

Universidad Central “Marta Abreu” de Las Villas

Facultad de Matemática, Física y Computación

Sistemas de Información

Tarea Extraclase I



AUTORES:

JORGE ABREU PERAZA

ALFRED AGUILAR RAMOS

AICENIS M. CASTRO OVES-GARCÍA

QUINTO DE CIENCIA DE LA COMPUTACIÓN

CURSO 2020-2021

Índice general

Introducción	3
Desarrollo	4
1. Trabajo con el almacenamiento de los datos	5
1. Descripción del problema a modelar	5
2. Tecnologías usadas para el trabajo con los datos	5
3. Diseño lógico del almacén de datos	6
4. Almacén de datos en la base de datos	7
2. Procesamiento de los datos	9
1. Operaciones de extracción, transformación y carga(ETL)	9
1.1. Transformaciones	9
1.1.1. Transformación para poblar la dimensión client_dim	10
1.1.2. Transformación para poblar la dimensión age_dim .	11
1.1.3. Transformación para poblar la dimensión time_dim .	12
1.1.4. Transformación para poblar la dimensión transport_dim	13
1.1.5. Transformación para poblar la dimensión destination_dim	14
1.1.6. Transformación para poblar la tabla de hechos arrival_facts	15
1.1.7. Transformación para poblar la tabla de hechos departure_facts	17
1.2. Trabajos	19
2. Cubo de datos OLAP	20
2.1. Consulta general	21
2.2. Slice & dice	22
2.3. Drill-down	22
2.4. Roll-up	23
2.5. Pivot	23
3. Reportes	23
Conclusiones	27

INTRODUCCIÓN

El Ministerio del Turismo (MINTUR) es el organismo estatal rector del Sistema de Turismo en Cuba, para desarrollar de forma eficiente sus tareas de control se hace necesario el procesamiento de la información mediante operaciones sobre bases de datos.

El presente informe tiene como objetivo principal describir el proceso de trabajo con los datos referente al MINTUR. Es decir, ilustrar el uso de las tecnologías empleadas para el almacenamiento y transformación, así como las variantes de extracción de información a partir de los datos almacenados para llegar a conclusiones y reportes estadísticos.

DESARROLLO

Capítulo 1

Trabajo con el almacenamiento de los datos

1. Descripción del problema a modelar

El Ministerio del Turismo (MINTUR) es el organismo estatal rector del Sistema de Turismo en Cuba, en el cual participan otras entidades del país. En este sentido, el MINTUR elabora las políticas y controla su aplicación en las entidades que administran directamente las propiedades del sector. Sus objetivos se basan en diseñar y concretar una comercialización más eficiente del producto turístico, incrementar y diversificar una oferta turística más competitiva, elevar el nivel de eficiencia económica del Sistema de Turismo y ampliar la proyección del horizonte temporal del desarrollo del turismo.

En función de cumplir los objetivos propuestos, la dirección del ministerio realizó una solicitud para desarrollar un sistema de información sustentado en la construcción de un almacén de datos que se alimente de las bases de datos relacionales existentes en el ministerio y de otras fuentes. El sistema debe permitir realizar análisis dinámicos del comportamiento de las entradas y salidas de extranjeros al país.

Las entradas y salidas de los extranjeros al país se analizan de manera diaria. Para el análisis de las entradas (arribos a Cuba) se utilizan los siguientes elementos: lugar de procedencia (continente, país), género (masculino, femenino), rango de edad (niños 0...12, jóvenes 13...20, adultos 22.60, tercera edad > 60), medio de transporte (aéreo, marítimo) y el nombre de la compañía.

En el caso de las salidas se tiene además el lugar destino (continente, país).

Esto permitirá al MINTUR orientar mejor las campañas publicitarias teniendo en cuenta, regiones específicas, temporadas del año, rango de edades y género de los clientes.

2. Tecnologías usadas para el trabajo con los datos

Para el modelado del almacén de datos se utiliza el ER/Studio, este software facilita el diseño del modelo lógico del problema dado, así como la generación del

código SQL que implementa la estructura de base de datos.

Para la gestión y manejo de base de datos se utiliza el PostgreSQL y el PG Admin, ahí se ejecutó el código generado a partir del diseño lógico y se obtuvo la estructura de base de datos.

Para poblar la base de datos y generar información a partir de los datos se utilizó la suite de programas de Pentaho Business Intelligence:

1. Pentaho Data Integration(PDI) para poblar la estructura de base de datos obtenida a partir del modelo lógico. Además, hace posible filtrar, modificar y limpiar los datos que serán introducidos en la base de datos, así como la extracción de los mismos. EN el PDI se realizan las operaciones de extracción, transformación y carga (ETL).
2. Pentaho Schema Workbench(PSW) facilita la transformación de datos, procesamiento y publicación de información. Permite el diseño de cubos de datos a partir de los cuales mediante consultas se obtiene información.
3. Pentaho User Console(PUC) para la publicación de la información obtenida a partir de consultas, posibilita de forma gráfica y sencilla el trabajo con los datos agrupados en los cubos de datos y la obtención de información.
4. Pentaho Report Designer(PRD) para la creación y el trabajo con reportes. Los reportes muestran de manera formal y clara la información obtenida a partir de consultas sobre los cubos de datos.

3. Diseño lógico del almacén de datos

El software ER/Studio permitió el diseño lógico del problema dado en un almacén de datos. Se muestra a continuación el diseño:

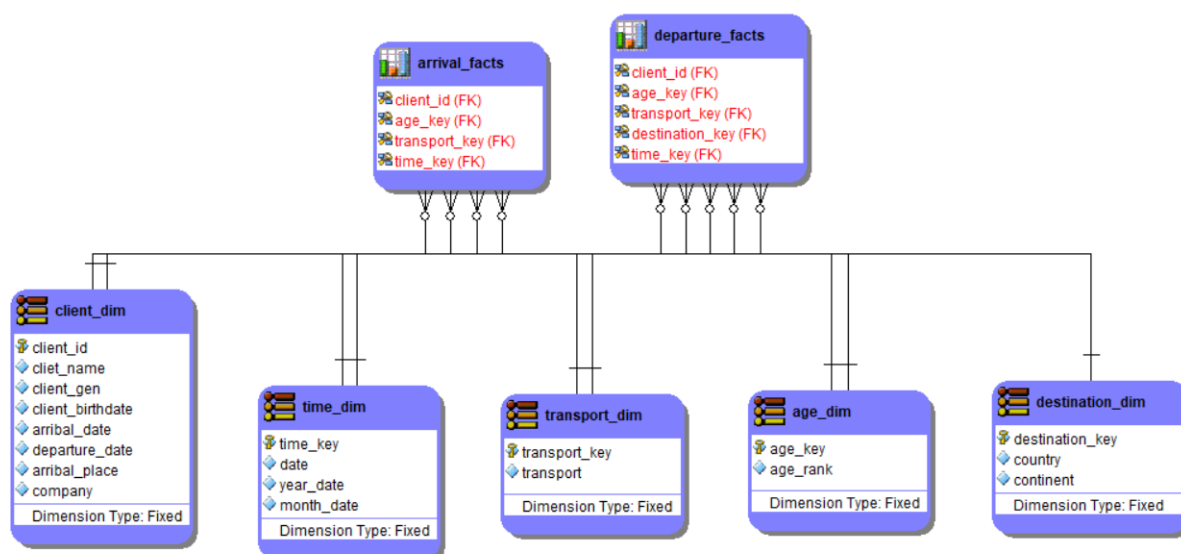


Figura 1.1: Diseño lógico

A partir del diseño lógico mediante el ER/Studio se genera el código SQL que implementa la estructura de la base de datos a utilizar.

