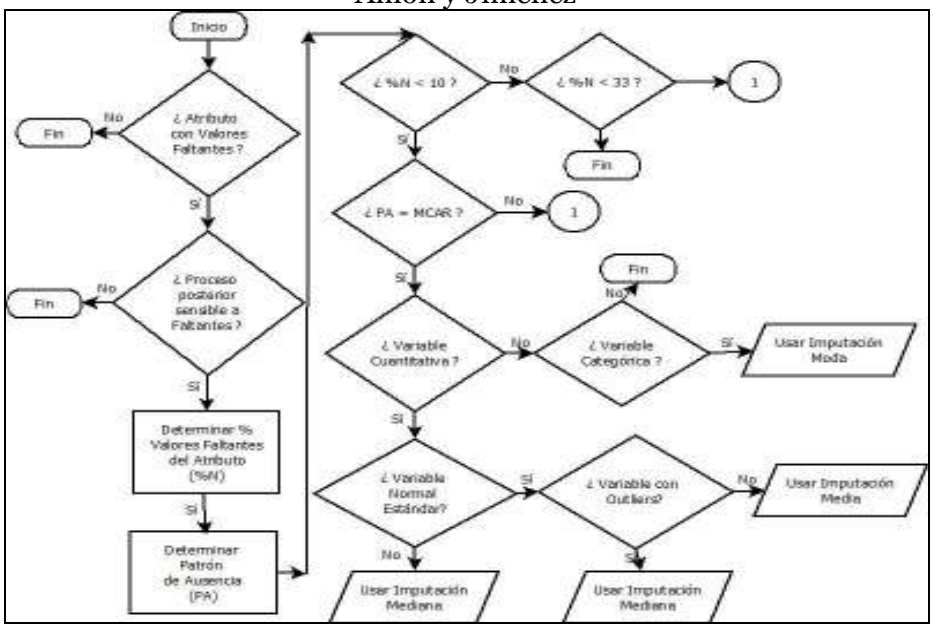
Como se puede apreciar en el registro “EF0010499935” el registro real pertence a una compra con factura porque su total es 20 el subtotal es 16.81 y el igv es de 3.19, luego de la imputación el total paso a ser 20, el subtotal 20 y el igv quedo en cero, esto nos da a entender que la variabilidad de los valores luego de haber usado una técnica de imputación, puede generar distorsión en la información final que se recoja de estos almacenes de datos, lo que va a generar que se tome una decisión frente a datos que no muestren la realidad de una organización.

VI. PROPUESTA

a) Mejoras en la Metodología de Depuración de Datos de Amón y Jiménez para su uso durante el proceso ETL.

i. Para los datos de tipo Cualitativos.- Los autores proponen un algoritmo que ayuda a detectar y depurar datos erróneos o nulos dentro de las bases de datos, como se muestra en la figura 134, dicho algoritmo está hecho para depurar bases de datos que en su mayoría son estadísticos o de sistemas expertos, luego de haber realizado este estudio se comprobó que se deben de adaptar y mejorar algunos pasos, para que se pueda usar durante el proceso ETL en la implementación de los data mart en bases de datos comunes.

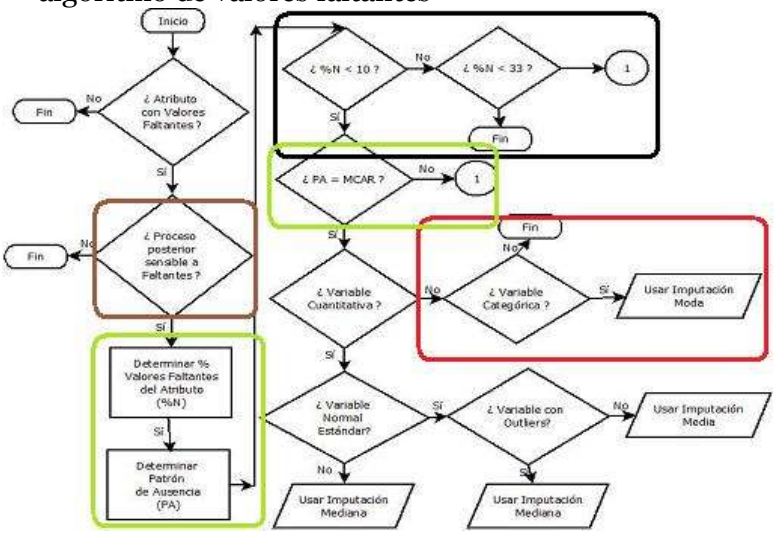
Figura 134: Diagrama para Valores Faltantes según guía metodológica de Amón y Jiménez



