



UTMACH

UNIDAD ACADÉMICA DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

IMPLEMENTACIÓN DE UN MÓDULO DE SOPORTE DE DECISIONES
PARA AGRICULTURA UTILIZANDO BIGDATA E INTELIGENCIA DE
NEGOCIOS INTEGRADO AL SISTEMA IOTMACH

JIMÉNEZ RUIZ FERNANDA ELIZABETH

MACHALA
2016



UTMACH

UNIDAD ACADÉMICA DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

IMPLEMENTACIÓN DE UN MÓDULO DE SOPORTE DE
DECISIONES PARA AGRICULTURA UTILIZANDO BIGDATA E
INTELIGENCIA DE NEGOCIOS INTEGRADO AL SISTEMA
IOTMACH

JIMÉNEZ RUIZ FERNANDA ELIZABETH

MACHALA
2016



UTMACH

UNIDAD ACADÉMICA DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

TRABAJO DE TITULACIÓN
PROPUESTAS TECNOLÓGICAS

IMPLEMENTACIÓN DE UN MÓDULO DE SOPORTE DE DECISIONES PARA
AGRICULTURA UTILIZANDO BIGDATA E INTELIGENCIA DE NEGOCIOS
INTEGRADO AL SISTEMA IOTMACH

ESPINOZA VEGA ANGEL ANDRÉS
INGENIERO DE SISTEMAS

JIMÉNEZ RUIZ FERNANDA ELIZABETH
INGENIERA DE SISTEMAS

MAZÓN OLIVO BERTHA EUGENIA

Machala, 20 de octubre de 2016

MACHALA
2016

Nota de aceptación:

Quienes suscriben MAZÓN OLIVO BERTHA EUGENIA, ZEA ORDOÑEZ MARIUXI PAOLA, REDROVAN CASTILLO FAUSTO FABIAN y JUMBO CASTILLO FREDDY ANIBAL, en nuestra condición de evaluadores del trabajo de titulación denominado IMPLEMENTACIÓN DE UN MÓDULO DE SOPORTE DE DECISIONES PARA AGRICULTURA UTILIZANDO BIGDATA E INTELIGENCIA DE NEGOCIOS INTEGRADO AL SISTEMA IOTMACH, hacemos constar que luego de haber revisado el manuscrito del precitado trabajo, consideramos que reúne las condiciones académicas para continuar con la fase de evaluación correspondiente.




MAZÓN OLIVO BERTHA EUGENIA
0603100512
TUTOR



ZEA ORDONEZ MARIUXI PAOLA
0702801598
ESPECIALISTA 1



REDROVAN CASTILLO FAUSTO FABIAN
0702739228
ESPECIALISTA 2



JUMBO CASTILLO FREDDY ANIBAL
0704167949
ESPECIALISTA 3

Machala, 20 de octubre de 2016

Urkund Analysis Result

Analysed Document: ESPINOZA VEGA ANGEL ANDRES - JIMENEZ RUIZ FERNANDA
ELIZABETH.docx (D21637499)
Submitted: 2016-09-07 06:37:00
Submitted By: aaespinozavega@gmail.com
Significance: 2 %

Sources included in the report:

TESIS KARINA CHONG ESCOBAR.docx (D14094802)
SANCHEZ_PABLO_S4A.docx (D14420458)
http://ucumari.unc.edu.ar/repositorio/archivos/Tutorial_de_Saiku-Pentaho.pdf
<http://www.slideshare.net/cuate18/diseo-de-un-datamart>
<https://elentornodehadoop.wordpress.com/>
<http://slideplayer.es/slide/4335865/>

Instances where selected sources appear:

6

CLÁUSULA DE CESIÓN DE DERECHO DE PUBLICACIÓN EN EL REPOSITORIO DIGITAL INSTITUCIONAL

El que suscribe, ESPINOZA VEGA ANGEL ANDRÉS, en calidad de autor del siguiente trabajo escrito titulado IMPLEMENTACIÓN DE UN MÓDULO DE SOPORTE DE DECISIONES PARA AGRICULTURA UTILIZANDO BIGDATA E INTELIGENCIA DE NEGOCIOS INTEGRADO AL SISTEMA IOTMACH, otorga a la Universidad Técnica de Machala, de forma gratuita y no exclusiva, los derechos de reproducción, distribución y comunicación pública de la obra, que constituye un trabajo de autoría propia, sobre la cual tiene potestad para otorgar los derechos contenidos en esta licencia.

El autor declara que el contenido que se publicará es de carácter académico y se enmarca en las disposiciones definidas por la Universidad Técnica de Machala.

Se autoriza a transformar la obra, únicamente cuando sea necesario, y a realizar las adaptaciones pertinentes para permitir su preservación, distribución y publicación en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

El autor como garante de la autoría de la obra y en relación a la misma, declara que la universidad se encuentra libre de todo tipo de responsabilidad sobre el contenido de la obra y que él asume la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

Aceptando esta licencia, se cede a la Universidad Técnica de Machala el derecho exclusivo de archivar, reproducir, convertir, comunicar y/o distribuir la obra mundialmente en formato electrónico y digital a través de su Repositorio Digital Institucional, siempre y cuando no se lo haga para obtener beneficio económico.

Machala, 20 de octubre de 2016



ESPINOZA VEGA ANGEL ANDRÉS
0704395508

CLÁUSULA DE CESIÓN DE DERECHO DE PUBLICACIÓN EN EL REPOSITORIO DIGITAL INSTITUCIONAL

El que suscribe, JIMÉNEZ RUIZ FERNANDA ELIZABETH, en calidad de autor del siguiente trabajo escrito titulado IMPLEMENTACIÓN DE UN MÓDULO DE SOPORTE DE DECISIONES PARA AGRICULTURA UTILIZANDO BIGDATA E INTELIGENCIA DE NEGOCIOS INTEGRADO AL SISTEMA IOTMACH, otorga a la Universidad Técnica de Machala, de forma gratuita y no exclusiva, los derechos de reproducción, distribución y comunicación pública de la obra, que constituye un trabajo de autoría propia, sobre la cual tiene potestad para otorgar los derechos contenidos en esta licencia.

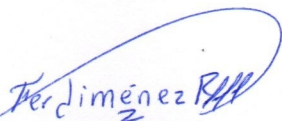
El autor declara que el contenido que se publicará es de carácter académico y se enmarca en las disposiciones definidas por la Universidad Técnica de Machala.

Se autoriza a transformar la obra, únicamente cuando sea necesario, y a realizar las adaptaciones pertinentes para permitir su preservación, distribución y publicación en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

El autor como garante de la autoría de la obra y en relación a la misma, declara que la universidad se encuentra libre de todo tipo de responsabilidad sobre el contenido de la obra y que él asume la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

Aceptando esta licencia, se cede a la Universidad Técnica de Machala el derecho exclusivo de archivar, reproducir, convertir, comunicar y/o distribuir la obra mundialmente en formato electrónico y digital a través de su Repositorio Digital Institucional, siempre y cuando no se lo haga para obtener beneficio económico.

Machala, 20 de octubre de 2016



JIMÉNEZ RUIZ FERNANDA ELIZABETH
1104810518

AGRADECIMIENTO

A nuestros queridos padres que con su entereza y amor incondicional han sabido guiarnos por el sendero del optimismo, de la fe y de la férrea voluntad que vuestras aspiraciones jamás se vean truncadas, gracias padres los amamos.

A nuestros hermanos y familiares por habernos dado su apoyo y fortaleza en el transcurso de la formación profesional.

A Dios por darnos fuerzas y fe para concluir con éxito esta propuesta tecnológica.

A la Universidad Técnica de Machala, a sus directivos y en especial a nuestros queridos tutores Ing. Bertha Mazón e Ing. Dixys Hernández, por sus conocimientos impartidos, por saber educar y ante todo formar, bases firmes para nuestros futuros aprendizajes.

A nuestra tutora Ing. Sist. Bertha Eugenia Mazón Olivo, Mg. por su paciencia, motivación, esperanzas y por difundir optimismo en el alma de estos jóvenes adolescentes que sin lugar a duda asumiremos el reto de ser útil a la patria, a la sociedad y familiares.

Y a todas las personas que de una u otra forma contribuyeron en cada una de las fases de desarrollo de esta propuesta tecnológica.

Fernanda y Angel

DEDICATORIA

A mis amados padres Ramiro Alberto y María Inés, quienes son mi pilar fundamental, por sus consejos, comprensión, amor, ayuda en los momentos difíciles, y por ayudarme con los recursos necesarios para estudiar. Me han dado inspiración de superación para desarrollarme como profesional y tener mis propios valores, principios, carácter, empeño, perseverancia y coraje para conseguir mis objetivos.

A mí querido hermano Wellington, por sus consejos y apoyo incondicional. A Dios quién supo guiarme por el camino del bien, darme fuerzas para seguir adelante y no desmayar en los problemas que se presentaban, enseñándome a enfrentar las adversidades sin perder nunca la fe ni desfallecer en el intento.

Fernanda

Dedico este trabajo de titulación a mis padres Angel Espinoza y Cecilia Vega, a mi hermana Valeria Espinoza, que con la bendición de Dios me apoyan en todas mis decisiones y esfuerzos por ser un profesional.

Angel

RESUMEN

La presente propuesta tecnológica, tiene como objetivo principal implementar un módulo de soporte de decisiones para agricultura utilizando Big Data e Inteligencia de Negocios (BI) integrado al sistema IoT Mach. El presente trabajo se sustenta en el análisis e investigación de las técnicas de Inteligencia de Negocios (BI), combinado con el Internet de las cosas aplicadas a la agricultura de precisión, se propone mediante procesos de integración de datos de distintas fuentes, realizar el procesamiento analítico en línea (OLAP), donde los múltiples datos que se utilizan para el diseño del Data Warehouse (DW) provienen de la red de sensores inalámbricos (WSN) ubicados en parcelas y del sistema transaccional de monitoreo y control. Los grandes volúmenes de datos, que se generan a través de la red de sensores inalámbricos (WSN) llegan a gran velocidad en tiempo real a las bases de datos NoSQL (Mongo Data Base - MongoDB) y los datos obtenidos mediante el software de monitoreo y control son almacenados en la base de datos relacional (PostgreSQL). Una vez que los datos pasan por un proceso de extracción, transformación y carga (ETL) mediante Pentaho Data Integration (Spoon), se crea el almacenamiento de datos, elaborando un Big Data de alto rendimiento en Oracle donde se almacenará información relacionada al último año de producción y Apache Hive como Big Data de bajo costo que incluirá los datos históricos de cultivo. Para el diseño, análisis y construcción del sistema de Inteligencia de Negocios (BI), se utilizó la Metodología Hefesto, luego de esta fase, se diseñan los cubos (OLAP), identificando los Indicadores clave de desempeño (KPIs) (Cultivo: Superficie de producción; Cosecha: Superficie de producción cosechada, número de unidades producidas, rendimiento, costo de producción, precio de venta y utilidad neta; Plagas y Enfermedades: Número de plantas afectadas, número de hectáreas afectadas y porcentaje de afectación; Haciendas y Parcelas: Área neta dedicada a cultivo, número de hectáreas cultivadas y porcentaje de superficie dedicada a cultivo; y Medidas de Sensores: Promedio de medida de sensores, valor mínimo, valor máximo, desviación estándar y varianza). Mediante el proceso de Extracción, transformación y carga (ETL) se carga la información a las bases de datos antes definidas según el año de producción, posteriormente se realiza un proceso de virtualización de los datos, con el fin de obtener una combinación satisfaciendo las necesidades de negocio por medio de Denodo Data Platform (VDP), este proceso de virtualización se lo realiza combinando los datos mediante Joins, creando vistas para su posterior lectura, luego

estas vistas son leídas en Pentaho, para posteriormente hacer un Análisis de cubos con los visores Pivot4J y Saiku. La interacción de todos los cubos conforma el Sistema de Soporte de Decisiones (DSS). También es posible visualizar un panel de control (Dashboard) que integra varios Indicadores claves de desempeño (KPI's) de cubos en el Sistema de Información Ejecutiva (EIS).

Como resultado de esta propuesta se crea una aplicación que brinda al agrónomo o empresario, la posibilidad de realizar un análisis descriptivo de sus cultivos, cosechas, cantidad de plagas o enfermedades, información de parcelas y haciendas y lecturas proporcionadas por la red de sensores inalámbricos, y así toda esa información almacenada sea de gran importancia para la toma de decisiones orientadas a la mejora de la efectividad del cultivo y cosecha.

Palabras Claves: big data; agricultura de precisión, inteligencia de negocios; iot, data warehouse; data mart; metodología hefesto.

ABSTRACT

This technological proposal, whose main objective is to implement a decision support module for agriculture using Big Data and Business Intelligence (BI) integrated into IOTMACH system. This work is based on the analysis and research techniques Business Intelligence (BI), combined with the Internet of things applied to precision agriculture, it is proposed through processes of integration of data from different sources, perform processing online analytical (OLAP), where multiple data used to design the data Warehouse (DW) come from the wireless sensor network (WSN) located on plots and transactional system monitoring and control. Large volumes of data generated through the wireless sensor network (WSN) reach high speeds in real time to NoSQL databases (Mongo Data Base - MongoDB) time and data obtained by monitoring software and Control are stored in the relational database (PosgreSQL). Once the data passes through a process of extraction, transformation and loading (ETL) by Pentaho Data Integration (Spoon), data storage is created, developing a Big Data high performance Oracle where information is stored related to last year production and Apache Hive as Big data inexpensive data including historical culture. Surface: For the design, analysis and system construction Business Intelligence (BI), the Hephaestus Methodology, after this phase, the cubes (OLAP) is designed, identifying key performance indicators (KPIs) (culture was used production Harvest: Surface production harvested, number of units produced, yield, production cost, selling price and net income; Pests and diseases: number of affected plants, number of hectares affected and percentage of affectation; Farms and Plots: area net dedicated to cultivation, number of hectares and percentage of area under cultivation; sensors and Measurements: average measurement sensors, minimum value, maximum value, standard deviation and variance). Through the process of extraction, transformation and loading (ETL) loads the information to the databases previously defined by year of production, then a process of virtualization of data is performed in order to obtain a combination satisfying the needs business through Denodo data Platform (VDP), this process of virtualization is done by combining data with Joins, creating views for later reading, then these views are read in Pentaho, later to analyze cubes with viewers Pivot4J and Saiku. The interaction of all cubes forms the Decision Support System (DSS). You can also display a control panel (Dashboard) that integrates several key performance indicators (KPI's) cubes in the Executive Information System (EIS).

As a result of this proposal an application that provides the agronomist or entrepreneur, the possibility of a descriptive analysis of their crops, crops, amount of pests or diseases, information fields and farms and readings provided by the wireless sensor network is created, and so all that information stored is of great importance for making oriented improving the effectiveness of cultivation and harvesting decisions.

Keywords: big data; business intelligence; iot; data warehouse; data mart; hefesto methodology.

CONTENIDO

	pág.
RESUMEN	1
ABSTRACT	5
CONTENIDO	7
LISTA DE FIGURAS	9
LISTA DE TABLAS	12
GLOSARIO	14
INTRODUCCIÓN	17
1. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS	19
1.1 Ámbito de Aplicación: descripción del contexto y hechos de interés	19
1.2 Establecimiento de requerimientos	20
1.3 Justificación del requerimiento a satisfacer.....	21
2. DESARROLLO DEL PROTOTIPO	23
2.1. Definición del prototipo tecnológico	23
2.2. Fundamentación teórica del prototipo	24
2.2.1 Internet de las cosas (IoT).	25
2.2.2 Red de sensores inalámbricos.	26
2.2.3 Agricultura.	26
2.2.4 Agricultura de precisión.	27
2.2.5 Big data.	27
2.2.6 Data warehouse.....	28
2.2.7 Apache hadoop.....	29
2.2.8 Apache hive.	29
2.2.9 Data mart.....	30
2.2.10 Inteligencia de negocios.	31
2.2.11 Pentaho.....	31
2.2.12 MongoDB.	32
2.2.13 Postgresql.	32
2.2.14 Comparación de rendimiento.....	32
2.3 Objetivos del prototipo	33
2.3.1 Objetivo General	33
2.3.2 Objetivos Específicos.....	33
2.4. Diseño del prototipo	34
2.4.1. Diseño del Data Warehouse.....	36

2.4.2.	Diseño de aplicaciones BI	57
2.4.3	Herramientas utilizadas.	68
2.6	Ejecución del prototipo.....	72
3.	EVALUACIÓN DEL PROTOTIPO	82
3.1	Plan de evaluación.....	82
3.1.1	Pruebas de Ejecución de Procesos de Carga (Unitarias).....	83
3.1.2	Pruebas de Usabilidad.....	83
3.2	Resultados de la evaluación.....	84
3.2.1	Resultados ejecución de Procesos de Carga (Unitarias)	84
3.2.2	Resultados pruebas de usabilidad	97
3.3	Conclusiones.....	99
3.4	Recomendaciones	100
	REFERENCIAS BIBLIOGRÁFICAS.....	101
	ÍNDICE COMPLEMENTARIO	104
	ANEXOS	105

LISTA DE FIGURAS

pág.

Figura 1 Arquitectura Data Warehouse.....	20
Figura 2 Arquitectura del Data Warehouse Agricultura de Precisión.....	24
Figura 3 Modelo BigData.....	28
Figura 4 Elementos básicos de un Data warehouse.....	29
Figura 5 Proceso de generación de un Data mart desde una bóveda de datos.....	30
Figura 6 Comparación de Inserción.....	33
Figura 7 Diseño del prototipo.....	35
Figura 8 Modelo Conceptual Data Mart 1.....	43
Figura 9 Modelo Conceptual DM2.....	44
Figura 10 Modelo Conceptual DM3.....	45
Figura 11 Modelo Conceptual DM4.....	46
Figura 12 Modelo Conceptual DM5.....	47
Figura 13 Modelo Lógico del DM 1.....	48
Figura 14 Modelo Lógico del DM 2.....	49
Figura 15 Modelo Lógico del DM 3.....	50
Figura 16 Modelo Lógico del DM 4.....	51
Figura 17 Modelo Lógico del DM 5.....	52
Figura 18 Proceso Extracción, Transformación y Carga al Data Warehouse 1.....	54
Figura 19 Proceso Extracción, Transformación y Carga al Data Warehouse 2.....	54
Figura 20 Proceso Extracción, Transformación y Carga al Data Warehouse 3.....	55
Figura 21 Proceso Extracción, Transformación y Carga al Data Warehouse 4.....	55
Figura 22 Vista combinación lecturas sensores.....	56
Figura 23 Proceso Extracción, Transformación y Carga al Data Warehouse 5.....	56
Figura 24 Vista tabla de hechos Data Mart 1: cultivos.....	58
Figura 25 Vista tabla de hechos Data Mart 2: Cosechas.....	59
Figura 26 Vista tabla de hechos Data Mart 3: Plagas y Enfermedades.....	60
Figura 27 Vista tabla de hechos Data Mart 4: Haciendas y Parcelas.....	61
Figura 28 Vista tabla de hechos Data Mart 5: Medida de Sensores.....	61
Figura 29 Model Editor Cultivo.....	63
Figura 30 Model Editor Cosecha.....	63
Figura 31 Model Editor Plagas y Enfermedades.....	64
Figura 32 Model Editor Haciendas y Parcelas.....	64
Figura 33 Model Editor Medida de Sensores.....	65
Figura 34 Diseño login.....	66
Figura 35 Diseño Data Mart 1.....	67
Figura 36 Herramientas utilizadas.....	68
Figura 37 Conexión BD Destino.....	70
Figura 38 Actualización Denodo.....	70
Figura 39 Conexión Denodo a Hive.....	71
Figura 40 Vista Cultivo.....	71
Figura 41 Conexión Pentaho a Denodo.....	72

Figura 42 Sistema Autenticación IOTMACH	72
Figura 43 Página principal pentaho	73
Figura 44 Log In	73
Figura 45 Dashboard	74
Figura 46 Panel de selección dashboard.....	74
Figura 47 Dashboard Cultivo	75
Figura 48 Opciones Dashboard.....	76
Figura 49 Bar chart Superficie de producción	77
Figura 50 Selector Categoría.....	77
Figura 51 Selector Especie y Año.....	78
Figura 52 Superficie de producción por categoría	78
Figura 53 Gounge Ivy Component.....	79
Figura 54 Superficie de producción por tipo de riego.....	79
Figura 55 Indicadores Mes, Semestre, Trimestre	80
Figura 56 Superficie de producción por tipo de suelo	80
Figura 57 Indicadores Tipo de Suelo por Mes, Semestre, Trimestre.....	81
Figura 58 Resultado Procesos de carga.....	85
Figura 59 Resultado carga de datos Cultivo - Hive	86
Figura 60 Resultado tiempo carga de datos Cultivo - Hive	86
Figura 61 Resultado carga de datos Cosechas - Hive.....	87
Figura 62 Resultado tiempo carga de datos Cosechas - Hive.....	88
Figura 63 Resultado carga de datos Plagas y Enfermedades - Hive	88
Figura 64 Resultado tiempo carga de datos Plagas y Enfermedades - Hive.....	89
Figura 65 Resultado carga de datos Hacindas y Parcelas - Hive	89
Figura 66 Resultado tiempo carga de datos Haciendas y Parcelas - Hive	90
Figura 67 Resultado carga de datos Medida de Sensores - Hive	90
Figura 68 Resultado tiempo carga de datos Medida Sensores - Hive.....	91
Figura 69 Resultado carga de datos Cultivo - Oracle	92
Figura 70 Resultado tiempo carga de datos Cultivo - Oracle.....	92
Figura 71 Resultado carga de datos Cosechas - Oracle	93
Figura 72 Resultado tiempo carga de datos Cosechas - Oracle	94
Figura 73 Resultado carga de datos Plagas y Enfermedades - Oracle.....	94
Figura 74 Resultado tiempo carga de datos Plagas y Enfermedades - Oracle.....	95
Figura 75 Resultado carga de datos Hacindas y Parcelas - Oracle.....	95
Figura 76 Resultado tiempo carga de datos Haciendas y Parcelas - Oracle.....	96
Figura 77 Resultado carga de datos Medida de Sensores - Oracle	96
Figura 78 Resultado tiempo carga de datos Medida Sensores - Oracle	97
Figura 79 Resultado Pruebas de Usabilidad.....	98

LISTA DE CUADROS

pág.

Cuadro 1 Procesos de la metodología Hefesto.....	36
Cuadro 2 Análisis de indicadores y perspectivas DM1	40
Cuadro 3 Análisis de indicadores y perspectivas DM2	41
Cuadro 4 Análisis de indicadores y perspectivas DM3	41
Cuadro 5 Análisis de indicadores y perspectivas DM4	42
Cuadro 6 Análisis de indicadores y perspectivas DM5	42
Cuadro 7 Matriz Pruebas ETL	83
Cuadro 8 Matriz Pruebas Usabilidad	84

LISTA DE TABLAS

pág.

Tabla 1 Contenidos Teóricos.....	25
Tabla 2 Matriz Resultados Pruebas ETL	84
Tabla 3 Carga de datos Cultivo - Hive.....	85
Tabla 4 Carga de datos Cosechas - Hive	87
Tabla 5 Carga de datos Plagas y Enfermedades - Hive.....	88
Tabla 6 Carga de datos Haciendas y Parcelas - Hive.....	89
Tabla 7 Carga de datos Medida de Sensores - Hive.....	90
Tabla 8 Carga de datos Cultivo - Oracle.....	91
Tabla 9 Carga de datos Cosechas - Oracle	93
Tabla 10 Carga de datos Plagas y Enfermedades - Oracle	94
Tabla 11 Carga de datos Haciendas y Parcelas - Oracle.....	95
Tabla 12 Carga de datos Medida de Sensores - Oracle.....	96
Tabla 13 Matriz resultados pruebas de usabilidad	97

LISTA DE ANEXOS

	pág.
Anexo A	105
Anexo B	106
Anexo C	107
Anexo D	108
Anexo E	109
Anexo F	110
Anexo G	111
Anexo H	112
Anexo I	113
Anexo J	114
Anexo K	115
Anexo L	116
Anexo M	117
Anexo N	121
Anexo O	123
Anexo P	126
Anexo Q	143
Anexo R	151
Anexo S	152
Anexo T	153
Anexo U	154
Anexo V	159
Anexo W	160
Anexo X	162
Anexo Y	165

GLOSARIO

API: La Interfaz de Programación de Aplicaciones o API (por sus siglas en inglés, *Application Programming Interface*), es un conjunto de servicios o funciones que las aplicaciones deben seguir con la intención de poder comunicarse entre ellas.

Base de Datos: Es un conjunto de datos relacionados entre sí, que se encuentran almacenados en un disco que permite que una aplicación o sistema informático pueda acceder a ellos de forma directa.

BI: Acrónimo de *Business Intelligence* o Inteligencia de Negocios. Son estrategias y herramientas que tienen como objetivo generar conocimiento a partir de datos e información en busca de optimizar el proceso de toma de decisiones de una organización.

Cargar: Función propia del proceso ETL. Tiene como objetivo escribir los resultados de los procesos anteriores (Extracción y Transformación) en una base de datos o Datawarehouse.

Cubo: Es una base de datos multidimensional que almacena la información de forma jerarquizada y estandarizada, permitiendo que el análisis de los datos sea más rápido y eficiente.

Dashboard: Un Dashboard (Tablero de comando integrado) es una herramienta de visualización de datos que indica el estado actual de los KPI (Indicadores clave de rendimiento) de una empresa. Las empresas recurren a distintas herramientas especializadas en el manejo de Dashboard para tener un control visual del estado de la organización.

DataMart: Es un conjunto de hechos y datos que se encuentran alineados y organizados y sirven como apoyo en la toma de decisiones de un área o departamento.

Data warehouse: Es una tecnología que agrega datos estructurados desde diversas fuentes de datos, de manera que pueda comparar y analizar dicha información generando inteligencia de negocios. Los Data warehouse se diferencian de las bases de datos estándar en que los primeros están diseñados para dar una visión de largo alcance de los datos a través del tiempo, mientras que las bases de datos estándares

mantiene una estructura estricta de los datos, permitiendo su rápida actualización en tiempo real.

Dimensión: En el contexto de los Data warehouse, se conoce como dimensión a una colección de información de referencia sobre un indicador medible. Aquí, los eventos son también llamados “hechos”. Las dimensiones buscan categorizar y describir esos “hechos” de forma que sirvan de apoyo para dar respuesta a las preguntas de negocio. Consisten en la agrupación de elementos con similares características, por ejemplo, producto, proveedor, línea, clientes, servicios, etc.

DIM: Es un modelo que se emplea para el manejo de las tablas de dimensión, alojadas en el Data warehouse, y que servirán para el diseño de las tablas de “hecho”. Las tablas de dimensión poseen una clave principal que al llegar a la tabla de “hechos” se convierte en un componente de la clave compuesta de dicha tabla.

DSS: Un Sistema de Soporte de Decisiones o DSS (por sus siglas en inglés, *Decision Support System*) es cualquier sistema informático que permite realizar un análisis a los indicadores de un negocio para la toma de decisiones.

ETL: El proceso para Extraer, Transformar y Cargar o ETL (por sus siglas en inglés, *Extract, Transform and Load*) permite a las organizaciones extraer datos desde diversas fuentes, para darles un nuevo formato, limpiarlos y volverlos a cargar en una nueva base de datos o Data warehouse para realizar un análisis de apoyo para la toma de decisiones.

Extraer: Función propia del proceso ETL y tiene como finalidad leer datos de una base de datos cuyo origen es conocido desde donde extrae un subconjunto de datos específico.

IDE: Un entorno de desarrollo integrado o IDE (por sus siglas en inglés, *Integrated Development Environment*) es un entorno de trabajo que posee una interfaz de usuario (GUI) equipada con un conjunto de herramientas que ha sido diseñada para ayudar a un programador en la construcción de una aplicación de software.

JDBC: Java Database Connectivity es una API que sirve para la conexión de programas escritos en Java con bases de datos independientes. JDBC utiliza un pequeño programa “puente” para realizar las conexiones.

Metadata: Los metadatos son datos que describen de manera resumida la información básica de otros datos, haciendo más fácil la búsqueda de los mismos. Por ejemplo,

autor, fecha de creación y fecha de modificación son los metadatos de un documento que contiene más datos. Estos metadatos hacen que la búsqueda de los documentos, en este caso, sea más rápida.

Minería de Datos: Minería de datos o *Data Mining*, es una técnica muy utilizada en distintas áreas de investigación, puesto que se encarga de la búsqueda automática de patrones luego de efectuar un análisis a grandes almacenes de datos. La minería de datos emplea algoritmos matemáticos muy sofisticados para segmentar la información y poder predecir eventos futuros.

ODBC: Open Database Connectivity es una interfaz estándar de programación de aplicaciones para acceder a una base de datos. Gracias al uso de ODBC se puede realizar conexiones a varias bases de datos. El principal proveedor y promotor de este módulo de soporte para la programación es Microsoft.

OLAP: El Procesamiento Analítico en Línea o más conocido como OLAP (por sus siglas en inglés, *Online Analytical Processing*) permite que un usuario pueda extraer y visualizar los datos desde distintos puntos de vista con mucha facilidad y de forma selectiva. Los datos OLAP se guardan en una base de datos multidimensional.

Performance Management: La administración de desempeño es la actividad que permite efectuar monitoreos para la toma de decisiones a partir de datos que son fundamentales en una organización.

PYMES: Es el nombre que se le da a las pequeñas o medianas empresas que poseen características que las distinguen de las demás; aunque suelen presentar ciertos límites de financiamiento prefijados por los estados o regiones en las que se sitúan.

Sistema Operativo: Es un programa o software que permite al hardware de la computadora comunicarse y operar con el software de la misma.

Transformar: Función propia del proceso ETL. Trabaja directamente con los datos extraídos, para convertir esos datos en los deseados.

INTRODUCCIÓN

La agricultura de precisión (Precision agriculture PA) nació en la década de los ochenta en los Estados Unidos ante la necesidad de mejorar la producción agrícola de ese país. Es una de las técnicas que en la actualidad se usan para mejorar el proceso de producción agrícola basado en la observación[1], la medida y la actuación de grandes volúmenes de datos recogidos a través de sistemas de información geográficos y redes de sensores inteligentes interconectados entre sí (Internet de las cosas) ubicados en las parcelas.

La ingeniería de sistemas y electrónica ofrecen a usuarios la posibilidad de incrementar la producción y calidad de productos, incorporando sensores inteligentes capaces de procesar y transmitir información inalámbricamente a una base de datos, para su posterior análisis estadístico, usando técnicas de inteligencia de negocios o BI para la toma de decisiones importantes en alta gerencia y de esta manera aumentar la producción a costos de implementación elevados recuperables a mediano o largo plazo.

El creciente desarrollo de los mercados y la exigente competitividad entre empresas agrícolas Ecuatorianas, ha llevado a que las compañías modernicen su capacidad de producción. Es por esto que se propone una solución innovadora Big Data, para la recolección de información referente a sus cultivos, cosechas, plagas y enfermedades, parcelas y haciendas, valores mínimos y máximos de temperatura, humedad, etc. Para así lograr un cultivo óptimo, garantizando el uso eficiente de recursos como: la cantidad de agua o fertilizante a usar en las plantaciones, representando un ahorro significativo de dinero por parte de los sectores que los implementen.

Para llegar a esta solución, se requiere la extracción, transformación y carga (ETL) de los datos obtenidos mediante la red de sensores inalámbricos y el sistema transaccional que proporciona toda la información referente a la empresa, áreas, parcelas, cultivos, cosechas, etc, que son almacenados en estructuras Big Data, para su posterior análisis de Inteligencia de negocios que se centra en la recopilación de información necesaria para implementar nuevas y novedosas estrategias de mercado analizando la información que se encuentran en cubos OLAP en áreas específicas que justifican la toma de decisiones de las empresas.

Aunque su uso en la actualidad es a nivel de países industrializados hemos determinado realizar la implementación de un módulo de soporte de decisiones para agricultura utilizando Big Data e Inteligencia de Negocios BI integrado al sistema IoT Mach.

Este documento de propuesta tecnológica se ha estructurado en tres capítulos los cuales se describen a continuación:

El Primer capítulo se centra en el desarrollo de las necesidades y requerimientos para la implantación correcta del prototipo a desarrollar.

En el Segundo capítulo se presenta las distintas partes que conforman el prototipo como la definición de objetivos y la fundamentación teórica. También se establecen el análisis, desarrollo, implantación y ejecución del proyecto.

En el Tercer capítulo se presenta el plan de evaluación, los resultados de la evaluación y finalmente las conclusiones y recomendaciones del prototipo terminado.

1. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS

1.1 **Ámbito de Aplicación: descripción del contexto y hechos de interés**

La diversidad de necesidades hace que los sistemas de información existentes, sean heterogéneo y variados en cuanto a plataformas, bases de datos y lenguajes de programación[2]. Internet de las cosas (IoT) es un invento tecnológico que está influyendo en el contexto actual y futuro. La idea de IoT se refiere a la creación de una red de objetos que se comunica con otros, la integración de sensores embebidos, RFID, GPRS, computadoras, actuadores, teléfonos móviles, etc [3]. IoT Mach integra tecnología (IoT) para el intercambio de información y el proceso de objetos de acuerdo a las tareas definidas, como sensores de humedad y actuadores de riego, con capacidades de interconexión entre sí por medio de una Red de Sensores Inalámbricos (WSN), (WSN) es un sistema que consiste en una colección de nodos y una estación base. Un nodo se compone de un procesador, la memoria, los sensores, radio y una batería. Una estación de base recibe la información y procesa los datos recibidos por el nodo[4]. IoT Mach propone un software de monitoreo y control ya que las aplicaciones en agricultura son aún incipientes. Las condiciones de los cultivos se deben supervisar cuidadosamente para producir una mejor cosecha. La presencia de sensores y actuadores son esenciales para controlar y vigilar los factores ambientales, etc. Por lo tanto, (WSN) garantiza monitorización en tiempo real[4]. Estos datos son almacenados en MongoDB el cual tiene un esquema flexible y cuenta con colecciones[5].

Toda esta información obtenida mediante la red de sensores inalámbricos y los datos procesados mediante la aplicación de monitoreo y control de agricultura de precisión, no podrían ser almacenados en estructuras de datos convencionales por tal razón IoT Mach implementará BigData sobre Hadoop y Oracle. Grandes volúmenes de datos agrícolas se recogen en forma de tipo estructurado y no estructurado. Los dispositivos de detección homogéneos y heterogéneos y las nuevas tecnologías son esenciales para obtener los diversos tipos de datos[6]. Hadoop es un marco de software de código abierto que soporta aplicaciones distribuidas de datos intensivos, autorizado debajo Apache. Dispone de dos capas; la primera es la capa de almacenamiento de datos y la segunda es la capa de procesamiento de datos. Hadoop tiene su propio sistema de archivo llamado sistema de archivos distribuidos Hadoop (HDFS) y el algoritmo de procesamiento de datos llamada MapReduce[7]. Cuando los datos llegan

a un ciclo, es necesario eliminar los datos antiguos para cumplir con los nuevos datos para almacenar, ya que afectaría el rendimiento de toda la base de datos.[8]

1.2 Establecimiento de requerimientos

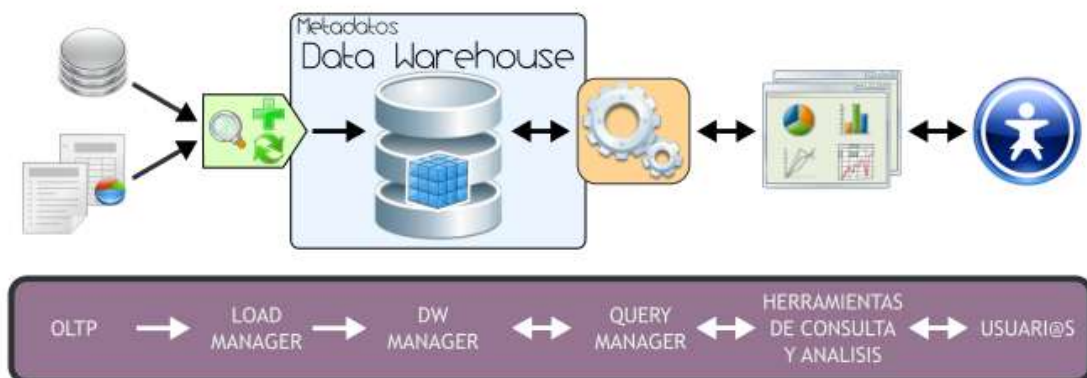
Los sistemas transaccionales generan muchos datos, pero ninguno llega a ser estadística, ni realizar una integración de medidas de los indicadores de rendimiento de la producción agrícola. Entonces el problema que se pretende resolver, es la generación de información estadística descriptiva que ayude en la toma de decisiones a través de un sistema de soporte de decisiones (DSS) y un Sistema de Información Ejecutiva (EIS).

Elaborar una estructura Big Data de bajo costo que almacenará los datos históricos de las producciones agrícolas y medidas de los sensores ubicados en las parcelas. A su vez una estructura Big Data de alto rendimiento que incluirá la información del último año de producción.

Los datos serán extraídos, transformados, cargados y enviados para su análisis estadístico en minería de datos y su tratamiento en línea en inteligencia de negocios, finalmente alta gerencia requiere de estos resultados a través de reportes para la toma de decisiones teniendo como más importantes:

- Cantidad de cosecha por hectárea.
- Especie de mayor producción.
- Costos de producción, etc.