El almacén de datos representado como estructura para la base de datos cuenta con:

1. Dimensiones

- **client_dim** (*Almacena los datos referentes al cliente, su nombre, género, fecha de nacimiento, arribo, partida, lugar de arribo y compañía en que viajaron*)
- **time_dim** (*Almacena la fecha, el año y mes de arribo de los clientes, así como la llave identificadora de la dimensión*)
- **transport_dim** (*Almacena la vía de transporte utilizada por cada cliente, agrupa los clientes por vía de transportación aérea o marítima, así como la llave identificadora de la dimensión*)
- **age_dim** (*Almacena el rango de edades en el que se agrupa cada cliente (niños, jóvenes, adultos, tercera edad), así como una llave identificadora de la dimensión*)
- **destination_dim** (*Almacena el país y continente de destino de los clientes al terminar su estancia en el país, así como la llave identificadora de la dimensión*)

2. Hechos

- **arrival_facts** (*Almacena las llaves de cada dimensión con datos de llegada de los clientes, ejemplo, el id del cliente, la llave de el rango de edad, vía de transporte y fechas de llegada de ese cliente*)
- **departure_facts** (*Almacena las llaves de cada dimensión con datos de salida de los clientes, ejemplo, el id del cliente, la llave de el rango de edad, vía de transporte, fechas de llegada y lugar de destino de ese cliente*)

4. Almacén de datos en la base de datos

El ER/Studio facilitó la generación del código sql para la creación de la estructura de base de datos. El script .sql con el código ejecutado en el Pg Admin de postgresql para la creación de la estructura de base de datos, está adjunto en los archivos del proyecto.

A continuación se muestra como quedaron las tablas de hechos y dimensiones en el PG Admin tras ejecutar el código sql generado por el ER/Studio a partir del diseño lógico.

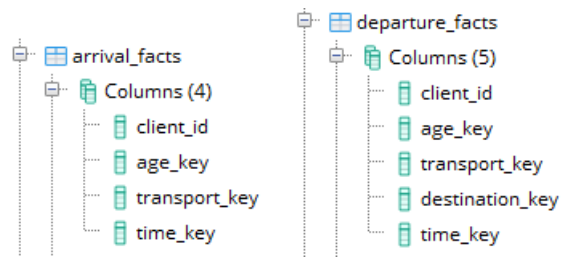


Figura 1.2: Tablas de hechos **arrival_facts**, **departure_facts**

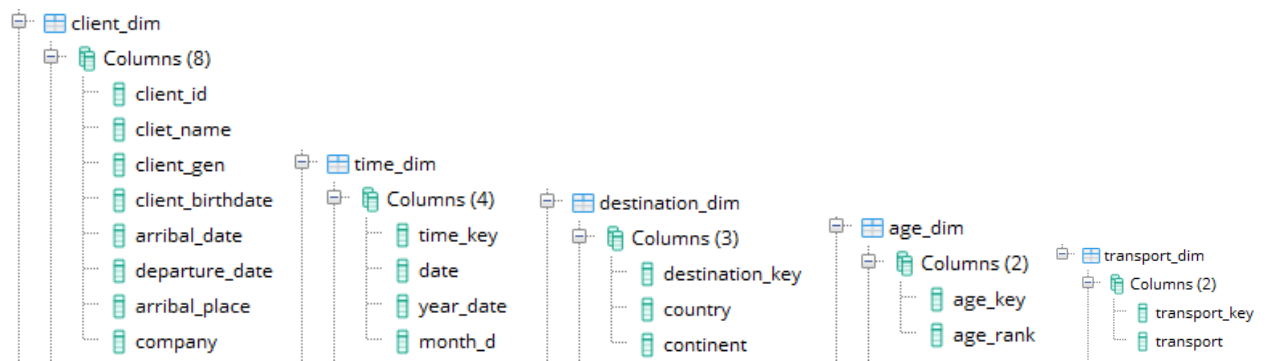


Figura 1.3: Tablas de dimensiones **client_dim**, **time_dim**, **destination_dim**, **age_dim**, **transport_dim**

Capítulo 2

Procesamiento de los datos

1. Operaciones de extracción, transformación y carga(ETL)

El PDI facilita implementar las operaciones básicas de ETL sobre la base de datos. Para poblar la estructura de base de datos obtenida a partir del modelo lógico se hizo necesario filtrar los datos a ingresar en cada tabla de dimensiones o hechos, pues muchos se encuentran corruptos, ejemplo, en el nombre de un cliente pueden estar contenidos dígitos o espacios. En cualquiera de los campos los datos a guardar necesitan ser filtrados.

Las **Transformaciones** tienen el objetivo de poblar cada tabla de la base de datos. En las transformaciones se filtran los datos y se van introduciendo teniendo en cuenta las reglas necesarias para evitar conflictos de llaves o llaves foráneas etc.

Los **Trabajos** son los encargados de que se desarrolle de forma organizada y sin conflictos cada una de las **Transformaciones** que pueblan la base de datos. En este caso bastó con solo un **Trabajo** que fuera ejecutando cada una de las **Transformaciones** implementadas.

1.1. Transformaciones

A continuación se muestran las transformaciones implementadas para la realización de operaciones ETL en la base de datos de MINTUR.

1.1.1. Transformación para poblar la dimensión `client_dim`

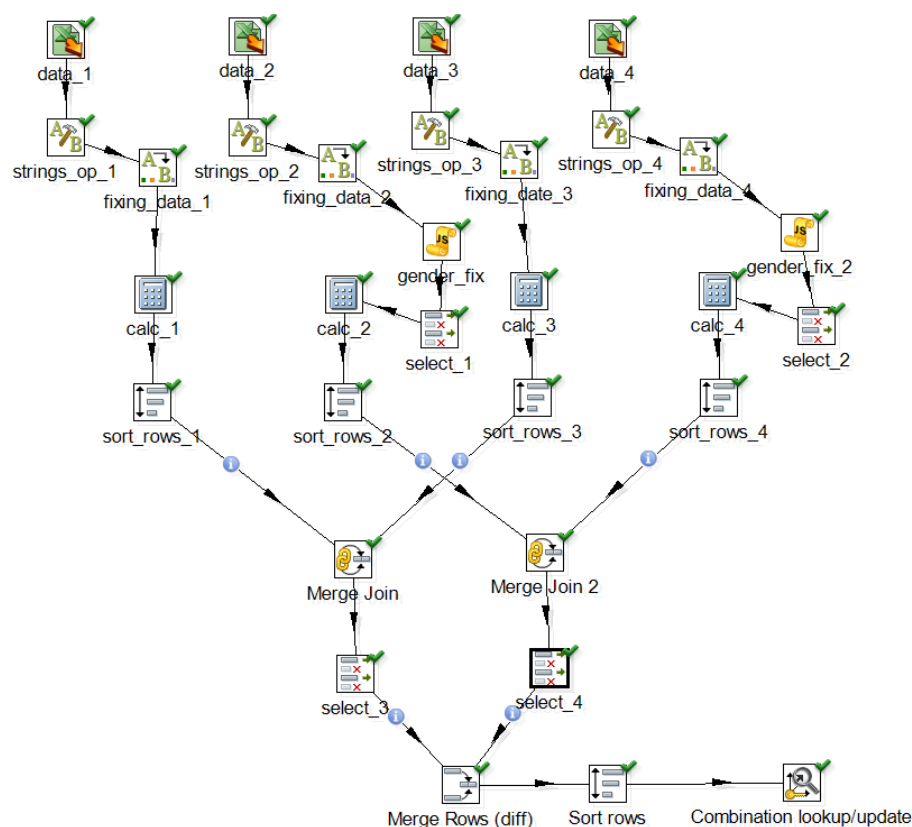


Figura 2.1: Transformación para poblar la dimensión `client_dim`

La transformación para poblar la dimensión `client_dim` con los datos de los clientes sigue el siguiente orden:

