



# DASHBOARD

## Trabajo Final – Base de Datos Masivas [11088]

### Descripción breve

Este documento contiene el desarrollo completo de un Dashboard a implementar en una empresa, así como también el objetivo de ser el trabajo final de la asignatura Base de Datos Masivas, Lic. en Sistemas de Información, UNLu.

Luis Andrés Giordano  
andresgiordano.unlu@gmail.com



## ÍNDICE

<b>INTRODUCCIÓN</b> .....	<b>3</b>
<b>OBJETIVO</b> .....	3
<b>¿QUÉ ES UN DASHBOARD?</b> .....	3
<b>PASOS</b> .....	<b>4</b>
<b>INDICADORES</b> .....	<b>5</b>
<b>NRO. 1: COSTO DE HORAS HOMBRE PRODUCTIVAS</b> .....	5
Análisis .....	6
<b>NRO. 2: COSTO DE HORAS HOMBRE IMPRODUCTIVAS</b> .....	6
Análisis .....	6
<i>Relación entre Nro. 1 y Nro. 2</i> .....	6
<b>NRO. 3: RELACIÓN CANTIDAD PRODUCTOS ENTREGADOS Y PRODUCIDOS</b> .....	7
<b>NRO. 4: RELACIÓN PRODUCTOS ENTREGADOS Y PEDIDOS</b> .....	7
<b>NRO. 5: UNIDADES POR HORA HOMBRE</b> .....	8
<b>NRO. 6: OBJETIVO DE ENTREGA DE HOJAS</b> .....	8
<b>NRO. 7: OBJETIVO DE VENTAS SEMANAL</b> .....	8
<i>Relación entre Nro. 5, Nro. 6 y Nro. 7</i> .....	9
<b>NRO. 8: COMPARACIÓN ENTRE FACTURACIÓN E INGRESOS</b> .....	9
Análisis .....	9
<b>NRO. 9: MAPA DE VENTAS POR ZONA</b> .....	10
<b>ESTRUCTURA DEL DATAWAREHOUSE</b> .....	<b>11</b>
<b>ESQUEMAS POR TEMA</b> .....	11
<b>Ventas</b> .....	12
Modelo conceptual .....	12
Modelo lógico .....	12
<b>Pedidos</b> .....	13
Modelo conceptual .....	13
Modelo lógico .....	13
<b>Mov_stock</b> .....	14
Modelo conceptual .....	14
Modelo lógico .....	14
<b>Ingreso</b> .....	14
Modelo conceptual .....	14
Modelo lógico .....	15
<b>Producción</b> .....	15
Modelo conceptual .....	15
Modelo lógico .....	16
<b>ESQUEMA DE CONSTELACIÓN</b> .....	16
<b>DEFINICIÓN DE CUBOS</b> .....	17
<b>ETL (EXTRACCIÓN, TRANSFORMACIÓN Y LOADING)</b> .....	<b>21</b>
<b>DESCRIPCIÓN</b> .....	21
<b>EJEMPLO</b> .....	22
Job .....	22
Actualización de dimensiones .....	22
Actualización de tabla de hechos .....	24



<b>DISEÑO E IMPLEMENTACIÓN DEL DASHBOARD .....</b>	<b>26</b>
<b>DISEÑO .....</b>	<b>26</b>
<i>Costos y ventas .....</i>	<i>26</i>
<i>Entregas y producción .....</i>	<i>27</i>
<b>IMPLEMENTACIÓN .....</b>	<b>28</b>
<i>Estructura de CDF .....</i>	<i>28</i>
<i>Ejemplo de definición de un componente.....</i>	<i>30</i>



# Introducción

## Objetivo

A partir de la base de datos de una empresa (la cual se dedica a la comercialización de carpintería metálica) generar un Data Warehouse el cual va a estar orientado a 5 (cinco) temas:

- a. Ventas
- b. Pedidos
- c. Producción
- d. Movimientos de stock (solo interesan altas)
- e. Ingresos (pagos registrados)

A partir de estos enfoques se diseñará un Dashboard que le provea al tomador de decisiones de la empresa diferentes indicadores y alertas, como el objetivo de la productividad en base a las ventas, objetivo de promedio de entrega mensual de hojas, objetivo de entrega de pedidos, entre otros.

El Dashboard se generará con la Suite Pentaho la cual brinda herramientas que ayudan a todo el proceso:

1. Pre-procesamiento e integración de los datos
2. Generación y visualización de los cubos
3. Generación del Dashboard

## ¿Qué es un Dashboard?

Se entiende por cuadro de mando o Dashboard al sistema que informa de la evolución de los parámetros fundamentales de negocio de una organización o de un área del mismo. Es una de las posibilidades de “explotar” datos multidimensionales y extraer información valdeira que ayude a la toma de decisiones en los diferentes niveles de la organización.

Los parámetros se miden a partir de:

- **Métricas:** valores sobre la actividad o resultado del negocio. Ej.: monto de ventas semanal.
- **KPI's (Key Performance Indicator):** medida del nivel de cumplimiento de un objetivo. Ej.: velocímetro que indica verde en caso de que se supere el monto de ventas estimado para la semana.

Formados por:

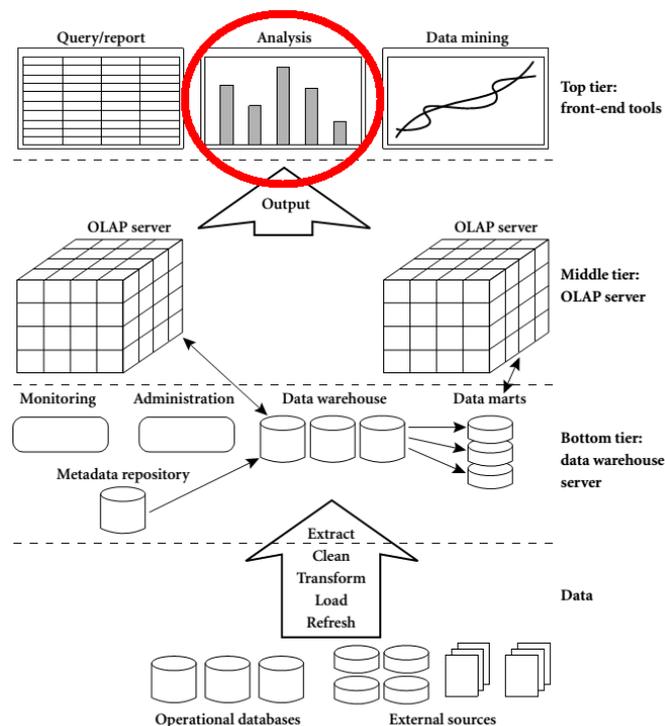
- **Gráficos,**
- **Mapas,**
- **Alertas visuales, etc.**

## Pasos

En esta sección la idea es mostrar brevemente cual va a ser el proceso y las actividades que se van a tener que llevar a cabo para llegar al producto final.

- I. **Identificación de Indicadores:** en este paso se realizan entrevistas con empleados de la empresa para encontrar aquellas métricas y KPI's que le resultan interesantes a las personas de la organización.
- II. **DW (Data Warehousing):** en este paso se diseña e implementa la estructura multidimensional que dará soporte a los datos que se obtengan del paso siguiente. También se agrega a este paso la generación de los cubos que se desarrollara con la herramienta Mondrian de la Suite.
- III. **ETL (Extracción, Transformación & Loading):** aquí es cuando se obtienen los datos de la base de datos, se los transforman para adaptarse a la nueva estructura multidimensional y luego se los cargan. Además, la nueva estructura debe ser recargada constantemente debido a la necesidad final de mostrar datos actualizados; sino, no tendría sentido que un Dashboard se base en dichos datos.
- IV. **Diseño e implementación del Dashboard:** en este paso se diseñara la forma en que se van a disponer y mostrar los diferentes indicadores. Luego se implementara el Dashboard usando la suite Pentaho dejándolo al servicio de la empresa.

En el grafico podemos ver rápidamente parte del proceso y las herramientas que implican la realización del Dashboard:





## Indicadores

A continuación se muestran cada uno de los indicadores que mostrará el Dashboard en términos de:

- a. **Descripción:** descripción breve del indicador.
- b. **Objetivo estratégico:** muestra que información valiosa genera para el tomador de decisiones.
- c. **Métrica:** fórmula sencilla de cómo se obtiene el valor del indicador.
- d. **Rango de tiempo:** a que rango de tiempo corresponde el valor que se muestra.
- e. **Meta:** cuáles son los valores para los cuales el indicador debería marcar un estado u otro; verde = bueno, amarillo = precaución, rojo = peligro.
- f. **Dirección:** indica si maximizar el valor de la métrica es lo buscado o viceversa.
- g. **Razón:** ¿Por qué es un buen indicador?

Los indicadores se obtuvieron realizando una entrevista con una persona encargada de la producción y del manejo del personal, así como también de mantener un objetivo de ventas apropiado para la empresa.

A continuación se enumeran los indicadores:

### Nro. 1: Costo de horas hombre productivas

**Descripción:** Mide el porcentaje del costo de horas hombre productivas sobre el monto total facturado.

**Objetivo Estratégico:** el tomador de decisiones necesita evaluar que exista una relación aceptada entre el monto facturado y el costo de horas productivas para determinar si el gasto en RRHH es el correcto.

**Métrica:**  $(\text{Costo horas hombre productivas} * 100) / \text{Monto facturado}$

**Rango de tiempo:** semana anterior a la fecha actual.

**Meta:** entre [0; 4,1) corresponde al valor azul debido a que se registra muy bajo costo productivo, [4,1; 6,5) para verde, [6,5; 10) para amarillo y [ 10; 15) para el rojo.

**Dirección:** Balancear, es decir, para vender más necesariamente voy a necesitar más productividad (más horas hombre produciendo), y viceversa, es por eso que la relación entre las variables debe mantenerse.

**Razón:** según el relevamiento que se realizó, el tomador de decisiones usa este indicador para controlar que **el costo productivo no se eleve más allá de lo esperado**. Para realizar este control necesita calcular el porcentaje del costo productivo sobre las ventas y ese es el valor que toman como referencia; y es el que muestra el tablero. El objetivo de este indicador puede cambiar, de hecho podría bajarse el umbral para el color azul, lo cual indicaría una producción más eficiente y no una alerta por tiempo ocioso en horas productivas.



## Análisis

Se realizó un análisis en cuanto a la correlación de las ventas con el costo productivo, semana a semana del año 2015. El resultado del coeficiente de correlación resultante fue de 0,76, es decir que al aumentar las ventas el costo productivo también aumenta, y viceversa.

## Nro. 2: Costo de horas hombre improductivas

**Descripción:** Mide el porcentaje del costo de horas hombre improductivas sobre el monto total facturado.

**Objetivo Estratégico:** el tomador de decisiones debe poder ver si la relación entre estas dos variables se da de forma negativa para analizar si el gasto en horas improductivas sobrepasa el umbral establecido.

**Métrica:**  $(\text{Costo horas hombre improductivas} * 100) / \text{Monto facturado}$

**Rango de tiempo:** semana anterior a la fecha actual.

**Meta:** [0; 0,7) = azul, [0,7; 2,4) = verde, [2,4; 5) = amarillo y [5; 10] = rojo.

**Dirección:** Mantener o minimizar, es decir, se quiere mayor monto en ventas a igual o menor cantidad de horas improductivas.