Fuente: Amón y Jiménez (2010)

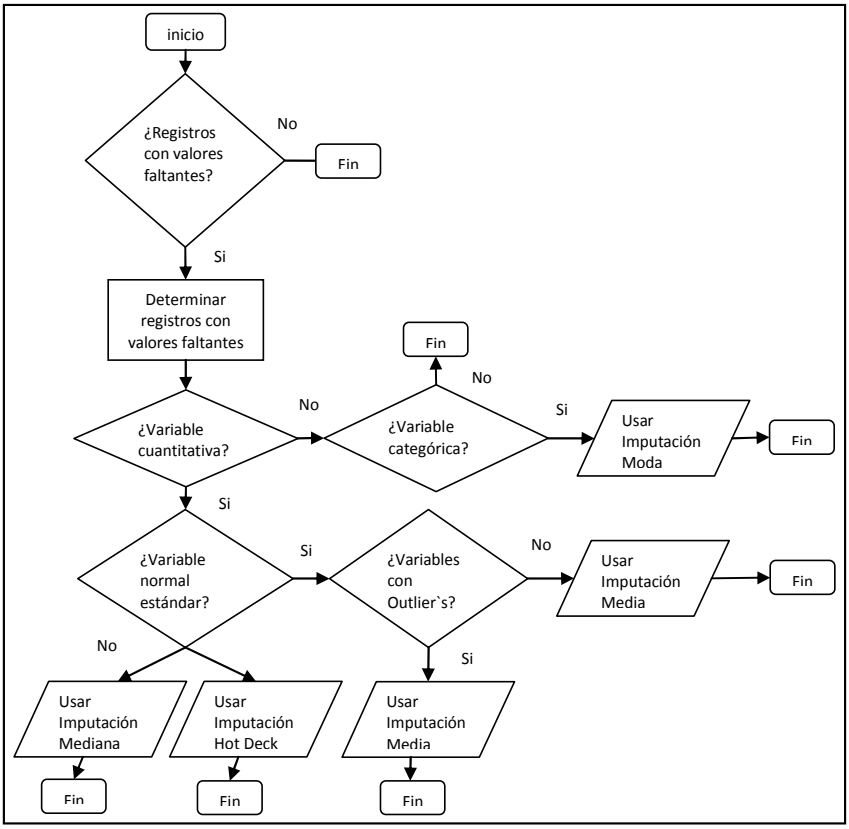
Lo que se debe de adaptar en el algoritmo se ha identificado en la figura 135.

Figura 135: Identificación de las adaptaciones a realizar en el algoritmo de valores faltantes



A partir del estudio del algoritmo se ha podido identificar los pasos que deben de adaptarse para que la metodología propuesta también se pueda usar durante el ETL en bases de datos comunes, reduciendo el tiempo en la depuración, sin dejar de lado el aseguramiento de la calidad de los registros que se encuentran en estos almacenes, quedando como se muestra en la figura 136.

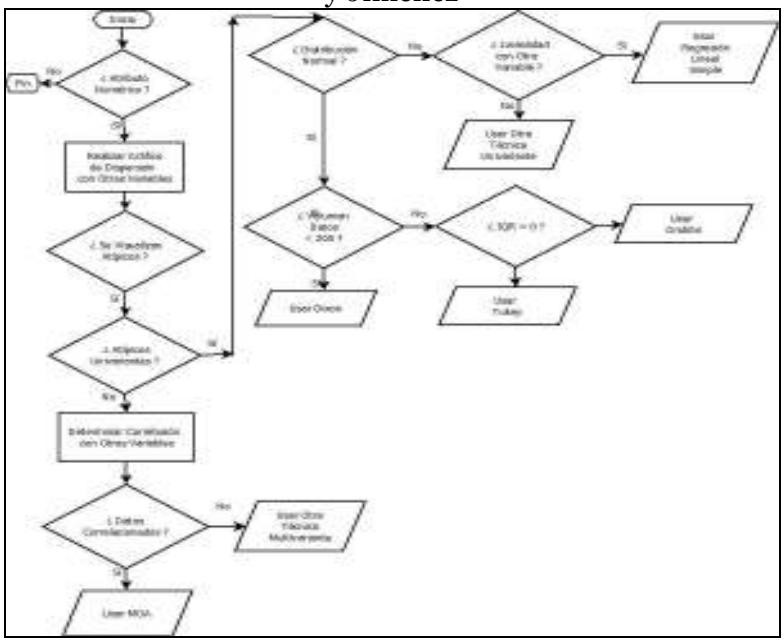
Figura 136: Algoritmo adaptado para la depuración de datos tipo cadena en bases de datos comunes.