1. Se toman datos de las cuatro fuentes .xls que se tienen, las entradas de clientes al país y salidas por vía aérea y marítima, **Personas_MINTUR_Entrada** y **Personas_MINTUR_Salida**(vía aérea) y **Personas_MINTUR_Entrada1** y **Personas_MINTUR_Salida1**(vía marítima).
2. Se checkea el campo con el nombre de los clientes para corregir errores como dígitos contenidos, espacios o caracteres especiales.
3. Se corrige la fecha de arribo y de nacimiento de forma que quede en el formato yyyy/mm/dd.
4. Se ordenan las columnas de forma ascendente por nombre para hacer un inner join entre las tablas **Personas_MINTUR_Entrada** y **Personas_MINTUR_Salida** vía aérea y marítima respectivamente. El propósito del inner join es que solo cumple objetivo tener en la base de datos los clientes con los cuales se cuente con todos sus datos de arribo, su fecha y datos de salida.

- Se unen con el Merge Rows los datos de clientes que vinieron por vía marítima y aérea y son los que se insertan en la tabla **client_dim**. La llave (**client_id**) identificadora de la dimensión se genera de forma automática.

1.1.2. Transformación para poblar la dimensión **age_dim**

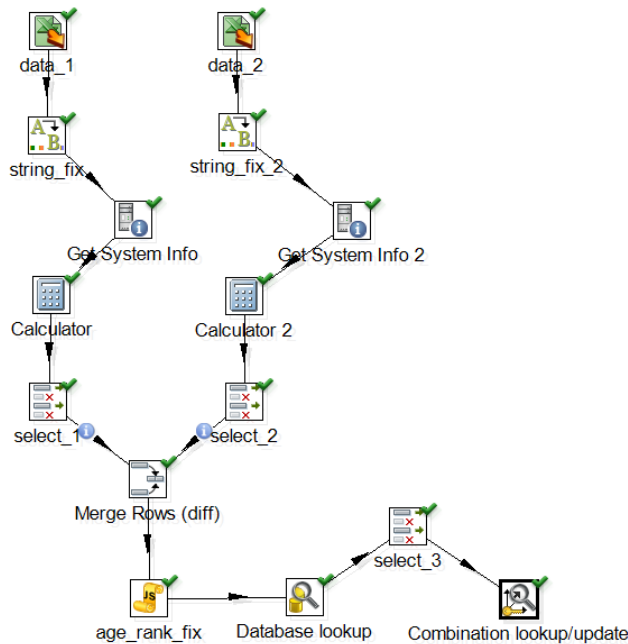


Figura 2.2: Transformación para poblar la dimensión **age_dim**

La transformación para poblar la dimensión **age_dim** con los rangos de edades de los clientes y la llave identificadora sigue el siguiente orden:

- Se toman datos de las dos fuentes .xls con las entradas de clientes al país por vía aérea y marítima, **Personas_MINTUR_Entrada**(vía aérea) y **Personas_MINTUR_Entrada1**(vía marítima), en este caso solo interesa guardar la fecha de nacimiento del cliente para determinar la edad y poder clasificar su rango: niño hasta 12 años, jóvenes de 13 a 21, adultos de 22 a 60 y tercera edad de 60 en adelante.
- Se corrige la fecha de nacimiento de forma que quede en el formato yyyy/mm/dd y a partir de ahí se calcula la edad en días usando la fecha actual proporcionada por el sistema y las funciones que brinda el **Calculator** para restar fechas (**Date A - Date B**). Se guarda de forma temporal la edad tipo entero.
 - Como lo que interesa es tener cada cliente clasificado en el rango de edades en que se ubica no es de interés guardar su edad. Se implementa un script sencillo que devuelve el rango de edad según su edad en días.

```

Script 1
//Script here

if (age_s <= 4380){
  rank = 'niños'
}else if (age_s >= 4745 && age_s <= 7665){
  rank = 'jóvenes'
}else if (age_s >= 8030 && age_s <= 21900){
  rank = 'adultos'
}else if (age_s > 21900){
  rank = 'tercera edad'
}

```

Figura 2.3: Script para determinar el rango de edad

3. Se busca en la base de datos el id del cliente según el nombre para verificar que se va a guardar el rango de edad de un cliente ya registrado y se guarda en la dimensión **age_dim** el rango de del cliente. La llave primaria(**age_key**) se genera automáticamente para cada cliente.

1.1.3. Transformación para poblar la dimensión **time_dim**

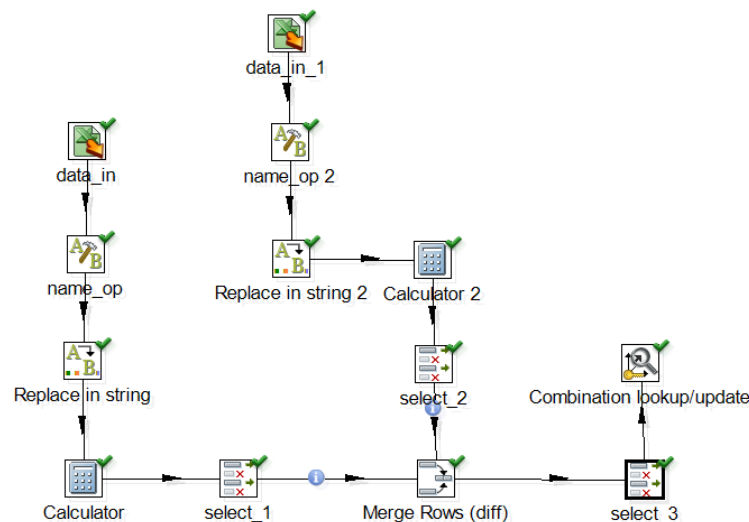


Figura 2.4: Transformación para poblar la dimensión **time_dim**

La transformación para poblar la dimensión **time_dim** con la fecha de llegada de los clientes, el año, el mes y la llave identificadora sigue el siguiente orden:

1. Se toman datos de las dos fuentes .xls con las entradas de clientes al país por vía aérea y marítima, **Personas_MINTUR_Entrada**(vía aérea) y **Personas_MINTUR_Entrada1**(vía marítima), en este caso solo interesa guardar la fecha de llegada del cliente y a partir de ahí determinar el año y mes de llegada.
2. Se corrige la fecha de llegada de forma que quede en el formato yyyy/mm/dd y a partir de ahí se determina el año y mes. Se utilizan las funciones que brinda

el **Calculator**, **Year of Date A** y **Month of Date A** que aíslan el año y mes respectivamente de una fecha dada.