**Razón:** la persona encargada de tomar decisiones desea simplemente controlar que el costo productivo se mantenga en los porcentajes establecidos. El objetivo puede cambiar, de hecho el valor de techo para el color verde podría ir disminuyendo según el criterio del tomador de decisiones para lograr una producción más eficiente.

## Análisis

Al igual que en el indicador anterior se realizó un análisis en cuanto a la correlación de las ventas con el costo improductivo. El resultado al que se llegó no es “bueno” debido a que el coeficiente de relación resultante fue de 0,14. Esto está indicando que en realidad existen otras variables que afectan al costo improductivo.

Como se trata de un indicador relevado directamente de la empresa y tomando en cuenta que lo utilizan para tomar decisiones, se consideró que tendrán razones suficientes para considerar la relación entre estas dos variables como válida, por lo cual solo queda a mi alcance mostrar este análisis para que lo tomen en cuenta como una advertencia.

## Relación entre Nro. 1 y Nro. 2

El tablero mostrará una relación entre el indicador N°1 y el N°2. Esta relación se mostrará en un rango de 0 a 1 de la siguiente manera:

- a. Cuando se acerca a 0 quiere decir que el gasto en horas improductivas es menor al gasto en horas productivas.
- b. Cuando se acerca a 1 quiere decir que el costo improductivo se acerca al costo productivo.



El objetivo de esta relación es bien claro, trata de mostrar que el gasto en la producción sea siempre inversamente proporcional al gasto en la improductividad de los empleados. La meta de esta relación es llegar lo más cerca posible de 0.

### Nro. 3: Relación cantidad productos entregados y producidos

**Descripción:** Mide la cantidad de productos entregados de los producidos

**Objetivo Estratégico:** el tomador de decisiones desea controlar la cantidad entregada del total producido para evaluar si se produce más de lo que se está entregando.

**Métrica:**  $(\text{Cantidad total de productos entregados} * 100) / \text{total productos producidos}$

**Rango de tiempo:** semana anterior a la fecha actual.

**Meta:** [0; 45) = rojo, [45; 60) = amarillo, [60; 100] = verde. Es decir, siempre se espera que se entregue el 60% de lo que se produce.

**Dirección:** Maximizar, siempre se intenta que la producción sea la justa para el cumplimiento con las entregas.

**Razón:** es un buen indicador debido a que si el valor es muy bajo (Ej.: 30%) puedo compararlo visualmente con el valor de la métrica 1 y evaluar si el gasto en productividad es demasiado, si esto ocurre posiblemente exista un gasto totalmente innecesario en materia prima y RRHH debido a que no estoy pudiendo entregar, por algún motivo, todo lo que produjo. El objetivo no es siempre llegar al 100%, de hecho podría establecerse el objetivo entre un 80 y 90% porque la nueva política de la empresa no es Just In Time (producir en base a los pedidos registrados, por ende, vender todo lo que se produce) sino tener stock extra.

### Nro. 4: Relación productos entregados y pedidos

**Descripción:** Mide la cantidad de productos entregados de los pedidos, tomando los pedidos con fecha de entrega a la semana anterior. Existe una diferencia entre los tipos de productos: los productos de 1 (una) sola hoja se entregan a 15 días, y los que tienen más de 1 (una) a 30, por lo cual siempre se trabaja con la fecha de entrega aproximada del producto.

**Objetivo Estratégico:** el tomador de decisiones debe analizar si la empresa puede cumplir normalmente con sus pedidos; si se atrasa o no.

**Métrica:**  $(\text{Cantidad total de productos entregados} * 100) / \text{total productos pedidos}$

**Rango de tiempo:** basado en la semana anterior a la fecha actual.

**Meta:** se busca como mínimo un 90% semanal de entrega. Por eso los rangos establecidos son: [0,60) = rojo, [60,90) = amarillo, [90,100) = verde y [100,150) = azul (lo que indicaría pedidos atrasados).

**Dirección:** Maximizar

**Razón:** el indicador tiene un grado alto de utilidad, debido a que el tomador de decisiones podrá evaluar a simple vista si la empresa se atrasará con los pedidos (rojo) o si ya se encuentra atrasada (azul).



## Nro. 5: Unidades por hora hombre

**Descripción:** Mide el número de hojas realizadas por hora de trabajo productivo. Una hora-hombre es la cantidad de trabajo que realiza una persona en una hora. Se mide en general tomando todas las horas productivas de todos los empleados y analizando cuántas hojas se produjeron.

**Objetivo Estratégico:** Para evaluar la productividad del trabajo en términos de producción por hora-hombre.

**Métrica:**  $(\text{Cantidad de unidades de producción} * 100) / \text{Cantidad de horas hombre productivas}$

**Rango de tiempo:** basado en los meses transcurridos del año.

**Meta:** se busca aproximadamente 1,9 unidades mensuales.

**Dirección:** Maximizar

**Razón:** permite evaluar mes a mes como se está llevando la producción, además esta métrica es comparable también con otras que muestra el dashboard.

## Nro. 6: Objetivo de entrega de hojas

**Descripción:** muestra el promedio mensual de hojas entregadas.

**Objetivo Estratégico:** permite analizar rápidamente al tomador de decisiones si la empresa está cumpliendo con el objetivo, esperado por él, de entrega de hojas.

**Métrica:**  $\text{sum}(\text{cantidad de hojas vendidas del mes}) / \text{cant. días trabajados en el mes}$

**Rango de tiempo:** basado en los meses transcurridos del año.

**Meta:** Actualmente se busca un promedio diario de 250 hojas entregadas, pudiendo bajar hasta un promedio de 235.

**Dirección:** Maximizar

**Razón:** este indicador es interesante para el tomador de decisiones ya que le permite ver la evolución de este objetivo en el año, además de poder compararlo contra otros indicadores que muestra el dashboard.

## Nro. 7: Objetivo de ventas semanal

**Descripción:** muestra el objetivo del monto de ventas semanal esperado

**Objetivo Estratégico:** permite analizar rápidamente al tomador de decisiones si la empresa está cumpliendo con el objetivo, esperado por él, de monto semanal de ventas.

**Métrica:** sumatoria total de ventas realizadas en la semana.

**Rango de tiempo:** basado siempre en la semana anterior a la fecha actual.

**Meta:** a determinar por el tomador de decisiones de la empresa, actualmente el objetivo (verde) se encuentra entre 1.5 y 3 millones.

**Dirección:** Maximizar

**Razón:** este indicador es interesante para el tomador de decisiones ya que le permite ver rápidamente si se está cumpliendo con el objetivo semanal de ventas esperado, además de poder comparar contra el objetivo de entrega de hojas.



## Relación entre Nro. 5, Nro. 6 y Nro. 7

Existen conclusiones que pueden surgir a partir del análisis y comparación de los indicadores 5, 6 y 7. La comparación se debe a que puedo, por ejemplo, la empresa puede estar cumpliendo con el objetivo de ventas de hojas, pero debido a que el producto se compone de 1 o más hojas no deja en claro qué producto final se está entregando (o cuántos de ellos). Con respecto a lo anterior el tomador de decisiones puede decidir tempranamente en cuanto a realizar campañas de marketing para vender más de un producto que de otro, con el objetivo de aumentar el monto de ventas y no necesariamente aumentar el objetivo de entrega de hojas.

Con respecto a la métrica número 5, se puede determinar, comparando con el objetivo de entrega de hojas si se está produciendo lo suficiente como para alcanzar ese objetivo.

Finalmente podría tomarse la decisión de definir un umbral para 5, debido a un nuevo objetivo en 6 y 7.

## Nro. 8: Comparación entre facturación e ingresos

**Descripción:** muestra el monto de dinero que ingresa con respecto a lo que se facturó. Según la empresa todas las facturas se deben abonar aproximadamente en un 80% dentro de los 15 días después de la facturación, es por eso que en el gráfico se muestran los ingresos de la semana anterior y la facturación de 3 semanas atrás contando desde la fecha actual.

**Objetivo Estratégico:** analizar posibles endeudamientos de los clientes.

**Métrica:** Monto ingreso y total facturado

**Rango de tiempo:** semana anterior a la fecha actual para los ingresos y para las ventas (facturación) la facturación de la tercera semana contando desde la fecha actual.

**Meta:** el objetivo solo es mostrarlas en forma de gráfico para comparar valores.

**Dirección:** Maximizar ambas

**Razón:** este indicador es interesante para el tomador de decisiones debido a que le permite evaluar de cierta forma el cumplimiento de sus clientes con la empresa y calidad tanto del producto como en la entrega del mismo. Además, si la empresa vende mucho más de lo que realmente le terminan pagando quiere decir que de una forma u otra está teniendo problemas con sus clientes.

### Análisis

Si bien se parte de la base que este indicador le es útil a la persona que toma las decisiones, cabe alertar la posibilidad de que los ingresos de la semana a analizar no se correspondan con las facturas que se toman para analizar las ventas. Es decir, puede haber ingresos que correspondan a pagos atrasados por parte de los clientes.

No obstante, la empresa reconoce esto pero advierte que el monto total de ingresos no debería representar menos del 80% de las ventas (siempre dentro de las semanas que se están analizando).



## Nro. 9: Mapa de ventas por zona

**Descripción:** muestra el monto de ventas del mes anterior a la fecha actual, agrupado por zonas que se marcaran en un mapa. El marcador varía en colores: amarillo, verde o rojo, para mostrar el grado de cumplimiento del objetivo. Además, si se selecciona uno de los marcadores aparece una burbuja con un gráfico de barras que muestra el top 5 de clientes de esa zona, indicando el monto de ventas por cada uno de ellos.

**Objetivo Estratégico:** simplemente lograr diferenciar las ventas por zonas para un mejor análisis.

**Métrica:** sum(Monto facturado por zona)

**Rango de tiempo:** basado en el mes anterior a la fecha actual.

**Meta:**  ventas menores a 200 Mil;  ventas entre 200 Mil y 500 Mil; ventas mayores a  500 Mil.

**Dirección:** Maximizar

**Razón:** este indicador es interesante para el tomador de decisiones debido a que le permite comparar las ventas por zona de los clientes.



## Estructura del Datawarehouse

Debido a que el Dashboard debe nutrirse de varios temas para mostrarse (por ende, de muchas tablas de hechos) el esquema correcto de Datawarehouse en términos del problema es de constelación. Este esquema consta de 5 tablas de hechos:

- a) **Venta**, refleja las ventas donde se indica la zona, el cliente, el producto vendido, el producto asociado vendido, la fecha, la cantidad vendida del producto, la cantidad de hojas vendidas y el monto total de la venta. Los valores son agregados a nivel de fecha, zona, cliente, producto.
- b) **Pedido**, refleja los pedidos donde se indica la zona, el cliente, el producto pedido, la fecha del pedido, la fecha prometida de entrega del pedido, la cantidad pedida del producto, la cantidad de hojas, el monto por unidad y el monto total del pedido. Los valores son agregados a nivel de fecha de pedido, zona, cliente, producto.
- c) **Mov\_Stock**, solamente almacena los movimientos de alta de un producto y la cantidad de hojas; ya que las bajas no interesan y además se ven reflejadas por la facturación de los productos. Los datos son agregados a nivel de fecha, producto.
- d) **Producción**, refleja la actividad realizada por un empleado. Se indica el empleado, fecha, cantidad de horas productivas e improductivas, cantidad de horas extra productivas e improductivas, cantidad de horas de descanso costo horas empleadas productivas e improductivas, costo horas extra empleadas productivas e improductivas y el costo de las horas de descanso.
- e) **Ingreso**, refleja los recibos registrados para un cliente en una fecha determinada. Se indica zona, cliente, fecha y monto pagado. Los datos son agregados a nivel de fecha, zona, cliente.