Como podemos apreciar en la figura 136, se ha reducido los pasos en donde se pregunta por el patrón de ausencia y el porcentaje de valores faltantes en los atributos, debido a que son pasos en donde se hallan patrones estadísticos que solo sirven para estudiar el comportamiento de los datos estadísticamente, los cuales en una base de datos no estadística no servirá de mucho.
Luego de determinar los registros con valores faltantes, procedemos a preguntar si es una variable cuantitativa o no, si lo es verificamos si es una variable normal estándar, en el caso que lo sea preguntamos si es una variable con valores incoherentes para poder usar una imputación mediana o caso contrario una imputación media.
Para los valores no cuantitativos, preguntamos si es una variable con relevancia, en el caso que lo sea podemos usar una imputación moda si no es el fin del algoritmo.

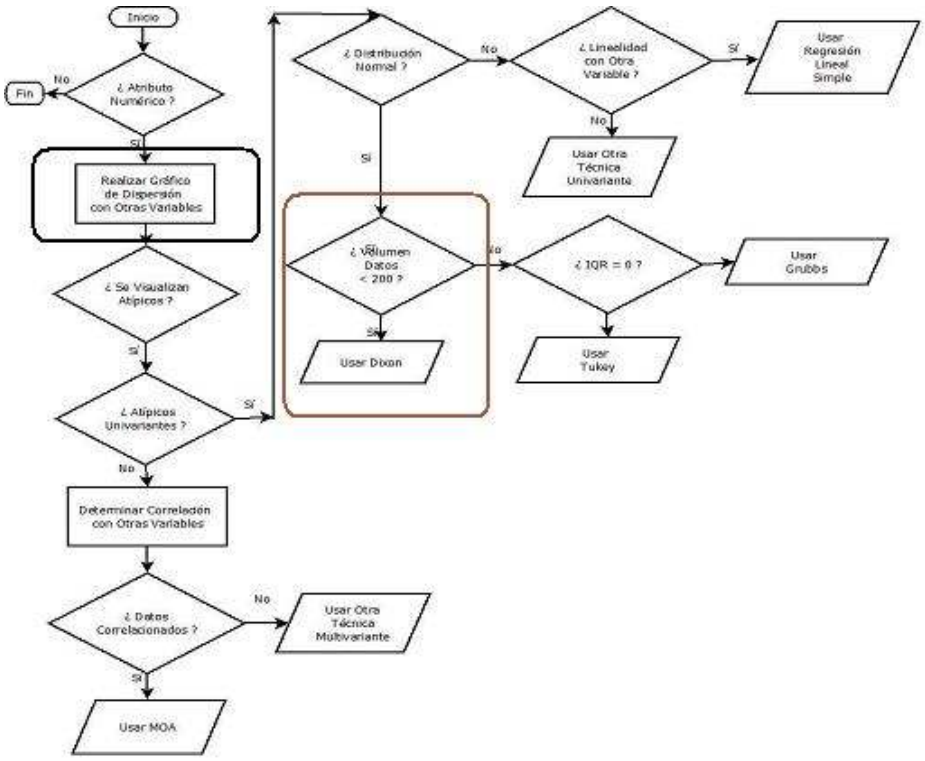
ii. **Para los tipo de datos Cuantitativos.-** Al igual como se hizo con los datos cualitativos, se procedió a adaptar algunos pasos para que el algoritmo de depuración de datos que nos plantea Amón y Jiménez pueda también ser utilizado con mucha mayor facilidad en bases de datos comunes. A continuación mostramos en la figura 137 el algoritmo original planteado en la guía metodológica.