3. Se unen con el Merge Rows las columnas con los datos determinados en cada tabla de entrada(**Personas_MINTUR_Entrada**(vía aérea) y **Personas_MINTUR_Entrada1**(vía marítima)).
4. Finalmente se guarda en la tabla **time_dim** la fecha de llegada de los clientes(**date**), el año(**year_date**) y el mes(**month_date**). La llave identificadora de los datos guardados para cada uno de los clientes(**time_key**) se genera de forma automática.

1.1.4. Transformación para poblar la dimensión **transport_dim**

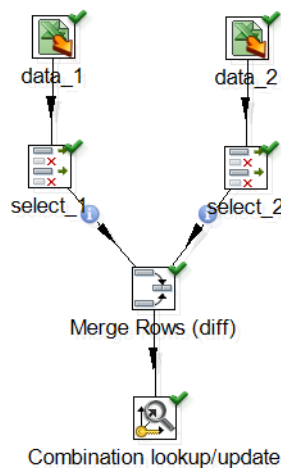


Figura 2.5: Transformación para poblar la dimensión **transport_dim**

La transformación para poblar la dimensión **transport_dim** con la vía de transporte(aérea o marítima) y la llave identificadora sigue el siguiente orden:

1. Se toman datos de las dos fuentes .xls con las entradas de clientes al país por vía aérea y marítima, **Personas_MINTUR_Entrada**(vía aérea) y **Personas_MINTUR_Entrada1**(vía marítima), en este caso solo interesa guardar la vía de transporte de los clientes que es el nombre del sheet del .xls. El nombre del sheet se guarda como un campo mediante las opciones que brinda la selección de campos a mostrar del .xls de entrada.
2. Se unen con el Merge Rows las columnas con los datos determinados en cada tabla de entrada(**Personas_MINTUR_Entrada**(vía aérea) y **Personas_MINTUR_Entrada1**(vía marítima)).
3. Finalmente se guarda en la tabla **transport_dim** la vía de transportación(**transport**). La llave identificadora de los datos guardados para cada uno de los clientes(**transport_key**) se genera de forma automática.

1.1.5. Transformación para poblar la dimensión destination_dim

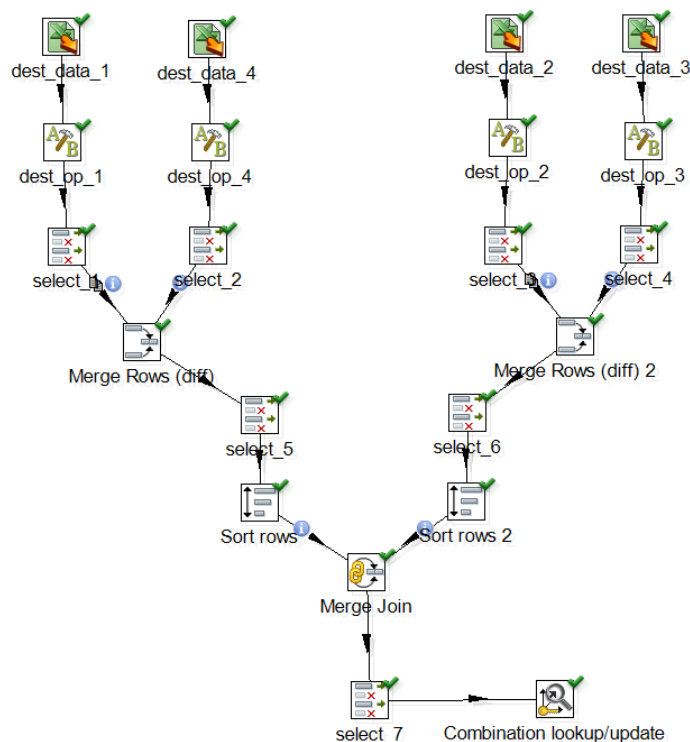


Figura 2.6: Transformación para poblar la dimensión **destination_dim**

La transformación para poblar la dimensión **destination_dim** con los datos de los clientes sigue el siguiente orden:

1. Se toman datos de las cuatro fuentes .xls que se tienen, las entradas de clientes al país y salidas por vía aérea y marítima, **Personas_MINTUR_Entrada** y **Personas_MINTUR_Salida**(vía aérea) y **Personas_MINTUR_Entrada1** y **Personas_MINTUR_Salida1**(vía marítima).
2. Se checkea el campo con el nombre de los clientes, el país y continente para corregir errores como dígitos contenidos, espacios o caracteres especiales.
3. El esquema muestra dos ramas principales, en la primera se analiza **Personas_MINTUR_Salida**(vía aérea) y **Personas_MINTUR_Salida1**(vía marítima) uniéndolos sus datos con el **Merge Row** y se seleccionan el país y continente de destino de los clientes al dejar el país, además de su nombre. En la segunda rama se analiza **Personas_MINTUR_Entrada**(vía aérea) y **Personas_MINTUR_Entrada1**(vía marítima) uniéndolos sus datos con el **Merge Row** y se seleccionan el nombre de cada cliente. El propósito de los merge realizados fue obtener todos los clientes con datos de salida del país(país y continente de destino) y en la segunda rama los clientes que entraron al país.

4. Se ordenan las columnas de forma ascendente por nombre para hacer un inner join entre los valores seleccionados, por una parte el nombre, país y continente de los clientes registrados con salida y en la otra los clientes que está registrados con entrada. El propósito del inner join es quedarse con el país y continente de destino solo de los clientes que estén registrados con entrada al país y salida. Es necesario este paso debido a que en los datos **Personas_MINTUR_Salida**(vía aérea) y **Personas_MINTUR_Salida1**(vía marítima) existen clientes que no está en las tablas **Personas_MINTUR_Entrada**(vía aérea) y **Personas_MINTUR_Entrada1**(vía marítima).
5. Finalmente se tienen los datos necesarios filtrados y se insertan en la dimensión **destination_dim**. La llave (**destination_key**) identificadora de la dimensión se genera de forma automática.