## Esquemas por tema

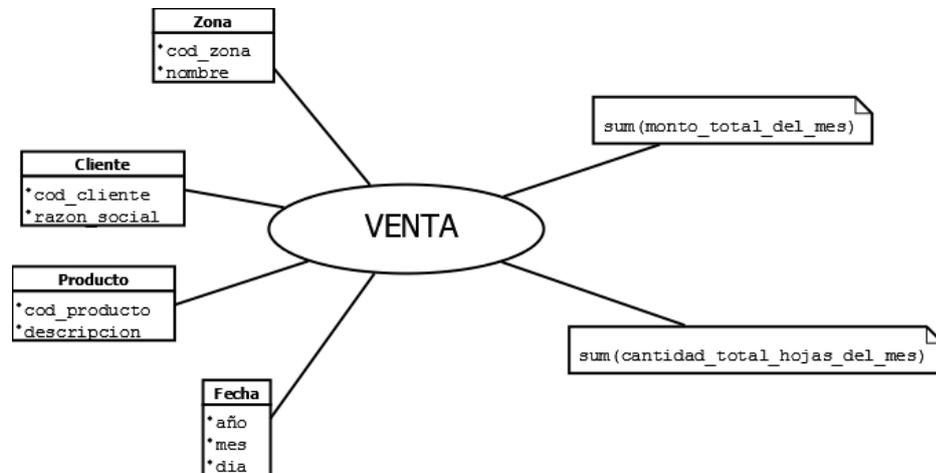
Primero se muestran los esquemas por cada tabla de hecho:

1. **Esquema conceptual**, tratando de identificar las dimensiones del hecho y cuáles son las consultas que se deben responder.
2. **Esquema lógico**, con sus respectivas dimensiones asociadas.

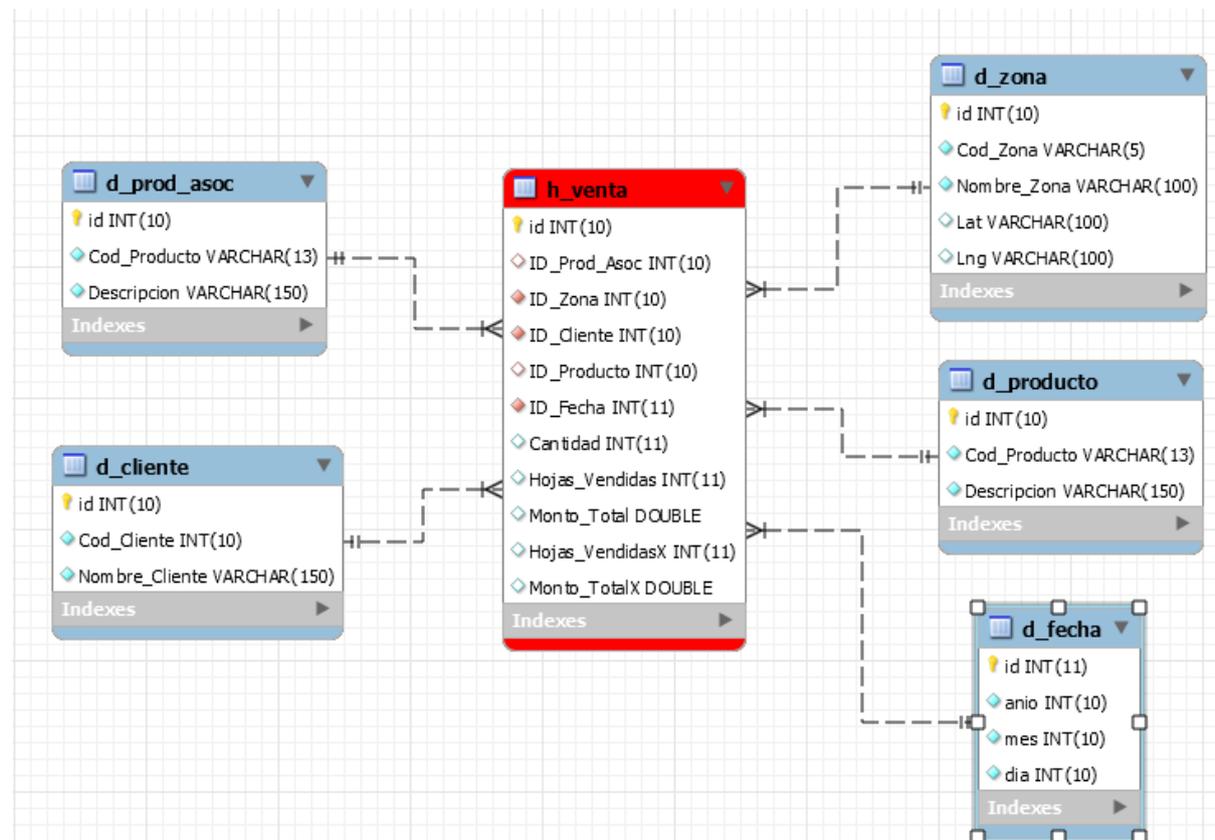
Luego, finalmente, se muestra el diagrama completo del esquema de constelación.