Figura 1 Arquitectura Data Warehouse



Fuente: [9]

1.3 Justificación del requerimiento a satisfacer

El almacenamiento de datos Big Data y la utilización de algoritmos capaces de identificar patrones importantes de las variables más representativas dentro de las empresas para establecer cuáles son las mejores decisiones que se puede tomar con el análisis de estos datos. Entonces, el problema del presente trabajo de investigación es: ¿Cómo implementar un módulo de soporte de decisiones para agricultura utilizando Big Data e inteligencia de negocios integrado al sistema IoT Mach? Para la solución se requiere que los datos sean obtenidos a partir de la transferencia inalámbrica de sensores (WSN) colocados en puntos estratégicos en las parcelas, estos sistemas embebidos envían información de variables como la humedad del suelo, acidez, temperatura, etc. Los datos proporcionados por (WSN) son transferidos a una base de datos MongoDB. A su vez los datos generados por el sistema transacción de monitoreo y control, son almacenados en una base de datos estructurada PostgreSQL, que contiene información de la empresa, hacienda, parcela, tipo de suelo, tipo de riego, etc. Los datos depurados son transferidos a los algoritmos de análisis en línea (Inteligencia de Negocios) y al tratamiento estadístico (minería de datos). Desarrollando finalmente los reportes concurrentes a las necesidades de información que serán utilizadas para justificar cualquier decisión de la empresa.

Las plataformas de grandes datos no son sólo características, también son tecnologías. No tiene que ser de la más nueva y más potente. A veces lo único que se necesita es una nueva configuración o acercamiento a una vieja. Todo depende de las necesidades de la organización o persona que desea utilizar la plataforma[10]. Hadoop es la plataforma que se adapta a los requerimientos, permitiendo usar Apache Hive como estructura Big Data de bajo costo y Oracle como estructura Big Data de alto rendimiento.

El sistema de soporte de decisiones (DSS) y el Sistema de Información Ejecutiva (EIS), serán implementados bajo herramientas Open Souce como Pentaho y sus Plugins: CDE, Saiku Analytcs, Pivot4J, JPivote.

La metodología implementada en el desarrollo del presente trabajo es Hefesto aplicada a Inteligencia de Negocios, debido a que presenta muchas ventajas por que se fundamenta en una amplia investigación en comparación a metodologías existentes

como la Kimball. Además, que los objetivos son fáciles de entender, usa modelos conceptuales lógicos y cuando se culmina una fase los resultados obtenidos se convierte en punto de partida para continuar su desarrollo.

Los beneficiarios de esta investigación son los estudiantes de ingeniería de sistemas de la UTMACH por que el presente documento proporciona las bases fundamentales para la utilización la Inteligencia de Negocios y la minería de datos en el campo de la producción agrícola de especies, la cual puede servir de guía para desarrollar futuras implementaciones en diferentes áreas como la medicina, sitios Web, análisis financiero, etc. Sin olvidar que otro beneficiario son las personas encargadas de la administración y cuidado de las parcelas, dando como resultado final un mejoramiento en la calidad de producción de las diferentes especies a través de los reportes generados por esta implementación.

2. DESARROLLO DEL PROTOTIPO

2.1. Definición del prototipo tecnológico

La estructura fundamental del proyecto consta de los siguientes bloques:

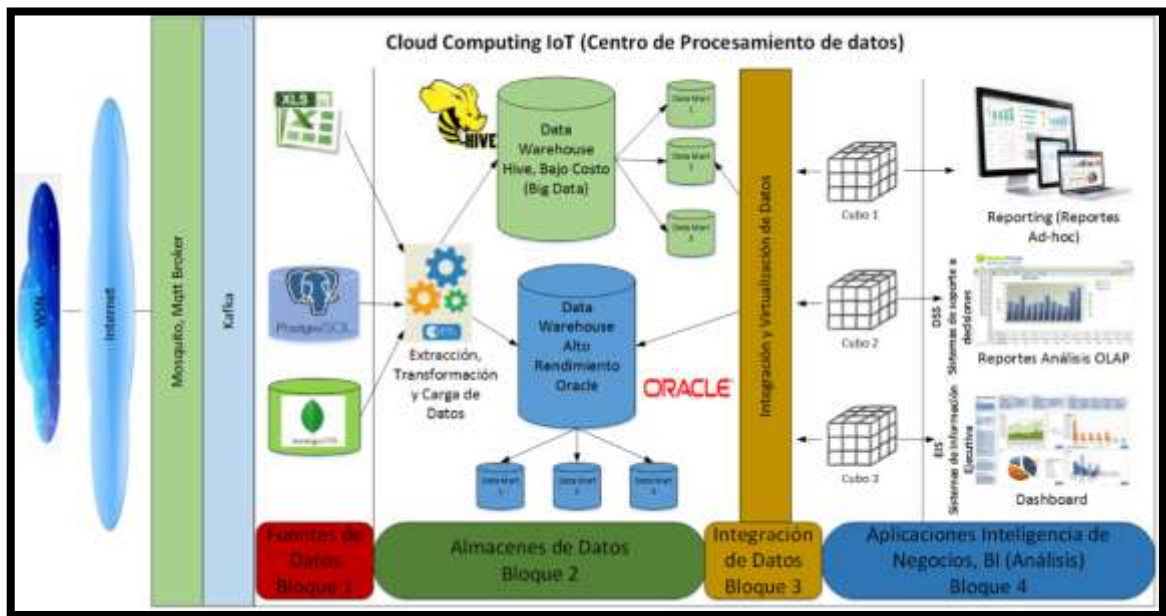
Bloque 1. Fuente de Datos: Acceso a distintas fuentes de datos como PostgreSQL y MongoDB encargados de almacenar los datos enviados por la WSN, mediante Pentaho Data Integration y sus respectivos conectores JDBC.

Bloque 2. Almacenes de Datos: Luego se usan dos motores de base de datos Apache Hive y Oracle para diseñar los Data Warehouse que están organizados en DataMarts.

Bloque 3. Integración y virtualización de Datos: La Virtualización de los datos se lleva a cabo usando el software Denodo encargado de asegurar la integridad y agilidad de los datos.

Bloque 4. Aplicaciones de Inteligencia de Negocios: La información pasa a los cubos multidimensionales para finalmente crear los reportes que soportan las decisiones empresariales.

Figura 2 Arquitectura del Data Warehouse Agricultura de Precisión

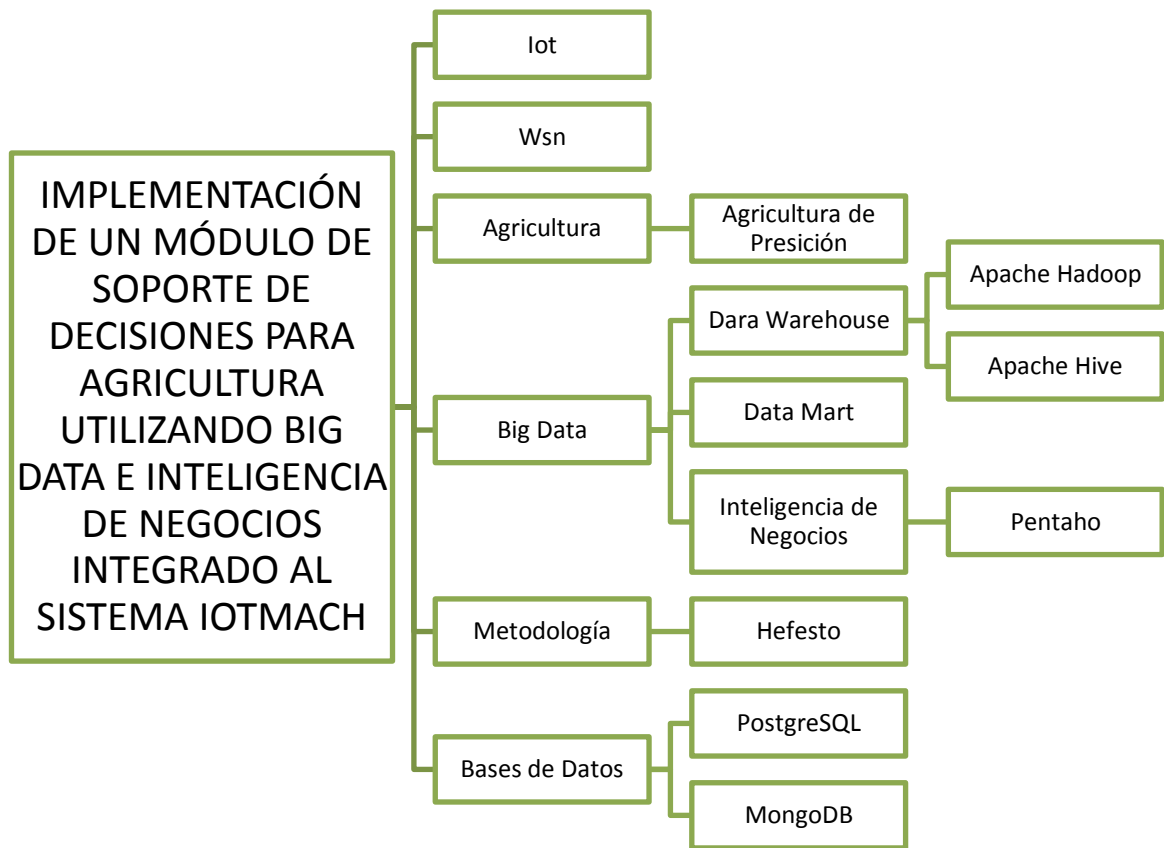


Fuente: Elaboración propia

2.2. Fundamentación teórica del prototipo

La fundamentación teórica para este trabajo de titulación se centra en el estado del arte concerniente a los temas agricultura, agricultura de precisión, internet de las cosas, redes de sensores, Data Warehouse, Big Data, e Inteligencia de negocios. A continuación, en la Tabla 1 se realiza un esquema de los contenidos teóricos.

Tabla 1 Contenidos Teóricos



Fuente: Elaboración propia

2.2.1 *Internet de las cosas (IoT)*. Internet de las cosas es un término que ha aparecido en nuestro entorno en los últimos años y que hace referencia a objetos que parecen simples, pero al relacionarse con la tecnología se convierten en objetos inteligentes. Entre los años 2008 y 2009 fue cuando apareció por primera vez el término Internet de las cosas, debido a que el número de dispositivos electrónicos superó el número de personas conectadas a internet.

Internet of Things (IoT), se compone de objetos inteligentes, tanto celulares (Smartphone), tabletas, sistemas de alarma, electrodomésticos y maquinas industriales, que están conectados a internet en todo momento. Con el pasar del tiempo, la gran mayoría de dispositivos, permanecerán enganchados a internet. Por ejemplo, una cámara, podría identificar y compartir su ubicación en los metadatos de las fotografías. La escala de IoT permitiría lograr interconectar miles de millones de dispositivos, la totalidad de los datos que comparten, y son capaces de llevar a cabo

acciones basadas en la información que facilitan. Las aplicaciones desarrolladas son la primordial potencia promotora del futuro de la IoT.[11]

¿Qué implica tener tantos dispositivos conectados a internet? Que se puede administrar o controlar estos dispositivos de manera remota mediante otros dispositivos o aplicaciones a través de internet.

2.2.2 Red de sensores inalámbricos. La red de sensores inalámbricos (WSN, por sus siglas en inglés Wireless Sensor Network) es una plataforma que facilita el monitoreo de una serie de nodos implementados en una red para realizar mediciones confiables. Esta red puede abarcar decenas, cientos, miles o millones de dispositivos pequeños, que están distribuidos geográficamente y son autónomos, también se los conoce como nodos sensores que se instalan alrededor de un objetivo que se desea monitorear permitiendo realizar mediciones y almacenar los datos que se producen a través de una red inalámbrica.

Red de sensores inalámbricos (WSN) consiste en nodos de sensores distribuidos en lugares remotos y se utilizan para medir datos de los sensores remotamente. Cada nodo de WSN consta de una interfaz de un microcontrolador inalámbrico con sensores. El usuario trabaja de forma independiente y diversos dispositivos electrónicos como la televisión, PDAs, computadoras portátiles, etc. Middleware se requiere para pegar todos estos dispositivos heterogéneos.[12]

Es importante para el sondeo de elementos asociados a la clase de suelo, la calidad de especies, utilizar redes de sensores inalámbricos (WSN), dará lugar a asistir a determinar tácticas para llevar un registro del riego en cada clase de especies, impulsando a un mejor y mayor rendimiento de los recursos naturales.[13]

2.2.3 Agricultura. La agricultura es el arte que tiene el hombre para cultivar la tierra y representa una de las actividades más longevas y principales de la humanidad. Es una actividad que tiene como objetivo cultivar el suelo en búsqueda de desarrollar y cosechar alimentos, aprovechar los bosques y selvas; además de la cría de ganado.

Los países consideran que tener a la agricultura entre las actividades principales desarrollada por el sector primario; una parte de los productos agrícolas son para el consumo directo de las personas, mientras que otro porcentaje está destinado a las industrias para que los procesen y se obtengan productos derivados.

Las herramientas y métodos usados en agricultura han tenido una considerable evolución con el paso de los años, desde el uso de arados con bueyes manejados por los campesinos, pasando por la implementación de molinos de viento e hidráulicos hasta el uso de sensores que indican el estado del clima o del suelo, buscando siempre mejorar el nivel de producción. En lo relacionado a la producción agrícola, se debe tener en consideración varios factores que intervienen en este proceso tales como el suelo, clima, inversión, la propiedad territorial y, últimamente, la tecnología.

2.2.4 Agricultura de precisión. Agricultura de precisión (PA), asignado a la gestión de parcelas rurales agrícolas considerando en todo momento la presencia de la diversa versatilidad espacial y temporal, por tal motivo es necesario realizar estudiado con el propósito de buscar la mejor manera de emplear los recursos naturales e insumos. De tal forma que podamos sortear la sobre aplicación que acaso llegara a deteriorar el desarrollo del cultivo. Así analizando desde esta perspectiva se acepta el AP poner el tipo de entrada variables que producen óptimo rendimiento que la aplicación en la media, lo que faculta llevar al máximo de los beneficios económicos y/o también mejor calidad del fruto.[14]

2.2.5 Big data. Se denomina como Big Data al proceso de análisis y gestión de grandes volúmenes de datos que por su magnitud no pueden ser tratados de manera tradicional, debido a que las herramientas de software que usualmente se emplean para la gestión de los datos ven superados sus límites y capacidades. Características:

Volumen: generación de grandes cantidades de datos, el procesamiento y el ahorro.[15]

Velocidad: Velocidad: El agotamiento del mercado, es demasiado para perder el sentido de la predicción, por lo cual el procedimiento del envejecimiento de la big

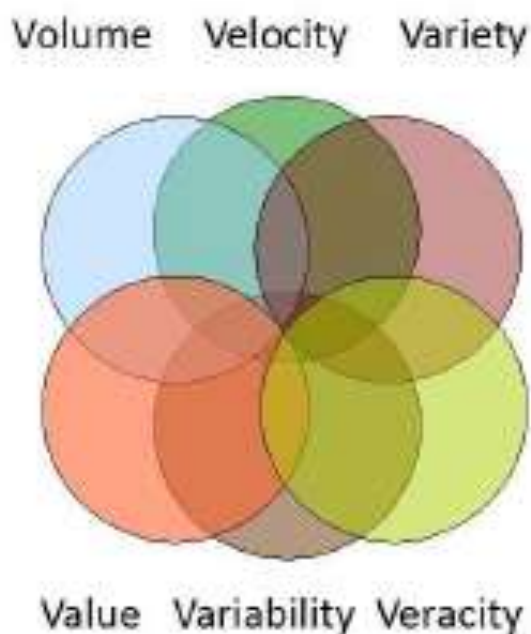
apunta a ser muy crítica, diez millones de datos deben ser examinados de manera amplia en solo diez minutos.[15]

Variedad: Hace mención a la forma de datos variables, pueden ser, texto, audio, video, web y streaming, etc. Datos estructurales, semiestructurales y no estructurales.[15]

Veracidad: Pone en relación a las fuentes de datos y la calidad en sí, si no hay calidad, significa que el estudio de los resultados no es veraz.[15]

Big Data como asistencia se refiere a la entrega de herramientas de análisis descriptivo o la información de un agente externo que apoya a las organizaciones a mantener la estructura, a descifrar y emplear las competencias adquiridas mediante amplios grupos de datos con el propósito de lograr una utilidad capaz de competir.[16]

Figura 3 Modelo BigData



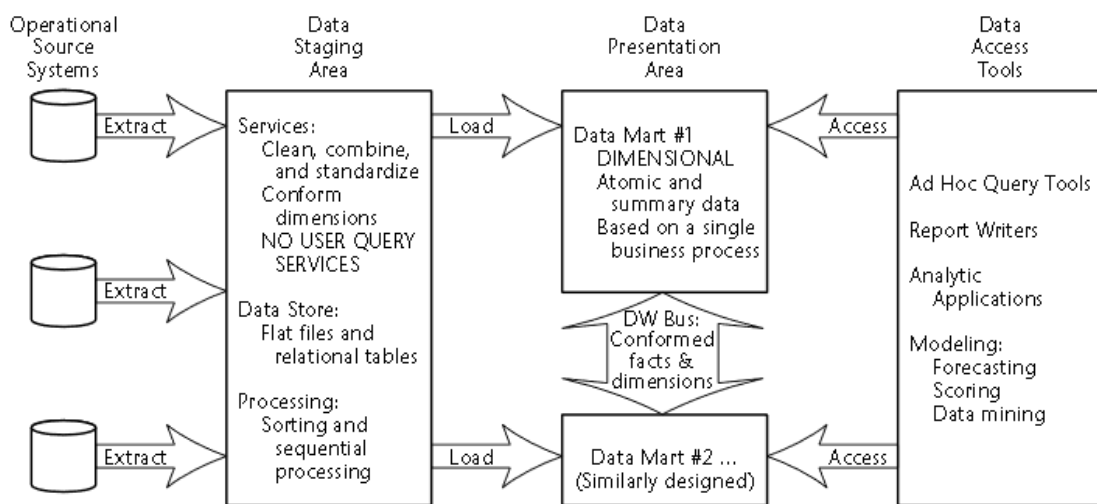
Fuente: [10]

2.2.6 Data warehouse. Ralph Kimball, puntualiza al almacenamiento de datos como un duplicado de los datos de transacciones sistematizadas definidas para consultas y análisis. Un depósito de datos de una compañía afianza los datos de orígenes diversos en ayuda a la determinación, propagación de informes y estudios. Los almacenes de datos con frecuencia emplean los esquemas de estrella y copo de nieve

para ofrecer los tiempos de respuesta más ágil posible para consultas complicadas [17]. Bill Inmon enuncia que el almacén de datos es Subject-Oriented, Integrated, Nonvolatile, Time-Variant en apoyo a la toma de decisiones de gestión.[15]

Los Data warehouse son bases de datos que sirven para manipular grandes volúmenes de datos, por lo general están alojadas en clúster de servidores, y suele actuar como un repositorio de miles de datos generados por cada departamento de una organización. Se necesita de software especializado en minería de datos para poder extraer la información más significativa de la Data warehouse.

Figura 4 Elementos básicos de un Data warehouse



Fuente:[18]

2.2.7 *Apache hadoop*. Apache Hadoop, inventado por Doug Cutting en soporte a su trabajo en Nutch, que consistía en una Web de código abierto o motor de búsqueda, creando un marco de software de código abierto para datos distribuidos. Siendo en la actualidad una de las tecnologías más populares para el almacenamiento de los datos estructurados, semi-estructurados y no estructurados que forman Big Data. Se encuentra disponible bajo licencia Apache 2.0. Está diseñado para pasar de los servidores individuales a miles de máquinas, cada oferta local de computación y almacenamiento.[19]

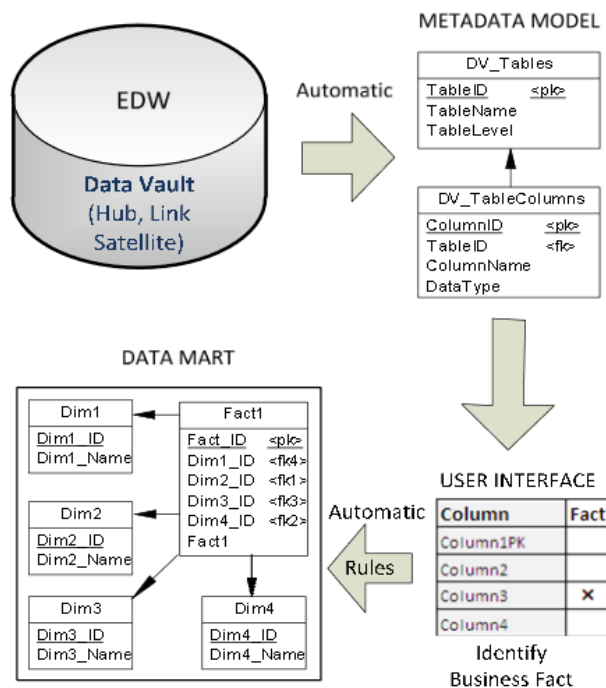
2.2.8 *Apache hive*. Apache Hive and Apache Pig son programas de código abierto para el análisis de grandes conjuntos de datos en un lenguaje de alto nivel. Hive es un

software de almacenamiento de datos que facilita la consulta y gestiona un gran conjunto de datos en el almacenamiento distribuido. Hive y Pig se ejecutan en la parte superior de Hadoop.[20]

2.2.9 *Data mart*. Una tabla de mercado de datos es la tabla principal en un modelo tridimensional donde se almacenan las mediciones de rendimiento numéricos de la empresa. Se da todas las medidas numéricas en un solo lugar en vez de ser duplicada en diferentes lugares en el almacén de datos. Actúan como medidas para el análisis a lo largo de varias dimensiones.[17]

Los data mart presentan varias ventajas que motivan su uso, por ejemplo, se disminuye el volumen de datos que se va a analizar, por ende las consultas de esos datos es más veloz; las consultas SQL y MDX se vuelven más sencillas en comparación a las realizadas con los Data warehouse y los registros históricos de la información son más fáciles de realizar y controlar.

Figura 5 Proceso de generación de un Data mart desde una bóveda de datos



Fuente: [21]

2.2.10 Inteligencia de negocios. Hoy en día, tanto las grandes como las pequeñas empresas, están encaminados a la reducción de costes y analizando sus gastos básicos. Esto incluye la reducción de costos en mano de obra, equipo de oficina y más importante, la tecnología utilizada. A medida que el ámbito de aplicación de los procesos de negocio se vuelve más amplio y abarca no sólo las empresas más grandes, si no, todo tipo de empresas, una coordinación eficaz requiere una función especializada. A medida que aumenta la complejidad de los procesos de negocio, se hace más difícil de integrarlos. Una solución puede ser la implementación de nuevos procesos de gestión, los flujos de nuevos negocios y la integración de tecnologías Open Source.[22]

Indicador clave de rendimiento (KPI) es una medida importante para las organizaciones. En particular, el KPI es una medida de un destino, incluyendo la dirección de mejora, de referencia o de destino, y el plazo que se asocian con actividades específicas para lograr objetivos a largo plazo.[23]

2.2.11 Pentaho. Pentaho BI Server es una suite de herramienta de código abierto y libre, orientada a la solución y centrada en procesos que engloba a los principales componentes que requiere para implementar soluciones basadas en procesos. Ofrece soluciones compuestas fundamentalmente de una infraestructura de herramientas para análisis e informes, además de que integra un motor de workflow para los procesos de negocio. La presencia de las reglas de negocios expresadas en forma de procesos y actividades permite que Pentaho presente y entregue la información adecuada en el momento oportuno.

Es un conjunto de herramientas de BI, incluyendo aplicaciones para la integración de datos, informes, análisis, cuadros de mando y la minería de datos. Al igual que las otras herramientas de BI de clase empresarial, módulo de informes de Pentaho es compatible con múltiples fuentes de datos y puede informar a los formatos Web, PDF y documentos de oficina. Las versiones están disponibles en el sitio de código abierto SourceForge.net o el código fuente está disponible directamente desde el repositorio de Subversion. La edición Enterprise de Pentaho tiene una funcionalidad adicional y apoyo. Pentaho ahora se integra con la plataforma de código abierto Hadoop.[24]

2.2.12 MongoDB. Es una base de datos de propósito general[25], y documental. Se utiliza un modelo flexible para proporcionar esquema dinámico. Los datos se almacenan en formato BSON, una serialización binaria codificada de documentos JSON. La estructura de MongoDB es diferente de los sistemas tradicionales de bases de datos relacionales. Ya que su estructura principal es un documento base, una colección podría ser comparable a una tabla mientras que un documento podría ser equivalente a un registro en un ser relacional[26].

MongoDB y HBase utilizan la estructura maestro-esclavo. Cuando el nodo maestro falla, también toman algunos mecanismos adecuados para la manipulación, para evitar el grave impacto en todo el sistema presentado por el punto único de fallo.[27]

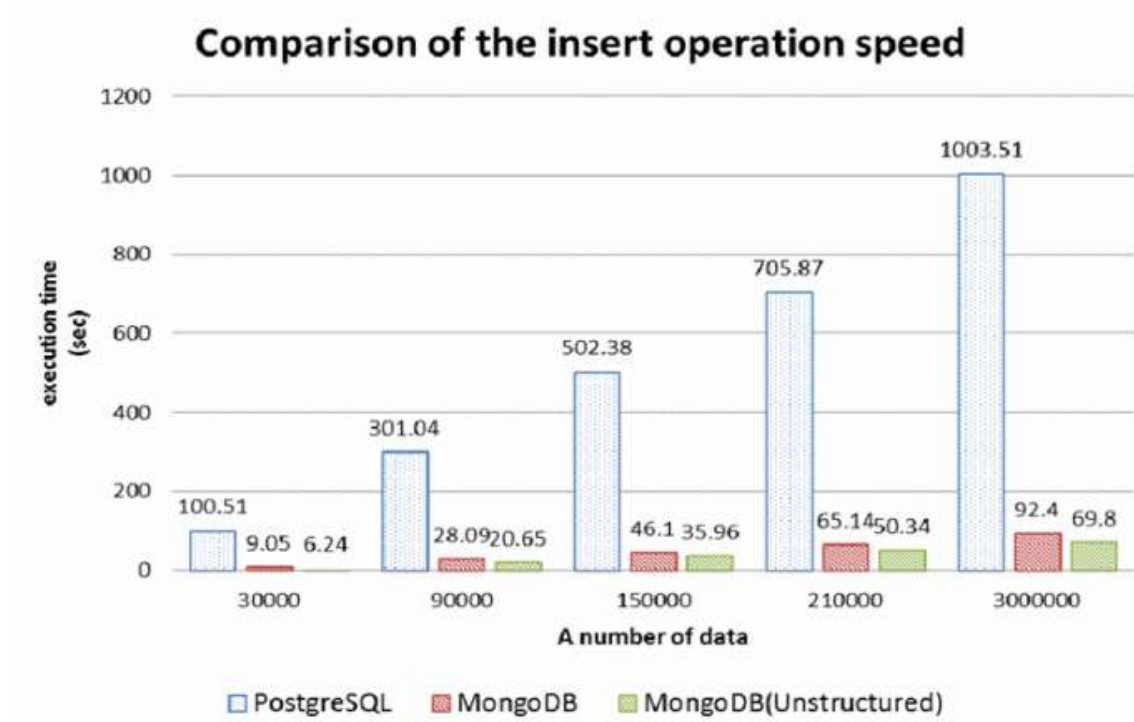
2.2.13 Postgresql. PostgreSQL se basa en POSTGRES, que fue desarrollado por la Universidad de California, en Berkeley. Es un sistema gestor de bases de relación de objeto (ORDBMS), es una base de datos de código abierto[28].

Es una de las más famosas RDBMS de código abierto desarrollado inicialmente por Michael Stonebraker de UC Berkeley. Ahora es desarrollado y mantenido por voluntarios.[29]

Es un sistema de bases de datos popular, que se ha utilizado ampliamente en los sistemas de las empresas. Históricamente, los sistemas de bases de datos se utilizaron principalmente para el procesamiento de transacciones en línea donde las transacciones acceder y procesar sólo algunas filas de las tablas de datos.[30]

2.2.14 Comparación de rendimiento. Debido al avance de la red social y la popularización de los dispositivos móviles, el sistema de gestión de base de datos relacional existente (RDBMS) de procesamiento masivo de datos se ha convertido en un problema. NoSQL es un sistema de gestión de base de datos que hace que el procesamiento de los datos no estructurados sean más fáciles y/o masivas, y muchas empresas hoy en día tienden a iniciar un proyecto utilizando NoSQL.[31]

Figura 6 Comparación de Inserción



Fuente: [31]

2.3 Objetivos del prototipo

2.3.1 Objetivo General

Implementar un módulo de soporte de decisiones para agricultura utilizando Big Data e Inteligencia de Negocios BI aplicando la metodología Hefesto, para la obtención de análisis descriptivos de los cultivos y cosechas, integrado al sistema IoT Mach.

2.3.2 Objetivos Específicos

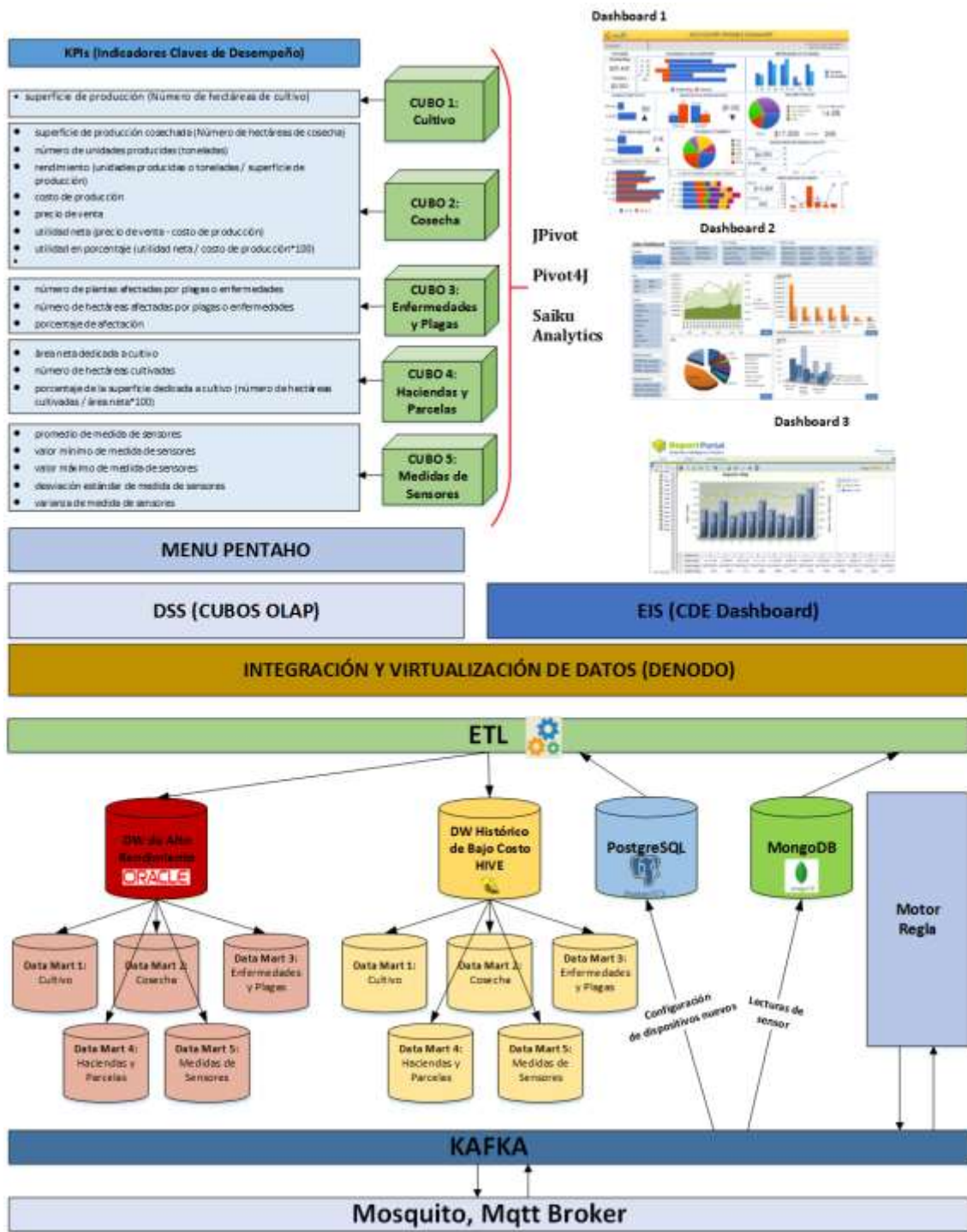
- Efectuar la investigación bibliográfica de las tecnologías y herramientas disponibles para el desarrollo de la implementación de un módulo de soporte de decisiones con inteligencia de negocios para agricultura.
- Diseñar DataMarts y Data warehouse para agricultura en busca de mejorar el proceso de toma de decisiones de una organización.

- Realizar el proceso ETL (Extracción, Transformación y Carga de datos) de la base de datos estructurada PostgreSQL y no estructurada MongoDB, provenientes de las redes sensores inalámbricos y el sistema transaccional.
- Utilizar la herramienta de Inteligencia de Negocios PENTAHO para la obtención de resultados que ayuden a la toma de decisiones.

2.4. Diseño del prototipo

La descripción detallada de las partes que conforman la propuesta tecnológica final, se muestran en la figura 7, el corazón del sistema se encuentra en la fuente de datos enviada por WSN y el sistema transaccional, los datos relacionales y no relacionales. Luego estos datos son extraídos, transformados y cargados en el motor de bases de datos Oracle de alto rendimiento y Hive de bajo costo, en este paso también se crean los Data Mart o datos de área de negocios específicos teniendo en cuenta los indicadores de desempeño. Después de su almacenamiento comienza la extracción dinámica de los datos significativos de la empresa usando cubos multidimensionales OLAP finalmente, se producen los Dashboards o análisis descriptivo usando reportes estadísticos que soporten la toma de decisiones o Inteligencia de Negocios.

Figura 7 Diseño del prototipo



Fuente: Elaboración propia.

2.4.1. Diseño del Data Warehouse

2.4.1.1. *Metodología Hefesto.* Gracias a su amplia investigación la herramienta a usar es “Hefesto”, ya que no proporciona procesos de creación de almacenes de datos. Lo principal, es entender paso a paso que se realizará, para saber exactamente que se está realizando.

Hefesto es una metodología fácil, que especifica el diseño y construcción de un Data Warehouse, siguiendo una serie de pasos ordenados e intuitivos.

En la siguiente tabla se muestran las fases y actividades que componen la arquitectura de HEFESTO y que conllevan a la construcción del Data Warehouse:

Cuadro 1 Procesos de la metodología Hefesto

1) ANÁLISIS DE REQUERIMIENTOS	Requisitos
a) Identificar preguntas	
b) Identificar indicadores y perspectivas de análisis.	
c) Modelo Conceptual.	Análisis
2) ANÁLISIS DE LOS OLTP	
a) Determinación de Indicadores.	
b) Establecer correspondencias.	
c) Nivel de granularidad	Diseño
d) Modelo Conceptual ampliado.	
3) MODELO LÓGICO DEL DW	
a) Tipo de Modelo Lógico del DW	
b) Tabas de dimensiones.	ETL
c) Tablas de hechos.	
d) Uniones	
4) INTEGRACIÓN DE DATOS	
a) Carga Inicial	
b) Actualización	
Fuente: [9]	

La tabla indica que se inicia recolectando información, con la finalidad de obtener una serie de preguntas que serán claves para el negocio, a los usuarios. Luego, hay que identificar los indicadores (KPIs) que son resultado de las preguntas junto con sus respectivos análisis, que serán base para la elaboración del modelo conceptual de datos del Data Warehouse.

El siguiente paso es analizar los OLTP en busca de definir de qué manera se construirán los indicadores, establecer correspondencias, fijar los niveles de granularidad de los datos y así obtener un modelo conceptual de datos más amplio.

Luego se realiza la elaboración del modelo lógico del almacén de datos con el tipo de modelo lógico “Copo de Nieve” ideal para almacén de datos, seguidamente se diseñan las tablas de dimensiones y de hechos, estableciendo las uniones de las tablas.

Al final, se define los procesos ETL (Extracción, transformación y carga) de los datos con los que se busca llenar el DW. Estos procesos ETL son ejecutados mediante la herramienta Spoon Pentaho Data Integration y mediante Jobs serán programados para ejecutarse cada cierto tiempo.

2.4.1.2. Pasos y aplicación metodológica

- **Análisis de requerimientos.** El primer paso será definir muy bien los requerimientos de negocio mediante preguntas que enfoquen totalmente los objetivos esperados de la empresa. Luego se realiza un valioso análisis de las preguntas para determinar cuáles serán los indicadores y medidas para la implementación del DW.

Finalmente establecemos la posibilidad de Integrar varios sistemas de analítica y procesamiento distribuido, visualizando y teniendo el control de los datos.

- **Identificar preguntas.** El objetivo principal en esta fase, es el de obtener e identificar las necesidades de información clave para la organización la cual será de vital importancia para llevar a cabo las metas y estrategias de la misma, permitiéndoles tomar una eficaz y eficiente toma de decisiones.

Aplicando a la propuesta tecnológica. A continuación, se narra los procesos más importantes que se tienen en consideración para la toma de decisiones empresariales.