Figura 137: Diagrama para Valores Atípicos según guía metodológica de Amón y Jiménez



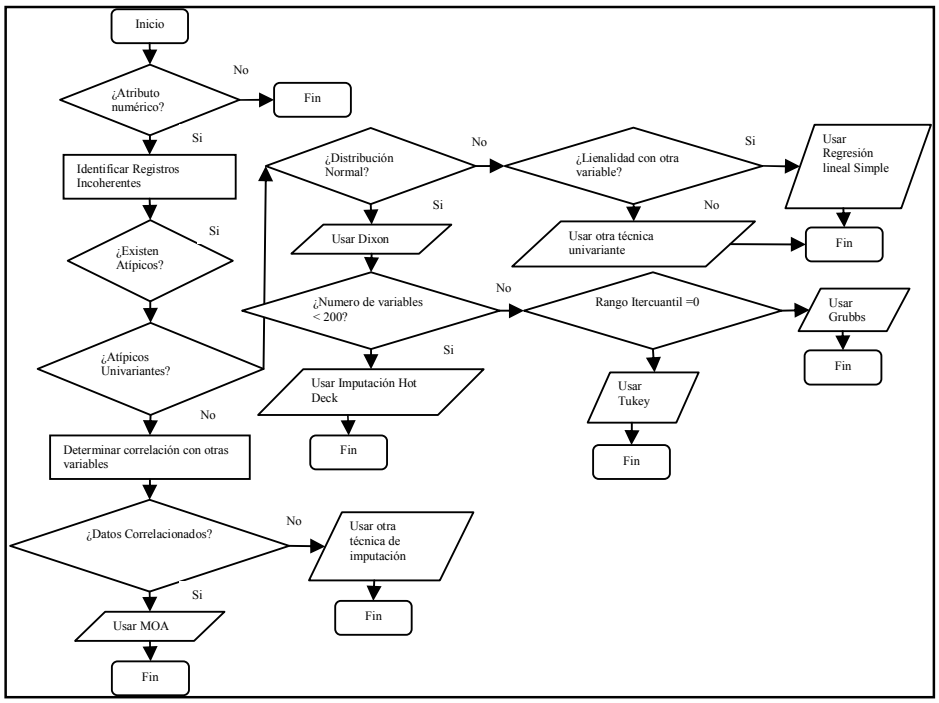
En la figura 138 mostramos las adaptaciones planteadas para mejorar el tiempo y la efectividad de la metodología en bases de datos comunes.

Figura 138: Identificación de las adaptaciones a realizar en el algoritmo de valores incoherentes



Como podemos apreciar en la figura 139, se ha reducido los pasos en donde se propone realizar gráfico de dispersión debido a que es utilizado en su mayoría en bases de datos estadísticas, lo cual en una base de datos común no será de mucha utilidad, Luego en otro punto recomendamos que si los datos tienen una distribución normal debemos usar la prueba de Dixon para determinar cuáles son los registros que presentan datos incoherentes y sea más fácil usar las técnicas de imputación, el algoritmo quedó tal como se muestra en la figura139.

Figura 139: Algoritmo adaptado para la depuración de datos tipo cuantitativo en bases de datos comunes.



Recomendaciones

a) Propuesta en el recojo de la información:

Situación Actual: Durante la investigación se pudo constatar que al momento de digitar la información, los sistemas transaccionales de la empresa no cuentan con mecanismos de control eficientes, por ejemplo al registrar una venta se carga automáticamente el código del cliente y su nombre, pero deja la posibilidad de que el usuario del sistema de ventas modifique el nombre del cliente, lo que puede ocasionar que un cliente con un mismo código pero con distintos nombres tengan varias compras.

Propuesta: Mejorar los mecanismos de control para el registro de la data a través de los sistemas transaccionales, que restrinjan acciones que generen duplicidad de información e incoherencia en los datos.

b) Propuesta en la Administración de Base de Datos

Situación Actual.- La administración y mantenimiento de la base de datos lo realiza un ingeniero de sistemas que no es permanente que solo asiste si hay algún problema en el sistema, en el caso no pueda el manda a un representante a solucionarlo. No existe una planificación en el mantenimiento de la base de datos y software lo que posibilita aun más que existan datos sucios.

Propuesta de Mejora.- Contar plan de mantenimiento mensual de la base de datos en donde y software que contemple la identificación de datos sospechosos y la corrección de los mismos con información real.

VII. CONCLUSIONES

Se logró desarrollar el proceso ETL aplicando la metodología de depuración de datos para asegurar la calidad de los datos en la construcción de data mart para la empresa MC EXPRESS de la ciudad de Chiclayo.

Se analizó el proceso ETL, identificando que la complejidad aumenta al trabajar con datos de archivos Excel y no de una base de datos propiamente dicha.

Se diseñó el proceso ETL para los data mart que se han construido, usando herramientas de Visual Studio BI Development.