# Ventas

## Modelo conceptual

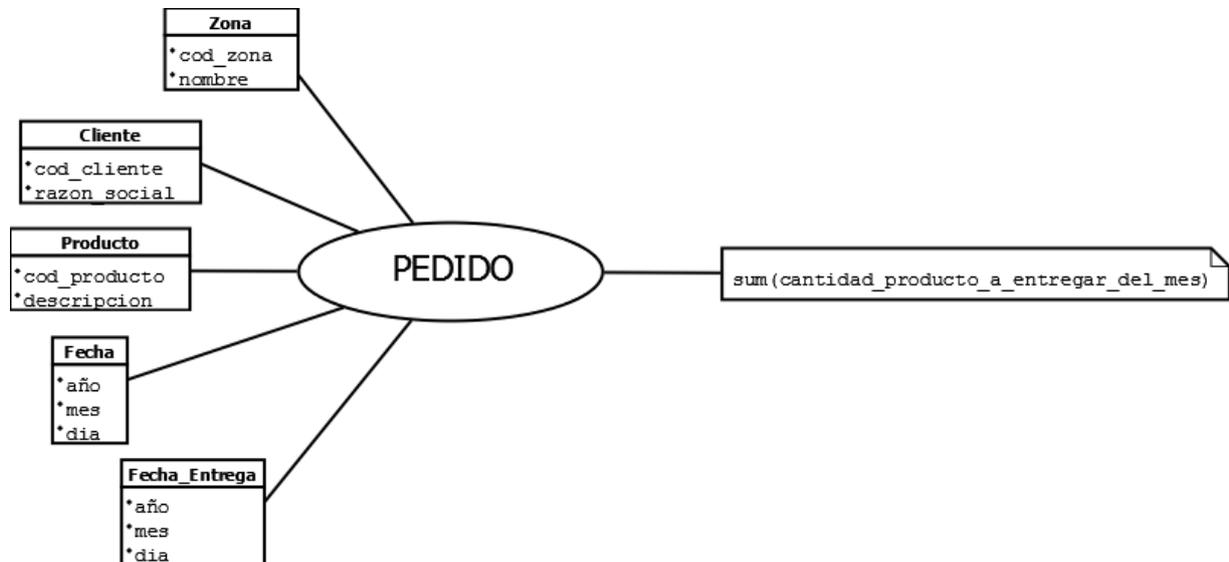


## Modelo lógico

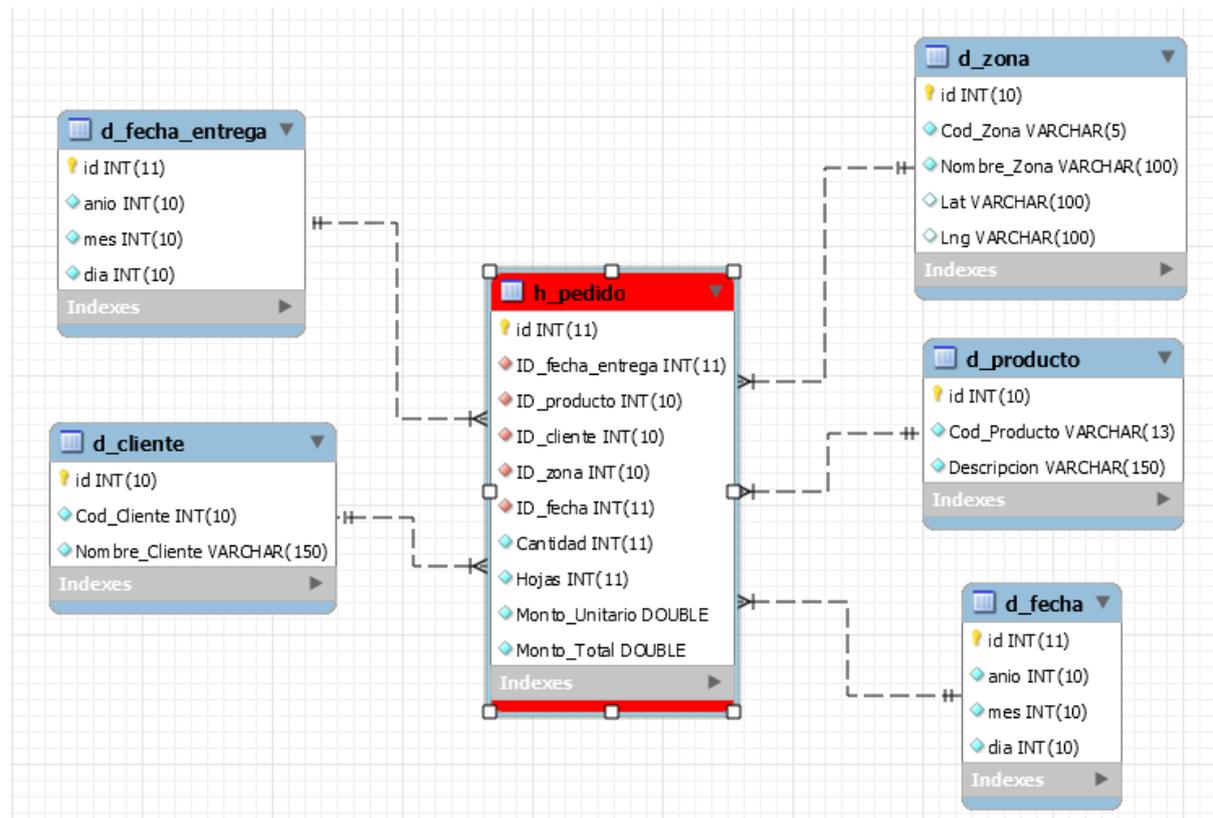


# Pedidos

## Modelo conceptual

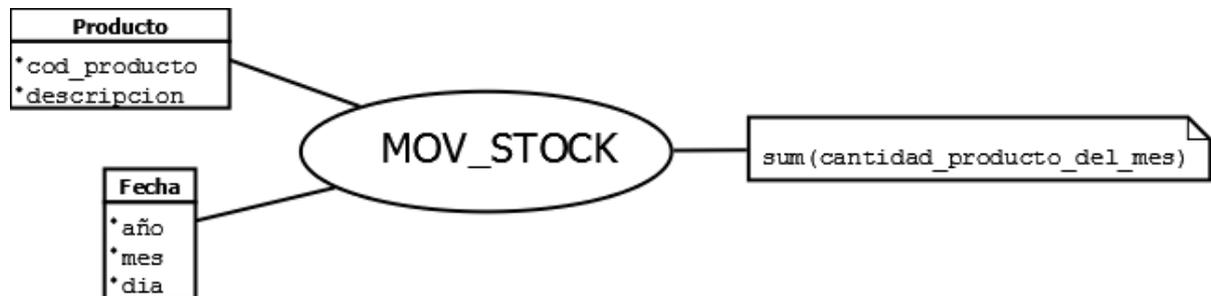


## Modelo lógico

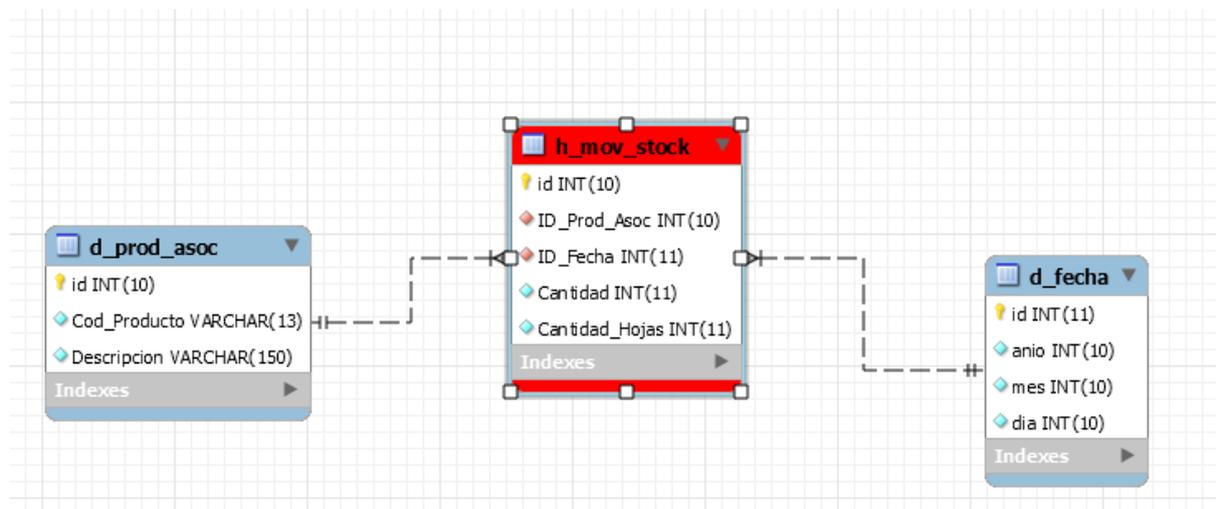


## Mov\_stock

### Modelo conceptual

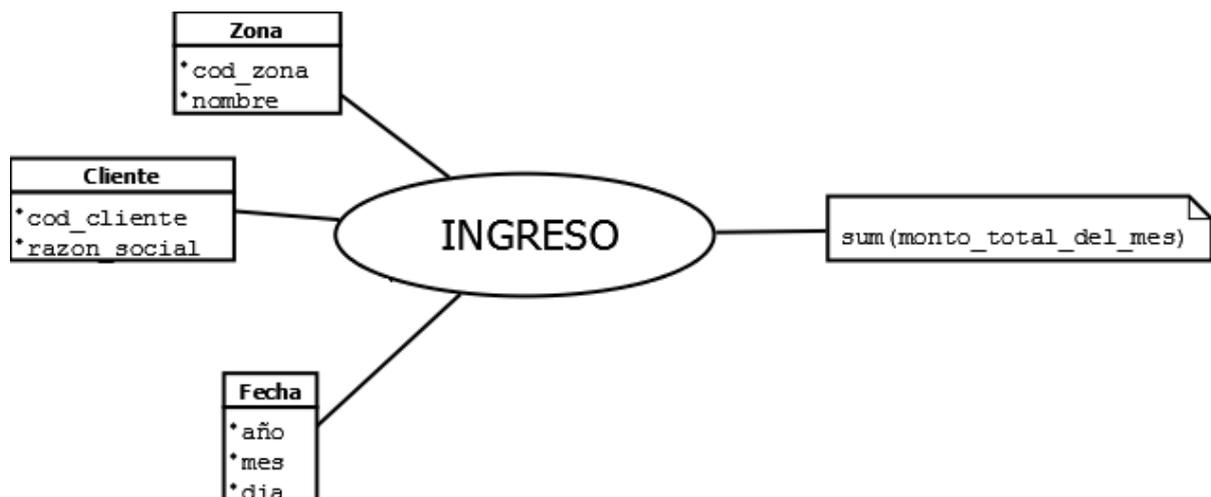


### Modelo lógico

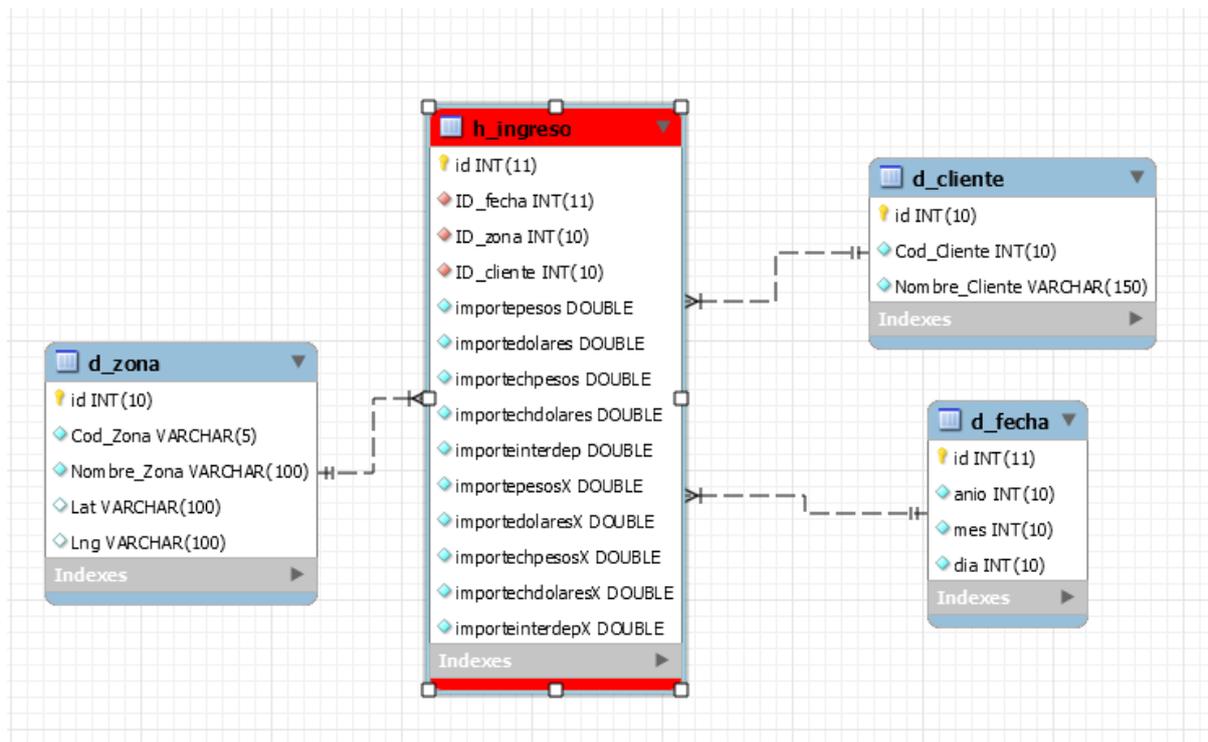


## Ingreso

### Modelo conceptual

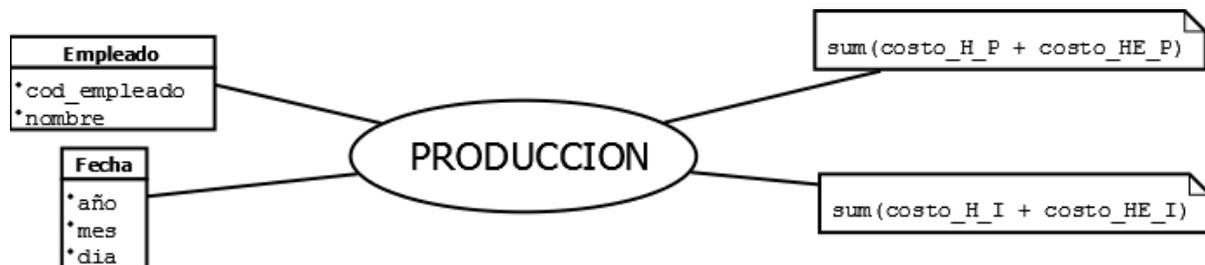


## Modelo lógico

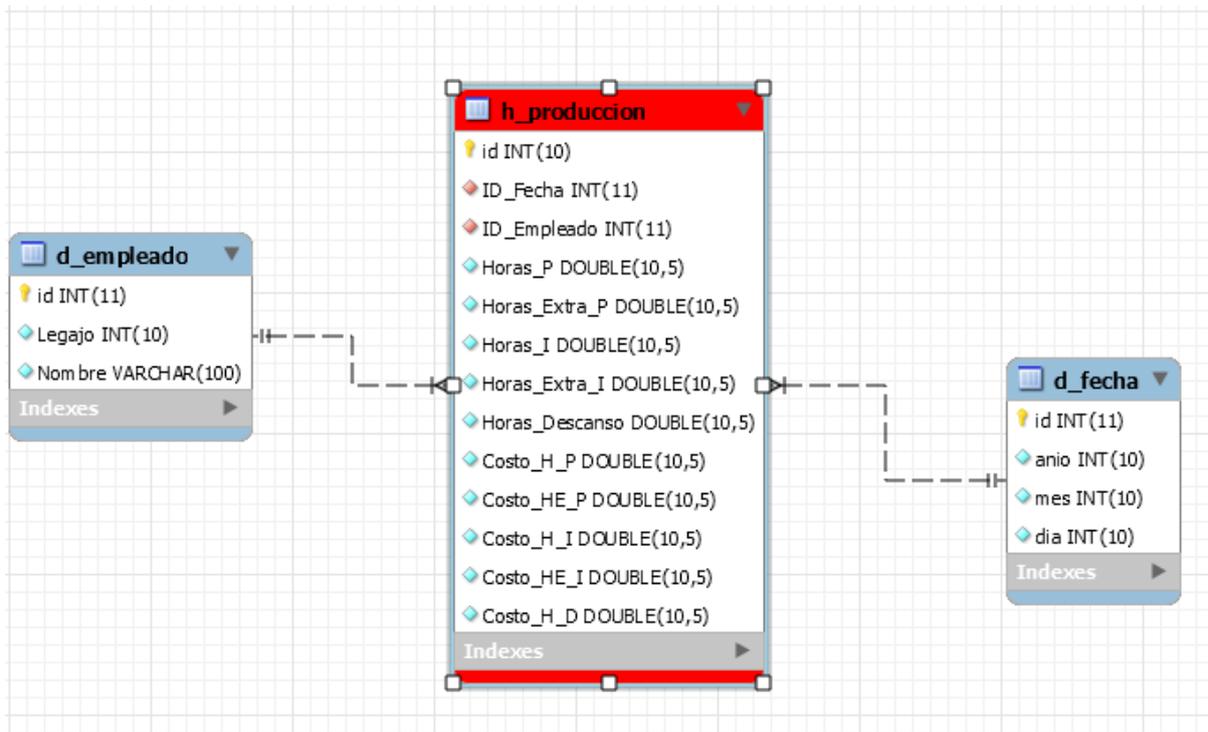


## Producción

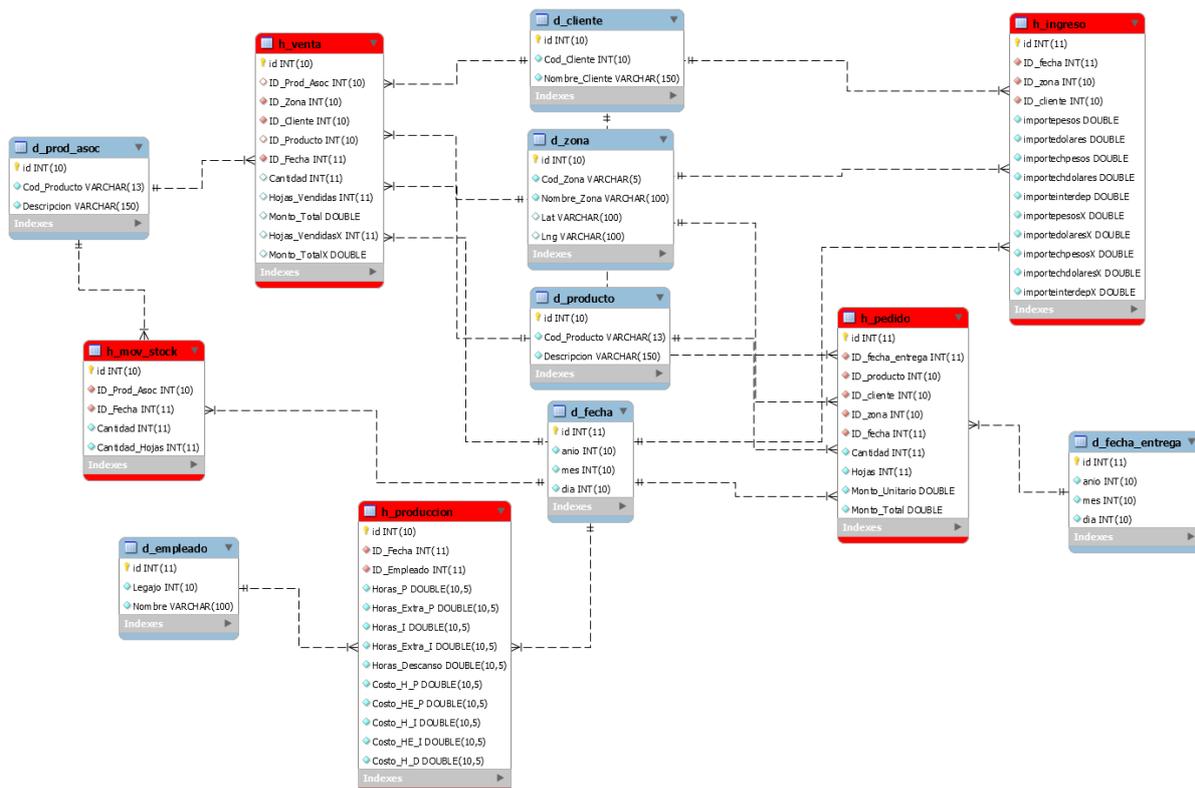
### Modelo conceptual



## Modelo lógico

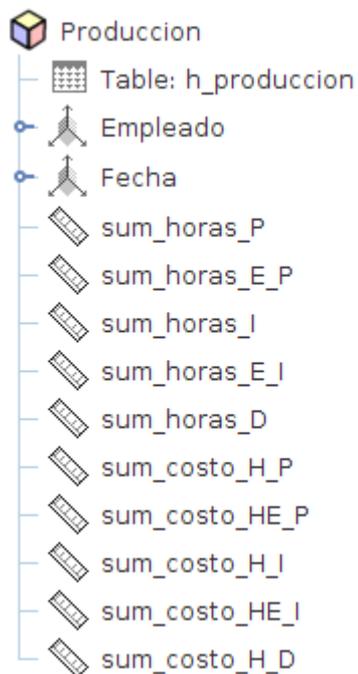


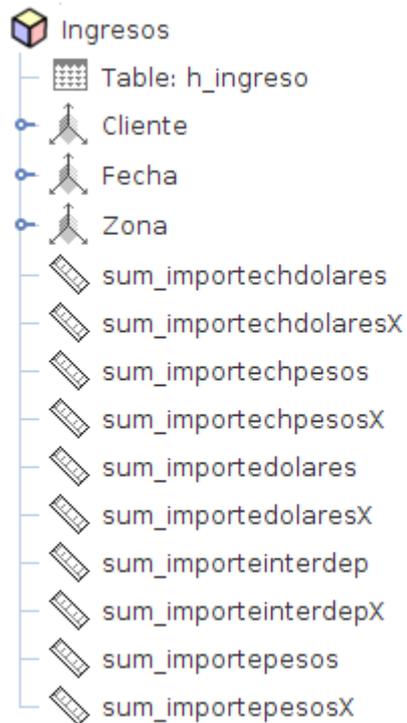
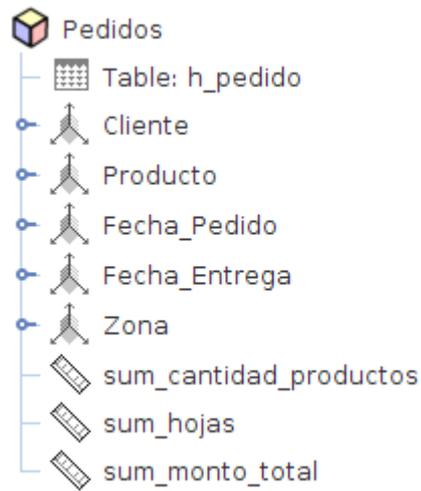
## Esquema de constelación



## Definición de cubos

Los cubos fueron definidos con la herramienta Mondrian. A continuación podemos ver los cubos definidos con sus dimensiones y sus métricas:





Una vez definidos los cubos se publicó el esquema en el servidor de Pentaho (bi-server):