Preguntas identificadas:

- ¿Cuál es la superficie de producción (número de hectáreas de cultivo) por especie, categoría de especie, empresa, día, mes, año, semestre, trimestre, temporada (estaciones del año), hacienda, parcela, tipo de suelo, tipo de riego y sector (país, provincia, cantón)?
- ¿Cuántas unidades (toneladas, cajas, quintales, etc.) se producen por especie, categoría de especie, empresa, día, mes, año, semestre, trimestre, temporada (estaciones del año), sector (país, provincia, cantón) y tipo de suelo?
- ¿Cuál es el rendimiento agrícola (unidades producidas o toneladas / superficie de producción), por especie, categoría de especie, empresa, hacienda, parcela, año, semestre, trimestre, temporada (estaciones del año) y sector (país, provincia, cantón)?
- ¿Cuál es la especie de mayor producción y en qué tipo de suelo?
- ¿Cuál es el costo de producción, precio de venta, utilidad neta y utilidad en porcentaje por especie, categoría de especie, empresa, hacienda, parcela, año, semestre, trimestre, temporada (estaciones del año) y sector (país, provincia, cantón)?
- ¿Cuántas plantas y hectáreas afectadas por plagas o enfermedades se tienen por especie, categoría de especie, empresa, día, mes, año, semestre, trimestre, temporada (estaciones del año), sector (país, provincia, cantón) y tipo de suelo?
- ¿Cuál es la especie de mayor afectación de plagas o enfermedades?
- ¿Cuál es el área neta (número de hectáreas), que se dedica a cultivos agrícolas, por empresa, hacienda, parcela, sector (país, provincia, cantón), tipo de suelo?
- ¿Cuántas hectáreas se cultivan por empresa, hacienda, parcela, sector (país, provincia, cantón) y tipo de suelo?
- ¿Cuál es el porcentaje de la superficie dedicada a cultivo por empresa, hacienda, parcela, sector (país, provincia, cantón) y tipo de suelo?
- ¿Cuál es el promedio, valor mínimo, valor máximo, desviación estándar, varianza por tipo de sensor que proporcionan la red de sensores inalámbricos (WSN) por empresa, hacienda, parcela, sector (país, provincia, cantón), tiempo día, mes, año, semestre, trimestre y temporada (estaciones del año)?

- Identificar indicadores y perspectivas de análisis. Luego de haber definido las preguntas clave, se continua con el análisis para definir los indicadores a utilizar y las perspectivas que intervengan. Los indicadores son valores numéricos por lo general y efectivos. Por otra parte, las perspectivas son los objetos donde se requiere examinar los indicadores, respondiendo las preguntas antes planteadas.

Determinando el siguiente listado:

- superficie de producción (Número de hectáreas de cultivo)
- superficie de producción cosechada (Número de hectáreas de cosecha)
- número de unidades producidas (toneladas)
- rendimiento (unidades producidas o toneladas / superficie de producción)
- costo de producción
- precio de venta
- área neta dedicada a cultivo
- número de hectáreas cultivadas
- porcentaje de la superficie dedicada a cultivo (número de hectáreas cultivadas / área neta*100)
- promedio
- valor mínimo
- valor máximo
- desviación estándar
- varianza

DATA MART 1: CULTIVOS

- 1) ¿Cuál es la superficie de producción (número de hectáreas de cultivo) por especie, categoría de especie, empresa, día, mes, año, semestre, trimestre, temporada (estaciones del año), hacienda, parcela, tipo de suelo, tipo de riego y sector (país, provincia, cantón)?

Cuadro 2 Análisis de indicadores y perspectivas DM1

INDICADOR	PERSPECTIVA
P1. Superficie de producción (número de hectáreas de cultivo).	Por especie, categoría de especie, empresa, día, mes, año, semestre, trimestre, temporada (estaciones del año), hacienda, parcela, tipo de suelo, tipo de riego y sector (país, provincia, cantón)
Fuente: Elaboración propia	

Indicador

Es vital para las organizaciones conocer la cantidad de hectáreas de producción poseen de una determinada especie; dicha información ayudará a que las empresas puedan determinar qué especie es cultivada en mayor número en comparación a las demás. La cantidad de hectáreas producidas puede filtrarse por el año y sector en que fueron cultivadas, ya que, si la empresa posee varios cultivos en distintas partes de un país o del mundo, el sistema facilitará la información detallada del cantón, distrito, departamento, cantón, provincia, país, etc. según sea lo que se busque.

El principal indicador para el desarrollo de esta consulta es el número de hectáreas de producción; las dimensiones son los siguientes campos: año, empresa (filtro según el rol), especie, categoría especie, país, provincia, cantón.

Data Warehouse nos permite la integración de datos corporativos en un único depósito donde los usuarios puedan consultar o analizar los datos para la toma de decisiones.

DATA MART 2: COSECHA

- 2) ¿Cuántas unidades (toneladas, cajas, quintales, etc.) se producen por especie, categoría de especie, empresa, día, mes, año, semestre, trimestre, temporada (estaciones del año), sector (país, provincia, cantón) y tipo de suelo?
- 3) ¿Cuál es el rendimiento agrícola (unidades producidas o toneladas / superficie de producción), por especie, categoría de especie, empresa, hacienda, parcela, año, semestre, trimestre, temporada (estaciones del año) y sector (país, provincia, cantón)?
- 4) ¿Cuál es la especie de mayor producción y en qué tipo de suelo?
- 5) ¿Cuál es el costo de producción, precio de venta, utilidad neta y utilidad en porcentaje por especie, categoría de especie, empresa, hacienda, parcela, año, semestre, trimestre, temporada (estaciones del año) y sector (país, provincia, cantón)?

Cuadro 3 Análisis de indicadores y perspectivas DM2

INDICADOR	PERSPECTIVA
P2. Unidades de (toneladas, cajas, quintales, etc.) se producen	por especie, categoría de especie, empresa, día, mes, año, semestre, trimestre, temporada (estaciones del año), sector (país, provincia, cantón) y tipo de suelo
P3. Rendimiento agrícola (unidades producidas o toneladas / superficie de producción),	por especie, categoría de especie, empresa, hacienda, parcela, año, semestre, trimestre, temporada (estaciones del año) y sector (país, provincia, cantón).
P4. Especie de mayor producción y en qué tipo de suelo	
P5. Costo de producción, precio de venta, utilidad neta y utilidad en porcentaje	
Fuente: Elaboración propia	

DATA MART 3: PLAGAS Y ENFERMEDADES

- 6) ¿Cuántas plantas y hectáreas afectadas por plagas o enfermedades se tienen por especie, categoría de especie, empresa, día, mes, año, semestre, trimestre, temporada (estaciones del año), sector (país, provincia, cantón) y tipo de suelo?
- 7) ¿Cuál es la especie de mayor afectación de plagas o enfermedades?

Cuadro 4 Análisis de indicadores y perspectivas DM3

INDICADOR	PERSPECTIVAS
P6. Número de plantas y hectáreas afectadas.	Por plagas o enfermedades se tienen por especie, categoría de especie, empresa, día, mes, año, semestre, trimestre, temporada (estaciones del año), sector (país, provincia, cantón) y tipo de suelo.
P7. Número de especies con mayor afectación de plagas o enfermedades.	Por plagas o enfermedades se tienen por especie, categoría de especie, empresa y tipo de suelo.
Fuente: Elaboración propia	

DATA MART 4: HACIENDAS Y PARCELAS

- 8) ¿Cuál es el área neta (número de hectáreas), que se dedica a cultivos agrícolas, por empresa, hacienda, parcela, sector (país, provincia, cantón), tipo de suelo?

- 9) ¿Cuántas hectáreas se cultivan por empresa, hacienda, parcela, sector (país, provincia, cantón) y tipo de suelo?
- 10) ¿Cuál es el porcentaje de la superficie dedicada a cultivo por empresa, hacienda, parcela, sector (país, provincia, cantón) y tipo de suelo?

Cuadro 5 Análisis de indicadores y perspectivas DM4

INDICADOR	PERSPECTIVA
P8. Área neta (número de hectáreas), que se dedica a cultivos agrícolas.	por empresa, hacienda, parcela, sector (país, provincia, cantón), tipo de suelo
P9. Número de hectáreas cultivadas.	
P10. Porcentaje de la superficie dedicada a cultivo.	
Fuente: Elaboración propia	

DATA MART 5: MEDIDAS DE SENSORES

- 11) ¿Cuál es el promedio, valor mínimo, valor máximo, desviación estándar, varianza por tipo de sensor que proporcionan la red de sensores inalámbricos (WSN) por empresa, hacienda, parcela, sector (país, provincia, cantón), tiempo día, mes, año, semestre, trimestre y temporada (estaciones del año)?

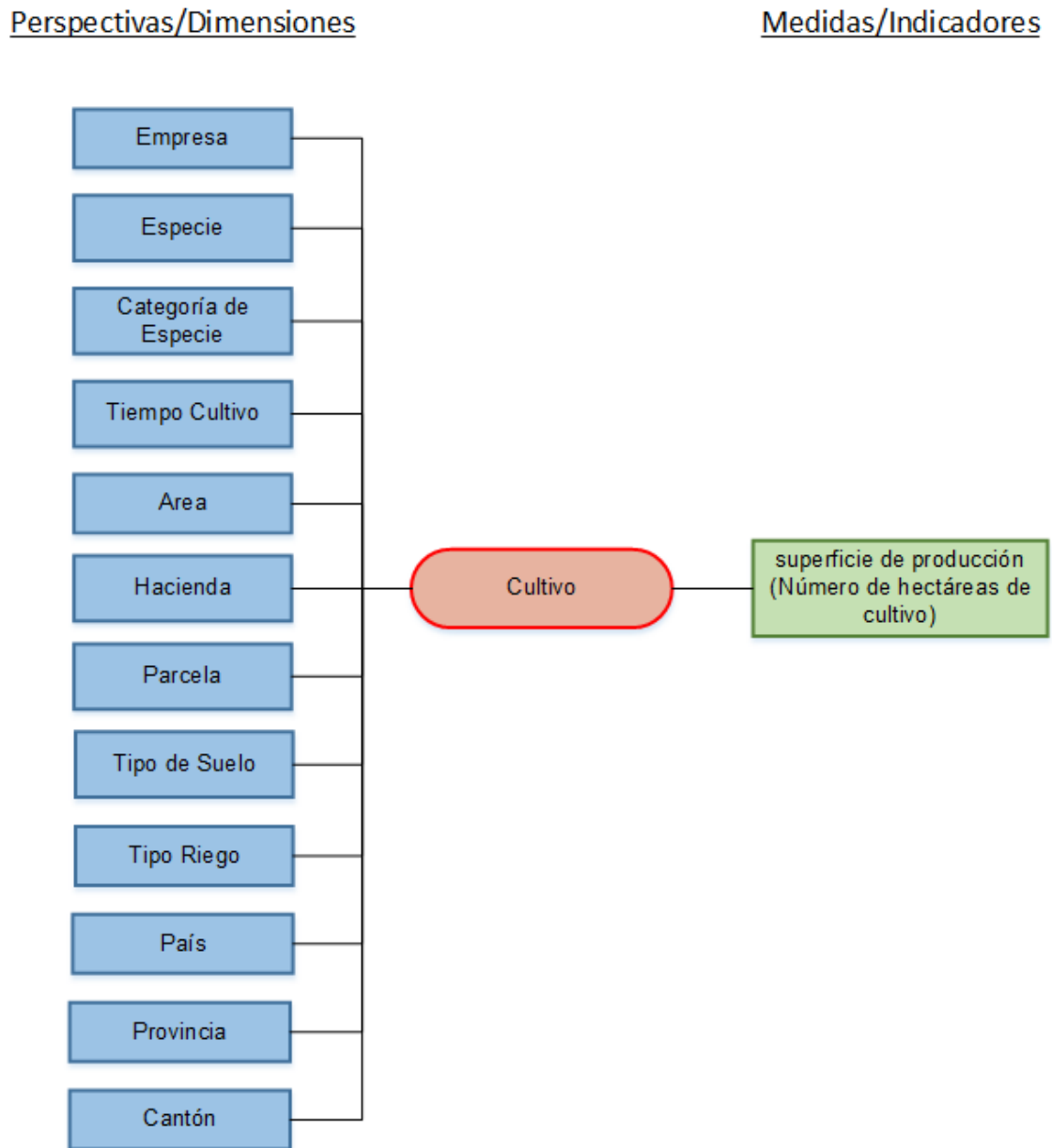
Cuadro 6 Análisis de indicadores y perspectivas DM5

INDICADOR	PERSPECTIVA
Promedio, valor mínimo, valor máximo, desviación estándar, varianza por tipo de sensor que proporcionan la red de sensores inalámbricos (WSN) (estaciones del año)	por empresa, hacienda, parcela, sector (país, provincia, cantón), tiempo día, mes, año, semestre, trimestre y temporada
Fuente: Elaboración propia	

La representación gráfica organizada por Data Marts en la siguiente:

DATA MART 1: CULTIVOS

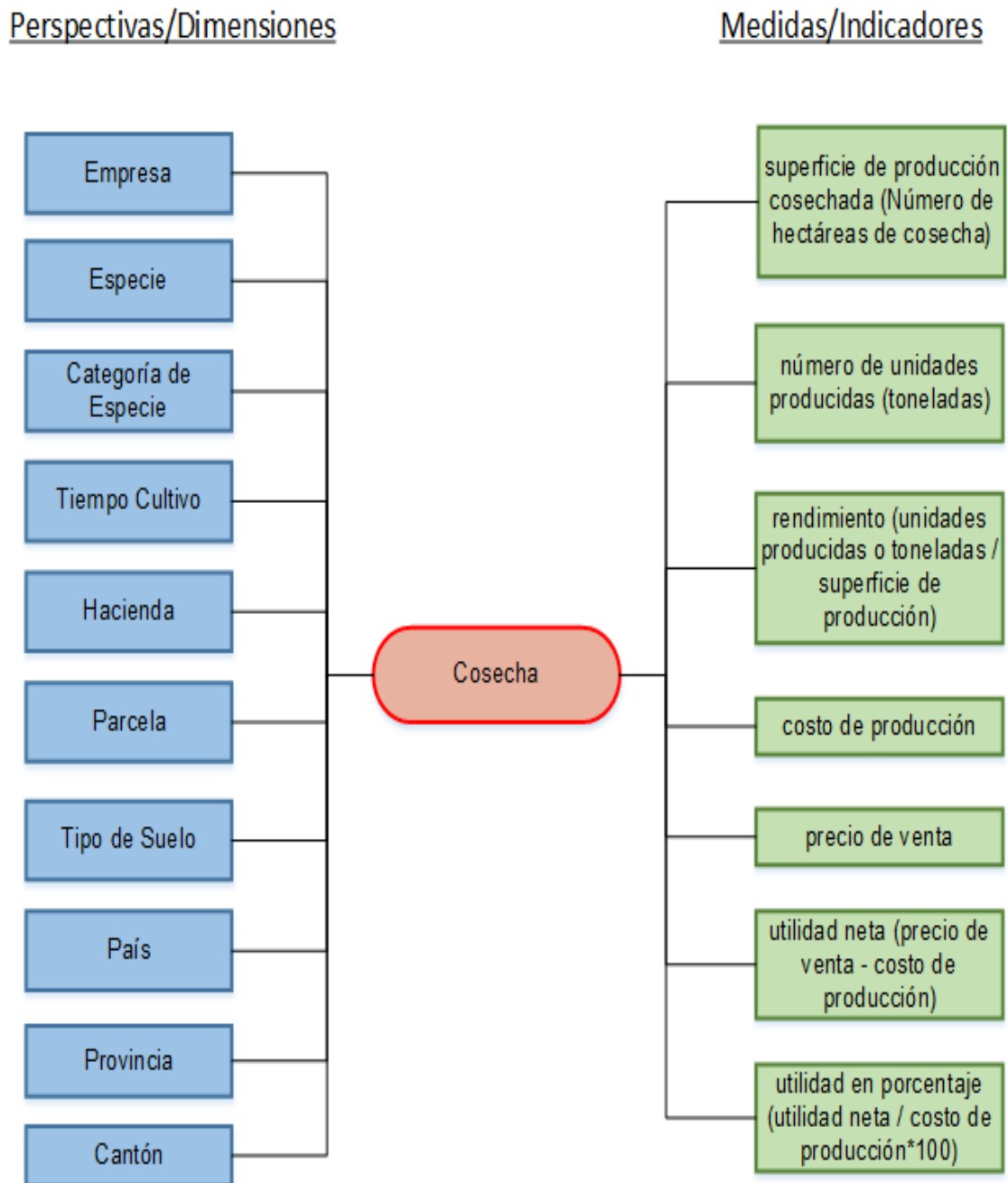
Figura 8 Modelo Conceptual Data Mart 1



Fuente: Elaboración propia

DATA MART 2: COSECHA

Figura 9 Modelo Conceptual DM2



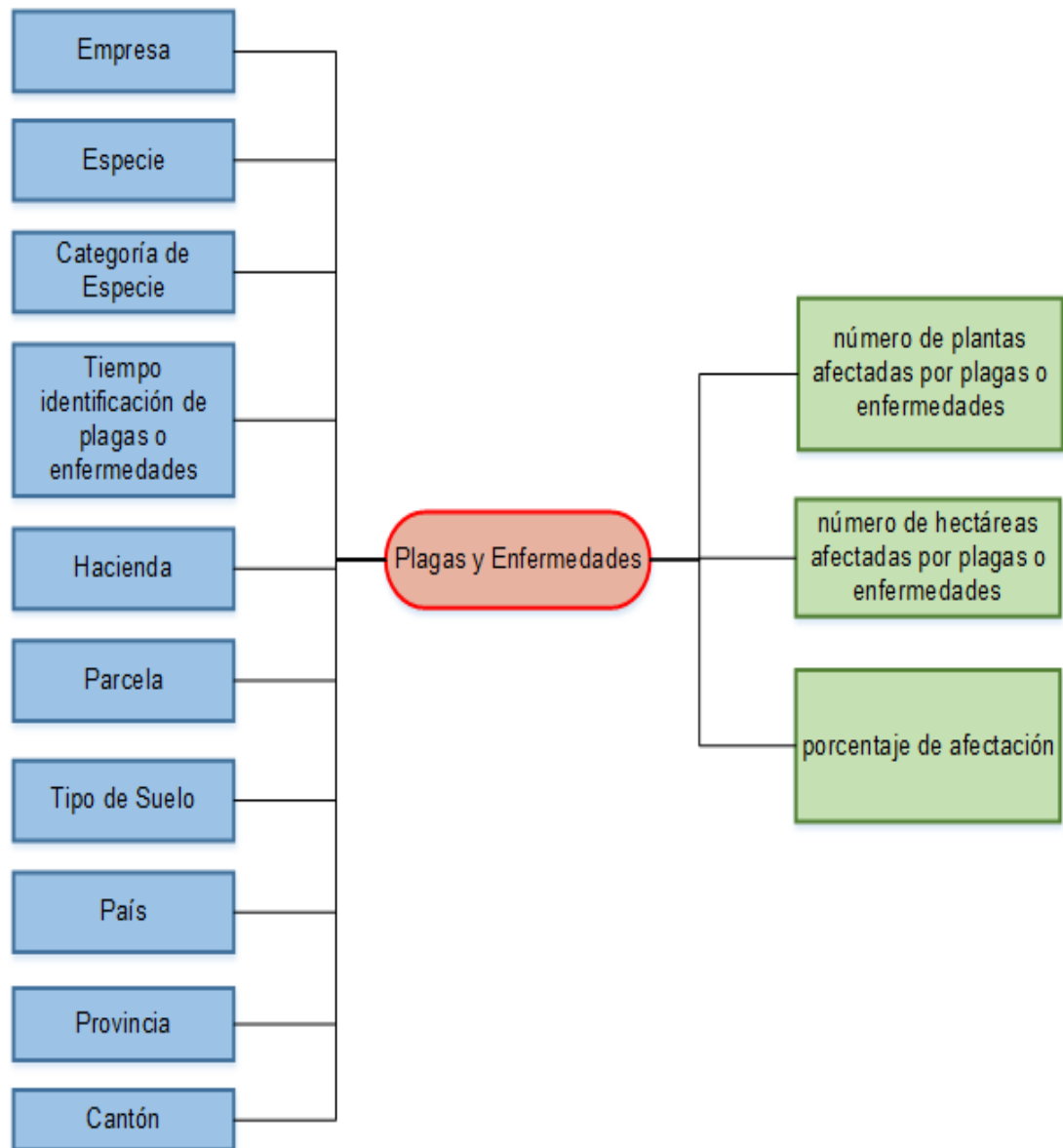
Fuente: Elaboración propia

DATA MART 3: PLAGAS Y ENFERMEDADES

Figura 10 Modelo Conceptual DM3

Perspectivas/Dimensiones

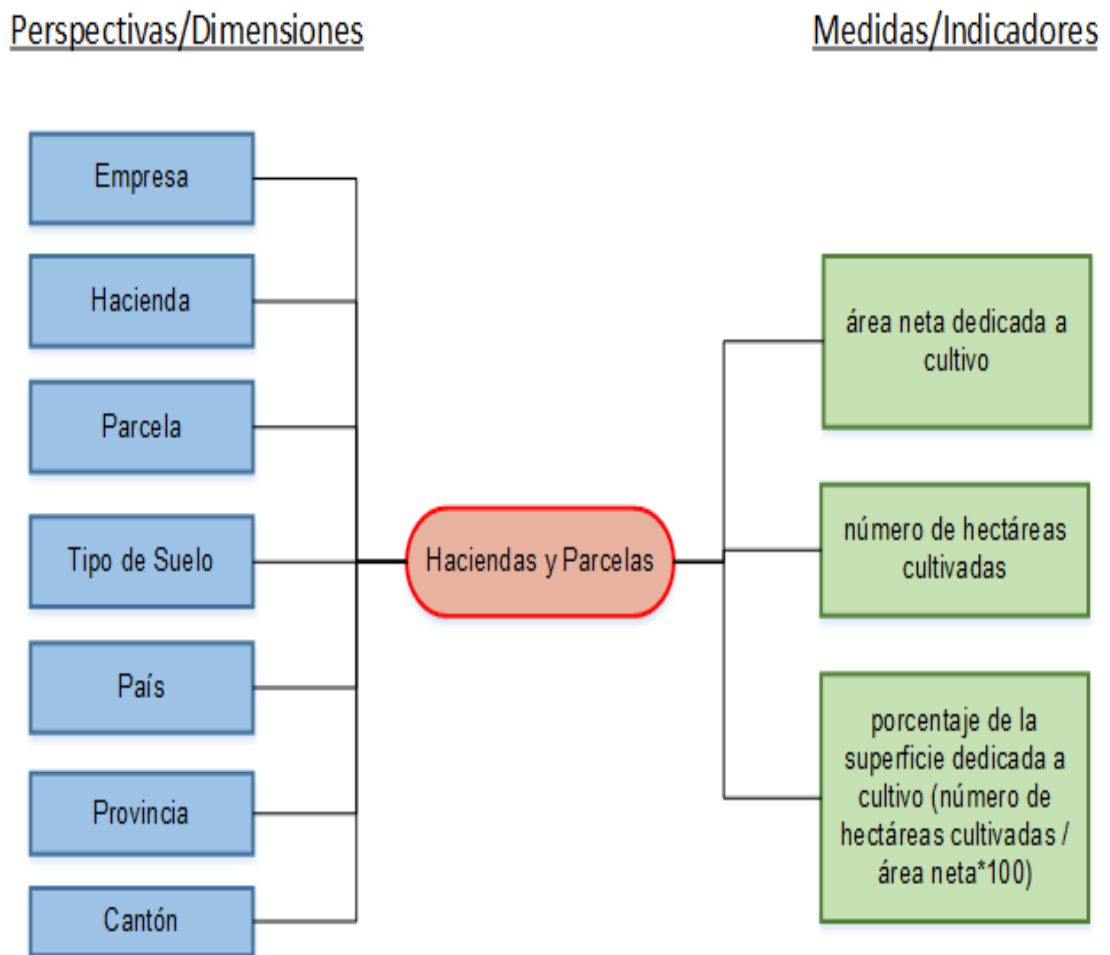
Medidas/Indicadores



Fuente: Elaboración propia

DATA MART 4: HACIENDAS Y PARCELAS

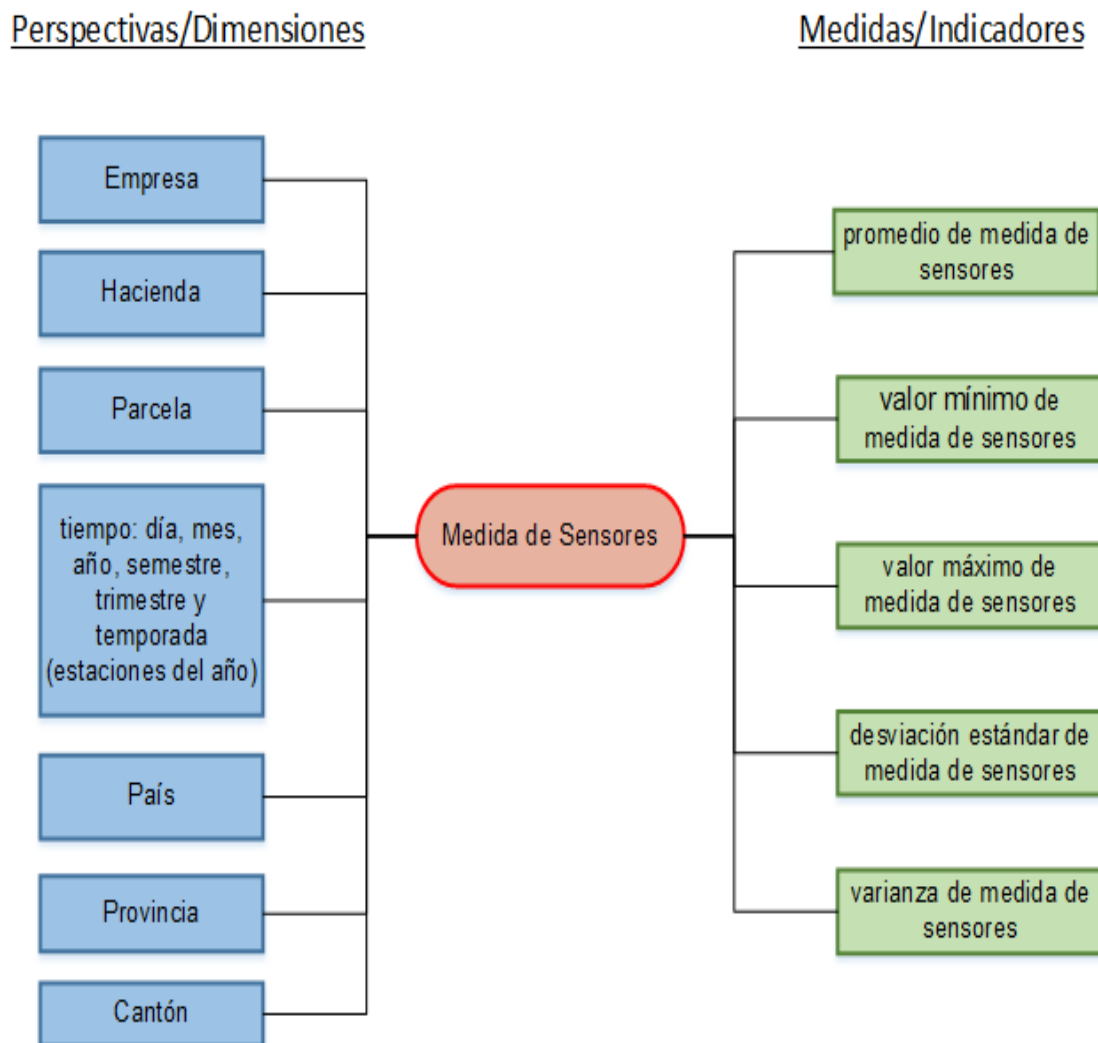
Figura 11 Modelo Conceptual DM4



Fuente: Elaboración propia

DATA MART 5: MEDIDAS DE SENSORES

Figura 12 Modelo Conceptual DM5



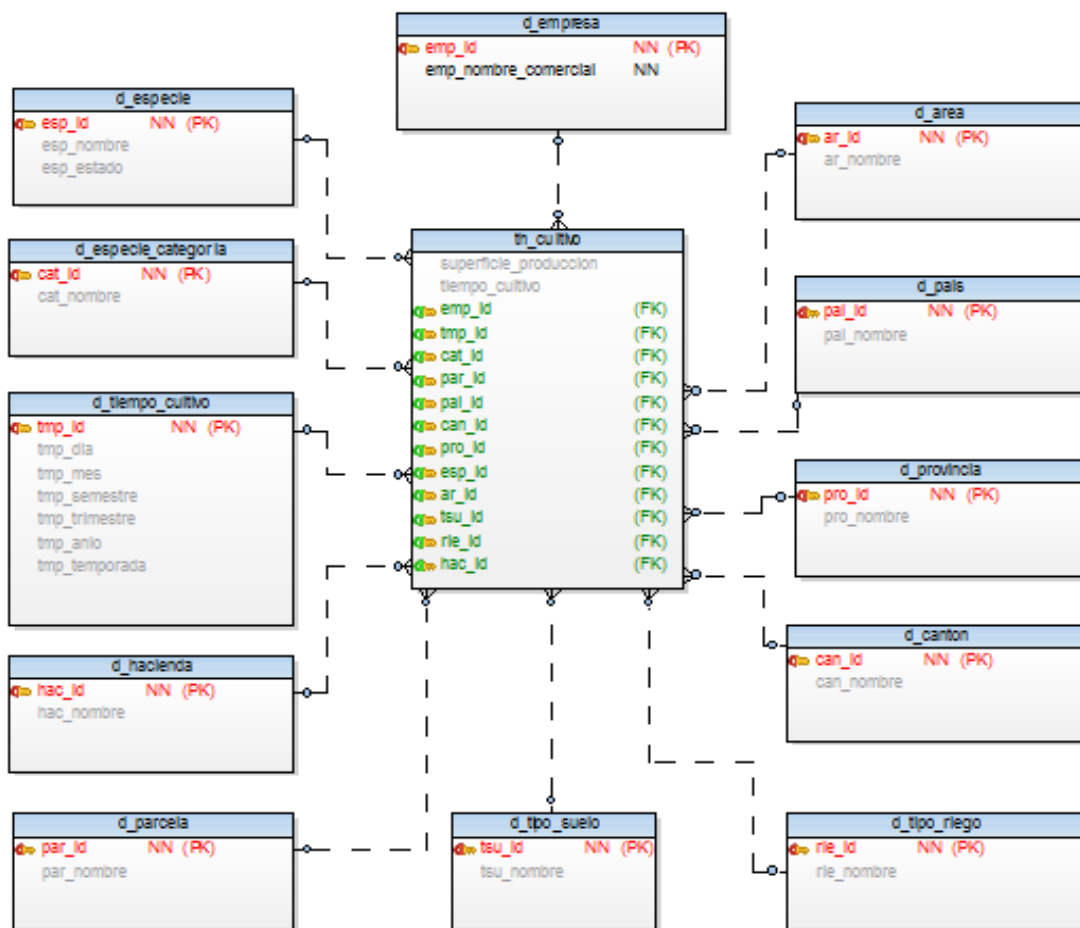
Fuente: Elaboración Propia

- Modelo lógico del DW. En esta fase, se elabora el modelo lógico de la estructura del DW para agricultura, tomando el modelo conceptual que fue elaborado en la fase anterior como base. Como primer punto se define el modelo que se usará, en este caso se usa el modelo de Copo de Nieve, para luego diseñar las dimensiones y hechos dentro del modelo.

Como punto final se realizan las uniones respectivas entre las tablas.

DATA MART 1: CULTIVOS

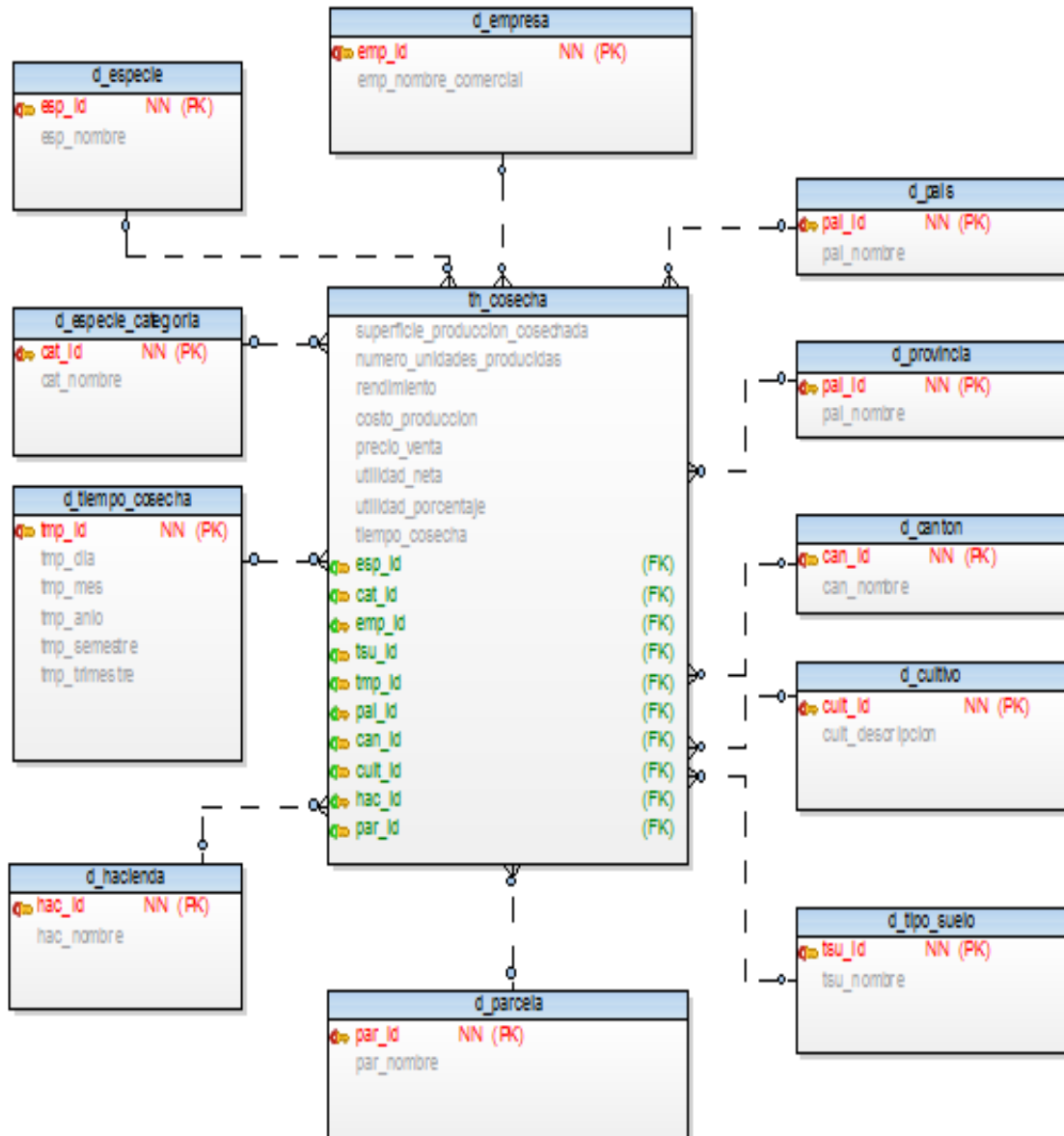
Figura 13 Modelo Lógico del DM 1



Fuente: Elaboración propia

DATA MART 2: COSECHA

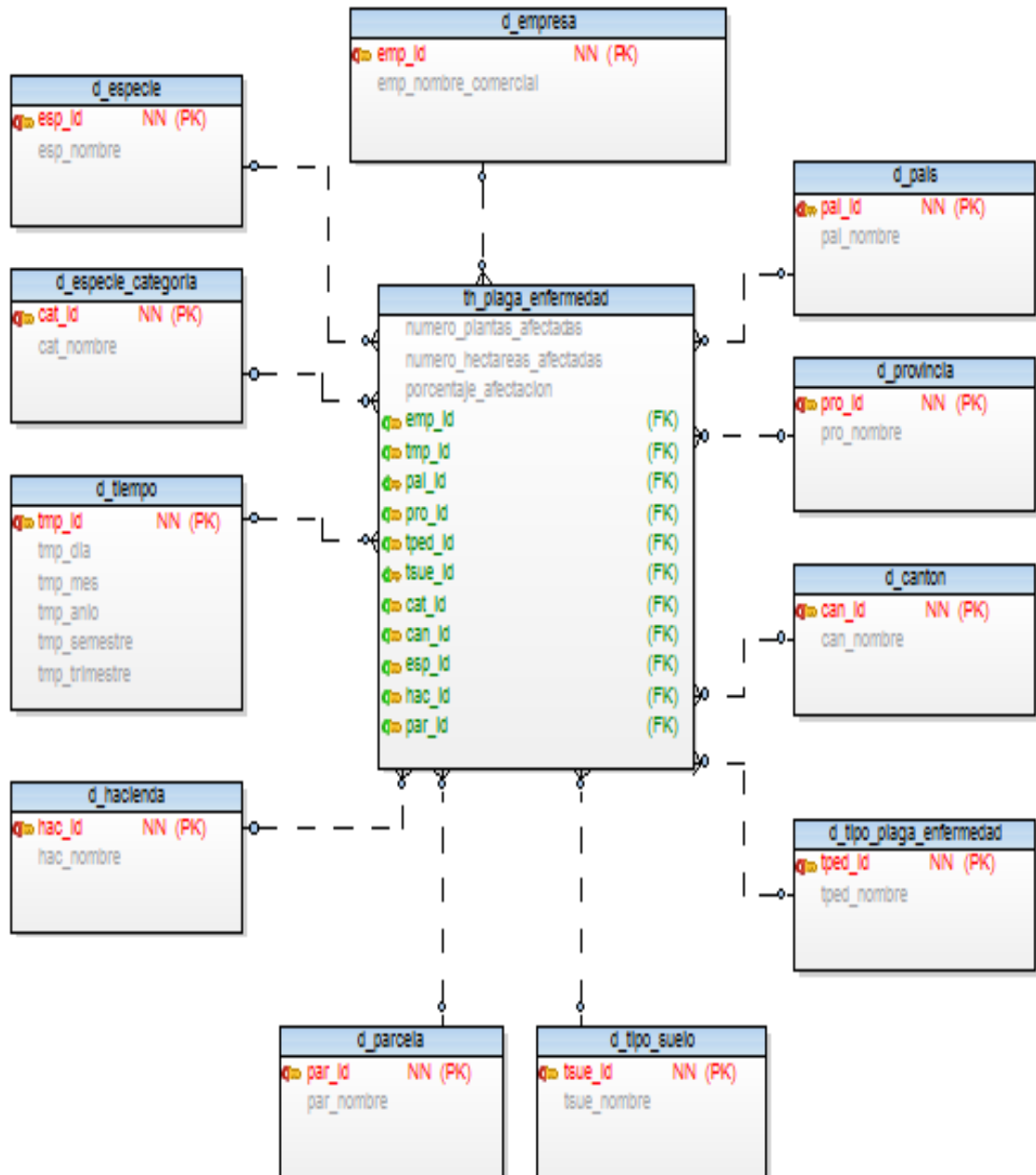
Figura 14 Modelo Lógico del DM 2



Fuente: Elaboración propia

DATA MART 3: PLAGAS Y ENFERMEDADES

Figura 15 Modelo Lógico del DM 3



Fuente: Elaboración propia

DATA MART 4: HACIENDAS Y PARCELAS

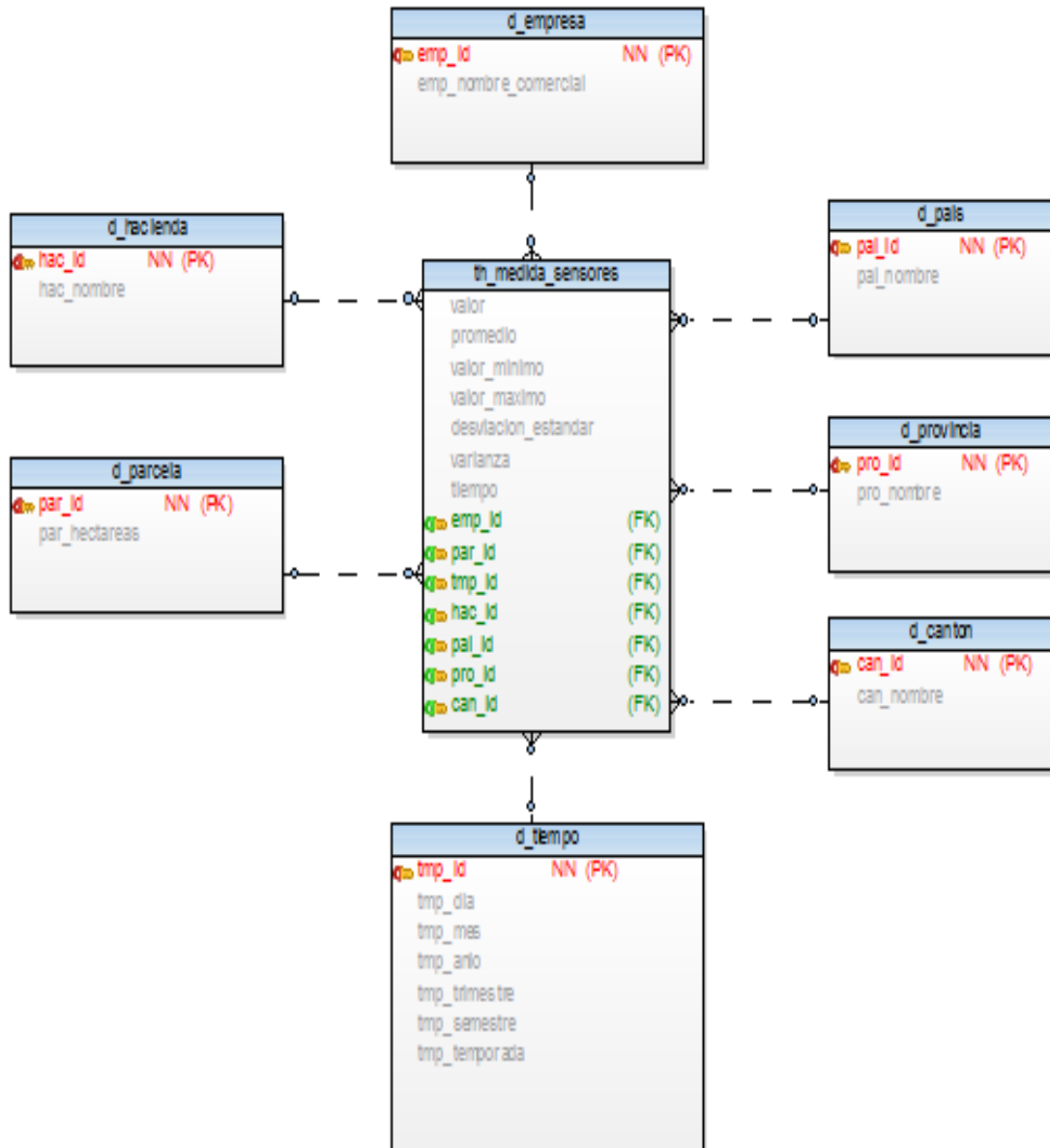
Figura 16 Modelo Lógico del DM 4



Fuente: Elaboración propia

DATA MART 5: MEDIDAS DE SENSORES

Figura 17 Modelo Lógico del DM 5



Fuente: Elaboración propia

- Diseño de los procesos extracción, transformación y Carga (ETL). Luego de haber construido el modelo lógico, se continua con esta fase comprobando con datos existentes en la base de datos mediante el proceso ETL. Luego se define las políticas de actualización del mismo.