Se desarrolló el proceso ETL para data mart, identificando las metodologías y plataforma usadas para dicho proceso, en donde resalta la facilidad en el uso de las herramientas usadas.

Se verificó los resultados del proceso ETL, la cual muestra una importante mejora en durante la aplicación de la metodología estudiada en comparación con la que propone Visual Studio.

VIII. REFERENCIAS BIBLIOGRÁFICAS

Amón I, Jiménez C. 2009. Towards a methodology for selection of data cleansing techniques, Grupo de Investigación y Desarrollo en Inteligencia Artificial Universidad Nacional de Colombia.

Amón I, Jiménez C. 2010. Guía Metodológica para la Selección de técnicas de Depuración de Datos, Universidad Nacional De Colombia, Medellín.

Babad Y.M., y J.A. Hoffer. 1984. *Even no data has a value, Communications of the ACM*, 27, 1984, 748757.

Bermudez, J. (sin Fecha). *Metodología de Implementación de Business Intelligence IBSS*. IBSS Consultiing.

Breuning,H. P. Kriegel, R. T. Ng, y J. Sander. 2000. *Lof: Identifying densitybased local outlier*. En: Proceedings SIGMOD, Dallas, Texas.

Calle Guglieri, José. 1997. *Reingeniería y Seguridad en el Ciberespacio*. Madrid, España: Ediciones Díaz de Santos S.A.

Chandola, V., Arindam, B., y Vipin K. 2007. *Outlier detection: A survey Technical Report Department of Computer Science and Engineering*. Minnesota, USA: University of Minnesota.

Codd, EF., Date, C. 1993. *Providing OLAP (On-line Analytical Processing) to User-Analysts: An IT Mandate*. Inc 1993

Curto Díaz, Josep y Conesa, Jordi. 2010. *Introducción al Business Intelligence*. Barcelona: Editorial UOC.

Date, Christopher. 2001. *Introducción a los Sistemas de Bases de Datos*. México: Pearson Educación.

De la Fuente, F. 2004. *Los Sistemas de Información en la Sociedad del Conocimiento*. Madrid: Esic Editorial.

Elmagarmid, A., Ipeirotis, P., y Verykios, V. 2007. Duplicate Record Detection: A Survey. IEEE Transactions on knowledge and data engineering. 19 (1). Enero, 2007.

English, L. 1999. *Improving Data Warehouse and Business Information Quality*. Jhon Wiley & Sons, Inc.

Giner de la Fuente, Fernando. 2004. *Los Sistemas de Información en la Sociedad del Conocimiento*. Madrid: ESIC Editorial.

Hancong L., Sirish S., y Wei, J. 2004. Online outlier detection and data cleaning. Computers & chemical engineering, 2004, 28 (9), 16351647.

Horváth & Partners. 2007. *Dominar el Cuadro de Mando Integral*. Barcelona: Ediciones Deusto.

Jaro, M. A. 1976. *Unimatch: A Record Linkage System: User's Manual, technical report*. Washington D.C.: US Bureau of the Census.

Jiménez Márquez, C., y J. Thibault. 2002. *Statistical data validation methods for large cheese plant database*. J.Dairy Sci.,85(9), 20812097.

Kedad, Z. and Métais E. 2002. Ontology-Based Data Cleaning. Lecture Notes in Computer Science, 2553. pp. 137 – 149.

Kendall Kenneth, [eta al]. 2005. *Análisis y Diseño de Sistemas*. México: Pearson Educación.

Kim, W., Choi, B.J., Hong, E.K., Kim, S.K., y Lee, D. 2003. A Taxonomy of Dirty Data. Data Mining and Knowledge Discovery, 7, 2003. 8199

Kimball, Ralph. 2008. *The Data Warehouse Lifecycle Toolkit*, Second Edition. New York: Wiley.

Khoshgoftaar, T. M., [et al]. 2005. *Detecting noisy instantes with the rulebased classification model*. Intel, Data Anal.

Kroenke, David M. 2003. *Procesamiento de Bases de Datos: Fundamentos, Diseño e Implementación*. México: Pearson Educación.

Little, J.A. y Donald Rubin. 1989. *Analysis of social science data with missing value*. Cambridge: Sociological Methods Research.

Lozano Pérez, María Dolores. 2006. *Ingeniería del Software y Bases de Datos: Tendencias Actuales*. Castilla-La Mancha: Universidad de Castilla-La Mancha.

McLeod, Raymond. 2000. *Sistemas de Información Gerencial*. México: Pearson Educación.