The screenshot shows the Pentaho Schema Designer interface. The main window displays a schema tree on the left with nodes: Ventas, Produccion, Pedidos, Mov\_Stock, and Ingresos. The right pane shows the 'Attribute' table with columns: name, description, measuresCaption, and defaultRole.

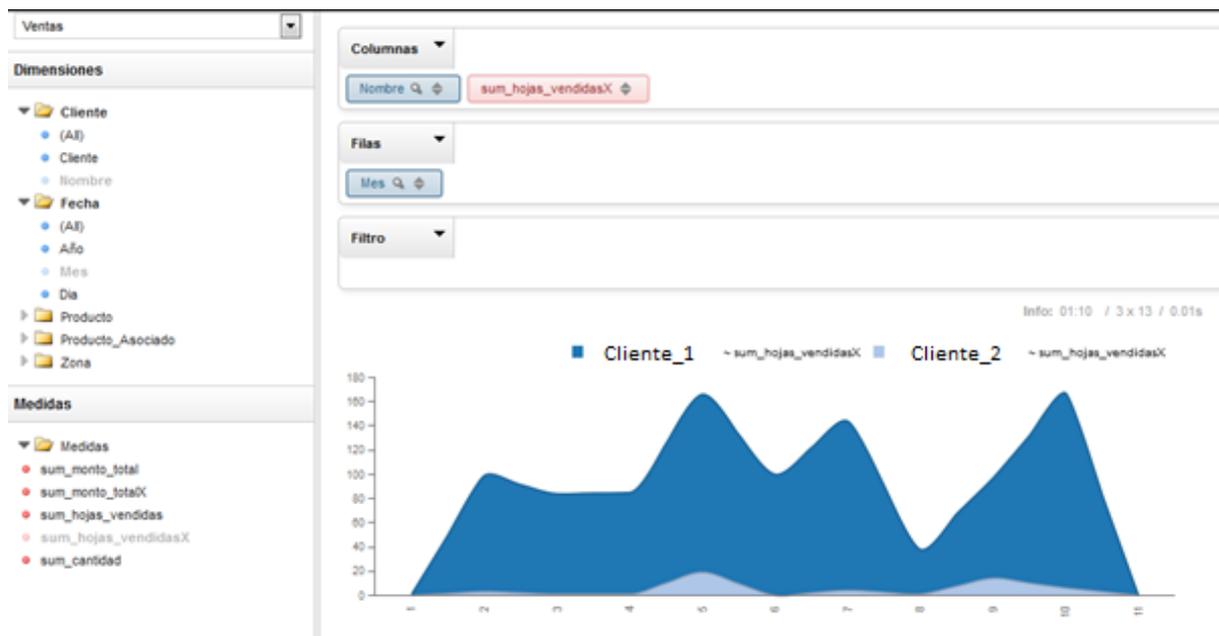
Two dialog boxes are overlaid on the main window:

- Repository Login:** A dialog for logging into the Pentaho repository. It includes fields for 'Server URL' (http://localhost:9999/pentaho/), 'Publish Password' (masked with dots), 'Pentaho Credentials' (User: joe, Password: masked), and a checked option 'Remember these Settings'. Buttons for 'OK' and 'Cancel' are at the bottom.
- Publish Schema:** A dialog for publishing the schema. It includes fields for 'Schema Name' (PavirYeah), 'Schema File' (PavirYeah.xml), and 'Location' (/MisCubos). Below these is a table showing the contents of the location:

Name	Type	Date Modified
DS_Ventas.saiku		01/25/2016 11:40 AM
prueba sumariza m...		01/23/2016 06:27 PM

At the bottom of the 'Publish Schema' dialog, there are 'Publish Settings' for the 'Pentaho or JNDI Data Source' (pavir) and a checked option 'Register XMLA Data Source'. Buttons for 'Publish' and 'Cancel' are at the bottom right.