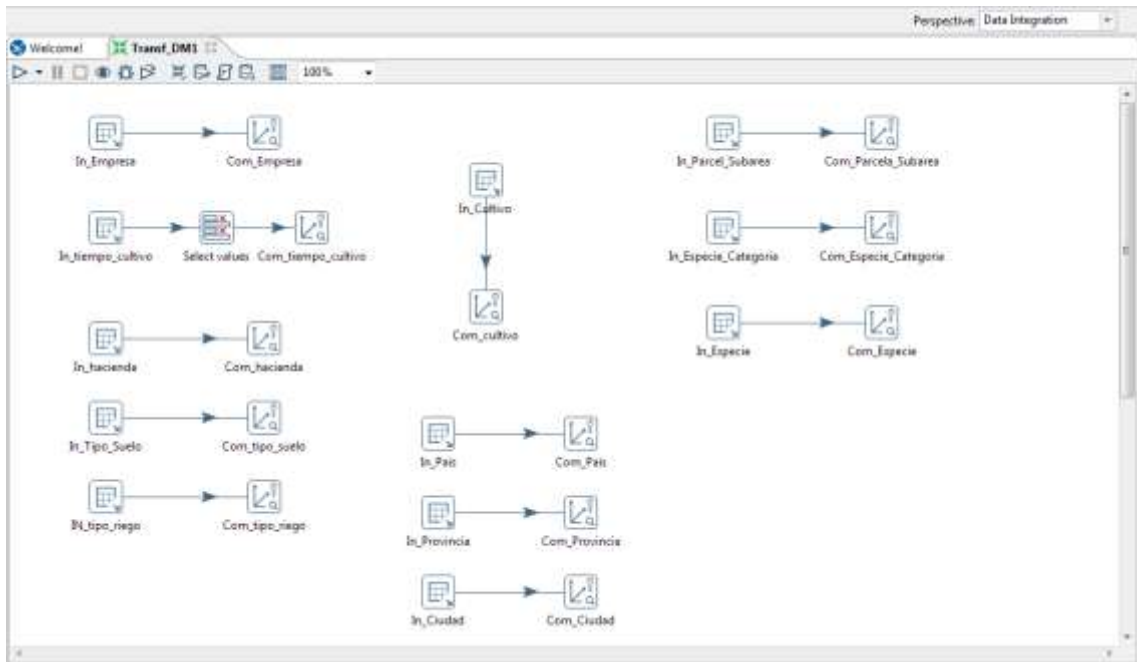
La carga inicial de datos en los ETL, se realiza por medio de JOBs, con una frecuencia de una vez por día, para así mantener actualizado el DW ya sea con los datos históricos y con los datos del último año, respectivamente. Luego ejecutan una Transformación, el cual carga los nuevos datos en las bases de datos correspondientes.

Se debe prevenir que el DW sea cargado con datos no existentes o sin saber su origen, así como definir condiciones y contraindicaciones para asegurar que se carguen los datos de interés. Revisar cómo crear procesos ETL en la herramienta Spoon Pentaho data integration, en Anexo N.

Los datos serán cargados empezando por las dimensiones y luego las tablas de hechos, se ha dividido por Data Marts los siguientes modelos de procesos ETL:

DATA MART 1: CULTIVOS

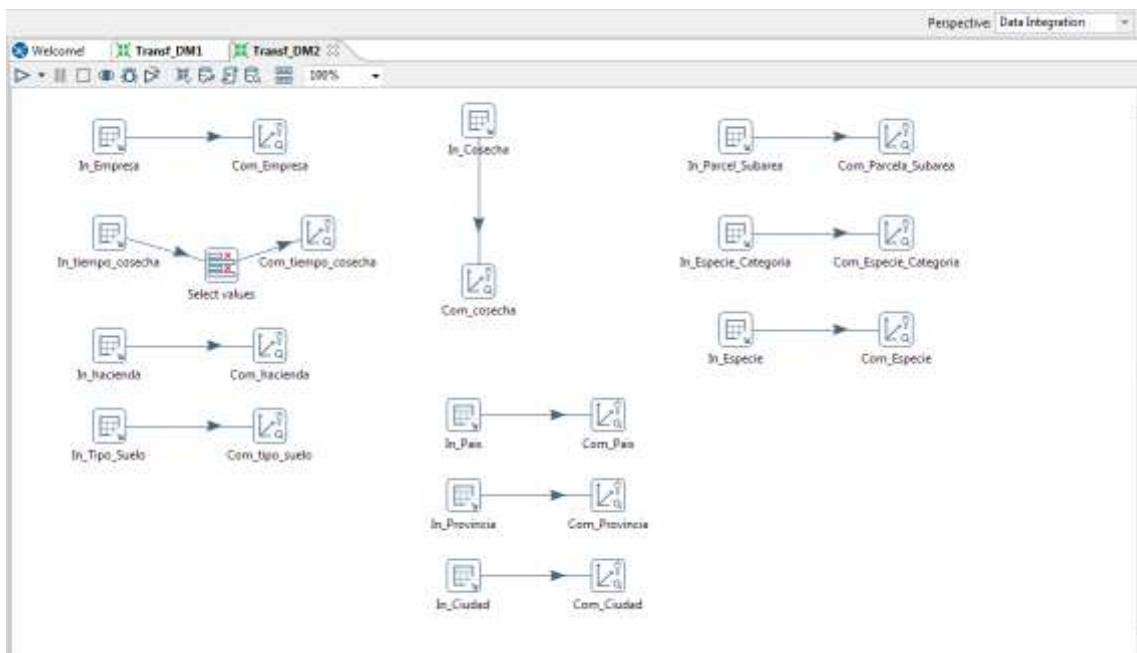
Figura 18 Proceso Extracción, Transformación y Carga al Data Warehouse 1



Fuente: Elaboración propia

DATA MART 2: COSECHA

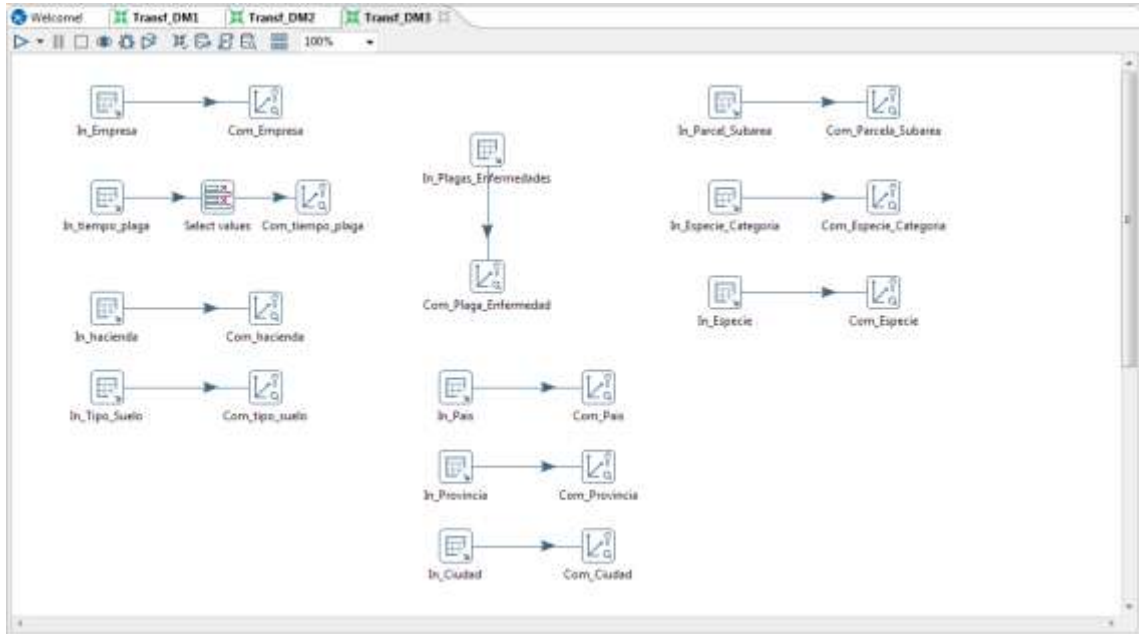
Figura 19 Proceso Extracción, Transformación y Carga al Data Warehouse 2



Fuente: Elaboración propia

DATA MART 3: PLAGAS Y ENFERMEDADES

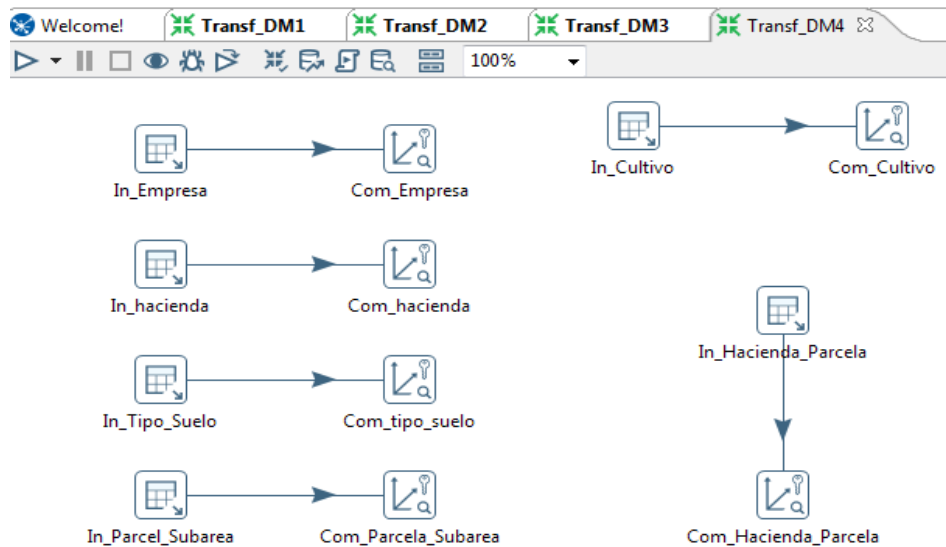
Figura 20 Proceso Extracción, Transformación y Carga al Data Warehouse 3



Fuente: Elaboración propia

DATA MART 4: HACIENDAS Y PARCELAS

Figura 21 Proceso Extracción, Transformación y Carga al Data Warehouse 4

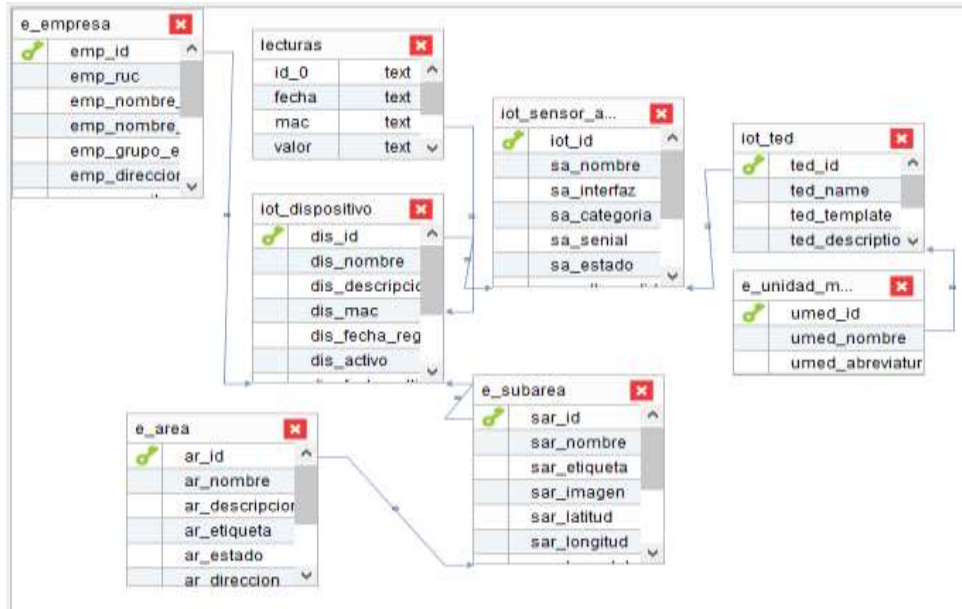


Fuente: Elaboración propia

DATA MART 5: MEDIDAS DE SENSORES

Para realizar el diseño del proceso ETL, es necesario primero realizar una combinación de fuentes de datos mediante la herramienta Denodo Platform, revisar Anexo O:

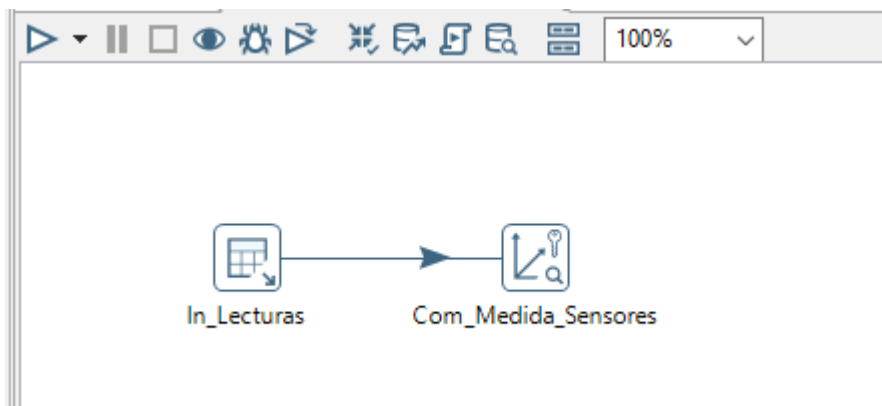
Figura 22 Vista combinación lecturas sensores



Fuente: Elaboración propia

Una vez creada la combinación de datos, se crea la vista correspondiente y se procede a hacer su ETL:

Figura 23 Proceso Extracción, Transformación y Carga al Data Warehouse 5



Fuente: Elaboración propia

2.4.2. *Diseño de aplicaciones BI*

2.4.2.1. *Sistema de Soporte de Decisiones (DSS) y Sistema de Información Ejecutiva (EIS) Dashboard.* Para la creación del sistema de soporte de decisiones (DSS) y el sistema de información ejecutiva (EIS) o Dashboard, se emplean una serie de consultas ejecutadas en Denodo Platform para formar los cubos y poder obtener la estructura necesaria que es necesaria para el análisis de datos mediante Pentaho, ya sea con los visores, Saiku, Pivote4J, JPivote o el mismo Dashboard. Las sentencias SQL las podemos revisar en Anexo P.

- Sentencias sql de creación de vistas

Cubo 1: Cultivos

Figura 24 Vista tabla de hechos Data Mart 1: cultivos

```

CREATE OR REPLACE VIEW dm_cultivo FOLDER = '/vistas' AS
SELECT th_cultivo.superficie_produccion AS superficie_produccion,
d_tiempo_cultivo.tcu_fecha AS fecha,
d_tiempo_cultivo.tcu_dia AS dia,
CASE
    WHEN d_tiempo_cultivo.tcu_mes= '1' THEN 'Enero'
    WHEN d_tiempo_cultivo.tcu_mes= '2' THEN 'Febrero'
    WHEN d_tiempo_cultivo.tcu_mes= '3' THEN 'Marzo'
    WHEN d_tiempo_cultivo.tcu_mes= '4' THEN 'Abril'
    WHEN d_tiempo_cultivo.tcu_mes= '5' THEN 'Mayo'
    WHEN d_tiempo_cultivo.tcu_mes= '6' THEN 'Junio'
    WHEN d_tiempo_cultivo.tcu_mes= '7' THEN 'Julio'
    WHEN d_tiempo_cultivo.tcu_mes= '8' THEN 'Agosto'
    WHEN d_tiempo_cultivo.tcu_mes= '9' THEN 'Septiembre'
    WHEN d_tiempo_cultivo.tcu_mes= '10' THEN 'Octubre'
    WHEN d_tiempo_cultivo.tcu_mes= '11' THEN 'Noviembre'
    WHEN d_tiempo_cultivo.tcu_mes= '12' THEN 'Diciembre'
    ELSE 'Desconocido'
END AS mes,
d_tiempo_cultivo.tcu_trimestre AS trimestre,
d_tiempo_cultivo.tcu_semestre AS semestre,
d_tiempo_cultivo.tcu_anio AS anio,
d_tiempo_cultivo.tcu_temporada AS temporada,
d_empresa.emp_nombre_comercial AS empresa,
d_empresa.emp_ruc AS ruc,
d_especie.esp_nombre AS especie,
d_especie_categoria.cat_nombre AS categoria,
d_hacienda.hac_nombre AS hacienda,
d_parcela.par_nombre AS parcela,
d_tipo_suelo.tsu_nombre AS tipo_suelo,
d_tipo_riego.tri_nombre AS tipo_riego,
d_ciudad.ciudad AS ciudad,
d_provincias.provincia AS provincia,
d_paises.pais AS pais
FROM ((((((((((th_cultivo AS th_cultivo INNER JOIN
d_tiempo_cultivo AS d_tiempo_cultivo ON th_cultivo.id_cultivo = d_tiempo_cultivo.id_tcu ) INNER JOIN
d_empresa AS d_empresa ON th_cultivo.emp_id = d_empresa.emp_id ) INNER JOIN
d_especie AS d_especie ON th_cultivo.esp_id = d_especie.esp_id ) INNER JOIN
d_especie_categoria AS d_especie_categoria ON th_cultivo.cat_id = d_especie_categoria.cat_id ) INNER JOIN
d_hacienda AS d_hacienda ON th_cultivo.hac_id = d_hacienda.hac_id ) INNER JOIN
d_parcela AS d_parcela ON th_cultivo.par_id = d_parcela.par_id ) INNER JOIN
d_tipo_suelo AS d_tipo_suelo ON th_cultivo.tsu_id = d_tipo_suelo.tsu_id ) INNER JOIN
d_tipo_riego AS d_tipo_riego ON th_cultivo.tri_id = d_tipo_riego.tri_id ) INNER JOIN
d_ciudad AS d_ciudad ON th_cultivo.ciudad_id = d_ciudad.ciudad_id ) INNER JOIN
d_provincias AS d_provincias ON th_cultivo.provincia_id = d_provincias.provincia_id ) INNER JOIN
d_paises AS d_paises ON th_cultivo.pais_id = d_paises.pais_id CONTEXT ('formatted' = 'yes');

```

Fuente: Elaboración propia

Cubo 2: Cosechas

Figura 25 Vista tabla de hechos Data Mart 2: Cosechas

```
CREATE OR REPLACE VIEW dm_cosecha FOLDER = '/vistas' AS
SELECT th_cosecha.superficie_produccion_cosechada AS superficie_produccion_cosechada,
th_cosecha.numero_unidades_producidas AS numero_unidades_producidas,
th_cosecha.rendimiento AS rendimiento,
th_cosecha.costos_produccion AS costos_produccion,
th_cosecha.precio_venta AS precio_venta,
th_cosecha.utilidad_neta AS utilidad_neta,
th_cosecha.utilidad_porcentaje AS utilidad_porcentaje,
th_cosecha.tiempo_cosecha AS tiempo_cosecha,
th_cosecha.tiempo_cosecha_dias AS tiempo_cosecha_dias,
d_tiempo_cosecha.tco_fecha AS fecha,
d_tiempo_cosecha.tco_dia AS dia,
CASE
    WHEN d_tiempo_cosecha.tco_mes= '1' THEN 'Enero'
    WHEN d_tiempo_cosecha.tco_mes= '2' THEN 'Febrero'
    WHEN d_tiempo_cosecha.tco_mes= '3' THEN 'Marzo'
    WHEN d_tiempo_cosecha.tco_mes= '4' THEN 'Abril'
    WHEN d_tiempo_cosecha.tco_mes= '5' THEN 'Mayo'
    WHEN d_tiempo_cosecha.tco_mes= '6' THEN 'Junio'
    WHEN d_tiempo_cosecha.tco_mes= '7' THEN 'Julio'
    WHEN d_tiempo_cosecha.tco_mes= '8' THEN 'Agosto'
    WHEN d_tiempo_cosecha.tco_mes= '9' THEN 'Septiembre'
    WHEN d_tiempo_cosecha.tco_mes= '10' THEN 'Octubre'
    WHEN d_tiempo_cosecha.tco_mes= '11' THEN 'Noviembre'
    WHEN d_tiempo_cosecha.tco_mes= '12' THEN 'Diciembre'
    ELSE 'Desconocido'
END AS mes,

d_tiempo_cosecha.tco_trimestre AS trimestre,
d_tiempo_cosecha.tco_semestre AS semestre,
d_tiempo_cosecha.tco_anio AS anio,
d_tiempo_cosecha.tco_temporada AS temporada,
d_empresa.emp_nombre_comercial AS empresa,
d_empresa.emp_ruc AS ruc,
d_especie.esp_nombre AS especie,
d_especie_categoria.cat_nombre AS categoria,
d_hacienda.hac_nombre AS hacienda,
d_parcela.par_nombre AS parcela,
d_tipo_suelo.tsu_nombre AS tipo_suelo,
d_ciudad.ciudad AS ciudad,
d_provincias.provincia AS provincia,
d_paises.pais AS pais

FROM (((((((th_cosecha AS th_cosecha INNER JOIN
d_tiempo_cosecha AS d_tiempo_cosecha ON th_cosecha.id_cosecha = d_tiempo_cosecha.id_tco ) INNER JOIN
d_empresa AS d_empresa ON th_cosecha.emp_id = d_empresa.emp_id ) INNER JOIN
d_especie AS d_especie ON th_cosecha.esp_id = d_especie.esp_id ) INNER JOIN
d_especie_categoria AS d_especie_categoria ON th_cosecha.cat_id = d_especie_categoria.cat_id ) INNER JOIN
d_hacienda AS d_hacienda ON th_cosecha.hac_id = d_hacienda.hac_id ) INNER JOIN
d_parcela AS d_parcela ON th_cosecha.par_id = d_parcela.par_id ) INNER JOIN
d_tipo_suelo AS d_tipo_suelo ON th_cosecha.tsu_id = d_tipo_suelo.tsu_id ) INNER JOIN
d_ciudad AS d_ciudad ON th_cosecha.ciudad_id = d_ciudad.ciudad_id ) INNER JOIN
d_provincias AS d_provincias ON th_cosecha.provincia_id = d_provincias.provincia_id ) INNER JOIN
d_paises AS d_paises ON th_cosecha.pais_id = d_paises.pais_id CONTEXT ('formatted' = 'yes');
```

Fuente: Elaboración propia

Cubo 3: Plagas y Enfermedades

Figura 26 Vista tabla de hechos Data Mart 3: Plagas y Enfermedades

```
CREATE OR REPLACE VIEW dm_plaga FOLDER = '/vistas' AS
SELECT th_plaga.cantidad_plantas_afectadas AS cantidad_plantas_afectadas,
th_plaga.cantidad_hec_afectadas AS cantidad_hec_afectadas,
th_plaga.porcentaje_hec_afectadas AS porcentaje_hec_afectadas,
d_tiempo_plaga.tpe_fecha AS fecha,
d_tiempo_plaga.tpe_dia AS dia,
CASE
    WHEN d_tiempo_plaga.tpe_mes= '1' THEN 'Enero'
    WHEN d_tiempo_plaga.tpe_mes= '2' THEN 'Febrero'
    WHEN d_tiempo_plaga.tpe_mes= '3' THEN 'Marzo'
    WHEN d_tiempo_plaga.tpe_mes= '4' THEN 'Abril'
    WHEN d_tiempo_plaga.tpe_mes= '5' THEN 'Mayo'
    WHEN d_tiempo_plaga.tpe_mes= '6' THEN 'Junio'
    WHEN d_tiempo_plaga.tpe_mes= '7' THEN 'Julio'
    WHEN d_tiempo_plaga.tpe_mes= '8' THEN 'Agosto'
    WHEN d_tiempo_plaga.tpe_mes= '9' THEN 'Septiembre'
    WHEN d_tiempo_plaga.tpe_mes= '10' THEN 'Octubre'
    WHEN d_tiempo_plaga.tpe_mes= '11' THEN 'Noviembre'
    WHEN d_tiempo_plaga.tpe_mes= '12' THEN 'Diciembre'
    ELSE 'Mes Desconocido'
END AS mes,
d_tiempo_plaga.tpe_trimestre AS trimestre,
d_tiempo_plaga.tpe_semestre AS semestre,
d_tiempo_plaga.tpe_anio AS anio,
d_tiempo_plaga.tpe_temporada AS temporada,
d_empresa.emp_nombre_comercial AS empresa,
d_empresa.emp_ruc AS ruc,
d_hacienda.hac_nombre AS hacienda,
d_parcela.par_nombre AS parcela,
d_especie.esp_nombre AS especie,
d_especie_categoria.cat_nombre AS categoria,
d_tipo_suelo.tsu_nombre AS tipo_suelo,
d_ciudad.ciudad AS ciudad,
d_provincias.provincia AS provincia,
d_paises.pais AS pais
FROM (((((((th_plaga AS th_plaga INNER JOIN
d_tiempo_plaga AS d_tiempo_plaga ON th_plaga.id_plaga = d_tiempo_plaga.id_tpe ) INNER JOIN
d_empresa AS d_empresa ON th_plaga.emp_id = d_empresa.emp_id ) INNER JOIN
d_hacienda AS d_hacienda ON th_plaga.hac_id = d_hacienda.hac_id ) INNER JOIN
d_parcela AS d_parcela ON th_plaga.par_id = d_parcela.par_id ) INNER JOIN
d_especie AS d_especie ON th_plaga.esp_id = d_especie.esp_id ) INNER JOIN
d_especie_categoria AS d_especie_categoria ON th_plaga.cat_id = d_especie_categoria.cat_id ) INNER JOIN
d_tipo_suelo AS d_tipo_suelo ON th_plaga.tsu_id = d_tipo_suelo.tsu_id ) INNER JOIN
d_ciudad AS d_ciudad ON th_plaga.ciudad_id = d_ciudad.ciudad_id ) INNER JOIN
d_provincias AS d_provincias ON th_plaga.provincia_id = d_provincias.provincia_id ) INNER JOIN
d_paises AS d_paises ON th_plaga.pais_id = d_paises.pais_id CONTEXT ('formatted' = 'yes');
```

Fuente: Elaboración propia

Cubo 4: Haciendas y Parcelas

Figura 27 Vista tabla de hechos Data Mart 4: Haciendas y Parcelas

```
CREATE OR REPLACE VIEW dm_hapa FOLDER = '/vistas' AS
SELECT th_hapa.area_dedicada_cultivo AS area_dedicada_cultivo,
th_hapa.hectareas_cultivadas AS hectareas_cultivadas,
th_hapa.superficie_dedicada_cul AS superficie_dedicada_culivo,
d_empresa.emp_nombre_comercial AS empresa,
d_empresa.emp_ruc AS ruc,
d_hacienda.hac_nombre AS hacienda,
d_parcela.par_nombre AS parcela,
d_tipo_suelo.tsu_nombre AS tipo_suelo,
d_cultivo.numero_hectareas AS numero_hectareas,
d_ciudad.ciudad AS ciudad,
d_provincias.provincia AS provincia,
d_paises.pais AS pais
FROM ((((((th_hapa AS th_hapa INNER JOIN
d_empresa AS d_empresa ON th_hapa.emp_id = d_empresa.emp_id ) INNER JOIN
d_hacienda AS d_hacienda ON th_hapa.hac_id = d_hacienda.hac_id ) INNER JOIN
d_parcela AS d_parcela ON th_hapa.par_id = d_parcela.par_id ) INNER JOIN
d_tipo_suelo AS d_tipo_suelo ON th_hapa.tsu_id = d_tipo_suelo.tsu_id ) INNER JOIN
d_cultivo AS d_cultivo ON th_hapa.cul_id = d_cultivo.cul_id ) INNER JOIN
d_ciudad AS d_ciudad ON th_hapa.ciudad_id = d_ciudad.ciudad_id ) INNER JOIN
d_provincias AS d_provincias ON th_hapa.provincia_id = d_provincias.provincia_id ) INNER JOIN
d_paises AS d_paises ON th_hapa.pais_id = d_paises.pais_id CONTEXT ('formatted' = 'yes');
```

Fuente: Elaboración propia

Cubo 5: Medida de Sensores

Figura 28 Vista tabla de hechos Data Mart 5: Medida de Sensores

```
CREATE OR REPLACE VIEW dm_lecturas FOLDER = '/vistas'
PRIMARY KEY ( 'mac' ) AS
SELECT
cast(lecturas.valor AS DOUBLE PRECISION) AS valor_promedio,
cast(lecturas.valor AS DOUBLE PRECISION) AS valor_maximo,
cast(lecturas.valor AS DOUBLE PRECISION) AS valor_minimo,
stdevp(cast(lecturas.valor AS DOUBLE PRECISION)) AS desviacion_estandar,
varp(cast(lecturas.valor AS DOUBLE PRECISION)) AS varianza,
lecturas.fecha AS tms_fecha,
extract(DAY from to_date(lecturas.fecha, 'DD MM YYYY')) AS tms_dia,
extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY')) AS tms_mes,
CASE
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '01' THEN 'Primer Trimestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '02' THEN 'Primer Trimestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '03' THEN 'Primer Trimestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '04' THEN 'Segundo Trimestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '05' THEN 'Segundo Trimestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '06' THEN 'Segundo Trimestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '07' THEN 'Tercer Trimestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '08' THEN 'Tercer Trimestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '09' THEN 'Tercer Trimestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '10' THEN 'Cuarto Trimestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '11' THEN 'Cuarto Trimestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '12' THEN 'Cuarto Trimestre'
    ELSE 'Trimestre Desconocido'
```

```

END AS tms_trimestre,
CASE
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '01' THEN 'Primer Semestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '02' THEN 'Primer Semestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '03' THEN 'Primer Semestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '04' THEN 'Primer Semestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '05' THEN 'Primer Semestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '06' THEN 'Primer Semestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '07' THEN 'Segundo Semestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '08' THEN 'Segundo Semestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '09' THEN 'Segundo Semestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '10' THEN 'Segundo Semestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '11' THEN 'Segundo Semestre'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '12' THEN 'Segundo Semestre'
    ELSE 'Semestre Desconocido'
END
extract(YEAR from to_date(cult_fecha, 'DD MM YYYY')) AS tms_anio,
CASE
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '01' THEN 'Invierno'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '02' THEN 'Invierno'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '03' THEN 'Invierno'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '04' THEN 'Invierno'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '05' THEN 'Invierno'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '06' THEN 'Invierno'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '07' THEN 'Verano'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '08' THEN 'Verano'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '09' THEN 'Verano'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '10' THEN 'Verano'
    WHEN extract(MONTH from to_date(lecturas.fecha, 'DD MM YYYY'))= '11' THEN 'Verano'
    WHEN extract(MONTH from to_date(cult_fecha, 'DD MM YYYY'))= '12' THEN 'Verano'
    ELSE 'Temporada Desconocida'
END AS tms_temporada,
e_empresa.emp_ruc AS ruc,
e_empresa.emp_nombre_comercial AS empresa,
e_area.ar_nombre AS hacienda,
e_subarea.sar_nombre AS parcela,
lecturas.mac AS mac,
iot_ted.ted_name AS unidad

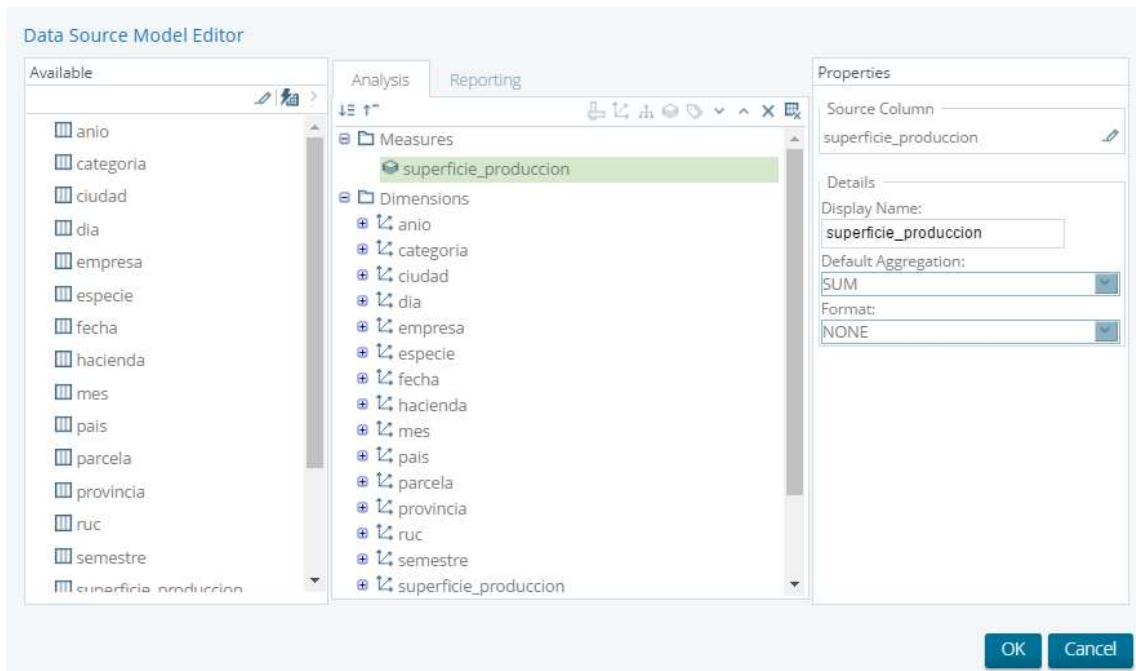
```

Fuente: Elaboración propia

- Crear modelos de fuentes de datos. Una vez creadas las vistas correspondientes, con los KPIs identificados en la metodología, procedemos a crear el modelo de fuente de datos mediante la herramienta que nos proporciona Pentaho, revisar cada uno de los componentes en Anexo Q.