1.1.6. Transformación para poblar la tabla de hechos **arrival_facts**

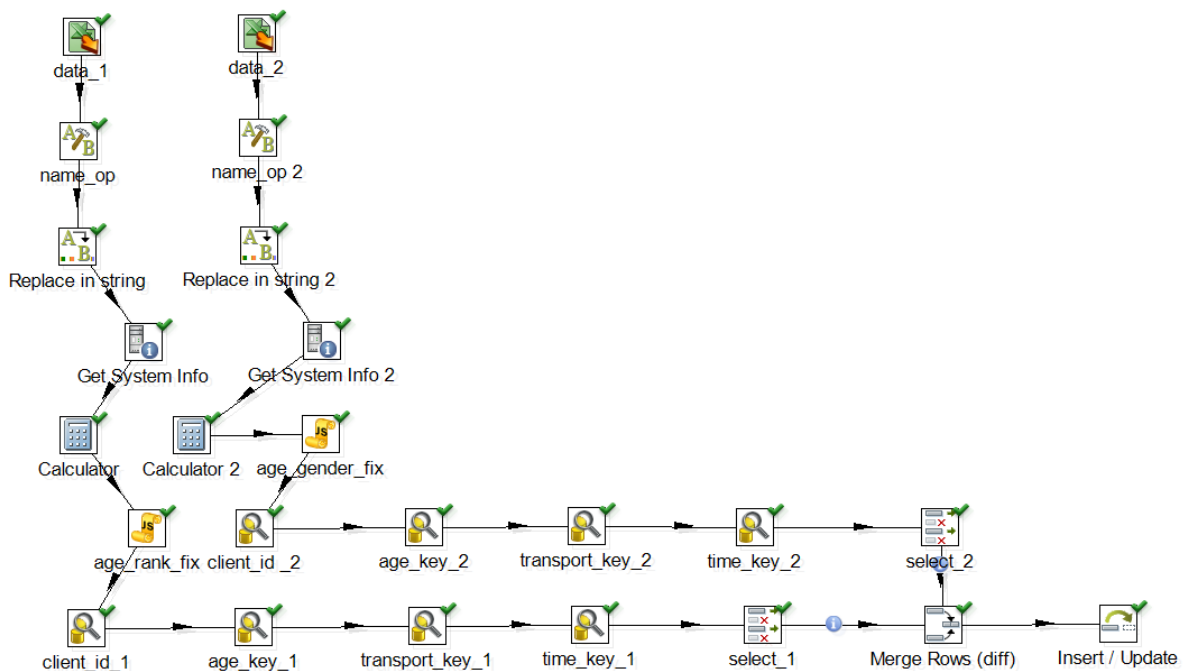


Figura 2.7: Transformación para poblar la tabla de hechos **arrival_facts**

La transformación para poblar la tabla de hechos **arrival_facts** con los datos de los clientes sigue el siguiente orden:

1. Se toman datos de las dos fuentes .xls con las entradas de clientes al país por vía aérea y marítima, **Personas_MINTUR_Entrada**(vía aérea) y **Personas_MINTUR_Entrada1**(vía marítima), en este caso interesa guardar las llaves identificadoras de de cada dimensión(**client_dim**, **age_dim**, **transport_dim**, **time_dim**), llaves que son foráneas y principales para la tabla de

hechos. El objetivo de esta transformación es poblar la tabla de hechos con los id de los clientes y a su vez, las llaves identificadoras de cada uno de los datos de ese cliente que se encuentran divididos en dimensiones.

2. Se checkea el campo con el nombre de los clientes para corregir errores como dígitos contenidos, espacios o caracteres especiales.
3. Se corrige la fecha de arribo y de nacimiento de forma que quede en el formato yyyy/mm/dd.
4. A partir de la fecha de nacimiento se calcula la edad en días usando la fecha actual proporcionada por el sistema y las funciones que brinda el **Calculator** para restar fechas (**Date A - Date B**). Se guarda de forma temporal la edad tipo entero. El proceso es el mismo descrito en la transformación para poblar **age_dim** obteniendo el rango de edad de cada cliente.
5. Se busca en la base de datos según el nombre, el id generado para los clientes cuando se ejecutó la transformación para poblar **client_dim**. A partir de que se tiene el id del cliente, se necesita el **age_key**, **transport_key** y **time_key** para poblar la tabla de hechos.
6. Se realizan búsquedas sucesivas en la base de datos por cada una de las tablas de dimensiones que ya se poblaron buscando las llaves generadas y guardándolas.
7. Se ejecuta un merge row para unir los datos encontrados a partir de **Personas_MINTUR_Entrada**(vía aérea) y **Personas_MINTUR_Entrada1**(vía marítima).
8. Finalmente se tienen los datos necesarios filtrados y se insertan en la tabla de hechos **arrival_facts** quedando poblada con todas las llaves correspondientes.

1.1.7. Transformación para poblar la tabla de hechos `departure_facts`

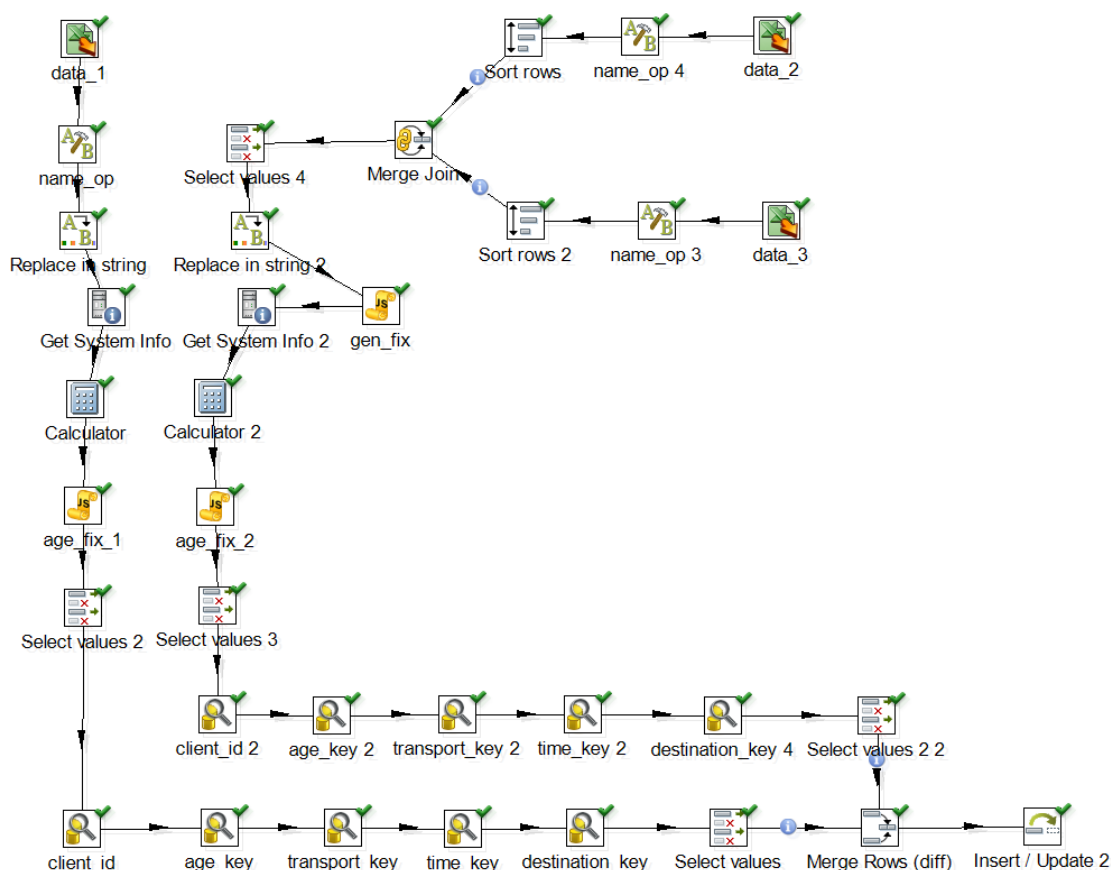
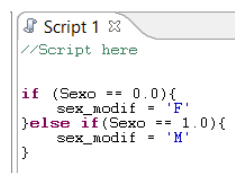


Figura 2.8: Transformación para poblar la tabla de hechos `departure_facts`

La transformación para poblar la tabla de hechos `departure_facts` con los datos de los clientes sigue el siguiente orden:

1. Se toman datos de las dos fuentes .xls con las salidas de clientes al país por vía aérea y marítima, **Personas_MINTUR_Salida**(vía aérea) y **Personas_MINTUR_Salida**(marítima). Además se necesita la fuente de entrada **Personas_MINTUR_Entrada1**. En este caso interesa guardar las llaves identificadoras de de cada dimensión(**client_dim**, **age_dim**, **transport_dim**, **time_dim**, **destination_dim**), llaves que son foráneas y principales para la tabla de hechos. El objetivo de esta transformación es poblar la tabla de hechos con los id de los clientes y a su vez, las llaves identificadoras de cada uno de los datos de ese cliente que se encuentran divididos en dimensiones. A diferencia de la tabla de hechos **arrival_facts**, esta tabla de hechos guarda el destino de partida de los clientes una vez que dejen el país.