Luego, con los cubos ya publicados podemos realizar una consulta con Saiku Analytics de la siguiente manera:



En dicha consulta se utiliza el cubo de Ventas, donde se genera un gráfico con la cantidad de hojas vendidas mes a mes del año 2015 para dos clientes específicos: Cliente\_1 y Cliente\_2



# ETL (Extracción, Transformación y Loading)

## Descripción

Este proceso se realizó con la herramienta Spoon de la Suite Pentaho. Para cada una de las tablas de hechos se siguieron los siguientes pasos:

- a) Crear una vista en la base de datos transaccional (la que usa el sistema de la empresa) para cada tema del Datawarehouse con el objetivo de traer los datos lo más resumido que sea posible en una primera instancia.
- b) Crear una transformación que actualice las dimensiones relacionadas con dicha tabla de hechos.
- c) Crear una transformación que actualice la tabla de hechos, haciendo las uniones correspondientes con las dimensiones ya actualizadas.
- d) Crear un job que ejecute a) y luego b). En el caso de algún error, se enviara un mail a la cuenta de correo andresgiordano.unlu@gmail.com.

Además, **se agregó una transformación más** que actualiza una tabla que se llama “ProductosIgnorados” que hace referencia a aquellos productos que la empresa no considera importantes a la hora de calcular las métricas y KPI’s; y que, por ende, tampoco considera validos a la hora de hacer otros tipos de análisis sobre los datos. Dicha transformación es la primera que se ejecuta.

**Existe una excepción con respecto a los pasos marcados anteriormente.** La transformación de la tabla de hechos del tema Producción se actualiza de una manera diferente. No existe una vista en la base de datos transaccional que se utilice en este proceso. La forma en que se termina actualizando esta tabla es mediante la ejecución de una script realizado en el lenguaje Java, al cual se le pasa la fecha de comienzo y de fin y realiza la consulta correspondiente a la tabla de la base de datos transaccional. Una vez realizada la consulta primero actualiza las dimensiones en la base de datos del Datawarehouse y luego actualiza la tabla de hechos.

**Para todas las transformaciones** se sigue una regla, la cual es la siguiente:

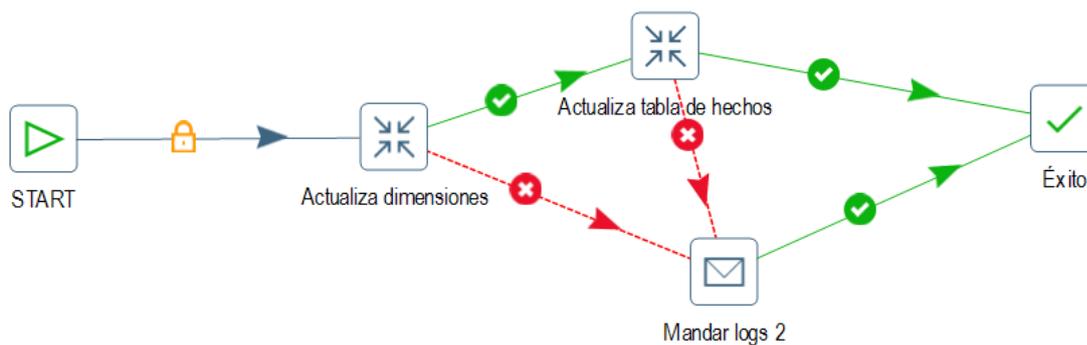
- a) Se lee una tabla que contiene las fechas en que se realizaron actualizaciones a las tablas de hechos y dimensiones. De dicha tabla se obtiene la última fecha.
- b) Esa última fecha es la que le indica a las consultas a la base de datos transaccional a partir de cuándo tiene que leer los datos.
- c) Particularmente, en la transformación que actualiza la tabla de hechos, una vez que el proceso resulto satisfactorio se actualiza (en la tabla que almacena las fechas) la última fecha procesada; la cual es el día anterior a la fecha actual.

Se planifico la ejecución de cada job para las 0:30 A.M, debido a que en ese horario ya no se realizan actualizaciones en el sistema; incluso, no se realizan desde aproximadamente las 10 PM del día anterior.

## Ejemplo

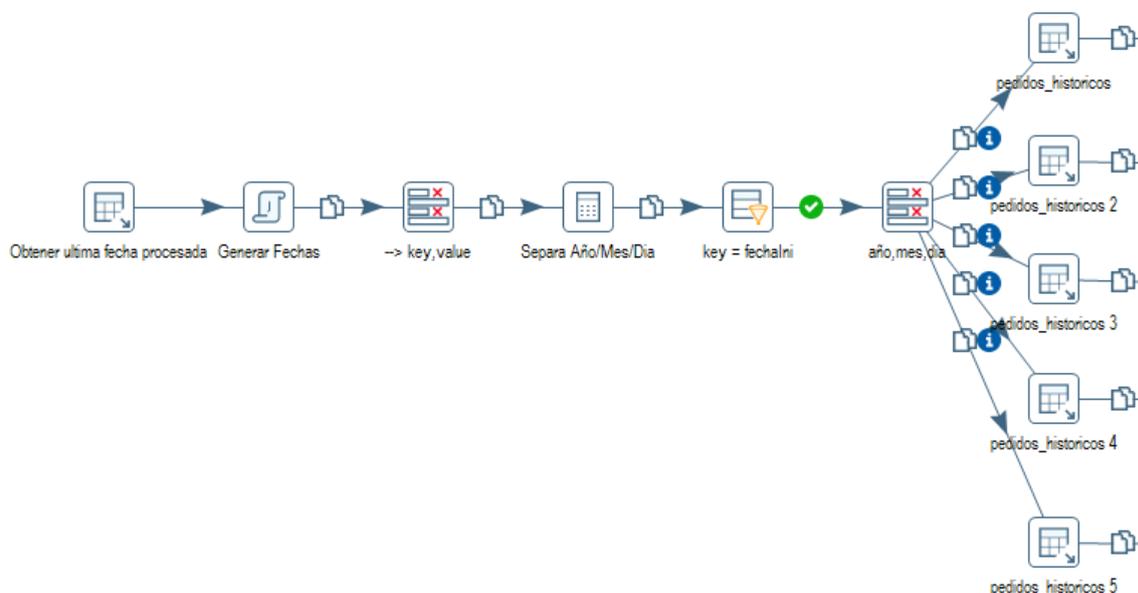
A continuación se muestra un ejemplo de la secuencia de pasos que se realizan tanto en el job como en las transformaciones del tema Pedidos.

### Job



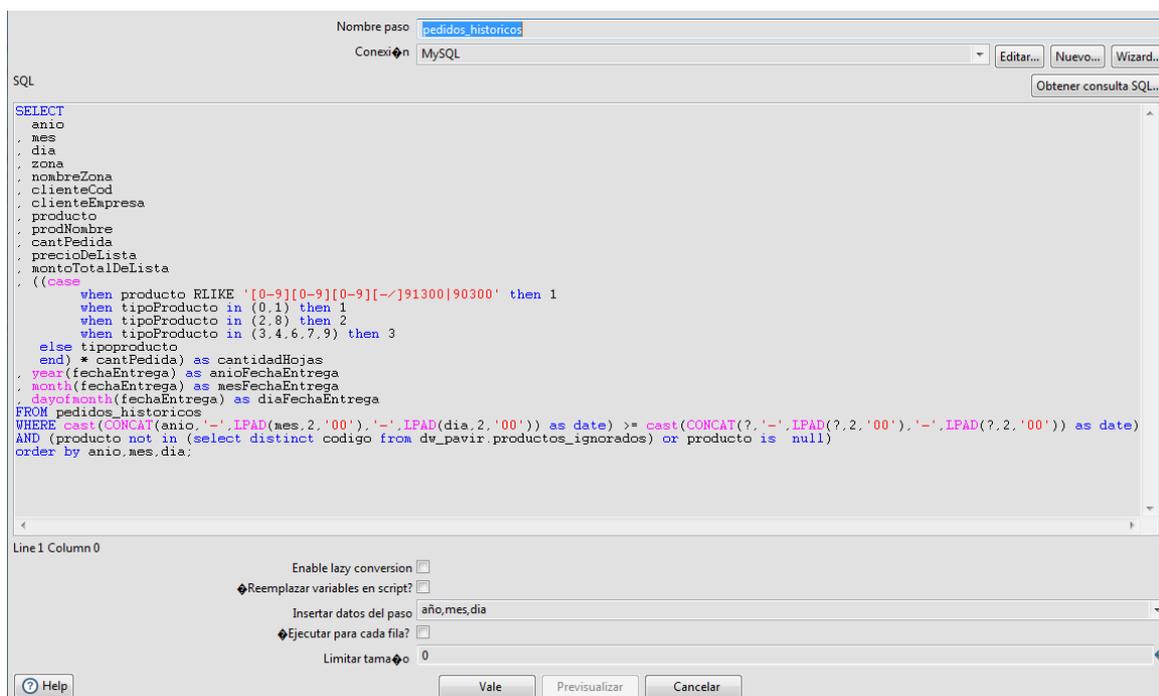
Como se puede ver en la imagen arriba, en caso de que todo marche correctamente se ejecutara primero la actualización de dimensiones, luego la actualización de la tabla de hechos y luego finaliza el job. En caso de falla se enviaran por correo los logs generados por la herramienta.

### Actualización de dimensiones



En la figura de arriba se puede ver que el primer paso es obtener la última fecha procesada como ya había mencionado. Si bien en la actualización de dimensiones no interesa, el próximo paso es generar la última fecha de actualización, esto queda así ya que se reutilizaron los mismos pasos para todas las transformaciones, y ya veremos luego que la última fecha procesada se utiliza solo en la actualización de tabla de hechos.

Luego se filtran algunas columnas, se genera el año/mes/día por separado, se aplica un filtro para que solo pase la fecha de inicio de la consulta, luego se aplica un filtro para que solo pase el año el mes y el día por separado para poder pasar estos datos como parámetro a la consulta.