Cubo 1: Cultivos

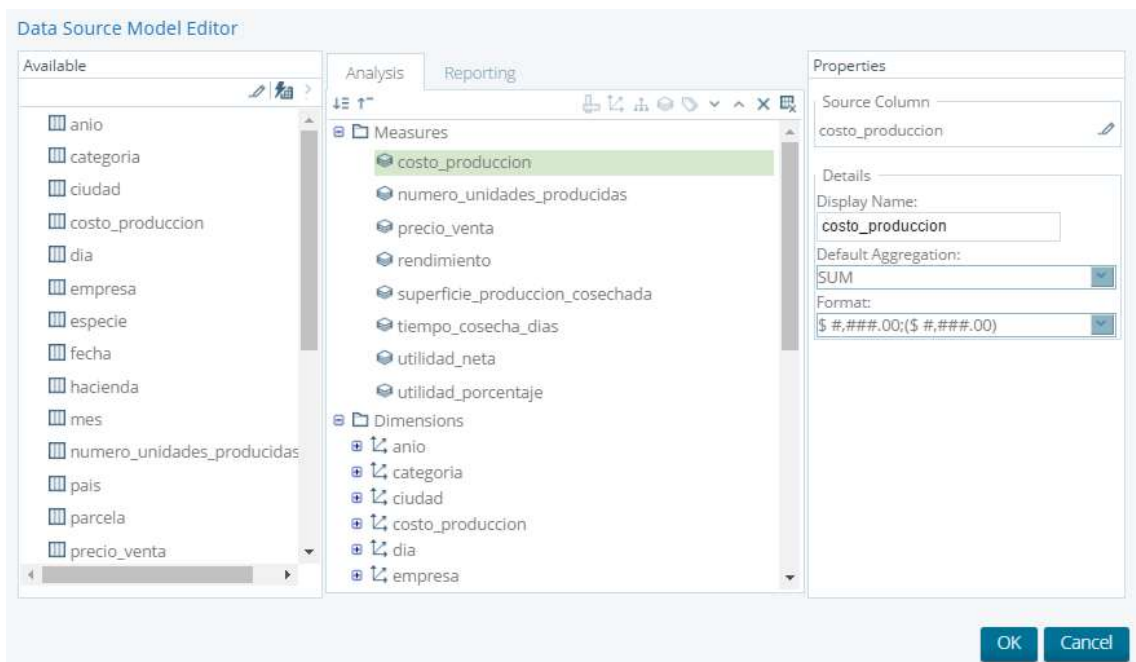
Figura 29 Model Editor Cultivo



Fuente: Elaboración propia

Cubo 2: Cosechas

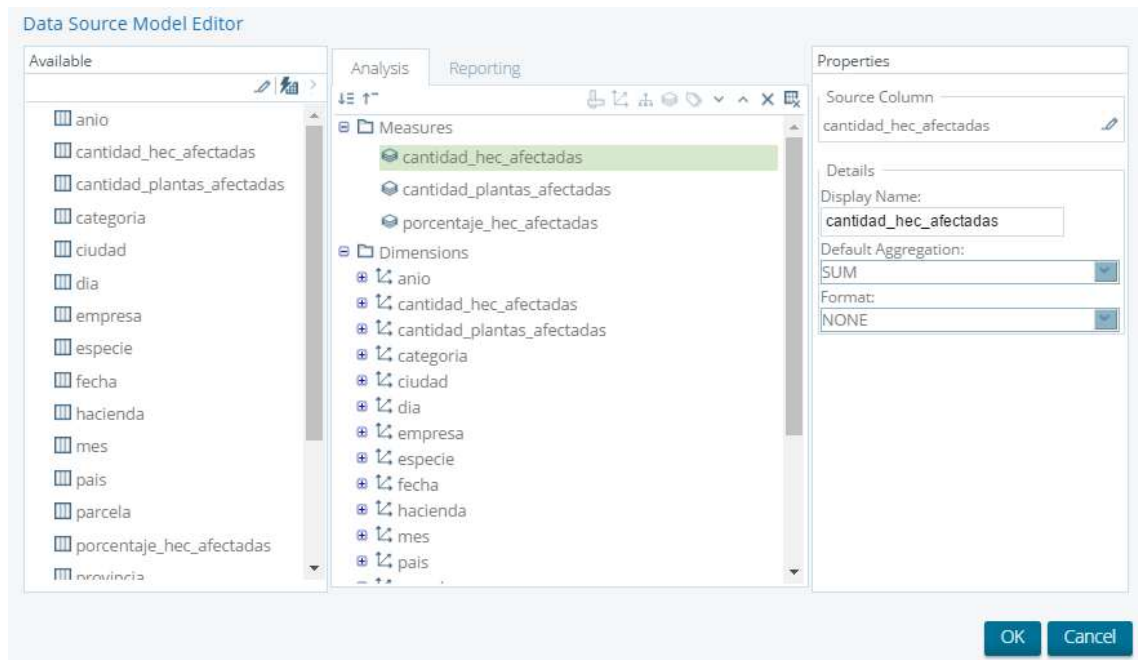
Figura 30 Model Editor Cosecha



Fuente: Elaboración propia

Cubo 3: Plagas y Enfermedades

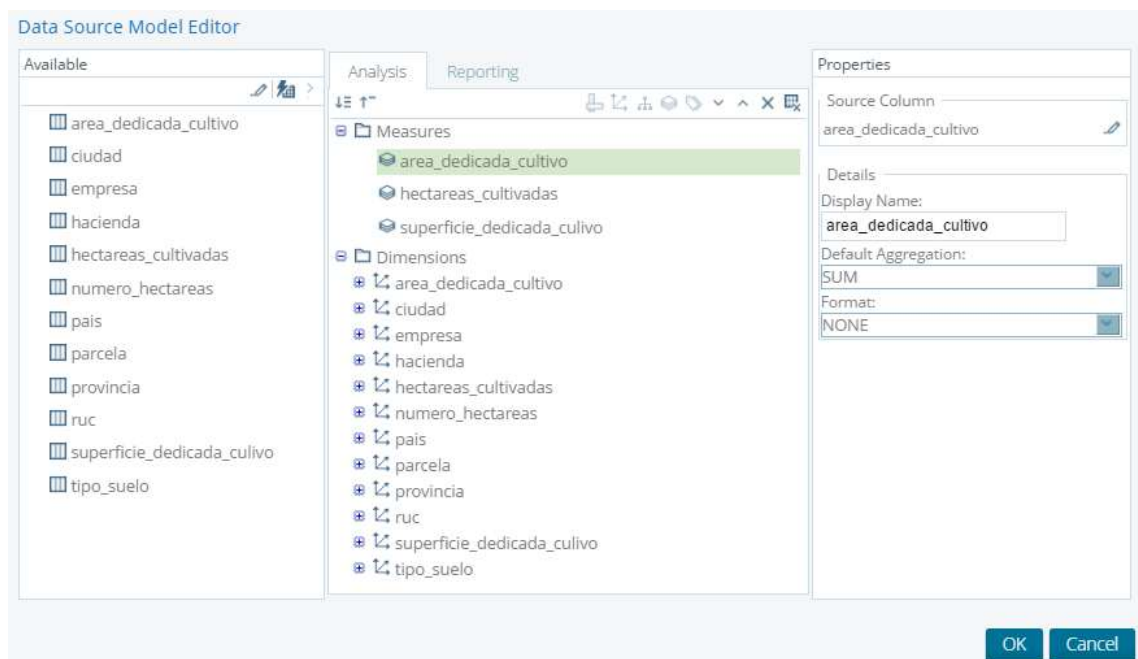
Figura 31 Model Editor Plagas y Enfermedades



Fuente: Elaboración propia

Cubo 4: Haciendas y Parcelas

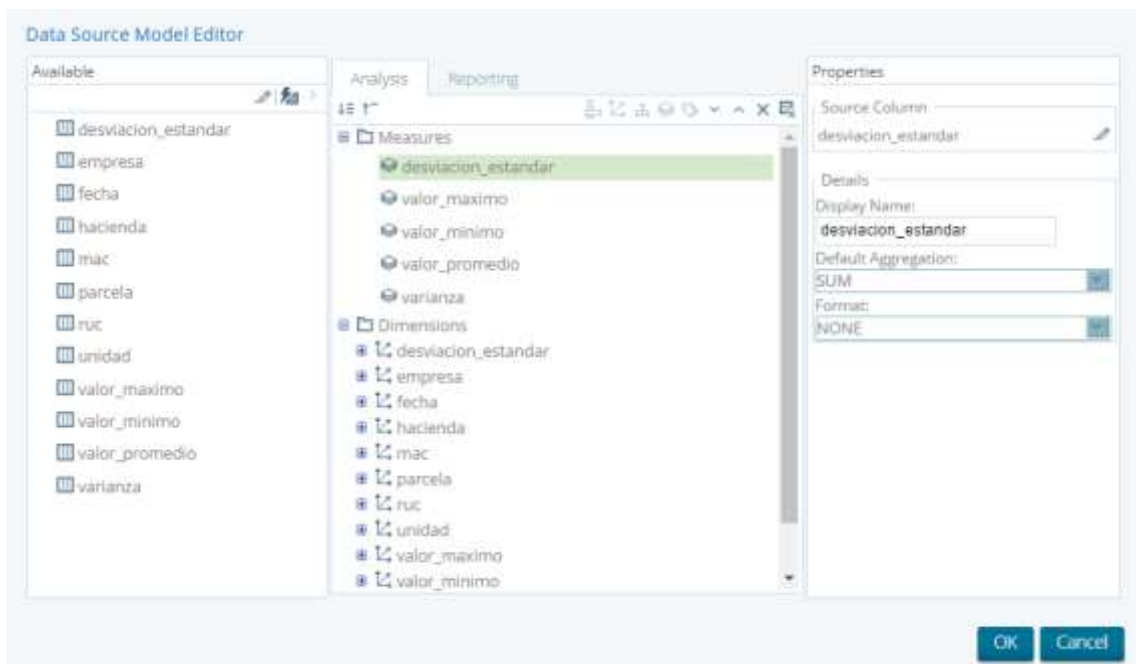
Figura 32 Model Editor Haciendas y Parcelas



Fuente: Elaboración propia

Cubo 5: Medida de Sensores

Figura 33 Model Editor Medida de Sensores

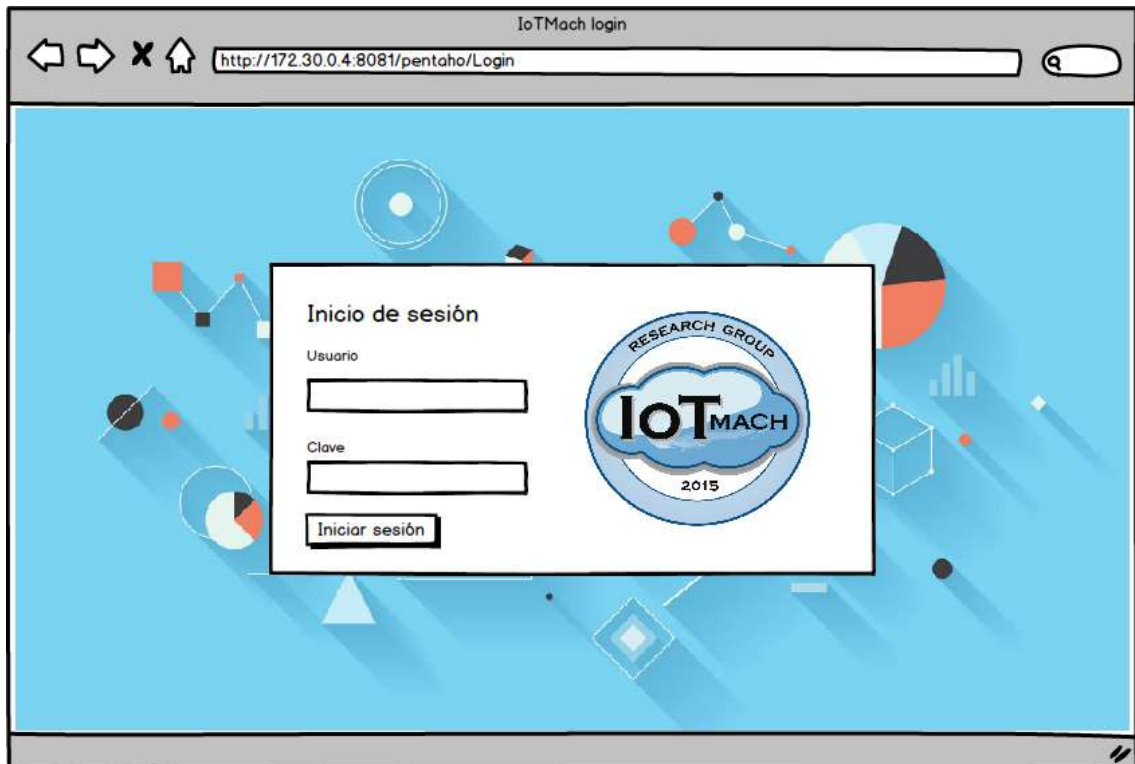


Fuente: Elaboración propia

- Diseño del Dashboard. Para el diseño del Dashboard, se usó la herramienta Balsamiq Mockups 3. Dando como resultado los siguientes:

Página principal o login, se debe acceder según su acceso asignado, si es usuario para visualizar o usuario administrador.

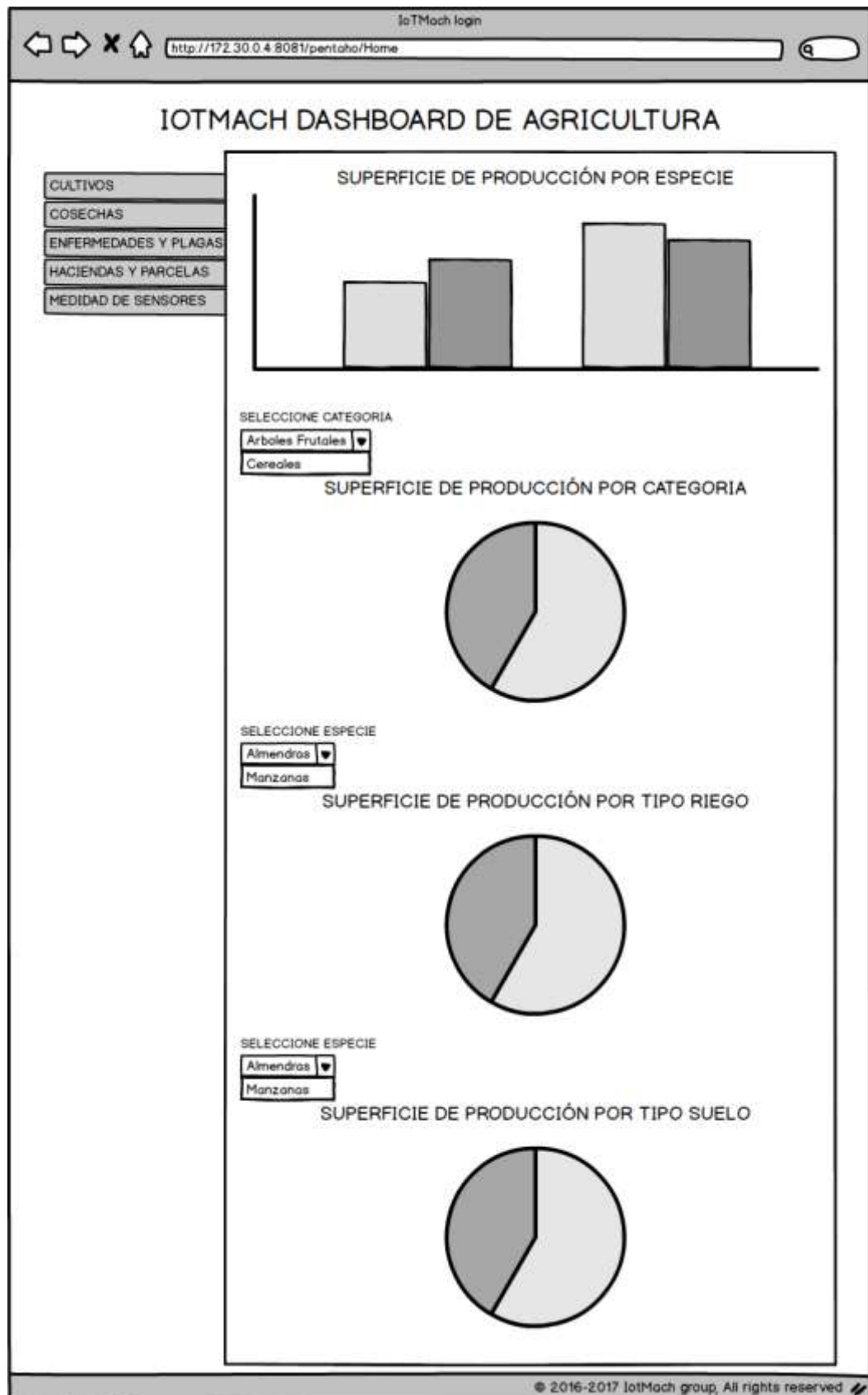
Figura 34 Diseño login



Fuente: Elaboración propia

Para el diseño del Dashboard se ha tratado de obtener toda la información importante en una sola ventana de visualización, la siguiente imagen representa el diseño del Data Mart 1: Cultivo:

Figura 35 Diseño Data Mart 1



Fuente: Elaboración propia

2.4.3 Herramientas utilizadas. La mayoría de herramientas utilizadas son open source. A continuación, se presenta y se describe cada herramienta utilizada.

Figura 36 Herramientas utilizadas



Fuente: Elaboración propia.

- ✓ Pentaho. Servidor de cubos OLAP y se integra a Jpivot.[32]
- ✓ Spoon. Herramienta para el diseño del ETL (Extracción, Transformación y Carga) de datos.[32]
- ✓ Apache-tomcat. Es el servidor web utilizado para aplicaciones JAVA.[32]
- ✓ Toad Data Modeler 5.4. Herramienta para crear diagramas Entidad-Relación.[33]

- ✓ Jpivot. Es el visor de los cubos que son administrados por un servidor OLAP.[32]
- ✓ CDE Dashboard. tablero de instrumentos editor de comunidades, es una herramienta gráfica para la creación, edición y vista previa de Pentaho. [32]
- ✓ Pivot4j. es una librería que nos permite pintar cubos, dispone también de una aplicación analítica. [34]
- ✓ Saiku Analytics. permite visualizar y realizar análisis de datos de forma fácil e intuitiva. Reemplazará Jpivot, mediante la cual se pueden construir vistas propias arrastrando y soltando campos.[35]
- ✓ PostgreSQL. Es el servidor de bases de datos donde se almacena del Data warehouse del sistema de Soporte de Decisiones.[36]
- ✓ Apache Hive. es una infraestructura de almacenamiento de datos construida sobre Hadoop para proporcionar agrupación, consulta, y análisis de datos. [37]
- ✓ Oracle. herramienta cliente/servidor para la gestión de Bases de Datos. [38]
- ✓ Hadoop. framework que permite el procesamiento de grandes volúmenes de datos a través de clusters. [39]
- ✓ Hortonworks. Permite el procesamiento distribuido de grandes conjuntos de datos a través de clusters de computadores.[40]
- ✓ Denodo. permite la virtualización de datos, proporciona agilidad de negocio mediante la integración de datos dispersos desde cualquier fuente de datos.[41]

2.5.1.1 Spoon Pentaho Data Integration. El proceso ETL se debe realizar mediante Spoon, herramienta libre bajo Pentaho. Se necesita crear una conexión a la Base de datos origen “Postgres”, revisar Anexo R, a la Base de datos destino “Hive” para los datos históricos, revisar Anexo S y a la base de datos destino Oracle con los datos correspondientes al último año de producción, revisar Anexo T.

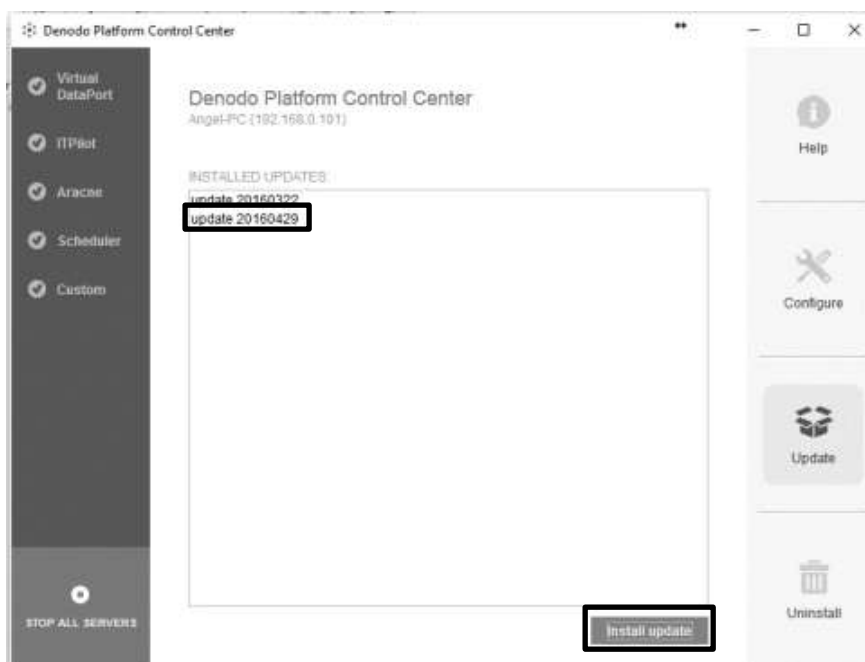
Figura 37 Conexión BD Destino



Fuente: Elaboración propia

2.5.1.2 *Denodo Platform*. Es necesario actualizar Denodo Platform para virtualizar los datos que Pentaho tiene que recibir y poner visualizar los datos mediante los visores y el Dashboard. Este procedimiento lo podemos revisar en el Anexo U. Debemos tener el Update del 29 de abril del 2016 para poder hacer la conexión:

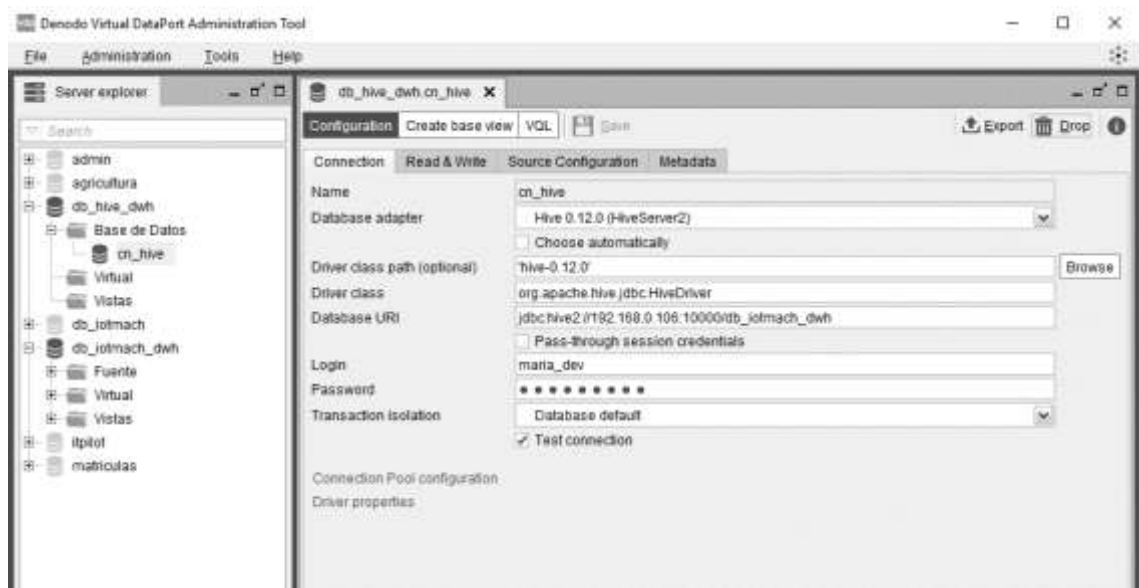
Figura 38 Actualización Denodo



Fuente: Elaboración propia

Se debe crear una conexión correspondiente para Apache Hive, revisar Anexo V. Luego de esto realizar las vistas correspondientes de acuerdo a los medidores de desempeño descritos en la metodología.

Figura 39 Conexión Denodo a Hive



Fuente: Elaboración propia

Para luego proceder con la creación de las vistas, logrando obtener las siguientes vistas:

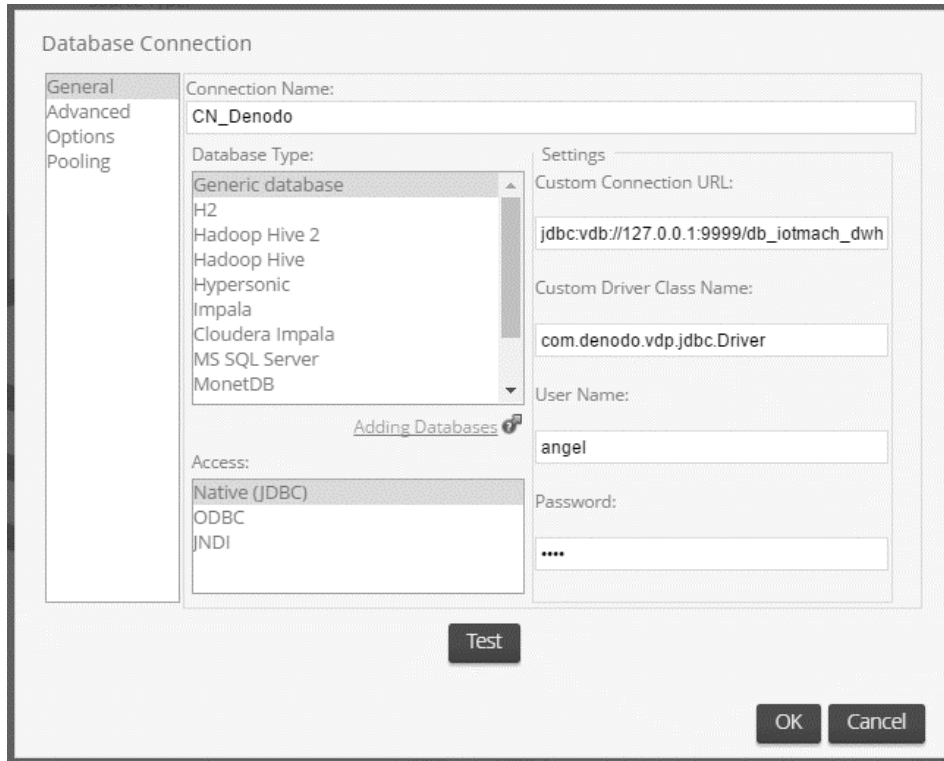
Figura 40 Vista Cultivo

superflide_produccion	emp_nombre_comercial	emp_ruc	esp_nombre	cat_nombre
247.0	DYUYAY	0704640333001	Almendras	Arboles Frutales
549.0	DYUYAY	0704640333001	Almendras	Arboles Frutales
375.0	DYUYAY	0704640333001	Caña de Azúcar	Caña de Azúcar
179.0	DYUYAY	0704640333001	Almendras	Arboles Frutales
285.0	DYUYAY	0704640333001	Malt	Cereales
537.0	DYUYAY	0704640333001	Malt	Cereales
667.0	DYUYAY	0704640333001	Manzanas	Arboles Frutales
25.0	DYUYAY	0704640333001	Almendras	Arboles Frutales
257.0	DYUYAY	0704640333001	Cebada	Cereales
235.0	DYUYAY	0704640333001	Cebada	Cereales
471.0	DYUYAY	0704640333001	Caña de Azúcar	Caña de Azúcar
234.0	PALMATECHO	0302262290001	Almendras	Arboles Frutales

Fuente: Elaboración propia

2.5.1.3 *Pentaho*. En esta herramienta es necesario crear un Data Source, el cual se obtiene desde Denodo, creando una sentencia dirigida a las vistas creadas anteriormente en Denodo, para esto debemos crear la conexión a Denodo, revisar Anexo W. Una vez realizada la conexión:

Figura 41 Conexión Pentaho a Denodo

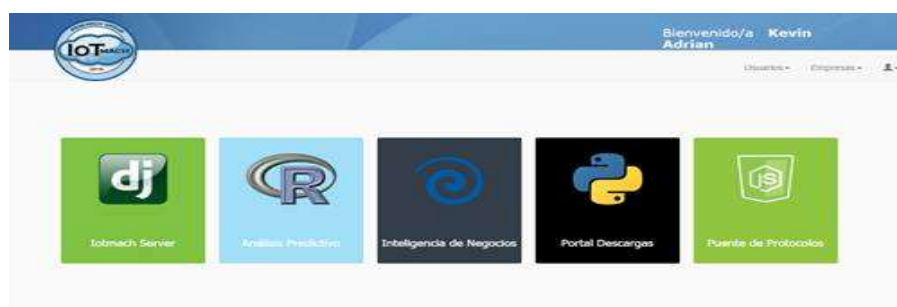


Fuente: Elaboración propia

2.6 Ejecución del prototipo

Se accede al menú principal ingresando al sistema de autenticación IOTMACH:

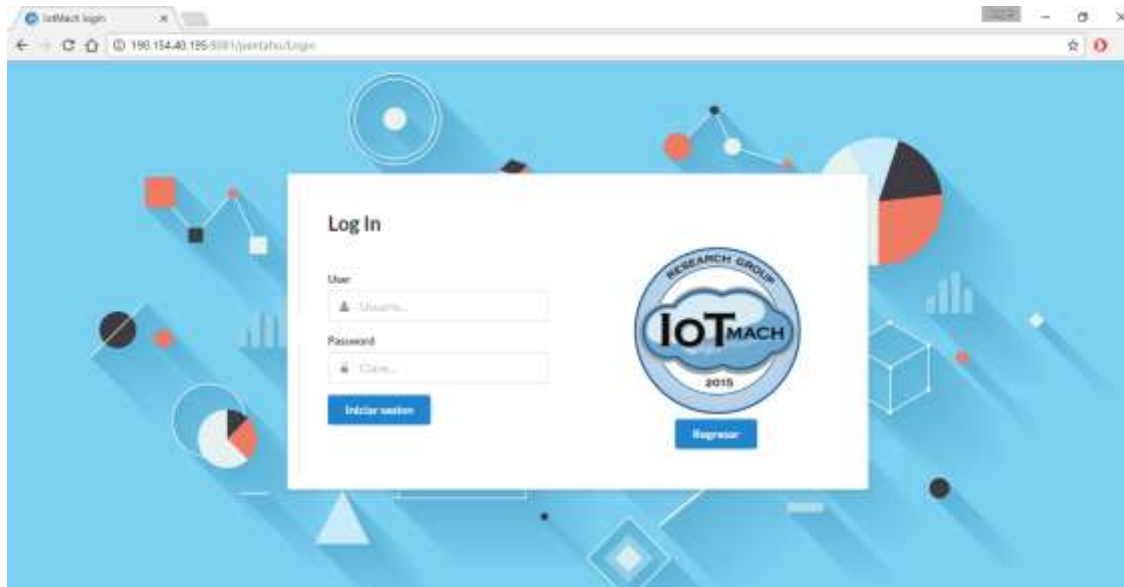
Figura 42 Sistema Autenticación IOTMACH



Fuente: Elaboración Propia

Luego de este acceso, obtenemos la información de la empresa que desea visualizar sus datos, Accediendo a Inteligencia de negocios, que nos lleva a la siguiente ventana:

Figura 43 Página principal pentaho



Fuente: Elaboración propia

Al ingresar se verán las diferentes opciones de medidas del dashboard, a continuación, se presenta brevemente los componentes del Dashboard.

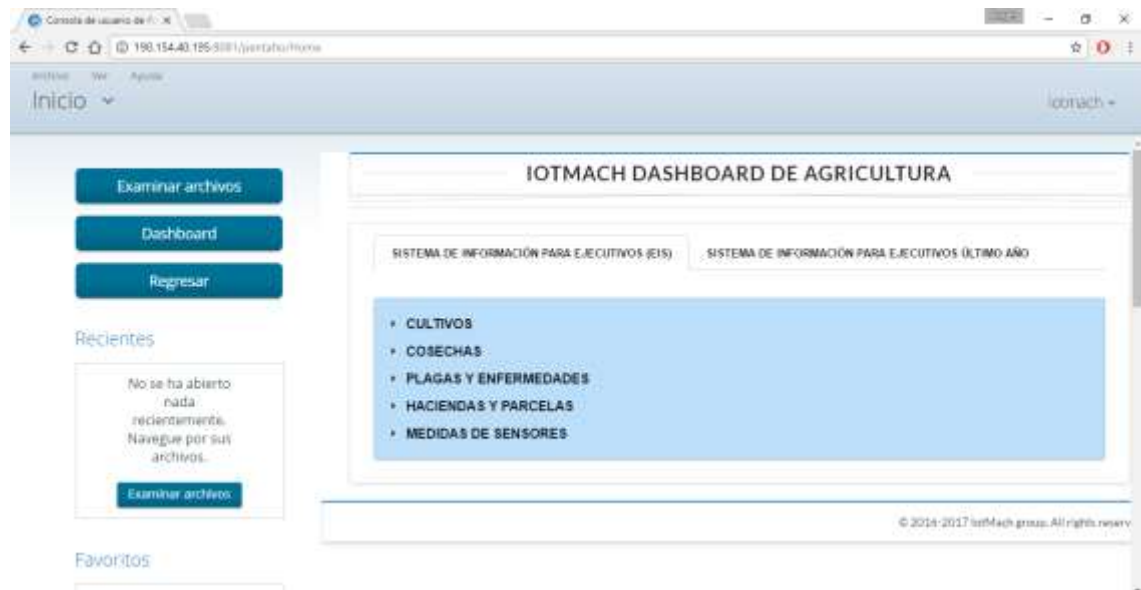
Figura 44 Log In



Fuente: Elaboración propia

Página Principal del Dashboard, por medio del panel ubicado en la parte izquierda de la ventana, podemos navegar en los diferentes indicadores de desempeño.

Figura 45 Dashboard



Fuente: Elaboración propia

Al desplegar las opciones se selecciona el Data Mart que se desea observar, se observa la opción indicando que tipo de estadísticas encontraremos dentro y damos click en Visualizar, el cual dirige a una ventana emergente para mostrar los gráficos.

Figura 46 Panel de selección dashboard



Fuente: Elaboración propia

Los datos mostrados en la siguiente figura, corresponden a los KPIs señalados anteriormente en la metodología, permitiendo agricultor o empresario la visualización de los datos históricos respecto a los cultivos, y tomar decisiones para mejoras de dichos cultivos.

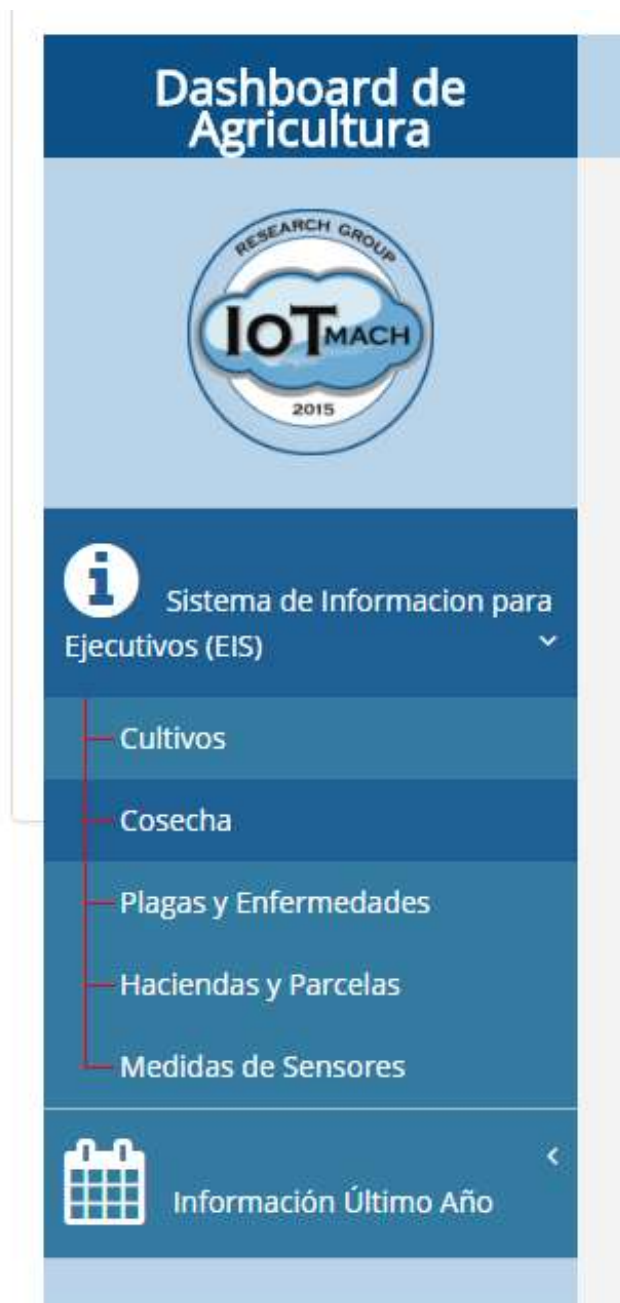
Figura 47 Dashboard Cultivo



Fuente: Elaboración propia

Por medio del menú desplegable, se puede seguir navegando en las otras opciones del Dashboard.