2. Se checkea el campo con el nombre de los clientes para corregir errores como dígitos contenidos, espacios o caracteres especiales.
3. Se corrige la fecha de arribo y de nacimiento de forma que quede en el formato yyyy/mm/dd.
4. En la segunda rama que se muestra en el gráfico, se ejecuta un inner join entre los clientes registrados con entrada por vía marítima(**Personas_MINTUR_Entrada1**) y los clientes registrados con salida por la misma vía(**Personas_MINTUR_Salida1**). El propósito del inner join es seleccionar solo los clientes que existan en ambas fuentes de datos, es decir clientes de los cuales se cuenta con todos los datos necesarios, de entrada y salida.
5. A los datos obtenidos de las fuentes de datos con vía de transporte marítima, es necesario corregir el campo con el género de los clientes, pues viene expresado como 0 para femenino y 1 para masculino, mientras se tiene como estándar en la base de datos guardarlos con género F o M. Esta corrección se realiza mediante un sencillo script.



```

Script 1
//Script here

if (Sexo == 0.0){
  sex_modif = 'F'
}else if (Sexo == 1.0){
  sex_modif = 'M'
}

```

Figura 2.9: Script para cambiar el género de 0 a F y 1 a M

6. Una vez que se tienen los clientes que tanto por una fuente de datos como por otra(vía aérea y marítima) tienen todos sus datos registrados correctamente, se puede continuar con el proceso de búsqueda de las llaves.
7. A partir de la fecha de nacimiento se calcula la edad en días usando la fecha actual proporcionada por el sistema y las funciones que brinda el **Calculator** para restar fechas (**Date A - Date B**). Se guarda de forma temporal la edad tipo entero. El proceso es el mismo descrito en la transformación para poblar **age_dim** obteniendo el rango de edad de cada cliente.
8. Se busca en la base de datos según el nombre, el id generado para los clientes cuando se ejecutó la transformación para poblar **client_dim**. A partir de que se tiene el id del cliente, se necesita el **age_key**, **transport_key**, **time_key**, **destination_key** para poblar la tabla de hechos.
9. Se realizan búsquedas sucesivas en la base de datos por cada una de la tablas de dimensiones que ya se poblaron buscando las llaves generadas y guardándolas.
10. Se ejecuta un merge row para unir los datos seleccionados.

11. Finalmente se tienen los datos necesarios filtrados y se insertan en la la tabla de hechos **departure_facts** quedando poblada con todas las llaves correspondientes.

1.2. Trabajos

El trabajo implementado para organizar la ejecución consecutiva de las transformaciones que pueblan la base de datos se muestra a continuación:

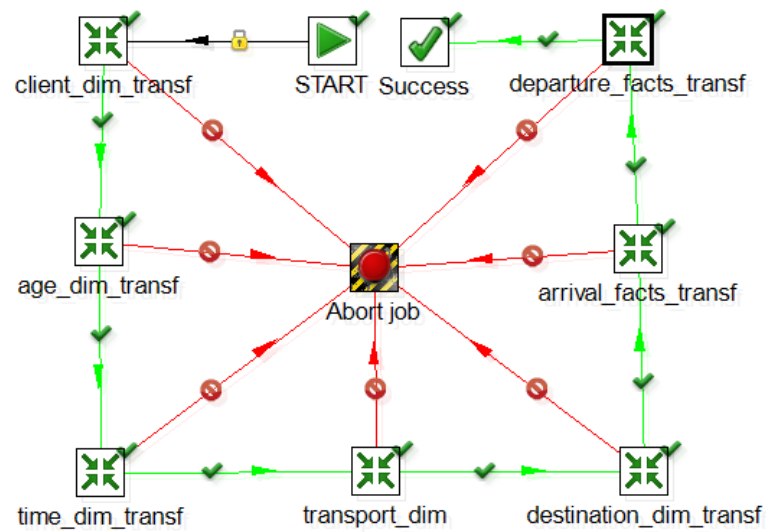


Figura 2.10: Trabajo

2. Cubo de datos OLAP

On-Line Analytical Processing (OLAP), es un método ágil y flexible para organizar datos en un sistema u organización multidimensional. Su objetivo es recuperar y manipular datos y combinaciones de los mismos a través de consultas o incluso reportes.

Con el PSW se modeló un cubo de datos a partir del cual se pueden realizar diferentes consultas sobre los datos.

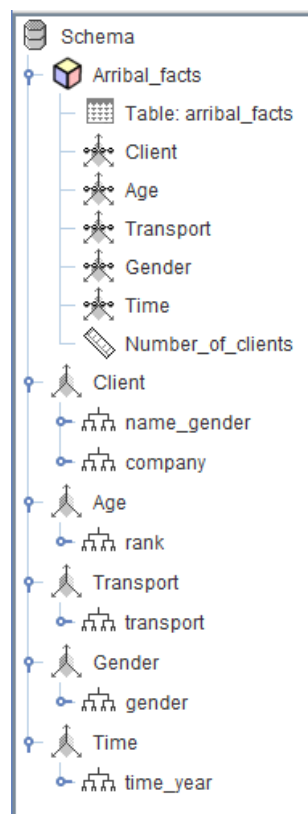


Figura 2.11: Cubo de datos en PSW

El cubo de datos modelado se centra en el número de clientes según las diferentes categorías. En este caso el número de clientes que entraron al país de una compañía X, mediante la vía de transporte Y, la cantidad según el rango de edad o el género. Así como la cantidad de clientes por año de llegada o mes.

El propósito del cubo de datos es facilitar la publicación de consultas sobre los datos en el Pentaho User Console. Además permite la ejecución de operaciones OLAP para la visualización de la información.

Las operaciones analíticas básicas OLAP son:

1. Slice & dice, permite hacer una selección de los valores de la dimensión que se deseen mostrar.
2. Drill-down, permite apreciar los datos en un mayor nivel de detalle, bajando por una jerarquía definida en el cubo OLAP.

3. Roll-up, proceso de visualización de datos con disminución de los detalles. Contrario a drill-down.
4. Pivot, permite seleccionar el orden de visualización de los atributos e indicadores para analizar la información desde diferentes perspectivas. Ejemplo hacer un swap a los campos de las columnas con las filas.

Al aplicar las operaciones básicas OLAP sobre el cubo de datos diseñado se obtuvieron los siguientes resultados en el PUC. Primero se muestra una consulta general de los datos guardados en el cubo diseñado. El código correspondiente a cada una de las consultas se encuentra adjunto a los archivos del proyecto.