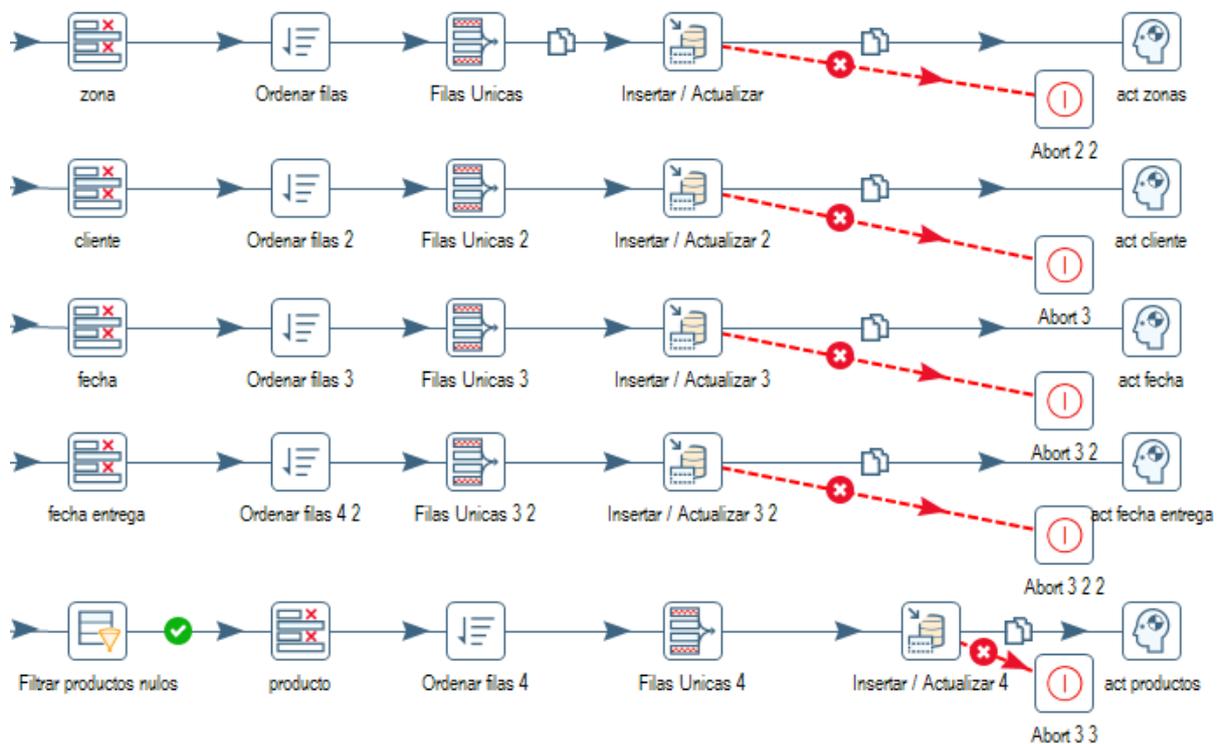
```
Nombre paso pedidos_historicos
Conexión MySQL
Editar... Nuevo... Wizard...
Obtener consulta SQL...

SQL

SELECT
  . anio
  . mes
  . dia
  . zona
  . nombreZona
  . clienteCod
  . clienteEmpresa
  . producto
  . prodNombre
  . cantPedida
  . precioDeLista
  . montoTotalDeLista
  . ((case
    when producto RLIKE '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]' then 1
    when tipoProducto in (0,1) then 1
    when tipoProducto in (2,8) then 2
    when tipoProducto in (3,4,6,7,9) then 3
    else tipoProducto
  end) * cantPedida) as cantidadHojas
  . year(fechaEntrega) as anioFechaEntrega
  . month(fechaEntrega) as mesFechaEntrega
  . dayofmonth(fechaEntrega) as diaFechaEntrega
FROM pedidos_historicos
WHERE cast(CONCAT(anio,'-',LPAD(mes,2,'00'),'-',LPAD(dia,2,'00')) as date) >= cast(CONCAT(?, '-',LPAD(?,2,'00'),'-',LPAD(?,2,'00')) as date)
AND (producto not in (select distinct codigo from dw_pavir.productos_ignorados) or producto is null)
order by anio,mes,dia;

Line1 Column0
Enable lazy conversion 
Reemplazar variables en script? 
Insertar datos del paso año,mes,dia
Ejecutar para cada fila? 
Limitar tamaño 0
Help Vale Previsualizar Cancelar
```

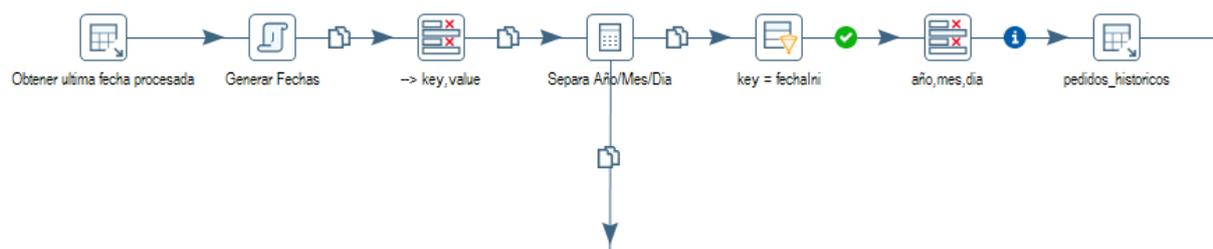
Y luego se ejecuta la consulta que puede observar en la imagen de arriba.



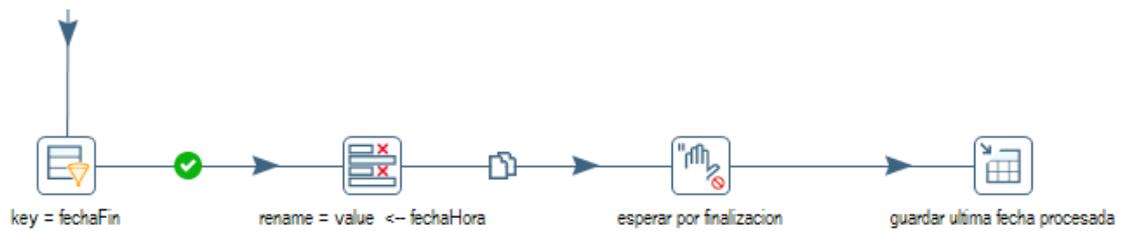
La imagen de arriba muestra la continuación de la transformación. Existe una excepción en con respecto a la actualización de los productos la cual es que no se aceptan productos nulos, por razones obvias. Si bien la consulta a la base de datos evita que lleguen productos nulos, igualmente se filtra por si acaso.

Habiendo aclarado ese paso, en general el siguiente paso es filtrar las columnas para quedarnos solo con los datos de la dimensión, luego se ordenan los datos para generar filas únicas. Luego de eso, se actualiza la dimensión.

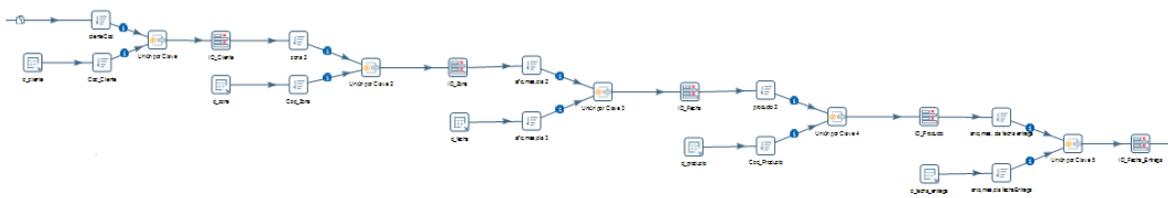
## Actualización de tabla de hechos



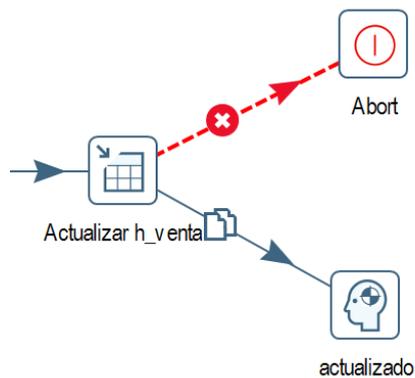
En una primera instancia, como se puede ver arriba, se ejecutan las mismas operaciones para obtener la última fecha procesada y pasar ese dato a la consulta que se realiza a la base de datos transaccional.



En el flujo que va hacia abajo se puede ver que se realiza un filtro para que solo quede la fecha de actualización (es decir, el día anterior a la fecha actual). Luego se espera la finalización con éxito de la actualización de la tabla de hechos y se actualiza la última fecha procesada.



Si seguimos el flujo a la derecha se puede ver la unión con las diferentes dimensiones para obtener la clave foránea a cada una de ellas.



Finalmente se actualiza la tabla de hechos.

# Diseño e implementación del Dashboard

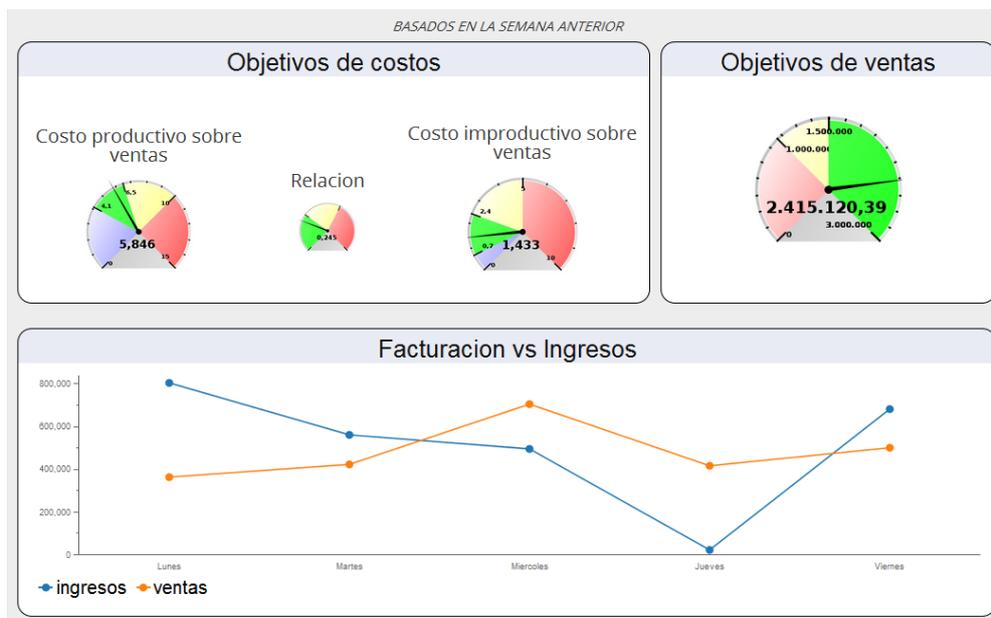
## Diseño

Debido a la cantidad de indicadores y métricas que tiene como objetivo mostrar este trabajo, es fundamental lograr un diseño que al menos agrupe los indicadores de tal manera que no genere agotamiento visual en el tomador de decisiones. Además, un buen diseño y agrupamiento de los componentes ayuda a que la información que se muestra sea más comprensible y comparable. Es decir, supongamos que muestro un gráfico sin título, en la esquina superior derecha de la pantalla que trata sobre las ventas y abajo ubico otro gráfico acerca de la totalidad de horas productivas del mes anterior. Un tablero con los indicadores dispuestos de esa manera nunca se entendería.

Tomando en cuenta lo dicho anteriormente se llegó a un diseño que agrupa los indicadores de una manera aproximada a como se agrupan los datos en el sistema de la empresa, dividiendo primero en **costos y ventas** y luego en **entregas y producción**.

## Costos y ventas

Se agruparon los indicadores de la siguiente manera:



Disponiendo primero los objetivos en cuanto a los costos y su relación, luego hacia la derecha el objetivo de ventas y abajo la evolución de facturación vs los ingresos. Todos estos indicadores muestran datos basados en la semana anterior. Se le puso un color de fondo específico haciendo alusión al rango de tiempo en que se basan los datos.