Figura 48 Opciones Dashboard

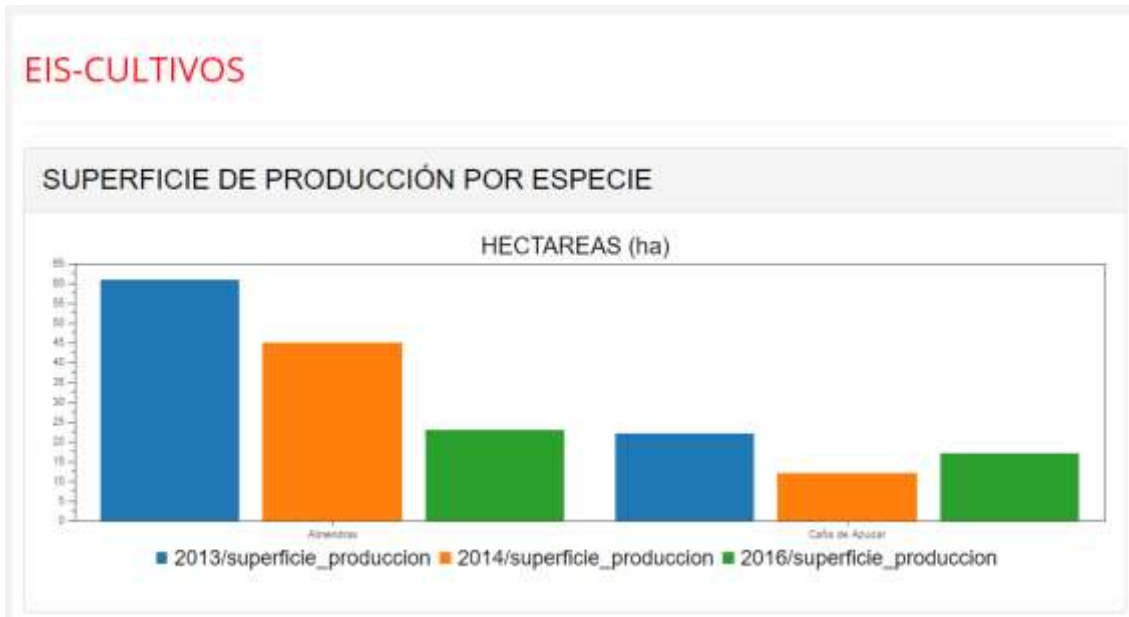


Fuente: Elaboración propia

A continuación, se detalla brevemente los componentes utilizados para la presentación de los datos en el Dashboard CDE Pentaho.

- 1) Componente Bar Chart, en la gráfica se muestra la medida (Superficie de producción), por años, separados por especies para obtener un valor y porcentaje referente a su medida.

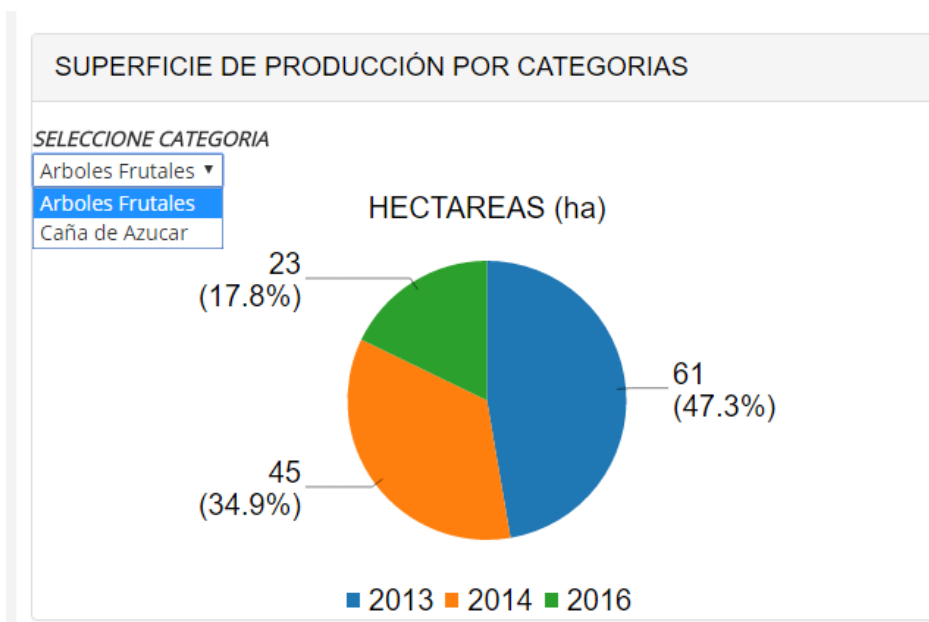
Figura 49 Bar chart Superficie de producción



Fuente: Elaboración propia

- 2) Select Component, por medio de una consulta MDX se obtiene el filtro de Categorías y lo ubicamos en el componente para obtener un gráfico dinámico de acuerdo a la selección del usuario.

Figura 50 Selector Categoría



Fuente: Elaboración propia

- 3) Select Component, de la misma forma se aplica este componente para colocar más filtros y poder presentar la información de manera interactiva por medio de consultas MDX ligadas con los componentes anteriores se obtiene el filtro de acuerdo a la selección del usuario.

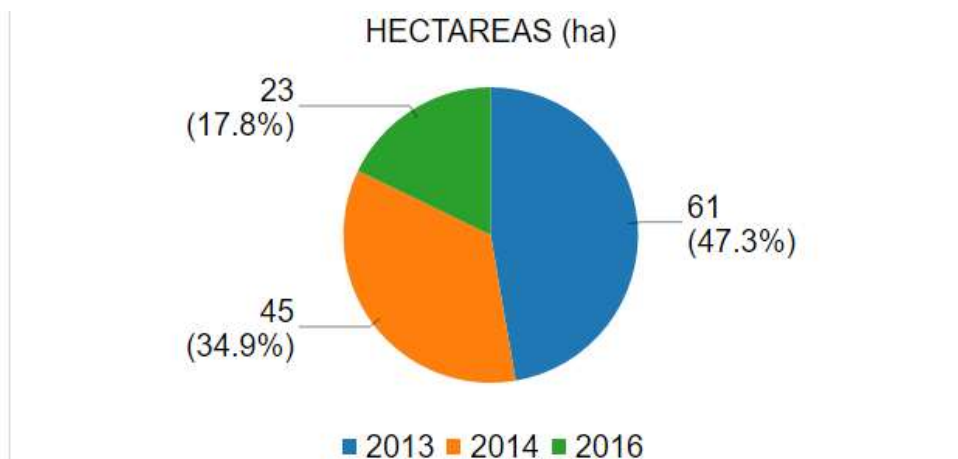
Figura 51 Selector Especie y Año



Fuente: Elaboración propia

- 4) Pie Chart, por medio de este componente podemos mostrar una medida con varias dimensiones, en este caso Superficie de producción por categorías.

Figura 52 Superficie de producción por categoría



Fuente: Elaboración propia

- 5) Gauge component, componente proporcionado por Ivy Dashboard component (IvyDC), con la característica de un medidor de combustible ideal para presentar indicadores por especie, año, mes, semestre y trimestre, así obtenemos un valor se superficie de producción personalizado.

Figura 53 Gauge Ivy Component



Fuente: Elaboración propia

- 6) En este caso se ha usado un Select Component, filtrado por tipo de riego, para que el gráfico cambie su medida cada vez que el usuario seleccione otro tipo de riego. Además, en la parte derecha una selección de Especie y año para mostrar los datos en el indicador Gauge Ivy Component.

Figura 54 Superficie de producción por tipo de riego



Fuente: Elaboración propia

Posterior a esto, luego de haber seleccionado Tipo de riego, Especie y año, se puede seguir filtrando la superficie de producción mediante los siguientes indicadores:

Figura 55 Indicadores Mes, Semestre, Trimestre



Fuente: Elaboración propia

- De la misma forma para el cuarto componente se ha usado un Select Component, filtrado por tipo de suelo, para que el gráfico cambie su medida cada vez que el usuario seleccione otro tipo de suelo. Además, en la parte derecha una selección de Especie y año para mostrar los datos en el indicador Gauge Ivy Component.

Figura 56 Superficie de producción por tipo de suelo



Fuente: Elaboración propia

Y posterior a esto, luego de haber seleccionado Tipo de suelo, Especie y año, se puede seguir filtrando la superficie de producción mediante los siguientes indicadores:

Figura 57 Indicadores Tipo de Suelo por Mes, Semestre, Trimestre



Fuente: Elaboración propia

3. EVALUACIÓN DEL PROTOTIPO

3.1 Plan de evaluación

Toda propuesta de investigación, sistema o herramienta a implantarse requiere ser evaluada para determinar si ésta cumple o no con la finalidad para el que ha sido desarrollada.

En esta sección de la propuesta tecnológica el plan de pruebas del módulo de soporte de decisiones para agricultura utilizando Big Data e inteligencia de negocios integrado al sistema IoT Mach, el mismo que arrojará una serie de resultados que va a permitir que el equipo de desarrollo del producto constante: errores de función, los mismos que deberán ser corregidos o el correcto funcionamiento del sistema para lo cual fue diseñado. Además, se presenta en este capítulo el propósito del plan de pruebas, sus objetivos y cada uno de los identificadores a ser evaluados en donde se describirá cada requisito a ser probado; para facilitar el proceso de evaluación se ha diseñado un formulario el mismo que registrara todo el proceso evaluativo.

El propósito fundamental del plan de pruebas es detallar que elementos influyen durante este proceso de evaluación del software, tales como: el alcance donde se identifica el tipo de prueba a emplear, los elementos del software a ser probados y las condiciones mismas que estas deben cumplir, los recursos necesarios que intervienen en el proceso de pruebas y las personas responsables de las evaluaciones.

Los objetivos que se plantean dentro del plan de prueba son:

- ✓ Encontrar errores en la Extracción, transformación y carga de los datos, tanto en Hive como en Oracle respectivamente.
- ✓ Verificar que los KPIs del EIS sean presentados correctamente.
- ✓ Determinar que el diseño del Dashboard cumpla con las expectativas del diseño antes modelado y presente toda la información obtenida mediante el ETL, y sea de fácil uso, incluso en la primera experiencia.

Para la ejecución de las pruebas de procesos de carga se ha seleccionado cuatro especialistas en Ingeniería de sistemas que con sus conocimientos darán su visto bueno del éxito de la ejecución de dichas pruebas, para la ejecución de las pruebas de

usabilidad se ha seleccionado cuatro especialistas en Ingeniería agropecuaria que evaluarán la experiencia de los datos mostrados mediante el dashboard de agricultura. Revisar Anexo A que contiene datos de los especialistas.

3.1.1 *Pruebas de Ejecución de Procesos de Carga (Unitarias).* Para revisar la funcionalidad del sistema de soporte de decisiones, es muy importante tener todos los datos en nuestro almacén de datos DW, por ello se ha elaborado el siguiente plan de pruebas con el objetivo de garantizar la carga correcta de los datos necesarios.

Cuadro 7 Matriz Pruebas ETL

Nº DE PRUEBA	FUNCIÓN	ACCIÓN DE LA PRUEBA	ANEXO
PU_1	Cultivos	ETL - HIVE	C
PU_2	Cosechas	ETL - HIVE	D
PU_3	Plagas y Enfermedades	ETL - HIVE	E
PU_4	Haciendas y Parcelas	ETL - HIVE	F
PU_5	Medidas de Sensores	ETL - HIVE	G
PU_6	Cultivos	ETL - ORACLE	H
PU_7	Cosechas	ETL - ORACLE	I
PU_8	Plagas y Enfermedades	ETL - ORACLE	J
PU_9	Haciendas y Parcelas	ETL - ORACLE	K
PU_10	Medidas de Sensores	ETL - ORACLE	L
Fuente: Elaboración Propia			

Las encuestas elaboradas son en base a la carga de los datos mediante la herramienta Spoon Data Integration, podemos observar su explicación en el Anexo B.

3.1.2 *Pruebas de Usabilidad.* Es primordial que el usuario se familiarice con la información presentada en el Dashboard y de gran importancia que su uso sea fácil desde la primera vista, para esto se ha diseñado una serie de preguntas que pueden comprobar que la información presentada es completamente entendible.

Cuadro 8 Matriz Pruebas Usabilidad

Nº	DESCRIPCIÓN
1	¿Es fácil utilizar el Dashboard por primera vez?
2	¿El panel de visualización es claro?
3	¿Las acciones a realizar son intuitivas?
4	¿El usuario puede controlar los gráficos de forma conveniente?
5	¿Los términos usados en el dashboard para describir medidas, indican de forma clara lo que representan?
6	¿Es posible observar de forma global lo que abarca el contenido de cada sección?
7	¿El texto de los mensajes de cada gráfico es significativo e identifica los valores necesarios para la toma de decisiones?
8	¿Los colores empleados para la presentación de los gráficos dentro del dashboard, van en la línea del logo del proyecto?
9	¿El diseño de la interfaz de cada sección, sigue la misma consistencia en todas las secciones del dashboard?
10	¿Existen elementos que representen los datos globales de cada sección?
11	¿Le han parecido claros y representativos los nombres y descripciones que aparecen en cada gráfico del dashboard?
12	¿La información que se presenta en cada gráfico tiene el tamaño adecuado?
13	¿Cree que este dashboard puede ser implementado para toma de decisiones en agricultura?
Fuente: Elaboración Propia	

3.2 Resultados de la evaluación

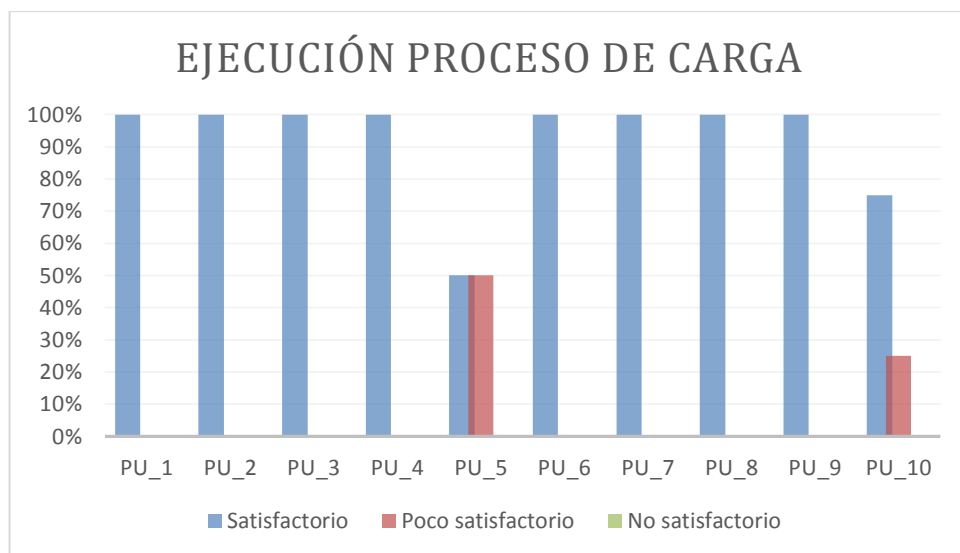
3.2.1 Resultados ejecución de Procesos de Carga (Unitarias)

Tabla 2 Matriz Resultados Pruebas ETL

Nº DE PRUEBA	FUNCIÓN	ACCIÓN DE LA PRUEBA	ANEXO	RESULTADOS			OBSERVACIONES
				Satisfactorio	Poco satisfactorio	No satisfactorio	
PU_1	Cultivos	ETL - HIVE	C	100%	0%	0%	
PU_2	Cosechas	ETL - HIVE	D	100%	0%	0%	
PU_3	Plagas y Enfermedades	ETL - HIVE	E	100%	0%	0%	
PU_4	Haciendas y	ETL -	F	100%	0%	0%	

	Parcelas	HIVE					
PU_5	Medidas de Sensores	ETL - HIVE	G	50%	50%	0%	Lags que causan demora
PU_6	Cultivos	ETL - ORACLE	H	100%	0%	0%	
PU_7	Cosechas	ETL - ORACLE	I	100%	0%	0%	
PU_8	Plagas y Enfermedades	ETL - ORACLE	J	100%	0%	0%	
PU_9	Haciendas y Parcelas	ETL - ORACLE	K	100%	0%	0%	
PU_10	Medidas de Sensores	ETL - ORACLE	L	75%	25%	0%	Lags que causan demora
Fuente: Elaboración Propia							

Figura 58 Resultado Procesos de carga



Fuente: Elaboración Propia

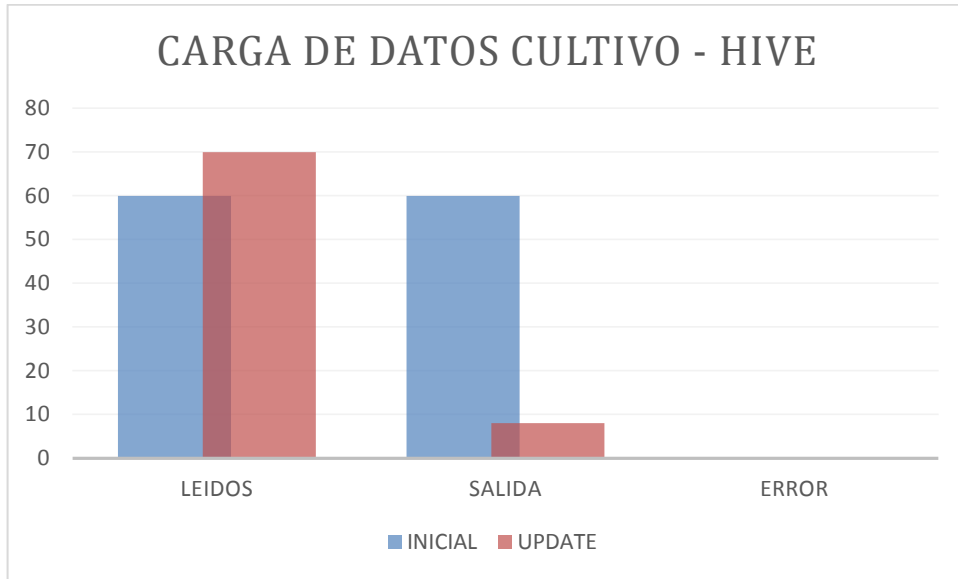
Mediante las pruebas de carga de datos, se obtuvo un 92% de satisfacción, un 8% de poca satisfacción y un 0% de no satisfacción. Dando como resultado la carga de datos exitosa en Hive y Oracle respectivamente.

Las pruebas de carga de datos, permite verificar la cantidad de datos leídos, cargados y erróneos, así como verificar el tiempo que toma el proceso ETL, revisar Anexo M:

Tabla 3 Carga de datos Cultivo - Hive

CARGA DE DATOS CULTIVO - HIVE				
DATOS	LEIDOS	SALIDA	ERROR	TIEMPO
INICIAL	60	60	0	504
UPDATE	70	8	0	321
Fuente: Elaboración Propia				

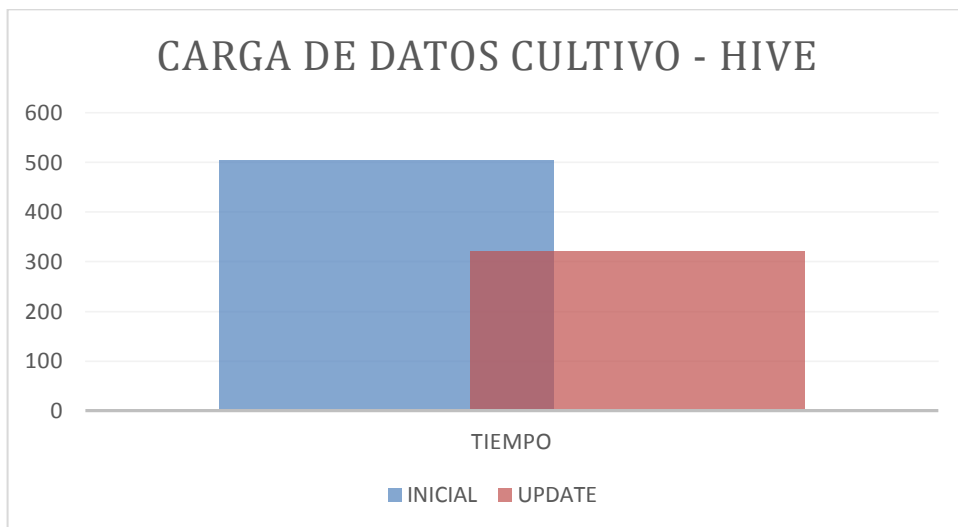
Figura 59 Resultado carga de datos Cultivo - Hive



Fuente: Elaboración Propia

Se puede observar en la figura 55 que los datos iniciales son leídos y cargados exitosamente sin ningún dato erróneo, de la misma forma, su actualización, leyendo más datos, pero cargando solo los nuevos.

Figura 60 Resultado tiempo carga de datos Cultivo - Hive



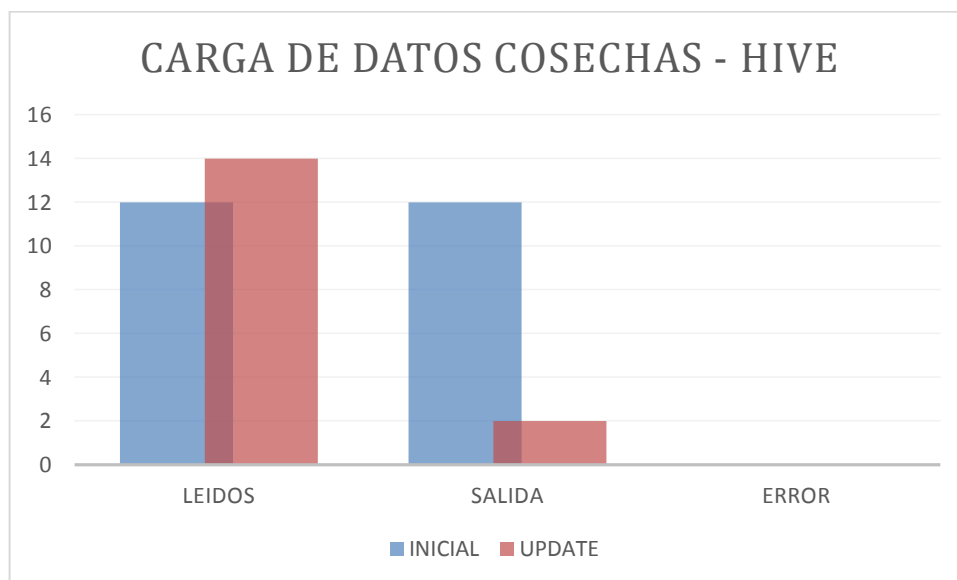
Fuente: Elaboración Propia

Se puede observar en la figura 56 que los datos iniciales son cargados en mayor tiempo, pero su actualización disminuyéndolo, estas actualizaciones son programadas en horas determinadas por el super usuario.

Tabla 4 Carga de datos Cosechas - Hive

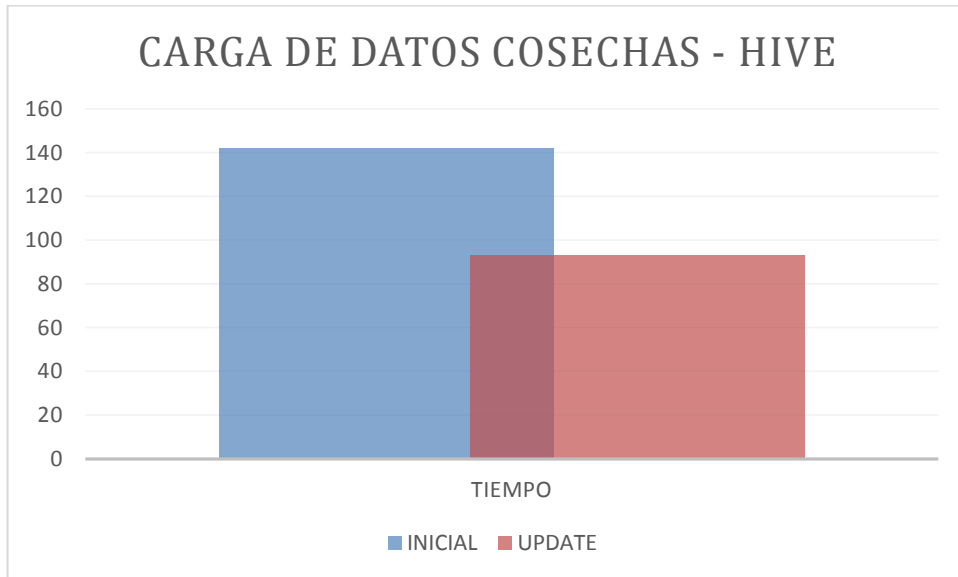
CARGA DE DATOS COSECHAS - HIVE				
DATOS	LEIDOS	SALIDA	ERROR	TIEMPO
INICIAL	12	12	0	142
UPDATE	14	2	0	93
Fuente: Elaboración Propia				

Figura 61 Resultado carga de datos Cosechas - Hive



Fuente: Elaboración Propia

Figura 62 Resultado tiempo carga de datos Cosechas - Hive

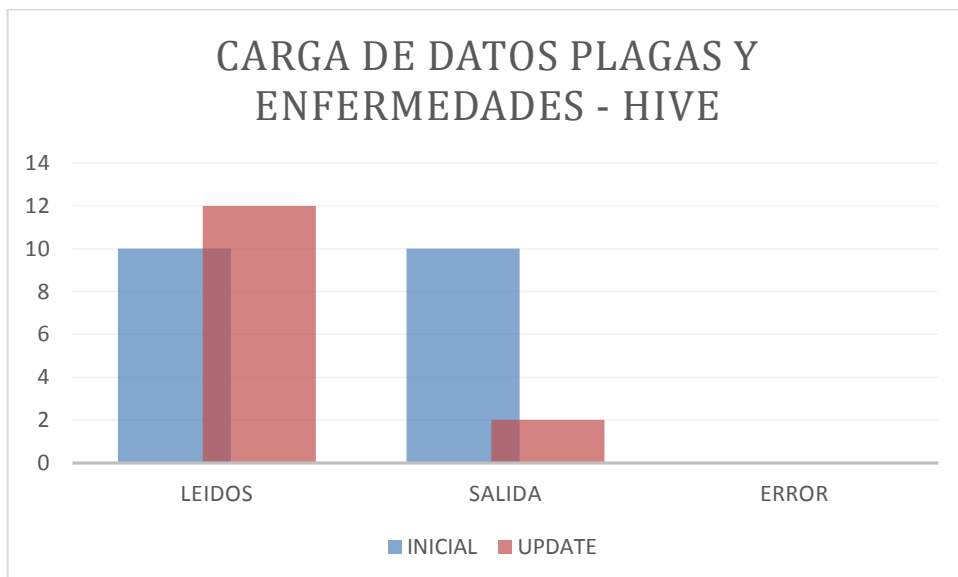


Fuente: Elaboración Propia

Tabla 5 Carga de datos Plagas y Enfermedades - Hive

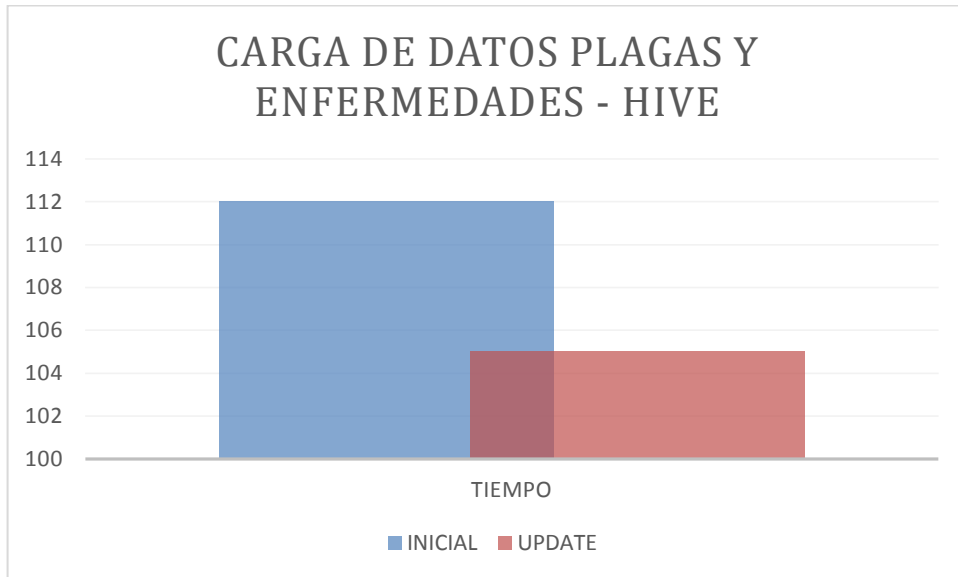
CARGA DE DATOS PLAGAS Y ENFERMEDADES - HIVE				
DATOS	LEIDOS	SALIDA	ERROR	TIEMPO
INICIAL	10	10	0	112
UPDATE	12	2	0	105
Fuente: Elaboración Propia				

Figura 63 Resultado carga de datos Plagas y Enfermedades - Hive



Fuente: Elaboración Propia

Figura 64 Resultado tiempo carga de datos Plagas y Enfermedades - Hive

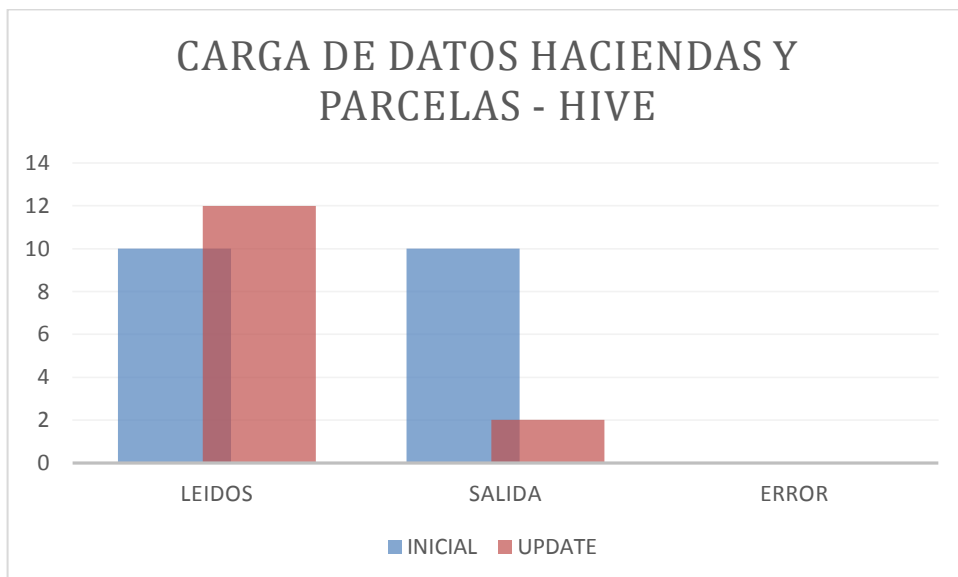


Fuente: Elaboración Propia

Tabla 6 Carga de datos Haciendas y Parcelas - Hive

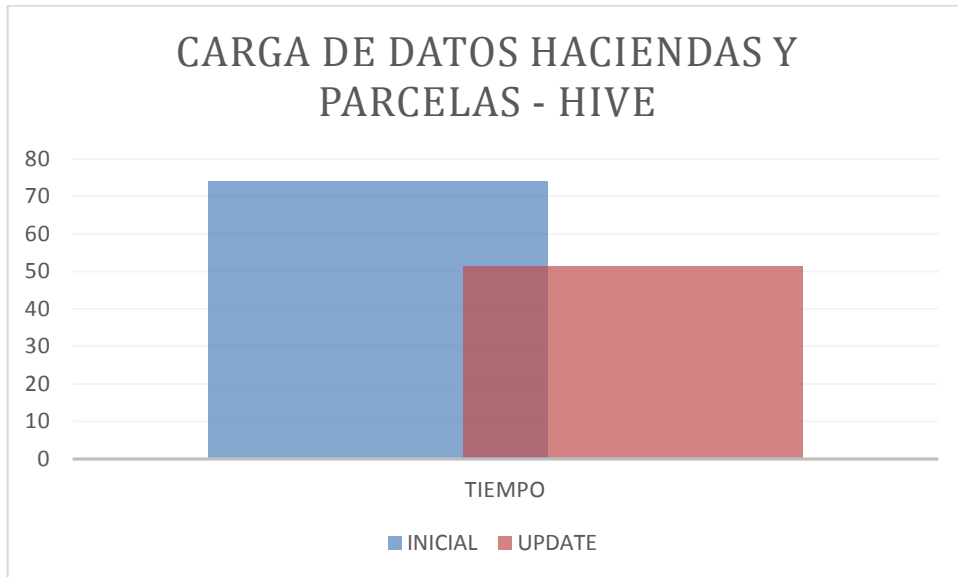
CARGA DE DATOS HACIENDAS Y PARCELAS - HIVE				
DATOS	LEIDOS	SALIDA	ERROR	TIEMPO
INICIAL	10	10	0	51,2
UPDATE	12	2	0	74
Fuente: Elaboración Propia				

Figura 65 Resultado carga de datos Haciendas y Parcelas - Hive



Fuente: Elaboración Propia

Figura 66 Resultado tiempo carga de datos Haciendas y Parcelas - Hive

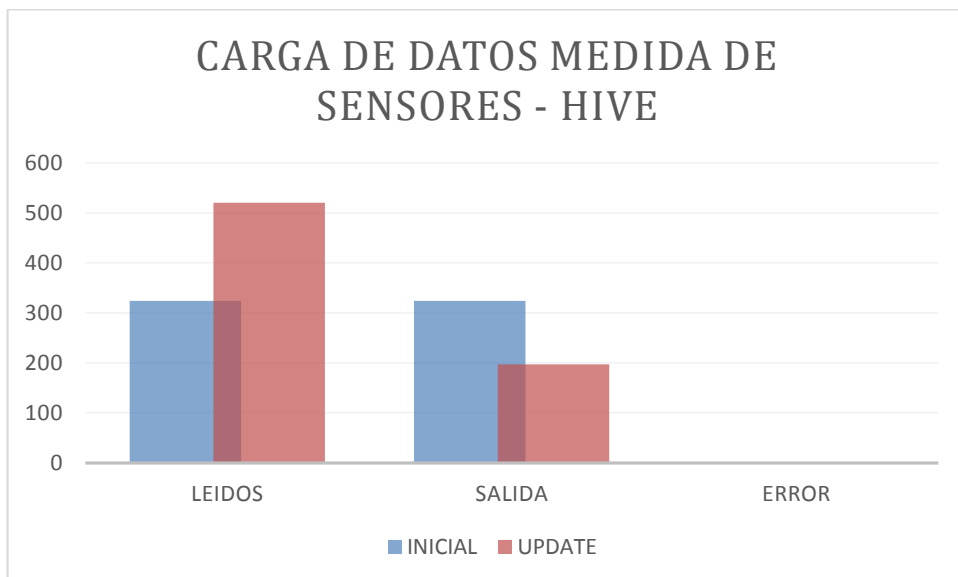


Fuente: Elaboración Propia

Tabla 7 Carga de datos Medida de Sensores - Hive

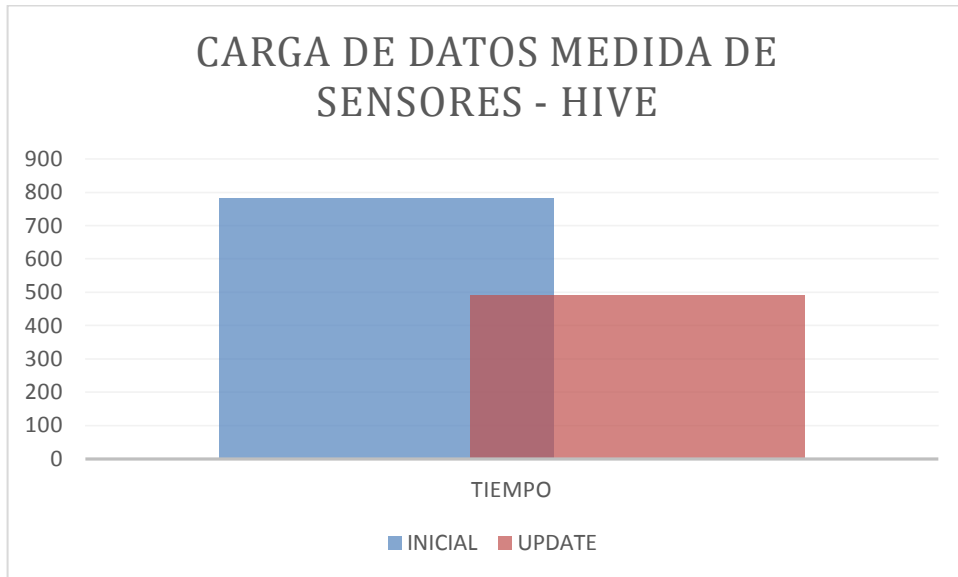
CARGA DE DATOS MEDIDA DE SENSORES - HIVE				
DATOS	LEIDOS	SALIDA	ERROR	TIEMPO
INICIAL	324	324	0	781
UPDATE	521	197	0	489
Fuente: Elaboración Propia				

Figura 67 Resultado carga de datos Medida de Sensores - Hive



Fuente: Elaboración Propia

Figura 68 Resultado tiempo carga de datos Medida Sensores - Hive



Fuente: Elaboración Propia

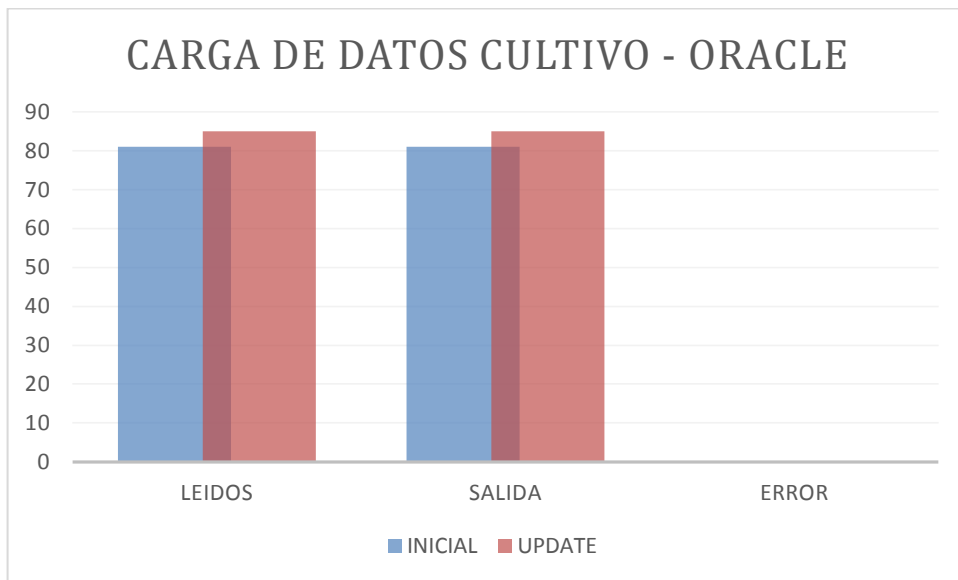
Las pruebas de carga de datos, permite verificar la cantidad de datos leídos, cargados y erróneos, así como verificar el tiempo que toma el proceso ETL, revisar Anexo M:

Tabla 8 Carga de datos Cultivo - Oracle

CARGA DE DATOS CULTIVO - ORACLE				
DATOS	LEIDOS	SALIDA	ERROR	TIEMPO
INICIAL	81	81	0	2
UPDATE	85	85	0	0,1

Fuente: Elaboración Propia

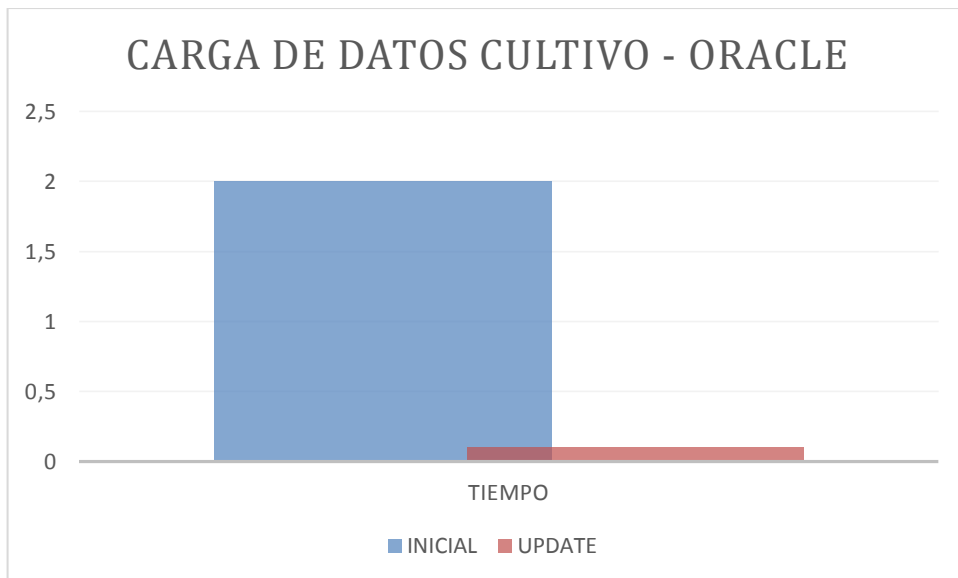
Figura 69 Resultado carga de datos Cultivo - Oracle



Fuente: Elaboración Propia

Se puede observar en la figura 65 que los datos iniciales son leídos y cargados exitosamente sin ningún dato erróneo, de la misma forma, su actualización, leyendo más datos, pero cargando solo los nuevos.