2.1. Consulta general

Columnas

Number_of_clients

Filas

year

rank

gender

transport

company

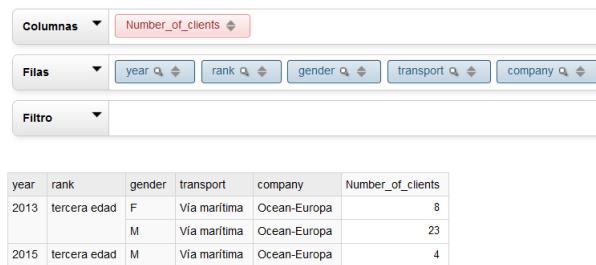
Filtro

year	rank	gender	transport	company	Number_of_clients
2013	tercera edad	F	Vía marítima	Ocean-Europa	8
		M	Vía marítima	Ocean-Europa	23
2014	adultos	F	Vía Aerea	air-france	27
				Copa	1
		M	Vía Aerea	air-europa	13
				air-france	3
				Copa	10
				Cubana	27
	jovenes	F	Vía Aerea	Cubana	49
		M	Vía Aerea	air-france	5
				Copa	19
	tercera edad	F	Vía Aerea	air-europa	69
			Vía marítima	Ocean-Europa	50
		M	Vía Aerea	air-europa	17
				Cubana	26
		Vía marítima	Ocean-Europa	55	

Figura 2.12: Consulta general

La consulta general que se realizó sobre el cubo de datos tiene el objetivo de mostrar el número de clientes que arribó al país por años, según el rango de edad, según el género, según la vía de transporte ya sea marítima o aérea y según la compañía por la cual realizaron el viaje.

2.2. Slice & dice

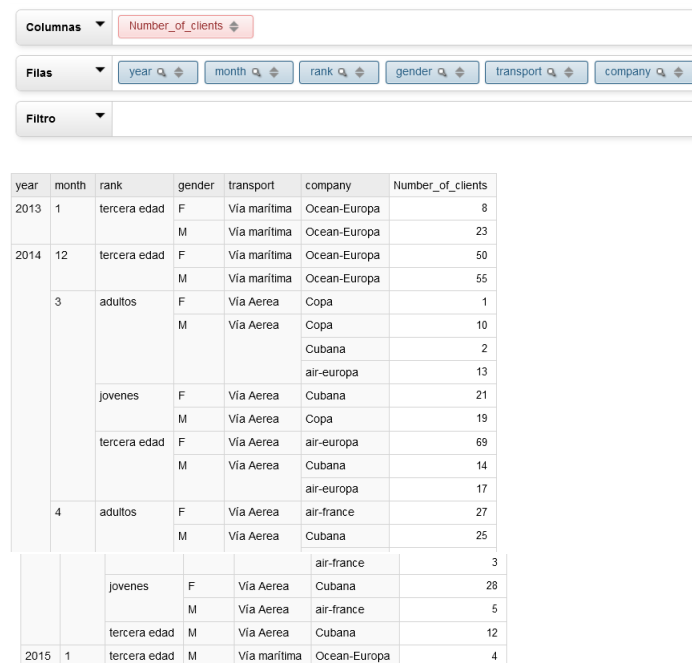


year	rank	gender	transport	company	Number_of_clients
2013	tercera edad	F	Via marítima	Ocean-Europa	8
		M	Via marítima	Ocean-Europa	23
2015	tercera edad	M	Via marítima	Ocean-Europa	4

Figura 2.13: Consulta slice and dice

En esta consulta se mostró solamente el número de clientes que arribaron en los años 2013 y 2015, agrupados según su rango de edad, género, vía de transporte y compañía.

2.3. Drill-down



year	month	rank	gender	transport	company	Number_of_clients
2013	1	tercera edad	F	Via marítima	Ocean-Europa	8
			M	Via marítima	Ocean-Europa	23
2014	12	tercera edad	F	Via marítima	Ocean-Europa	50
			M	Via marítima	Ocean-Europa	55
	3	adultos	F	Via Aerea	Copa	1
			M	Via Aerea	Copa	10
					Cubana	2
					air-europa	13
		jovenes	F	Via Aerea	Cubana	21
			M	Via Aerea	Copa	19
		tercera edad	F	Via Aerea	air-europa	69
			M	Via Aerea	Cubana	14
					air-europa	17
	4	adultos	F	Via Aerea	air-france	27
			M	Via Aerea	Cubana	25
					air-france	3
		jovenes	F	Via Aerea	Cubana	28
			M	Via Aerea	air-france	5
		tercera edad	M	Via Aerea	Cubana	12
2015	1	tercera edad	M	Via marítima	Ocean-Europa	4

Figura 2.14: Consulta drill-down

En esta consulta se bajó una jerarquía en el cubo OLAP, se añadió la columna mes(**month**) para conocer de forma más desglosada el número de clientes que arribaron al hotel. Es decir, se pasó de conocer solamente el número de clientes agrupado por años, a una agrupación por mes más detallada.

2.4. Roll-up

year	rank	gender	transport	Number_of_clients
2013	tercera edad	F	Vía marítima	8
		M	Vía marítima	23
2014	adultos	F	Vía Aerea	28
		M	Vía Aerea	53
	jovenes	F	Vía Aerea	49
		M	Vía Aerea	24
	tercera edad	F	Vía Aerea	69
			Vía marítima	50
		M	Vía Aerea	43
			Vía marítima	55
2015	tercera edad	M	Vía marítima	4

Figura 2.15: Consulta roll-up

La consulta roll-up es lo opuesto a drill-down, en este caso se muestra a menos detalle, es decir se sube en una jerarquía pues la compañía de viaje brinda más detalles que solo la vía de transporte, si aérea o marítima.

2.5. Pivot

Columnas

year

rank

gender

transport

company

Filas

Number_of_clients

Filtro

Info: 10.56 / 18 x 6 / 0.01s

MeasuresLevel	2013				2014								2015				
	tercera edad		adultos				jovenes				tercera edad				tercera edad		
	F	M	F	M			F	M	F	M			M				
	Via maritima	Via maritima	Via Aerea	Via Aerea			Via Aerea	Via Aerea	Via Aerea	Via maritima	Via Aerea	Via maritima	Via maritima				
	Ocean-Europa	Ocean-Europa	air-france	Copa	air-europa	air-france	Copa	Cubana	Cubana	air-france	Copa	air-europa	Ocean-Europa	air-europa	Cubana	Ocean-Europa	Ocean-Europa
Number_of_clients	8	23	27	1	13	3	10	27	49	5	19	69	50	17	26	55	4

Figura 2.16: Consulta pivot

La consulta pivot, aporta otro punto de vista a el análisis de los datos, en este caso intercambiamos los campos columna de la consulta general por los campos filas.

3. Reportes

Los reportes hacen posible el registro de la información obtenida a partir de las consultas aplicadas sobre los datos de una forma clara, comprensible y permite el uso de gráficos para visualizar la información.

Se pueden obtener los reportes en varios formatos, en este caso se guardaron en formato .pdf y se publicaron para ser visualizados con el PUC.

A continuación se muestran los reportes obtenidos para algunas de las consultas antes ejecutadas y un ejemplo de reporte dinámico.

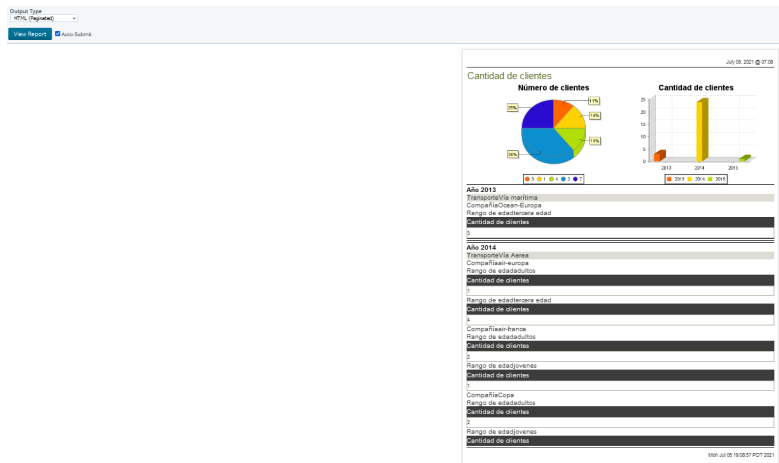


Figura 2.17: Reporte consulta general

Se mostró el reporte general que separa el número de clientes según el año en que arribaron al país, la vía de transporte, la compañía en que viajaron y el rango de edad en que se encuentran.