Méndez del Rio, Luis. 2000. *Más Allá del Business Intelligence*. Barcelona: Ediciones Gestión 2000.

Moliner López Francisco Javier. 2005. *Grupos A Y B Temario Bloque Específico*. España: Editorial Mad S.L.

Müller, H., y Freytag, J.C. 2003. Problems, Methods, and Challenges in Comprehensive Data Cleansing. Technical Report HUBIB164, Humboldt University, Berlin.

Oliveira, P., Rodrigues, F., Henriques, P., y Galhardas, H. 2005. A Taxonomy of Data Quality Problems. En: Second International Workshop on Data and Information Quality IQIS 2005 (Porto, Portugal, Junio 1317, 2005).

Parra Iglesias, Enrique. 1998. *Tecnologías de la Información en el Control de Gestión*. Madrid: Ediciones Díaz de Santos S.A.

Pérez L. César y Daniel Santín. 2008. *Minería de Datos Técnicas y Herramientas*. Madrid: Ediciones Paraninfo.

Rahm, E., y Do, H. H. 2000. Data Cleaning: Problems and Current Approaches. IEEE Bulletin of the Technical Committee on Data Engineering, 24 (4).

Rittman, Mark. 2006. Data Profiling and Automated Cleansing Using Oracle Warehouse Builder 10g Release 2, Septiembre, 2006.

Ristad, E. y P. Yianilos. 1998. *Learning string edit distance*. IEEE Trans. Pattern Analysis and Machine Intelligence, 20(5), 522532.

Rob,Peter y Coronel, Carlos. 2006. *Sistemas de Bases de Datos: Diseño, Implementación y Administración*. México: Thomson Editores S.A.

Rosenthal, A., Wood, D., y Hughes, E. 2001. Methodology for Intelligence Database Data Quality. Julio, 2001.

Russell. 1918: Index, U.S. Patent 1,261,167, http:// patft.uspto.gov/netahtml/srchnum.htm, Apr. 1918.

Smith, T. y Waterman, M. 1981. *Identification of common molecular subsequences*. J. Molecular Biology.

Stair, Ralph M. y. Reynolds, George W. 2000. *Principios de Sistemas de Información: Enfoque Administrativo*. México: Cengage Learning Editores.

Taft. 1970: Name Search Techniques. Technical Report Special Report No. 1, Nueva York State Identification and Intelligence. System, Albany, N.Y., Febrero, 1970.

Tierstein, Leslie. A Methodology for Data Cleansing and Conversion, White paper W R Systems, Ltd.

Tuya, Javier, [eta al]. 2007. *Técnicas Cuantitativas para la Gestión en la Ingeniería del Software*. La Coruña, España: Netbiblo.

Trujillo Mondejar, Juan Carlos, [eta al]. 2011. *Diseño y Explotación de Almacenes de Datos. Conceptos Básicos de Modelado*. Alicante, España: Editorial Club Universitario.

Viera Braga, Luis P., [et al]. 2009. *Introducción a la Minería de Datos*. Río de Janeiro: E-papers Servicios Editoriales Ltda.

Waterman, M. y T. Beyer. 1976. *Some biological sequence metrics. Advances in Math.*, 20(4), 367387.

Wixom, B.H. y H.J Watson. 2001. *An empirical investigation of the factors affecting data warehouse success*.MIS Quarieriy.

Linkografia

Diseñador SSIS de Visual Studio 2005 [En Línea]. http://msdn.microsoft.com/es-es/library/ms137973.aspx [Consulta: Noviembre 2010].

Gartner. Dirty Data is a Business Problem, Not an IT Problem. [En línea]. 2007. http://www.gartner.com/it/page.jsp?id=501733 [Consulta: Junio 12 de 2010]

IDC y SAS. La falta de calidad de los datos provoca pérdidas de productividad del 30 por ciento [En línea]. 2009. http://www.computing.es/Noticias/200906260002/La-falta-de-calidad-de-los-datos-provoca-perdidas-de-productividad-del-30-por-ciento.aspx [Consulta: Junio 11 de 2010].

Revista Marketing y Ventas, las pérdidas por una base de datos defectuosa pueden ascender al 20%. [En línea]. 2006. http://www.marketingdirecto.com/actualidad/bases-de-datos-y-crm/las-perdidas-por-una-base-de-datos-defectuosa-pueden-ascender-al-20/ [Consultado: Junio 11 de 2010]