Figura 70 Resultado tiempo carga de datos Cultivo - Oracle



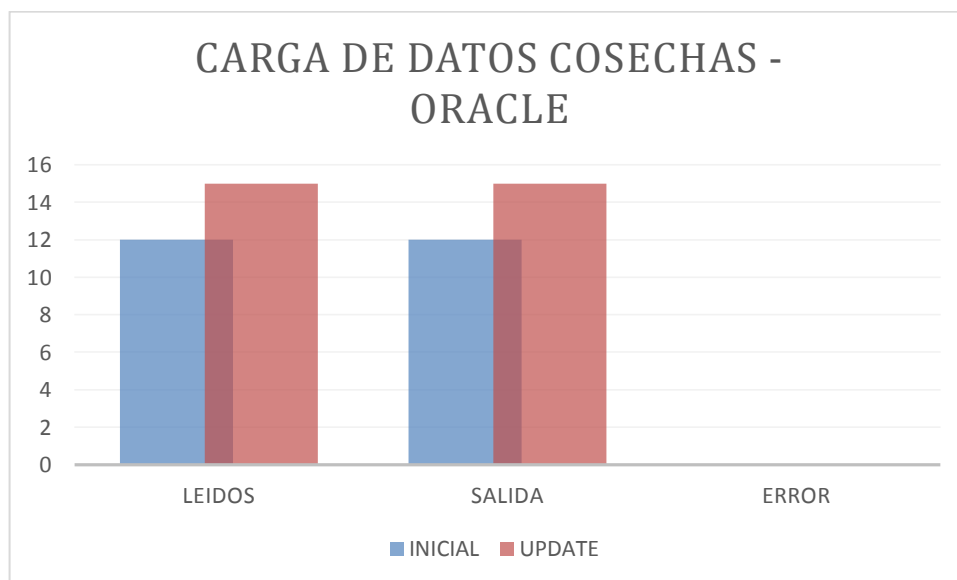
Fuente: Elaboración Propia

Se puede observar en la figura 65 que los datos iniciales son cargados en mayor tiempo, pero su actualización disminuyéndolo, estas actualizaciones son programadas en horas determinadas por el super usuario.

Tabla 9 Carga de datos Cosechas - Oracle

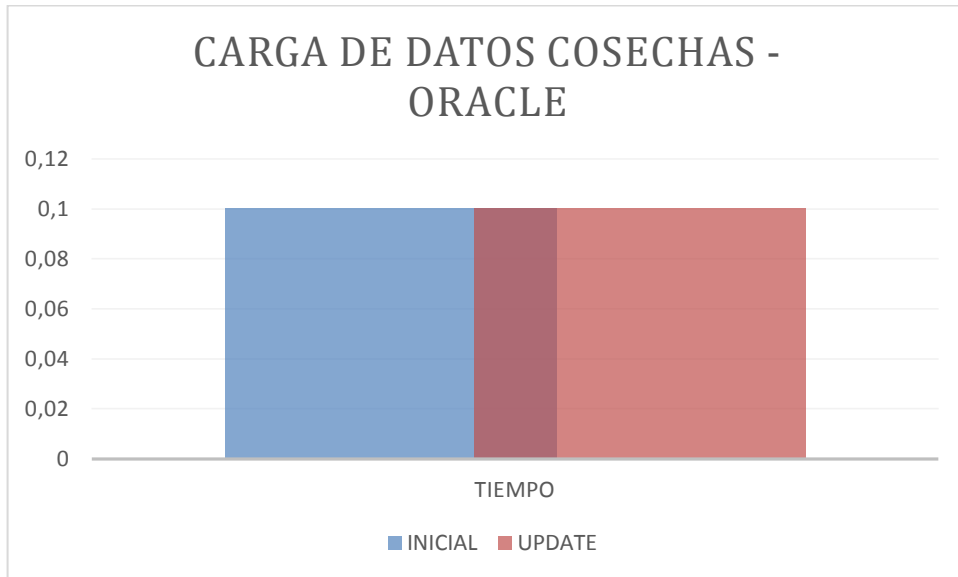
CARGA DE DATOS COSECHAS - ORACLE				
DATOS	LEIDOS	SALIDA	ERROR	TIEMPO
INICIAL	12	12	0	0,1
UPDATE	15	15	0	0,1
Fuente: Elaboración Propia				

Figura 71 Resultado carga de datos Cosechas - Oracle



Fuente: Elaboración Propia

Figura 72 Resultado tiempo carga de datos Cosechas - Oracle

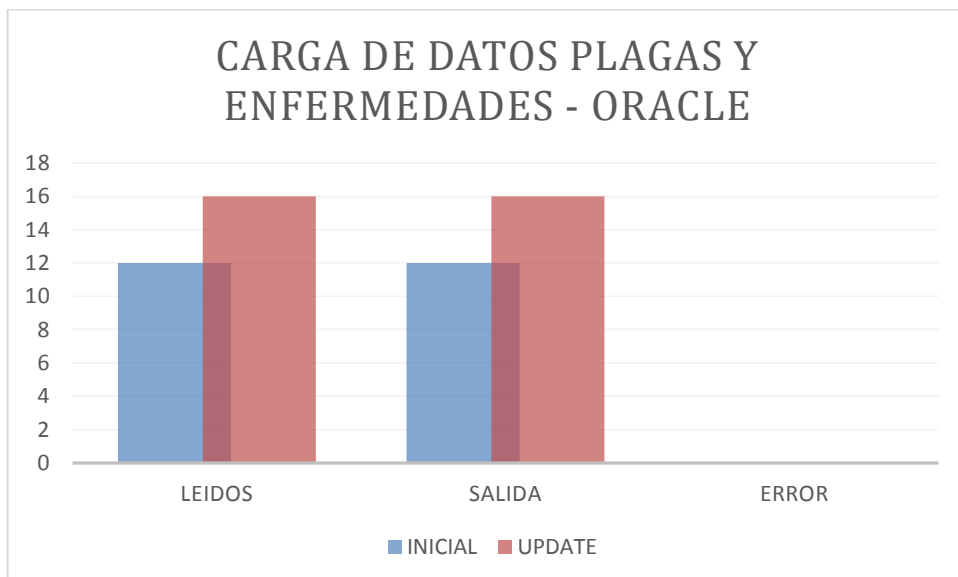


Fuente: Elaboración Propia

Tabla 10 Carga de datos Plagas y Enfermedades - Oracle

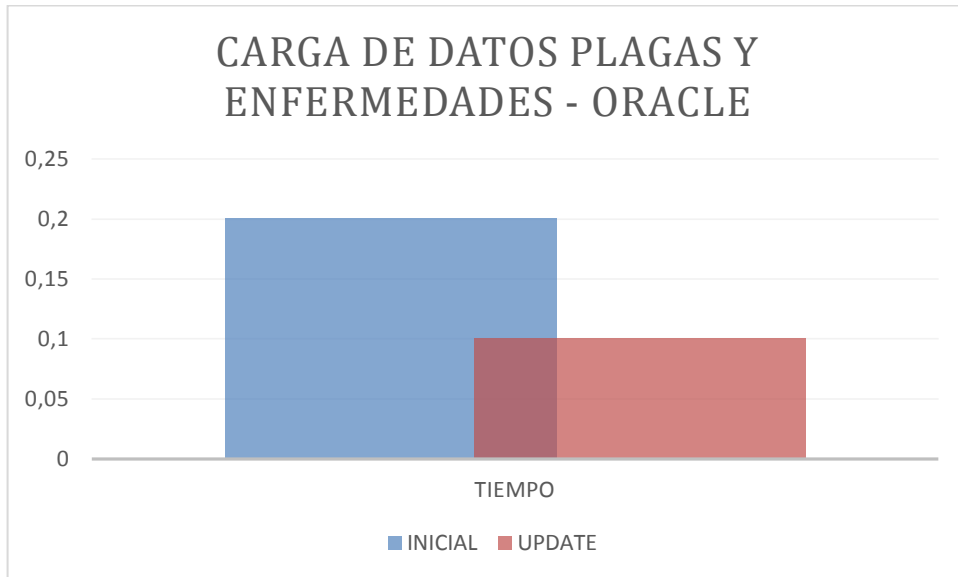
CARGA DE DATOS PLAGAS Y ENFERMEDADES - ORACLE				
DATOS	LEIDOS	SALIDA	ERROR	TIEMPO
INICIAL	12	12	0	0,2
UPDATE	16	16	0	0,1
Fuente: Elaboración Propia				

Figura 73 Resultado carga de datos Plagas y Enfermedades - Oracle



Fuente: Elaboración Propia

Figura 74 Resultado tiempo carga de datos Plagas y Enfermedades - Oracle

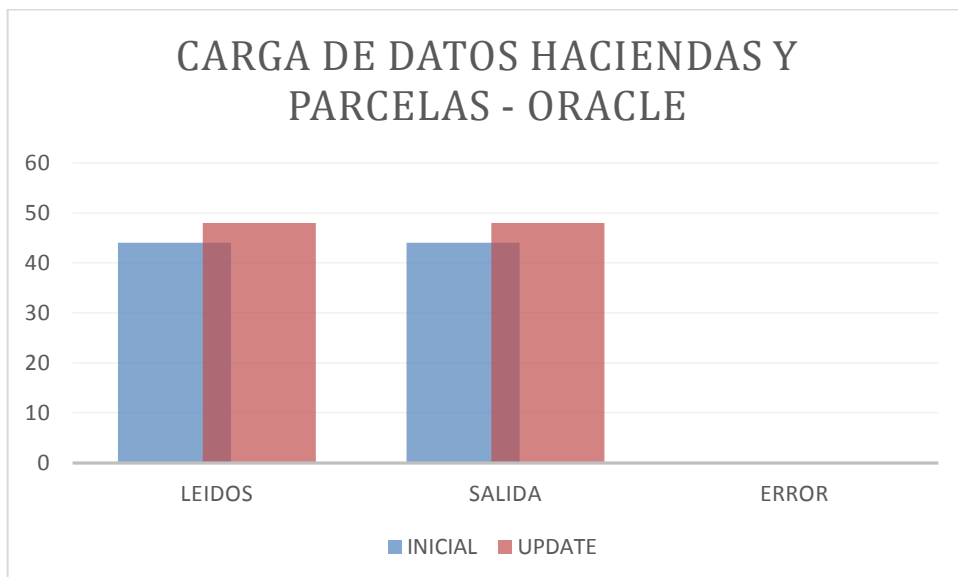


Fuente: Elaboración Propia

Tabla 11 Carga de datos Haciendas y Parcelas - Oracle

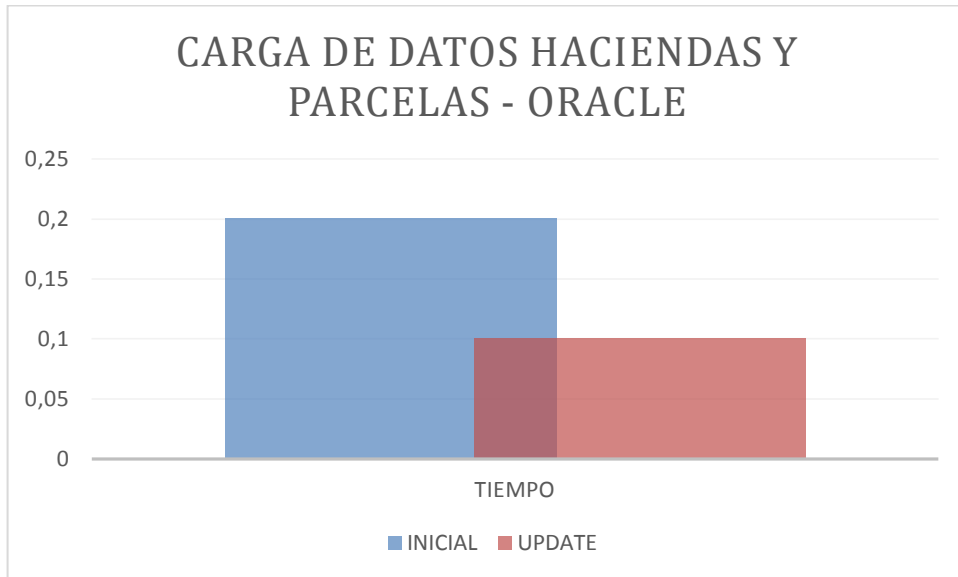
CARGA DE DATOS HACIENDAS Y PARCELAS - ORACLE				
DATOS	LEIDOS	SALIDA	ERROR	TIEMPO
INICIAL	44	44	0	0,2
UPDATE	48	48	0	0,1
Fuente: Elaboración Propia				

Figura 75 Resultado carga de datos Haciendas y Parcelas - Oracle



Fuente: Elaboración Propia

Figura 76 Resultado tiempo carga de datos Haciendas y Parcelas - Oracle



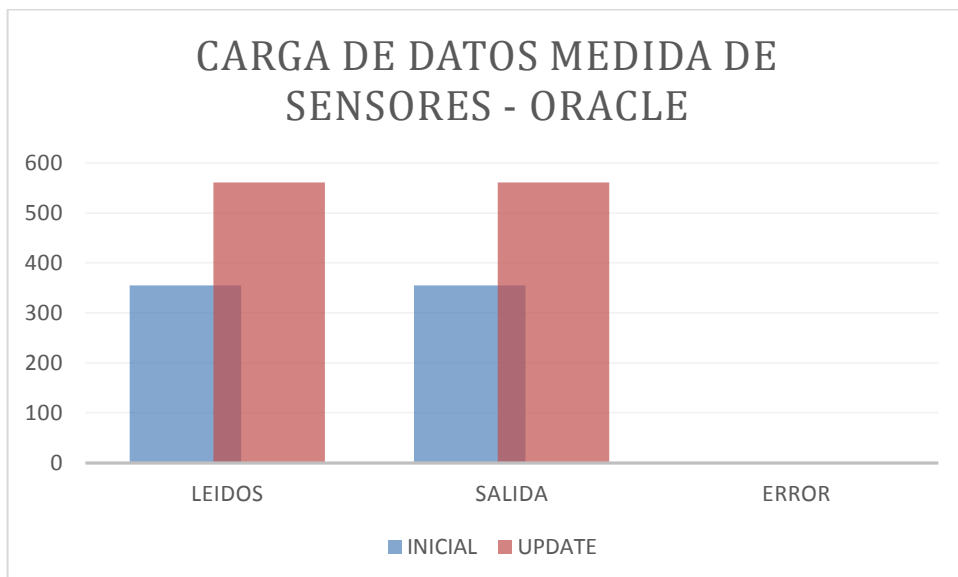
Fuente: Elaboración Propia

Tabla 12 Carga de datos Medida de Sensores - Oracle

CARGA DE DATOS MEDIDA DE SENSORES - ORACLE				
DATOS	LEIDOS	SALIDA	ERROR	TIEMPO
INICIAL	355	355	0	8
UPDATE	561	561	0	3

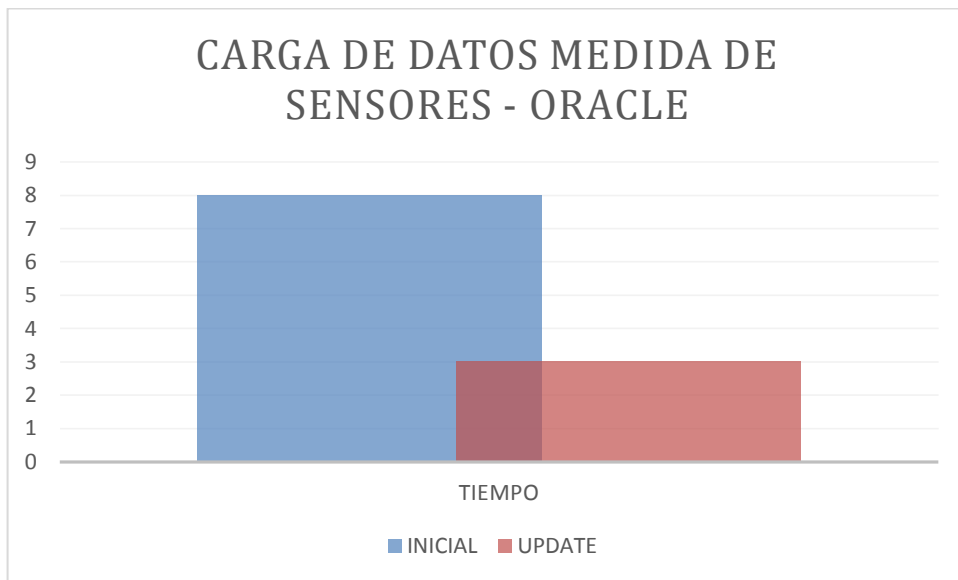
Fuente: Elaboración Propia

Figura 77 Resultado carga de datos Medida de Sensores - Oracle



Fuente: Elaboración Propia

Figura 78 Resultado tiempo carga de datos Medida Sensores - Oracle



Fuente: Elaboración Propia

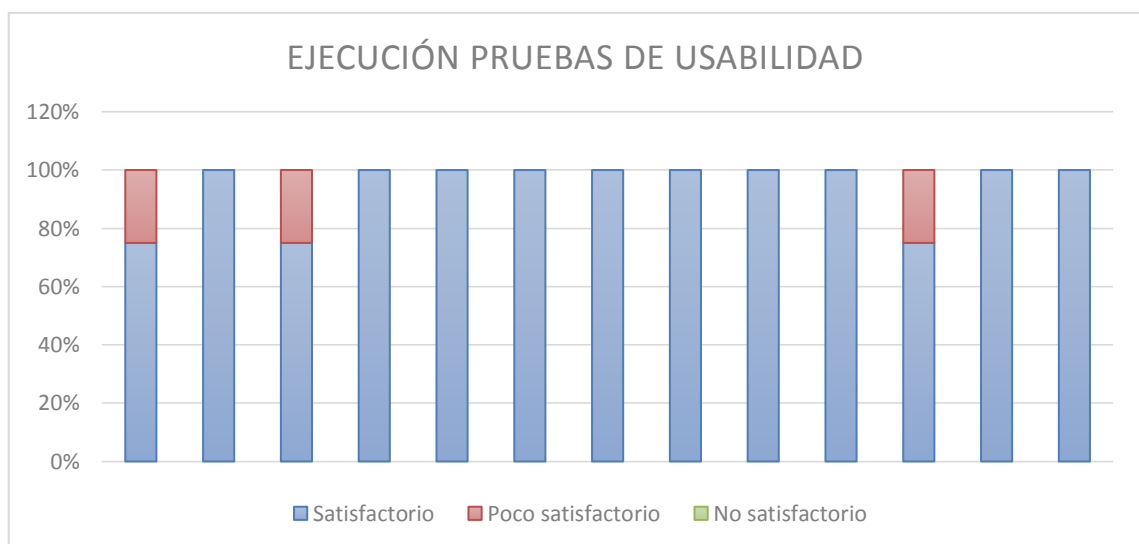
3.2.2 Resultados pruebas de usabilidad

Tabla 13 Matriz resultados pruebas de usabilidad

Nº	DESCRIPCIÓN	Satisfactorio	Poco satisfactorio	No satisfactorio	Observaciones
1	¿Es fácil utilizar el Dashboard por primera vez?	33%	66%	0%	
2	¿El panel de visualización es claro?	100%	0%	0%	
3	¿Las acciones a realizar son intuitivas?	66%	33%	0%	
4	¿El usuario puede controlar los gráficos de forma conveniente?	100%	0%	0%	
5	¿Los términos usados en el dashboard para describir medidas, indican de forma clara lo que representan?	100%	0%	0%	
6	¿Es posible observar de forma global lo que abarca el contenido de cada sección?	100%	0%	0%	
7	¿El texto de los mensajes de cada gráfico es significativo e identifica los valores necesarios para la toma de decisiones?	100%	0%	0%	

8	¿Los colores empleados para la presentación de los gráficos dentro del dashboard, van en la línea del logo del proyecto?	100%	0%	0%	
9	¿El diseño de la interfaz de cada sección, sigue la misma consistencia en todas las secciones del dashboard?	100%	0%	0%	
10	¿Existen elementos que representen los datos globales de cada sección?	100%	0%	0%	
11	¿Le han parecido claros y representativos los nombres y descripciones que aparecen en cada gráfico del dashboard?	66%	33%	0%	
12	¿La información que se presenta en cada gráfico tiene el tamaño adecuado?	100%	0%	0%	
13	¿Cree que este dashboard puede ser implementado para toma de decisiones en agricultura?	100%	0%	0%	
Fuente: Elaboración Propia					

Figura 79 Resultado Pruebas de Usabilidad



Fuente: Elaboración Propia

Mediante las pruebas de usabilidad, se obtuvo un 94% de satisfacción, un 6% de poca satisfacción y 0% de no satisfacción. Evidenciando que el Dashboard o EIS, ha cumplido con todas las expectativas, tanto del diseño como de negocio.

3.3 Conclusiones

Se logró la implementación de un módulo de soporte de decisiones para agricultura utilizando Big Data e Inteligencia de Negocios BI aplicando la metodología Hefesto, para la obtención de análisis descriptivos de los cultivos y cosechas, integrado al sistema IoT Mach, creando una aplicación que brinda al agrónomo o empresario, la posibilidad de realizar un análisis descriptivo de sus cultivos, cosechas, cantidad de plagas o enfermedades, información de parcelas y haciendas, y medidas de sensores en sus cultivos, y así toda esa información almacenada sea de gran importancia para la toma de decisiones orientadas a la mejora de la efectividad del cultivo y cosecha. Haciendo las pruebas respectivas y obteniendo como resultado la mayor cantidad de especialistas satisfechos con el resultado.

Se realiza con éxito la investigación bibliográfica de las tecnologías y herramientas para justificar la implementación de un módulo de soporte de decisiones.

Se crea una estructura Data warehouse compuesta por cinco Data Marts que corresponden a: Cultivos, Cosechas, Plagas y Enfermedades, Haciendas y Parcelas, y Lectura de Sensores utilizando la metodología Hefesto, logrando definir como destino la base de datos no sql Apache Hive para la información histórica y la base de datos estructurada Oracle 12c para información relacionada al último año.

Se desarrolla el proceso ETL usando la herramienta Pentaho Data Integration (Spoon) para los datos estructurados depositados en base de datos PostgreSQL y una combinación de datos por medio de Denodo Platform para lograr integrar las medidas de sensores con valores del sistema transaccional que guardan información de empresa, hacienda, parcela, etc. Que luego son cargados por medio de la herramienta Spoon a las bases de datos correspondientes.

Por medio de la herramienta de Inteligencia de Negocios Pentaho en su versión 6.1, se obtienen los resultados clave para las decisiones de negocio. Usando CDE que permite el desarrollo de Dashboards avanzados con la información ya almacenada en nuestro Big Data, dando uso a componentes propios de la herramienta y a componentes adicionales como Ivy Dashboard Component también Open Source.

3.4 Recomendaciones

Se implementa un módulo de soporte de decisiones para agricultura utilizando Big Data e Inteligencia de Negocios BI aplicando la metodología Hefesto, para la obtención de análisis descriptivos de los cultivos y cosechas, integrado al sistema IoT Mach. Con respecto a los datos no estructurados almacenados en MongoDB, no pueden ser tratados directamente por la herramienta Spoon, entonces se requieren alternativas como Denodo para realizar la combinación de información para luego generar su ETL, recomendando una herramienta capaz de integrar los datos de diferentes fuentes y generar su ETL en el mismo proceso.

El proceso ETL (Extracción, Transformación y Carga) se debe realizar al menos una vez por semana para mantener alimentada y actualizada la Data Warehouse, de tal manera que las estadísticas que se generen en la Inteligencia de Negocios tanto en los cubos como en el Dashboard se vea reflejada la última información que ha producido la empresa.

Para que el acceso a los Data Warehouse con Hive sea más rápido y eficiente se debe implementar un sistema de clúster con Hadoop e incrementar los nodos a medida que aumenta la cantidad de datos.

Pentaho 6.1 es una herramienta muy completa para la aplicación de soluciones de inteligencia de negocios, pero para modificar su diseño y sea más amigable al usuario, no basta con las opciones que dispone, entonces se deben aplicar conocimientos en programación y/o diseño usando java script, css, html.

El acceso a la información estadística generada por el Sistema de Inteligencia de Negocios debe restringirse para que únicamente sea visible a los usuarios que tengan los permisos correspondientes, por ejemplo, los dueños o gerentes de la empresa.

REFERENCIAS BIBLIOGRÁFICAS

- [1] D. Gómez-Candón, A. I. De Castro, and F. López-Granados, "Assessing the accuracy of mosaics from unmanned aerial vehicle (UAV) imagery for precision agriculture purposes in wheat," *Precis. Agric.*, vol. 15, no. 1, pp. 44–56, Feb. 2014.
- [2] K. Kurniawan and A. Ashari, "Service orchestration using enterprise service bus for real-time government executive dashboard system," in *2015 International Conference on Data and Software Engineering (ICoDSE)*, 2015, pp. 207–212.
- [3] N. Dlodlo and J. Kalezhi, "The internet of things in agriculture for sustainable rural development," in *2015 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC)*, 2015, no. April, pp. 13–18.
- [4] Santoshkumar and Udaykumar R.Y, "Development of WSN system for precision agriculture," in *2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, 2015, pp. 1–5.
- [5] S. Shin and H. Hashimoto, "Storage and Reuse of Each Student's Personalized Documents Using a Non-structured Database," *2015 Int. Conf. Comput. Appl. Technol.*, pp. 92–93, 2015.
- [6] M. R. Bendre, R. C. Thool, and V. R. Thool, "Big data in precision agriculture: Weather forecasting for future farming," in *2015 1st International Conference on Next Generation Computing Technologies (NGCT)*, 2015, no. September, pp. 744–750.
- [7] A. Khalid, H. Afzal, and S. Aftab, "Balancing scalability, performance and fault tolerance for structured data (BSPF)," in *16th International Conference on Advanced Communication Technology*, 2014, pp. 725–732.
- [8] Muxiu Zhu and Xianjiu Zhang, "The management of big data tables based on Oracle partition technology," in *2014 9th International Conference on Computer Science & Education*, 2014, no. Iccse, pp. 570–572.
- [9] R. Bernabeu, "Hefesto," p. 146, 2010.
- [10] P. D. C. De Almeida and J. Bernardino, "Big Data Open Source Platforms," *2015 IEEE Int. Congr. Big Data*, pp. 268–275, 2015.
- [11] N. Bari, G. Mani, and S. Berkovich, "Internet of Things as a Methodological Concept," in *2013 Fourth International Conference on Computing for Geospatial Research and Application*, 2013, pp. 48–55.
- [12] A. Narmada and P. S. Rao, "Zigbee Based WSN with IP Connectivity," in *2012 Fourth International Conference on Computational Intelligence, Modelling and Simulation*, 2012, pp. 178–181.
- [13] I. C. I. De Ciencia, I. Del, S. Por, H. Manuel, C. Guaicha, R. M. Jaramillo, C. Diferencial, D. E. B. Por, K. Stephanie, and E. Flores, "NUEVAS TECNOLOGÍAS APLICADAS AL TERRITORIO," 2015.
- [14] R. Teruel Santos and A. Mauro Saraiva, "A Reference Process for Management Zones Delineation in Precision Agriculture," *IEEE Lat. Am. Trans.*, vol. 13, no. 3, pp. 727–738, Mar. 2015.
- [15] H. Qin, Z.-M. Qian, and Y.-C. Zhao, "On the Research of Data Warehouse in Big

- Data,” in *2015 International Conference on Network and Information Systems for Computers*, 2015, pp. 354–357.
- [16] K. Park, M. C. Nguyen, and H. Won, “Web-based collaborative big data analytics on big data as a service platform,” in *2015 17th International Conference on Advanced Communication Technology (ICACT)*, 2015, pp. 564–567.
- [17] B. K. Seah and Nor Ezam Selan, “Design and implementation of data warehouse with data model using survey-based services data,” in *Fourth edition of the International Conference on the Innovative Computing Technology (INTECH 2014)*, 2014, pp. 58–64.
- [18] R. Kimball and M. Ross, *NEW YORK • CHICHESTER • WEINHEIM • BRISBANE • SINGAPORE • TORONTO*. 2002.
- [19] C. Strong, M. Fodor, M. Cerwin, J. José Camargo-Vega, J. Felipe Camargo-Ortega, L. Joyanes-Aguilar, E. Polit, L. J. Aguilar, U. D. C. Rica, J. Schroeck, Michael; Shockley, Rebecca; Smart, A. G. Suriol, B. Marr, C. Information, Puromarketing, and Gartner, “Los desafíos del marketing en la era del big data Marsy Dayanna Ortiz Morales Luis Joyanes Aguilar Lillyana María Giraldo Marín Digital desafíos marketing en del big data,” *Rev. Fac. Ing. (Fac. Ing.)*, Enero-Abril, vol. 24, no. 38, p. 22, 2015.
- [20] A. Fuad, A. Erwin, and H. P. Ipung, “Processing performance on Apache Pig, Apache Hive and MySQL cluster,” *Proc. 2014 Int. Conf. Information, Commun. Technol. Syst. ICTS 2014*, pp. 297–301, 2015.
- [21] D. Krneta, “An Approach to Data Mart Design from a Data Vault,” vol. 15, no. March, pp. 473–478, 2016.
- [22] M. Di Sano, “Business Intelligence as a Service: A New Approach to Manage Business Processes in the Cloud,” in *2014 IEEE 23rd International WETICE Conference*, 2014, pp. 155–160.
- [23] Meyliana, H. A. E. Widjaja, and S. W. Santoso, “University dashboard: An implementation of executive dashboard to university,” in *2014 2nd International Conference on Information and Communication Technology (ICoICT)*, 2014, pp. 282–287.
- [24] M. S. Gounder, V. V. Iyer, and A. Al Mazyad, “A survey on business intelligence tools for university dashboard development,” in *2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC)*, 2016, pp. 1–7.
- [25] A. M. C. M, D. L. H. R, B. E. M. O, U. Académica, D. I. Civil, U. Técnica, and K. V. Machala, “Cloud computing con herramientas open-source para Internet de las cosas,” pp. 173–182, 2015.
- [26] P. Kookarinrat and Y. Temtanapat, “Analysis of Range-Based Key Properties for Sharded Cluster of MongoDB,” in *2015 2nd International Conference on Information Science and Security (ICISS)*, 2015, pp. 1–4.
- [27] Shidong Huang, Lizhi Cai, Zhenyu Liu, and Yun Hu, “Non-structure Data Storage Technology: A Discussion,” in *2012 IEEE/ACIS 11th International Conference on Computer and Information Science*, 2012, pp. 482–487.
- [28] Mengying Zhang, “PMT:A Procedure Migration Tool from Oracle to PostgreSQL,” in *IET International Conference on Smart and Sustainable City 2013 (ICSSC 2013)*, 2013, pp. 384–389.

- [29] Zhengdong Xia and Tianming Bu, "The implementation of flash-aware buffer replacement algorithms in PostgreSQL," in *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, 2015, no. 61402178, pp. 1215–1219.
- [30] Y. Ujibashi, M. Nakamura, T. Tabaru, T. Hashida, M. Kawaba, and L. Harada, "Design of a Shared Memory mechanism for efficient parallel processing in PostgreSQL," in *2015 6th International Conference on Information, Intelligence, Systems and Applications (IISA)*, 2015, pp. 1–6.
- [31] M.-G. Jung, S.-A. Youn, J. Bae, and Y.-L. Choi, "A Study on Data Input and Output Performance Comparison of MongoDB and PostgreSQL in the Big Data Environment," *2015 8th Int. Conf. Database Theory Appl.*, pp. 14–17, 2015.
- [32] Pentaho Corporation, "Any Analytics, Any Data, Simplified," *Any Analytics, Any Data, Simplified*, 2016. [Online]. Available: <http://www.pentaho.com/product/product-overview>.
- [33] Dell Inc, "Toad Data Modeler," *Easy-to-use, cross-platform database modeling software*, 2016. [Online]. Available: <https://software.dell.com/products/toad-data-modeler/>.
- [34] Pivot4J Team, "Pivot4J," *Modern Ajax-based User Interface*, 2016. [Online]. Available: <http://www.pivot4j.org/>.
- [35] OSBI LTD, "Saiku Business Analytics," *Analyse and explore data. Wherever it is stored.*, 2016. [Online]. Available: <http://www.meteorite.bi/products/saiku>.
- [36] P. G. D. Group, "PostgreSQL," 2016. [Online]. Available: <https://www.postgresql.org/>.
- [37] The Apache Software Foundation, "APACHE HIVE TM," *Getting Started With Apache Hive Software*, 2014. [Online]. Available: <https://hive.apache.org/>.
- [38] Oracle, "Oracle Products and Services," 2016. [Online]. Available: <https://www.oracle.com/products/index.html>.
- [39] The Apache Software Foundation, "Welcome to Apache™ Hadoop®!," *What Is Apache Hadoop?*, 2014. [Online]. Available: <http://hadoop.apache.org/>.
- [40] Hortonworks Inc., "CONNECTED DATA PLATFORMS DATA-IN-MOTION, DATA-AT-REST IN THE CLOUD, IN THE DATA CENTER," *HORTONWORKS DATA CENTER SOLUTIONS*, 2016. [Online]. Available: <http://hortonworks.com/products/>.
- [41] Denodo Technologies, "Denodo Platform," *¿Qué es la virtualización de datos?*, 2016. [Online]. Available: <http://www.denodo.com/es/virtualizacion-de-datos/que-es>.