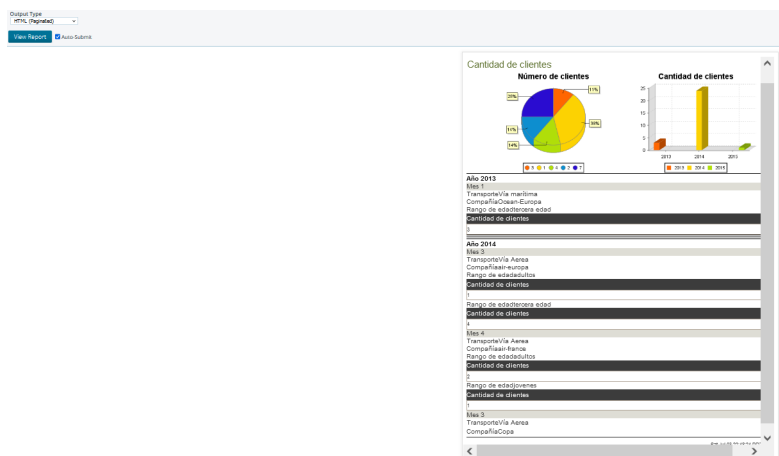


Figura 2.18: Reporte para la consulta drill-down

Se mostró el reporte correspondiente a la consulta drill-down que separa el número de clientes según el año en que arribaron al país, el mes, la vía de transporte, la compañía en que viajaron y el rango de edad en que se encuentran.

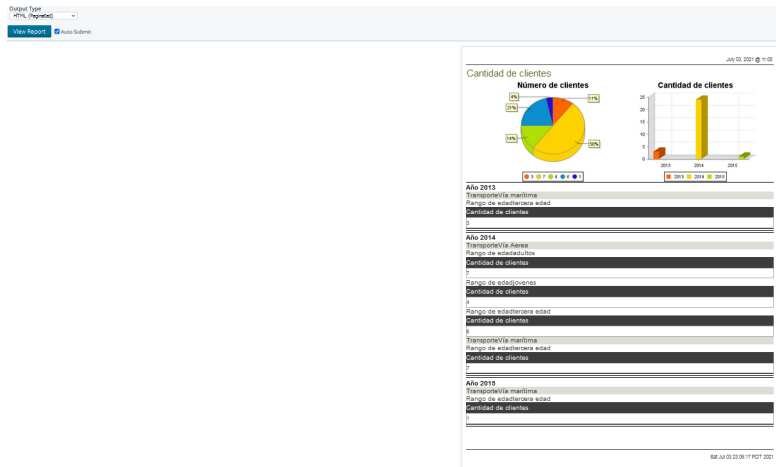


Figura 2.19: Reporte para la consulta roll-up

Se mostró el reporte correspondiente a la consulta roll-up que separa el número de clientes según el año en que arribaron al país, la vía de transporte en que viajaron y el rango de edad en que se encuentran.

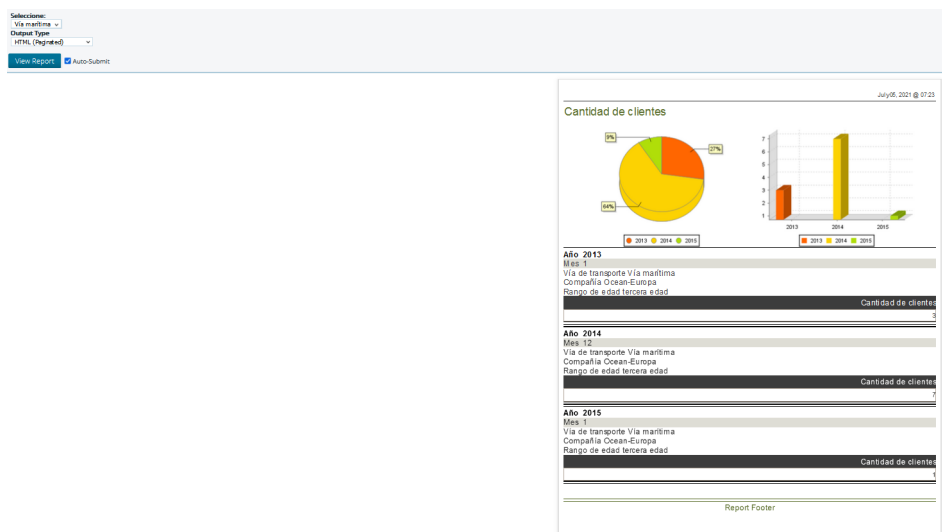


Figura 2.20: Reporte con parámetro, seleccionando vía marítima

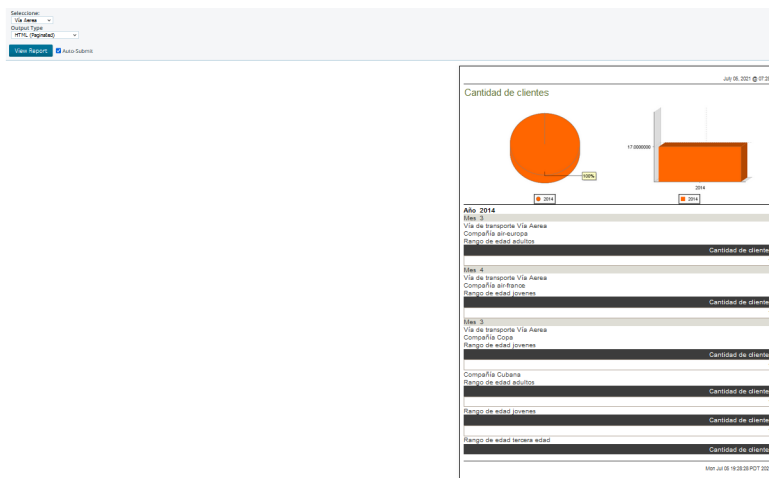


Figura 2.21: Reporte con parámetro, seleccionando vía aérea

Se mostró el reporte con parámetros que separa el número de clientes según el año en que arribaron al país, la vía de transporte, la compañía en que viajaron y el rango de edad en que se encuentran. El parámetro permite la selección de la vía de transporte, ya sea marítima o aérea.

CONCLUSIONES

Se estudió el proceso de trabajo con datos referentes al MINTUR, su extracción, transformación y carga para poblar una base de datos sobre la cual se realizaron consultas para la obtención y el análisis de información. Se utilizó el ER/Studio para diseñar el modelo lógico que representa el problema planteado y facilita la implementación de una estructura de base de datos. Además, se usaron las herramientas para el procesamiento de datos que brinda Pentaho y se ilustró paso a paso como se organizaron y trabajaron los datos a procesar, primero poblando la base de datos de forma correcta y limpia, luego agrupando los datos en un cubo de datos OLAP para así aplicar consultas básicas OLAP y obtener información en el PUC.