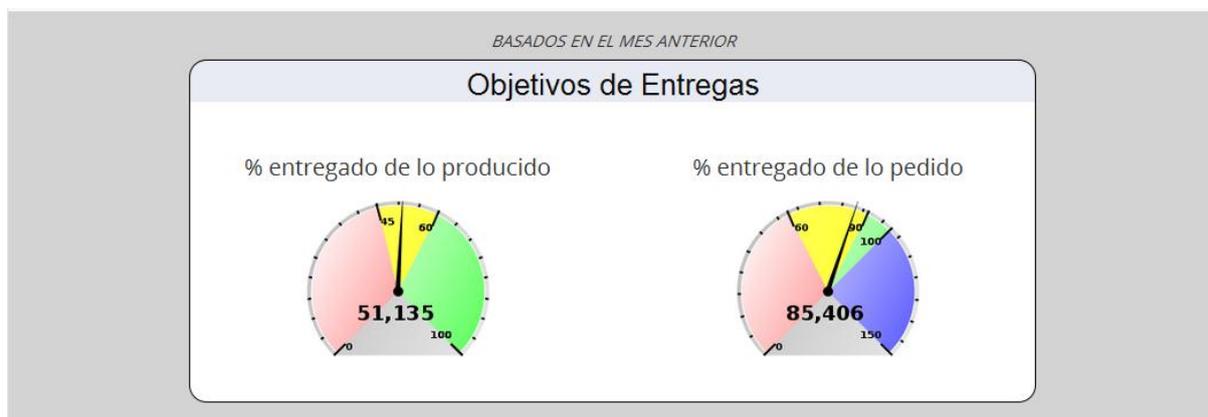
Luego, debajo de esa sección se ubica el mapa de ventas por zona:



Como se puede ver, se sigue el criterio anterior poniendo un color de fondo para indicar que ese elemento corresponde a un rango de tiempo diferente, aplicando un color más oscuro debido a que el rango abarca mayor cantidad de días.

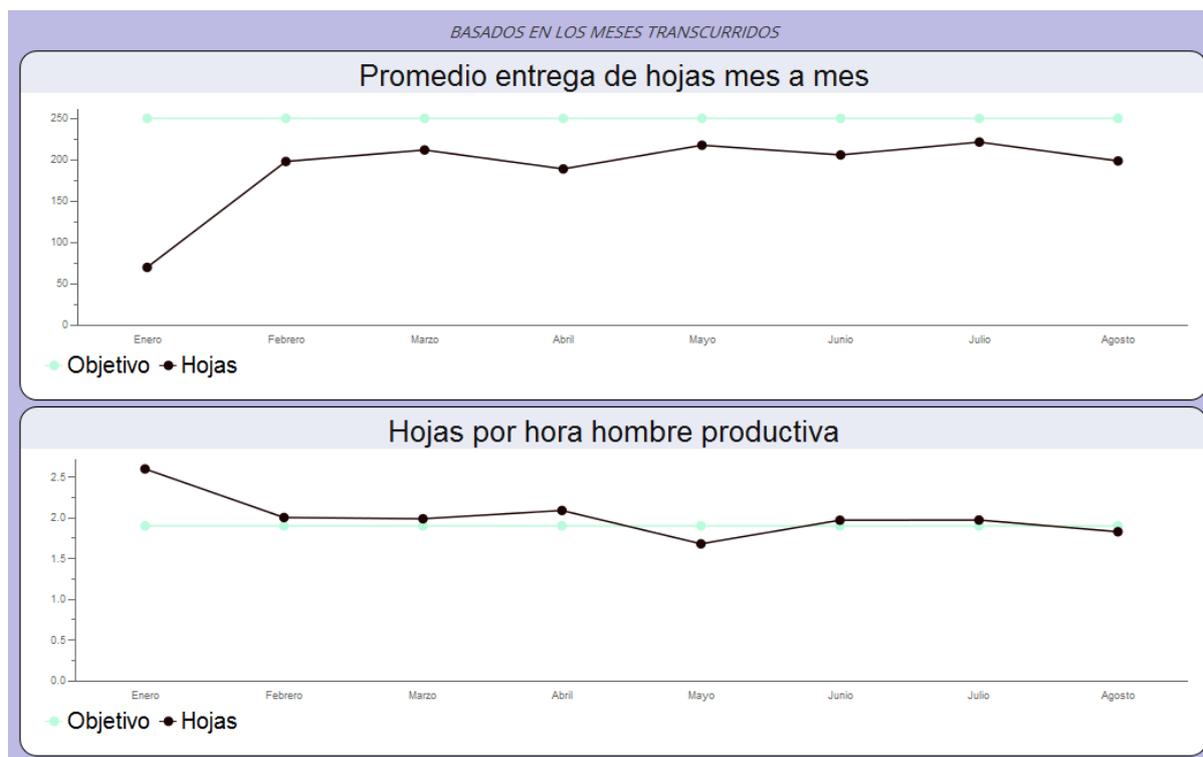
## Entregas y producción

Los indicadores fueron agrupados de la siguiente manera:



Primero, verticalmente, se dispuso una sección con los objetivos de entrega. Otra vez, se le puso un color de fondo indicando que corresponde a un rango de tiempo diferente.

Más abajo se puede ver la otra sección:



Esta agrupa en un color diferente los indicadores que muestran la evolución de la métrica a lo largo de los meses transcurridos. La decisión de agrupar estos dos indicadores es obvia por la cronología que muestran ambos y por estar estrechamente relacionada la información que muestran ambos.

## Implementación

La implementación fue realizada con la herramienta CDF (Community Dashboard Framework) de la Suite Pentaho. Dicha herramienta se basa en HTML, CSS y JavaScript para su funcionamiento.

Se reutilizaron los ejemplos que provee la Suite, tanto de la documentación como de los ejemplos concretos que provee para poder lograr mostrar los indicadores, todo lo demás es conocimiento previo de HTML, CSS y JavaScript.

## Estructura de CDF

Este framework establece una estructura de archivos necesarios para la ejecución del dashboard:

- a) Un archivo .html de base que debe contener en cierto lugar un parámetro al cual se lo referencia como {content}. A continuación se muestra el que se definió para éste trabajo:

```

114         <div class="container">
115             <span class="span-3">&nbsp;</span>
116             <div class="span-13 panelTitleText">&nbsp;</div>
117             <div class="span-8 last panelTitleImages"></div>
118
119             <div class="span-17">&nbsp;</div>
120
121
122         </div>
123
124     </div-->
125
126     <div id="content" class="webdetailsMainContent">
127         {content}
128     </div>
129
130 </div>
131 <div class="webdetailsPanelContainerRight">
132
133

```

A este documento el framework lo llama “estilo”.

- b) Una carpeta que contiene los siguiente archivos:
- Un archivo llamado index.xml el cual brinda los datos necesarios para que se muestre la carpeta del proyecto en la herramienta Saiku.

```

1  <?xml version='1.0' encoding='UTF-8' ?>
2  <index>
3      <name>%name</name>
4      <description>%description</description>
5      <icon>folder.png|solutions.png</icon>
6      <visible>true</visible>
7      <display-type>icons</display-type>
8  </index>
9

```

- Un archivo llamado index.properties el cual indica los parámetros para el archivo anterior:

```

1  # Copyright 2006 Pentaho Corporation. All rights reserved.
2  # This software was developed by Pentaho Corporation and is provided under the terms
3  # of the Mozilla Public License, Version 1.1, or any later version. You may not use
4  # this file except in compliance with the license. If you need a copy of the license,
5  # please go to http://www.mozilla.org/MPL/MPL-1.1.txt. The Original Code is the Pentaho
6  # BI Platform. The Initial Developer is Pentaho Corporation.
7  #
8  # Software distributed under the Mozilla Public License is distributed on an "AS IS"
9  # basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. Please refer to
10 # the license for the specific language governing your rights and limitations.
11 name=Pavir Dashboard
12 description=Trabajo Final BDM - Pavir Dashboard
13

```

- Un archivo .xcdf el cual tiene un formato .xml que es el que lee la herramienta Saiku para identificar que se trata de un Dashboard generado con la herramienta CDF:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <cdf>
3   <title>Pavir Dashboard</title>
4   <author>Andres</author>
5   <description>Este Dashboard muestra los indicadores de Pavir</description>
6   <icon></icon>
7   <template>template.html</template>
8   <style>mystyle</style>
9 </cdf>
10
```

En este documento se indica que archivo HTML se va a usar como “template” y que archivo se va a usar como “estilo”, éste último es el que se mencionó en el punto a).

- d. Un documento HTML que haga referencia al seleccionado como “template” en el archivo anterior. Este es el que va a reemplazar el parámetro {content} en el documento del punto a), insertando todo su contenido en lugar de {content}.

Este archivo básicamente contiene toda la estructura web definida para el dashboard. Es una opción (como se hizo en este trabajo) agregar tantas hojas de estilo como sean necesarias para darle forma a la página web resultante.

- e. Otra opción que da este framework, por el hecho de tratarse en definitiva de programación web (entendiendo ciertas cuestiones específicas del framework), es la de agregar scripts JS. En este trabajo se utiliza un script para la definición de los componentes del core de CDF.

## Ejemplo de definición de un componente

A continuación se muestra un ejemplo de definición de un componente en CDF.

- a) Como primer paso se debe definir, en algún script en el documento template, una variable que contenga:
- El nombre del objeto del dashboard
  - El tipo de componente que se quiere utilizar.
  - El objeto que contiene la definición el componente.
  - La referencia al id del tag HTML donde se cargara el objeto.

Aquí un ejemplo:

```
193 VentasProductivoDial =
194 {
195   name: "VentasProductivoDial",
196   type: "dialComponent",
197   listeners:[],
198   chartDefinition: MetaLayerCharts.definitionVentasProductivoDial,
199   htmlObject: "VentasProductivo_object",
200   executeAtStart: true
201 }
```

- b) La definición del componente. A continuación se muestra la definición correspondiente al componente que se muestra en el punto a) :

```
20  MetaLayerCharts = {
21
22
23  definitionVentasProductivoDial: {
24    width: 400,
25    height: 200,
26    chartType: "DialChart",
27    queryType: 'cda',
28    cdaFile: "bi-developers/dashboards/MySolution/Pavir.cda",
29    dataAccessId: 'Productivo_Ventas',
30    backgroundColor: "#FFFFFF",
31    needleColor: "#000000",
32    tickColor: "#000000",
33
34    colors: ["#0000FF", "#00FF00", "#FFFF00", "#FF0000"],
35    domainTickFontSize: 20,
36    intervals: [4.1, 6.5, 10.0, 15.0],
37    includeLegend: true
38  },
```

Como se puede ver en la imagen arriba, se especifican características especiales del componente como el ancho, el alto, el color de fondo. Al ser un gráfico tipo velocímetro se puede definir el color de la aguja, de las diferentes marcas, el tamaño de la letra, los colores a utilizar en los rangos y los mismos rangos que va a mostrar el gráfico.

También se debe definir la fuente de los datos, es decir, de donde se va a nutrir el componente para mostrar los datos. En la mayoría de los casos se puede utilizar la referencia a un archivo .cda. Este archivo contiene el query y la conexión a la base de datos que se utilizará. CDF utiliza este mecanismo para evitar ataques a la base de datos; CDF prohíbe el uso de sql directo desde un Script JS por defecto. Entonces el archivo CDA, el cual solo se puede acceder desde el servidor, contiene la definición del query, que conexión se utilizara y define un id para dicha consulta. En la definición del componente se utiliza el path al archivo CDA y el id de la consulta para identificar el query que se utilizará para obtener los datos con que se va a cargar el componente.