ÍNDICE COMPLEMENTARIO

Apache Hive, 1, 3, 19, 21, 27, 66, 68, 96, 99, 100

Big Data, 1, 3, 15, 18, 19, 22, 25, 26, 27, 31, 79, 96, 97, 98, 99, 100

Cosecha, 1, 60, 114, 116

Cultivo, 1, 60, 63, 68, 72, 82, 83, 88, 89, 114, 116

Data Warehouse, 1, 18, 21, 22, 34, 35, 38, 51, 52, 53, 54, 97, 98, 143

Denodo, 1, 3, 21, 53, 54, 66, 67, 68, 69, 96, 97, 100, 123, 151, 152, 153, 156, 157,
159, 160

ETL, 1, 3, 12, 13, 14, 15, 32, 35, 51, 53, 54, 65, 66, 79, 80, 81, 82, 88, 96, 97, 114,
140, 145

Haciendas y Parcelas, 1, 58, 61, 80, 81, 82, 86, 87, 92, 93

Hadoop, 17, 19, 27, 28, 29, 66, 97, 100

Hefesto, 1, 19, 31, 34, 96, 97, 98

Inteligencia de Negocios, 1, 12, 15, 19, 20, 21, 31, 32, 96, 97

KPIs, 1, 3, 35, 59, 72, 79

Medidas de Sensores, 1, 80, 82

OLAP, 1, 3, 14, 15, 32, 65

Oracle, 1, 3, 17, 19, 21, 32, 66, 79, 82, 88, 89, 90, 91, 92, 93, 94, 96, 98, 99, 100, 150

Plagas y Enfermedades, 1, 57, 61, 80, 81, 82, 85, 86, 91, 92, 96

Sistema de Información Ejecutiva, 2, 18, 19, 54

Sistema de Soporte de Decisiones, 2, 13, 54

Spoon, 1, 3, 35, 51, 65, 66, 80, 96, 97, 140, 148, 149, 150

ANEXOS

Anexo A

ESPECIALISTAS			
Nombre	Apellidos	Título	Rol
Rosario	Cuenca Romero	Ingeniera Agrónoma	Evaluador Dashboard
Angel	Zarate Encalada	Ingeniero Agrónomo	Evaluador Dashboard
Julio	Albuquerque Ávila	Ingeniero Agrónomo	Evaluador Dashboard
Eufemia	Niola Mora	Ingeniera Agrónoma	Evaluador Dashboard
Andrés	Tamayo Tagual	Ingeniero de Sistemas	Evaluador ETL
Cristian	Miro Pezo	Ingeniero de Sistemas	Evaluador ETL
Arturo	Sanmatín Lapo	Ingeniero de Sistemas	Evaluador ETL
Alexandra	Vargas Jaramillo	Ingeniera de Sistemas	Evaluador ETL

Anexo B

Caso de prueba: Nombre de caso de prueba	Nº: Numero de la prueba			
Escenario: Nombre del escenario a probar				
Responsable diseño: Nombre del responsable de diseño de la prueba				
Responsable ejecución: Nombre del responsable para ejecutar la prueba	Fecha: Fecha de la prueba			
Precondiciones	Condiciones especiales			
Datos de entrada	Datos que se han ingresado para ejecutar la prueba			
Descripción de pasos	Lista de pasos que se llevan a cabo para ejecutar la prueba			
Resultado esperado	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 60%; padding: 5px;">Descripción de lo que espera en la prueba</td> <td style="padding: 5px;"> Cumplimiento Se marca el casillero según se cumpla o no el resultado esperado </td> <td style="padding: 5px;"> <input type="checkbox"/> Satisfactorio <input type="checkbox"/> Poco satisfactorio <input type="checkbox"/> No Satisfactorio </td> </tr> </table>	Descripción de lo que espera en la prueba	Cumplimiento Se marca el casillero según se cumpla o no el resultado esperado	<input type="checkbox"/> Satisfactorio <input type="checkbox"/> Poco satisfactorio <input type="checkbox"/> No Satisfactorio
Descripción de lo que espera en la prueba	Cumplimiento Se marca el casillero según se cumpla o no el resultado esperado	<input type="checkbox"/> Satisfactorio <input type="checkbox"/> Poco satisfactorio <input type="checkbox"/> No Satisfactorio		
Resultado obtenido	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; padding: 5px;">Errores: Descripción de los errores obtenidos en caso no haber llegado al resultado esperado</td> <td style="padding: 5px;">Fallas Provocadas: Descripción de las fallas directas o indirectas que provoca el error</td> </tr> </table>	Errores: Descripción de los errores obtenidos en caso no haber llegado al resultado esperado	Fallas Provocadas: Descripción de las fallas directas o indirectas que provoca el error	
Errores: Descripción de los errores obtenidos en caso no haber llegado al resultado esperado	Fallas Provocadas: Descripción de las fallas directas o indirectas que provoca el error			
Recomendaciones u Observación	Recomendaciones u observaciones que pueden servir para corregir el error o mejorar algún aspecto			

Anexo C

Caso de prueba: Ejecutar ETL - HIVE		Nº: PU_1	
Escenario:		Data Mart Cultivo	
Responsable diseño: Grupo de investigación		Fecha: 15/08/2016	
Precondiciones:		Sesión iniciada Ingreso de datos en Sistema transaccional de monitoreo y control	
Datos de entrada		Empresa Tempo Cultivo Hacienda Tipo Suelo Tipo Riego País Provincia Ciudad Parcela Especie Categoría Especie Cultivo	
Descripción de pasos:		1. Acceder a la herramienta Spoon Pentaho Data Integraton 2. Abrir Job Cultivo 3. Dar clic en empezar	
Resultado esperado:		CUMPLIMIENTO	
		SATISFACTORIO	
		POCO SATISFACTORIO	
		NO SATISFACTORIO	
Resultado obtenido:		Errores:	Fallas Provocadas:
Recomendaciones u Observaciones:			
Nombre Evaluador:			
		FIRMA	

Anexo D

Caso de prueba: Ejecutar ETL - HIVE		Nº: PU_2	
Escenario:		Data Mart Cosecha	
Responsable diseño: Grupo de investigación		Fecha: 15/08/2016	
Precondiciones:	Sesión iniciada Ingreso de datos en Sistema transaccional de monitoreo y control		
Datos de entrada	Empresa Tempo Cultivo Hacienda Tipo Suelo País Provincia Ciudad Parcela Especie Categoría Especie Cosecha		
Descripción de pasos:	1. Acceder a la herramienta Spoon Pentaho Data Integraton 2. Abrir Job Cosecha 3. Dar clic en empezar		
Resultado esperado:	La transformación a finalizado	CUMPLIMIENTO	
		SATISFACTORIO	
		POCO SATISFACTORIO	
		NO SATISFACTORIO	
Resultado obtenido:	Errores:	Fallas Provocadas:	
Recomendaciones u Observaciones:			
Nombre Evaluador:			
		FIRMA	

Anexo E

Caso de prueba: Ejecutar ETL - HIVE		Nº: PU_3	
Escenario:		Data Mart Plagas y Enfermedades	
Responsable diseño: Grupo de investigación		Fecha: 15/08/2016	
Precondiciones:	Sesión iniciada Ingreso de datos en Sistema transaccional de monitoreo y control		
Datos de entrada	Empresa Tempo Cultivo Hacienda Tipo Suelo País Provincia Ciudad Parcela Especie Categoría Especie Plagas y Enfermedades		
Descripción de pasos:	1. Acceder a la herramienta Spoon Pentaho Data Integraton 2. Abrir Job Plagas y Enfermedades 3. Dar clic en empezar		
Resultado esperado:	La transformación a finalizado	CUMPLIMIENTO	
		SATISFACTORIO	
		POCO SATISFACTORIO	
		NO SATISFACTORIO	
Resultado obtenido:	Errores:	Fallas Provocadas:	
Recomendaciones u Observaciones:			
Nombre Evaluador:			
		FIRMA	

Anexo F

Caso de prueba: Ejecutar ETL - HIVE		Nº: PU_4	
Escenario:		Data Mart Haciendas y Parcelas	
Responsable diseño: Grupo de investigación		Fecha: 15/08/2016	
Precondiciones:	Sesión iniciada Ingreso de datos en Sistema transaccional de monitoreo y control		
Datos de entrada	Empresa Hacienda Tipo Suelo Parcela Cultivo Hacienda Parcela		
Descripción de pasos:	1. Acceder a la herramienta Spoon Pentaho Data Integraton 2. Abrir Job Haciendas y Parcelas 3. Dar clic en empezar		
Resultado esperado:	La transformación a finalizado	CUMPLIMIENTO	
		SATISFACTORIO	
		POCO SATISFACTORIO	
		NO SATISFACTORIO	
Resultado obtenido:	Errores:	Fallas Provocadas:	
Recomendaciones u Observaciones:			
Nombre Evaluador:			
		FIRMA	

Anexo G

Caso de prueba: Ejecutar ETL - HIVE		Nº: PU_5	
Escenario:		Data Mart Medidas de Sensores	
Responsable diseño: Grupo de investigación		Fecha: 15/08/2016	
Precondiciones:	Sesión iniciada Ingreso de datos en Sistema transaccional de monitoreo y control		
Datos de entrada	Empresa Hacienda Parcela Sector Lecturas		
Descripción de pasos:	1. Acceder a la herramienta Spoon Pentaho Data Integraton 2. Abrir Job Medidas de Sensores 3. Dar clic en empezar		
Resultado esperado:	La transformación a finalizado	CUMPLIMIENTO	
		SATISFACTORIO	
		POCO SATISFACTORIO	
		NO SATISFACTORIO	
Resultado obtenido:	Errores:	Fallas Provocadas:	
Recomendaciones u Observaciones:			
Nombre Evaluador:			
		FIRMA	

Anexo H

Caso de prueba: Ejecutar ETL - ORACLE		Nº: PU_6	
Escenario:		Data Mart Cultivo	
Responsable diseño: Grupo de investigación		Fecha: 15/08/2016	
Precondiciones:	Sesión iniciada Ingreso de datos en Sistema transaccional de monitoreo y control		
Datos de entrada	Empresa Tempo Cultivo Hacienda Tipo Suelo Tipo Riego País Provincia Ciudad Parcela Especie Categoría Especie Cultivo		
Descripción de pasos:	1. Acceder a la herramienta Spoon Pentaho Data Integraton 2. Abrir Job Cultivo 3. Dar clic en empezar		
Resultado esperado:	La transformación a finalizado	CUMPLIMIENTO	
		SATISFACTORIO	
		POCO SATISFACTORIO	
		NO SATISFACTORIO	
Resultado obtenido:	Errores:	Fallas Provocadas:	
Recomendaciones u Observaciones:			
Nombre Evaluador:			
		FIRMA	

Anexo I

Caso de prueba: Ejecutar ETL - ORACLE		Nº: PU_7	
Escenario:		Data Mart Cosecha	
Responsable diseño: Grupo de investigación		Fecha: 15/08/2016	
Precondiciones:	Sesión iniciada Ingreso de datos en Sistema transaccional de monitoreo y control		
Datos de entrada	Empresa Tempo Cultivo Hacienda Tipo Suelo País Provincia Ciudad Parcela Especie Categoría Especie Cosecha		
Descripción de pasos:	1. Acceder a la herramienta Spoon Pentaho Data Integraton 2. Abrir Job Cosecha 3. Dar clic en empezar		
Resultado esperado:	La transformación a finalizado	CUMPLIMIENTO	
		SATISFACTORIO	
		POCO SATISFACTORIO	
		NO SATISFACTORIO	
Resultado obtenido:	Errores:	Fallas Provocadas:	
Recomendaciones u Observaciones:			
Nombre Evaluador:			
		FIRMA	

Anexo J

Caso de prueba: Ejecutar ETL - ORACLE		Nº: PU_8	
Escenario:		Data Mart Plagas y Enfermedades	
Responsable diseño: Grupo de investigación		Fecha: 15/08/2016	
Precondiciones:	Sesión iniciada Ingreso de datos en Sistema transaccional de monitoreo y control		
Datos de entrada	Empresa Tempo Cultivo Hacienda Tipo Suelo País Provincia Ciudad Parcela Especie Categoría Especie Plagas y Enfermedades		
Descripción de pasos:	1. Acceder a la herramienta Spoon Pentaho Data Integraton 2. Abrir Job Plagas y Enfermedades 3. Dar clic en empezar		
Resultado esperado:	La transformación a finalizado	CUMPLIMIENTO	
		SATISFACTORIO	
		POCO SATISFACTORIO	
		NO SATISFACTORIO	
Resultado obtenido:	Errores:	Fallas Provocadas:	
Recomendaciones u Observaciones:			
Nombre Evaluador:			
		FIRMA	

Anexo K

Caso de prueba: Ejecutar ETL - ORACLE		Nº: PU_9	
Escenario:		Data Mart Haciendas y Parcelas	
Responsable diseño: Grupo de investigación		Fecha: 15/08/2016	
Precondiciones:	Sesión iniciada Ingreso de datos en Sistema transaccional de monitoreo y control		
Datos de entrada	Empresa Hacienda Tipo Suelo Parcela Cultivo Hacienda Parcela		
Descripción de pasos:	1. Acceder a la herramienta Spoon Pentaho Data Integraton 2. Abrir Job Haciendas y Parcelas 3. Dar clic en empezar		
Resultado esperado:	La transformación a finalizado	CUMPLIMIENTO	
		SATISFACTORIO	
		POCO SATISFACTORIO	
		NO SATISFACTORIO	
Resultado obtenido:	Errores:	Fallas Provocadas:	
Recomendaciones u Observaciones:			
Nombre Evaluador:			
		FIRMA	

Anexo L

Caso de prueba: Ejecutar ETL - ORACLE		Nº: PU_10	
Escenario:		Data Mart Medidas de Sensores	
Responsable diseño: Grupo de investigación		Fecha: 15/08/2016	
Precondiciones:	Sesión iniciada Ingreso de datos en Sistema transaccional de monitoreo y control		
Datos de entrada	Empresa Hacienda Parcela Sector Lecturas		
Descripción de pasos:	1. Acceder a la herramienta Spoon Pentaho Data Integraton 2. Abrir Job Medidas de Sensores 3. Dar clic en empezar		
Resultado esperado:	La transformación a finalizado	CUMPLIMIENTO	
		SATISFACTORIO	
		POCO SATISFACTORIO	
		NO SATISFACTORIO	
Resultado obtenido:	Errores:	Fallas Provocadas:	
Recomendaciones u Observaciones:			
Nombre Evaluador:			
		FIRMA	

Anexo M

Resultados de carga de datos ETL

CARGA INICIAL DE DATOS

Data Mart 1: Cultivo

Execution Results

Execution History | Logging | Step Metrics | Performance Graph | Metrics | Preview data

#	Stepname	Copynr	Read	Written	Input	Output	Updated	Rejected	Errors	Active	Time	Speed (r/s)
3	In_tiempo_cultivo	0	0	5	5	0	0	0	0	Finished	0.1s	41
4	In_Pais	0	0	3	3	0	0	0	0	Finished	0.1s	25
5	In_Especie	0	0	20	20	0	0	0	0	Finished	0.1s	177
6	In_Empresa 2	0	0	2	2	0	0	0	0	Finished	0.1s	18
7	In_Cultivo	0	0	5	5	0	0	0	0	Finished	0.1s	56
8	In_Provincia	0	0	3	3	0	0	0	0	Finished	0.1s	28
9	In_Especie_Categoria	0	0	7	7	0	0	0	0	Finished	0.1s	67
10	In_Tipo_Suelo	0	0	6	6	0	0	0	0	Finished	0.1s	92
11	In_Ciudad	0	0	3	3	0	0	0	0	Finished	0.1s	39
12	In_tipo_riego	0	0	1	1	0	0	0	0	Finished	0.1s	13
13	Com_cultivo	0	5	5	5	5	0	0	0	Finished	8mn 24s	0
14	Com_Pais	0	3	3	1	1	0	0	0	Finished	4mn 32s	0
15	Com_Provincia	0	3	3	1	1	0	0	0	Finished	3mn 22s	0
16	Com_Parcelsa_Subarea	0	3	3	3	3	0	0	0	Finished	2mn 49s	0
17	Com_Especie_Categoria	0	7	7	7	7	0	0	0	Finished	5mn 56s	0
18	Com_Especie	0	20	20	20	20	0	0	0	Finished	6mn 57s	0
19	Com_tipo_suelo	0	6	6	6	6	0	0	0	Finished	7mn 38s	0
20	Select values	0	5	5	0	0	0	0	0	Finished	0.1s	39
21	Com_tiempo_cultivo	0	5	5	5	5	0	0	0	Finished	4mn 0s	0
22	Com_Ciudad	0	3	3	3	3	0	0	0	Finished	1mn 25s	0
23	Com_Empresa	0	2	2	2	2	0	0	0	Finished	5mn 19s	0
24	Com_tipo_riego	0	1	1	1	1	0	0	0	Finished	46.7s	0
25	Com_hacienda	0	2	2	2	2	0	0	0	Finished	2mn 15s	0

Data Mart 2: Cosecha

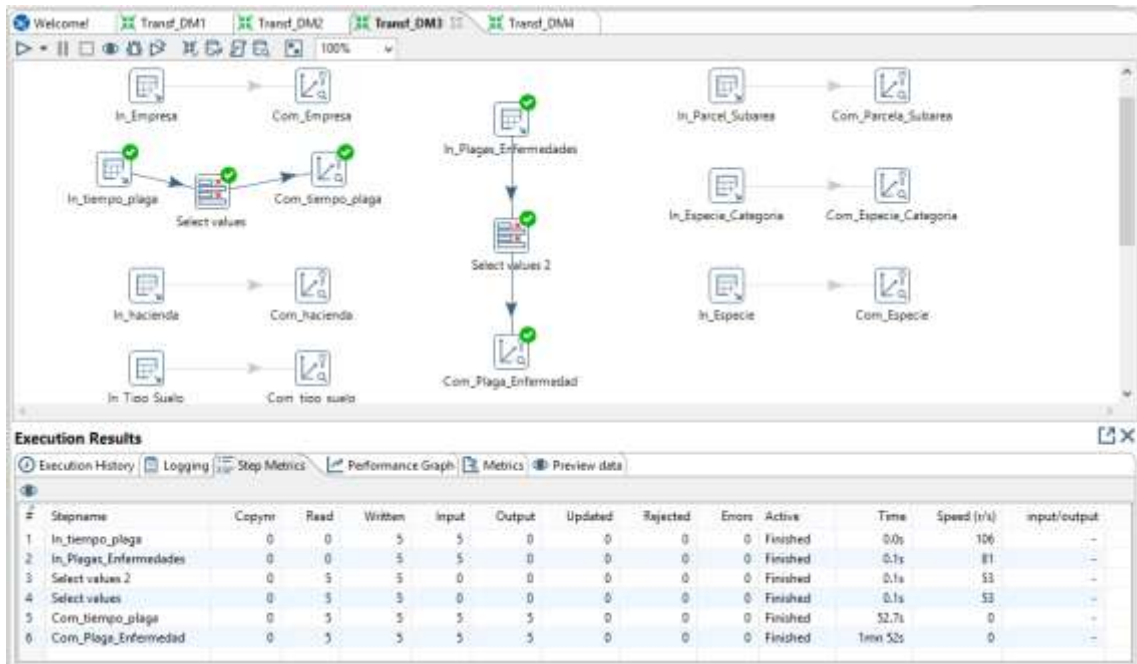
Transf_DM1 | Transf_DM2 | Transf_DM3 | Transf_DM4

Execution Results

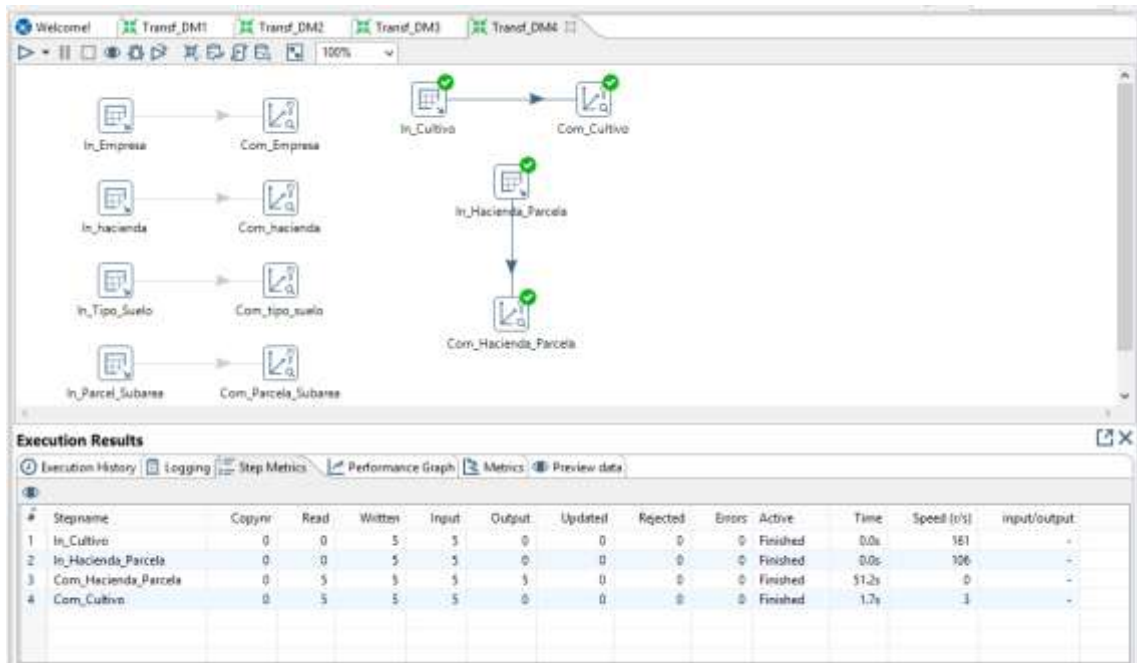
Execution History | Logging | Step Metrics | Performance Graph | Metrics | Preview data

#	Stepname	Copynr	Read	Written	Input	Output	Updated	Rejected	Errors	Active	Time	Speed (r/s)	input/output
1	In_tiempo_cosecha	0	0	6	6	0	0	0	0	Finished	0.2s	26	-
2	Select values	0	6	6	0	0	0	0	0	Finished	0.3s	24	-
3	In_Cosecha	0	0	6	6	0	0	0	0	Finished	0.3s	24	-
4	Com_tiempo_cosecha	0	6	6	6	6	0	0	0	Finished	1mn 15s	0	-
5	Com_cosecha	0	6	6	6	6	0	0	0	Finished	2mn 22s	0	-

Data Mart 3: Plagas enfermedades



Data Mart 4: Haciendas parcelas



ACTUALIZACIÓN DE DATOS

Data Mart 1: Cultivo

Execution Results

Execution History | Logging | Step Metrics | Performance Graph | Metrics | Preview data

#	StepName	Copysr	Read	Written	Input	Output	Updated	Rejected	Errors	Active	Time	Speed (r/s)	input/output
2	In_hacienda	0	0	2	2	0	0	0	0	Finished	0.0s	42	-
3	In_tiempo_cultivo	0	0	6	6	0	0	0	0	Finished	0.1s	95	-
4	In_Pais	0	0	3	3	0	0	0	0	Finished	0.1s	38	-
5	In_Especie	0	0	21	21	0	0	0	0	Finished	0.0s	447	-
6	In_Empresa 2	0	0	3	3	0	0	0	0	Finished	0.0s	64	-
7	In_Cultivo	0	0	6	6	0	0	0	0	Finished	0.1s	64	-
8	In_Provincia	0	0	3	3	0	0	0	0	Finished	0.1s	38	-
9	In_Especie_Categoria	0	0	7	7	0	0	0	0	Finished	0.1s	111	-
10	In_Tipo_Suelo	0	0	6	6	0	0	0	0	Finished	0.1s	95	-
11	In_Ciudad	0	0	3	3	0	0	0	0	Finished	0.1s	38	-
12	In_tipo_riego	0	0	1	1	0	0	0	0	Finished	0.1s	11	-
13	Com_cultivo	0	6	6	6	1	0	0	0	Finished	1mn 7s	0	-
14	Com_Pais	0	3	3	1	0	0	0	0	Finished	2.6s	1	-
15	Com_Provincia	0	3	3	1	0	0	0	0	Finished	2.8s	1	-
16	Com_Parcela_Subarea	0	3	3	3	1	0	0	0	Finished	2mn 41s	0	-
17	Com_Especie_Categoria	0	7	7	7	1	0	0	0	Finished	1mn 53s	0	-
18	Com_Especie	0	21	21	21	3	0	0	0	Finished	5mn 21s	0	-
19	Com_tipo_suelo	0	6	6	6	0	0	0	0	Finished	20.0s	0	-
20	Select values	0	6	6	0	0	0	0	0	Finished	0.1s	64	-
21	Com_tiempo_cultivo	0	6	6	6	1	0	0	0	Finished	4mn 29s	0	-
22	Com_Ciudad	0	3	3	3	0	0	0	0	Finished	4.5s	0	-
23	Com_Empresa	0	3	3	3	1	0	0	0	Finished	3mn 44s	0	-
24	Com_tipo_riego	0	1	1	1	0	0	0	0	Finished	1.2s	1	-
25	Com_hacienda	0	2	2	2	0	0	0	0	Finished	8.2s	0	-

Data Mart 2: Cosecha

Welcome! | Transf_DM1 | Transf_DM2 | Transf_DM3 | Transf_DM4

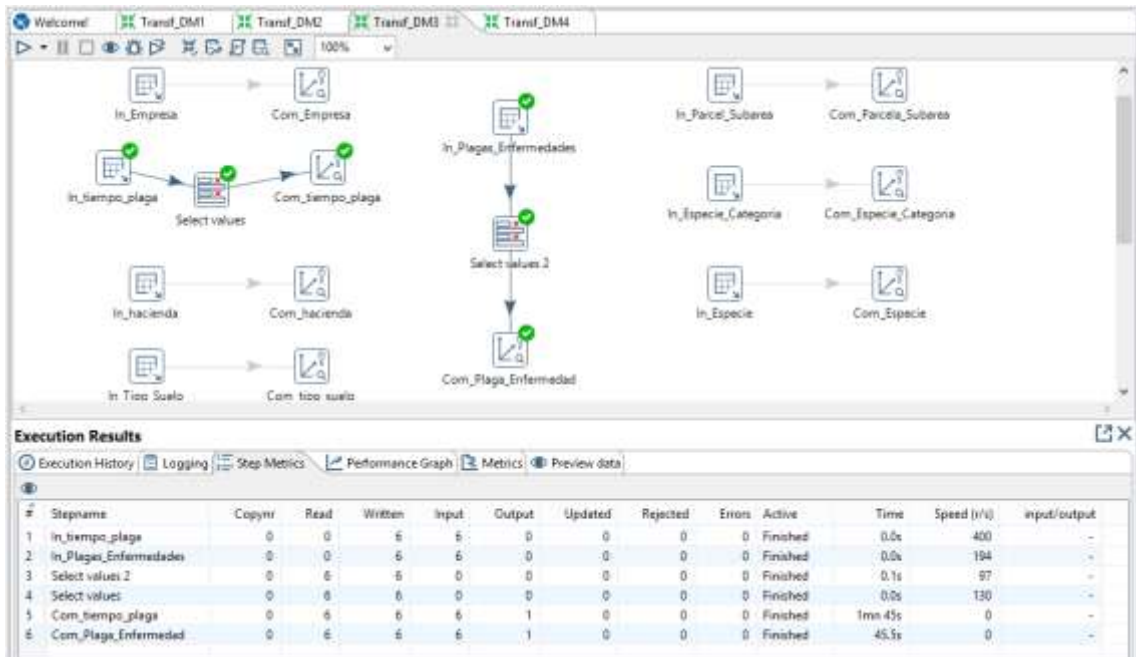
100%

Execution Results

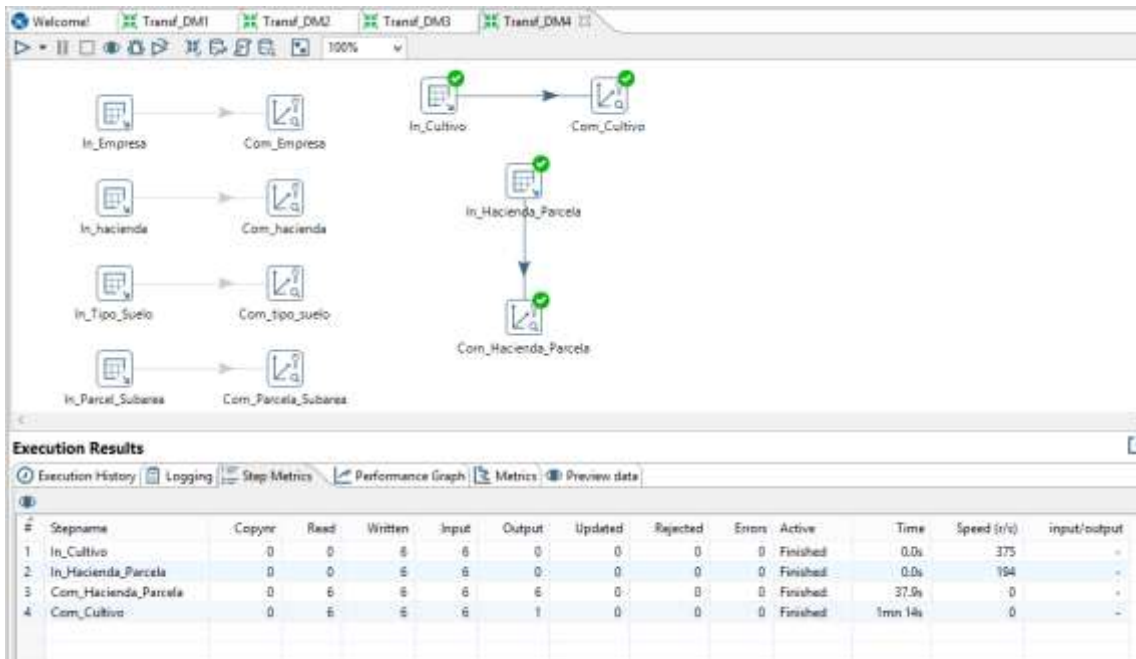
Execution History | Logging | Step Metrics | Performance Graph | Metrics | Preview data

#	StepName	Copysr	Read	Written	Input	Output	Updated	Rejected	Errors	Active	Time	Speed (r/s)	input/output
1	In_tiempo_cosecha	0	0	7	7	0	0	0	0	Finished	0.0s	467	-
2	Select values	0	7	7	0	0	0	0	0	Finished	0.0s	152	-
3	In_Cosecha	0	0	7	7	0	0	0	0	Finished	0.0s	226	-
4	Com_tiempo_cosecha	0	7	7	7	1	0	0	0	Finished	1mn 33s	0	-
5	Com_cosecha	0	7	7	7	1	0	0	0	Finished	47.8s	0	-

Data Mart 3: Plagas enfermedades



Data Mart 4: Hacienda parcela



Anexo N

Nº	DESCRIPCIÓN	RESULTADOS			Observaciones
		Satisfactorio	Poco satisfactorio	No satisfactorio	
1	¿Es fácil utilizar el Dashboard por primera vez?				
2	¿El panel de visualización es claro?				
3	¿Las acciones a realizar son intuitivas?				
4	¿El usuario puede controlar los gráficos de forma conveniente?				
5	¿Los términos usados en el dashboard para describir medidas, indican de forma clara lo que representan?				
6	¿Es posible observar de forma global lo que abarca el contenido de cada sección?				
7	¿El texto de los mensajes de cada gráfico es significativo e identifica los valores necesarios para la toma de decisiones?				
8	¿Los colores empleados para la presentación de los gráficos dentro del dashboard, van en la línea del logo del proyecto?				
9	¿El diseño de la interfaz de cada sección, sigue la misma consistencia en todas las secciones del dashboard?				
10	¿Existen elementos que representen los datos globales de cada sección?				
11	¿Le han parecido claros y representativos los nombres y descripciones que aparecen en cada gráfico del dashboard?				
12	¿La información que se presenta en cada gráfico tiene el tamaño adecuado?				
13	¿Cree que este dashboard puede ser implementado para toma de decisiones en agricultura?				

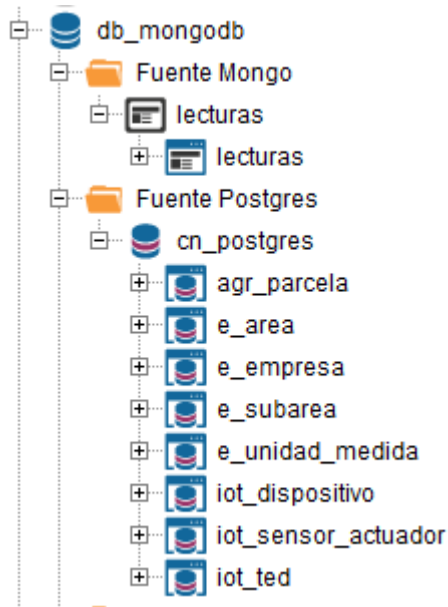
Nombre Evaluador:

FIRMA

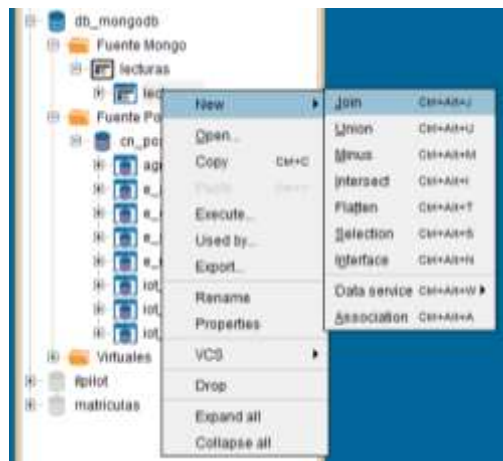
Anexo O

Combinación de datos Data Mart 5, que corresponde a los datos proporcionados por la red de sensores y los datos del sistemas transacción que contienen nombre de dispositivo, empresa, tipo de sensor, etc.

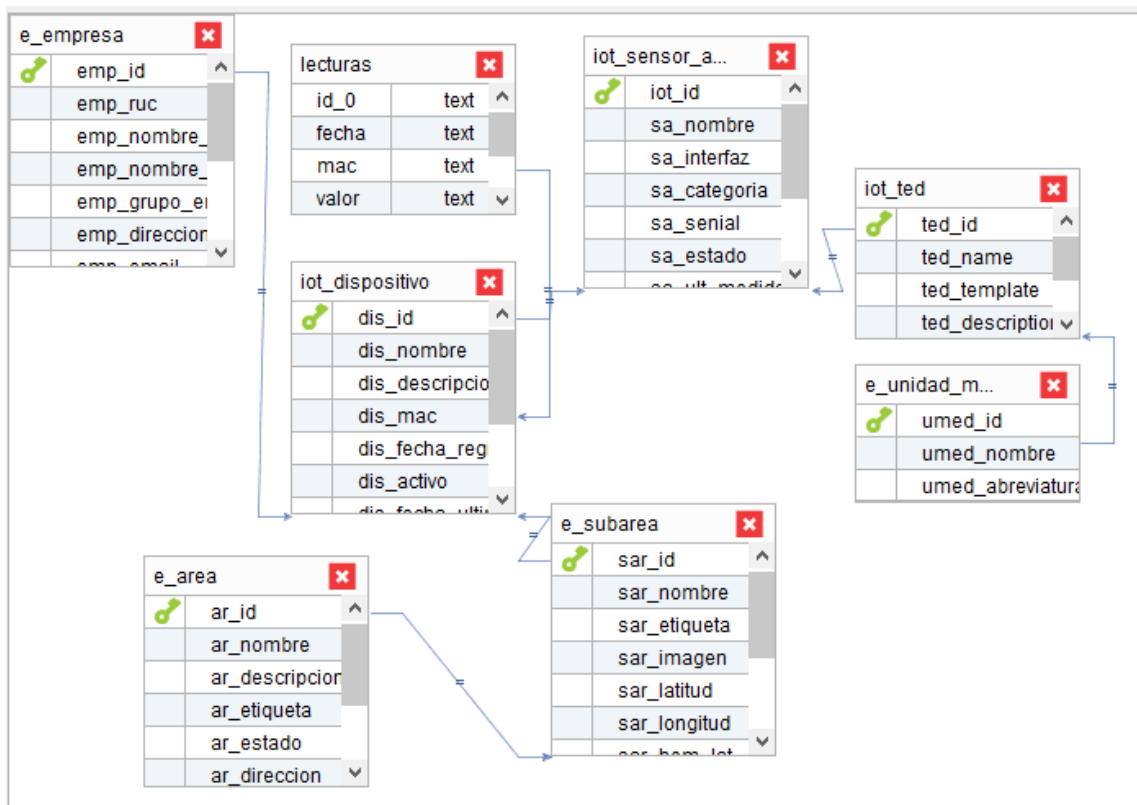
Como primer paso se debe crear las fuentes de datos respectivamente:



Una vez virtualizados los datos, se procede a crear la unión de tablas, dando click derecho en la tabla lecturas y seleccionando **Join**.



Una vez arrastradas todas las tablas necesarias, se procede a realizar las conexiones para poder generar la vista adecuada.



Luego, eliminamos los campos repetidos de tal forma que muestre solo los datos necesarios.

View name: dm_lecturass

PK	View Name	Field Name	Field Type
<input checked="" type="checkbox"/>	lecturas	mac	text
<input checked="" type="checkbox"/>	dm_lecturass	valor	double
<input type="checkbox"/>	lecturas	interfaz	text
<input type="checkbox"/>	iot_dispositivo	dis_nombre	text
<input type="checkbox"/>	iot_sensor_actuador	sa_nombre	text
<input type="checkbox"/>	e_empresa	emp_ruc	text
<input type="checkbox"/>	e_empresa	emp_nombre_comercial	text
<input type="checkbox"/>	iot_ted	ted_name	text
<input type="checkbox"/>	e_unidad_medida	umed_nombre	text
<input type="checkbox"/>	e_unidad_medida	umed_abreviatura	text
<input type="checkbox"/>	e_subarea	sar_nombre	text
<input type="checkbox"/>	e_area	ar_nombre	text

Una vez descartadas ciertas tablas, se procede a guardar y a ejecutar la vista para verificar los datos.

db_jotmach_dwh.dm_lecturas

Summary Edit Options VQL Execution panel Tree view Used by Data lineage Associations Publish Export Drop

Database: db_jotmach_dwh

View type: Derived

View schema:

PK	Field Name	Field Type
	valor_promedio	double
	valor_maximo	double
	valor_minimo	double

Execute Query Results

Results Execution Trace Refresh Save Query: SELECT * FROM dm_lecturas CONTEXT ('18/8/20', 'cache_wait_for_load=true') TRACE

Total rows received: 18308 (shown 150)

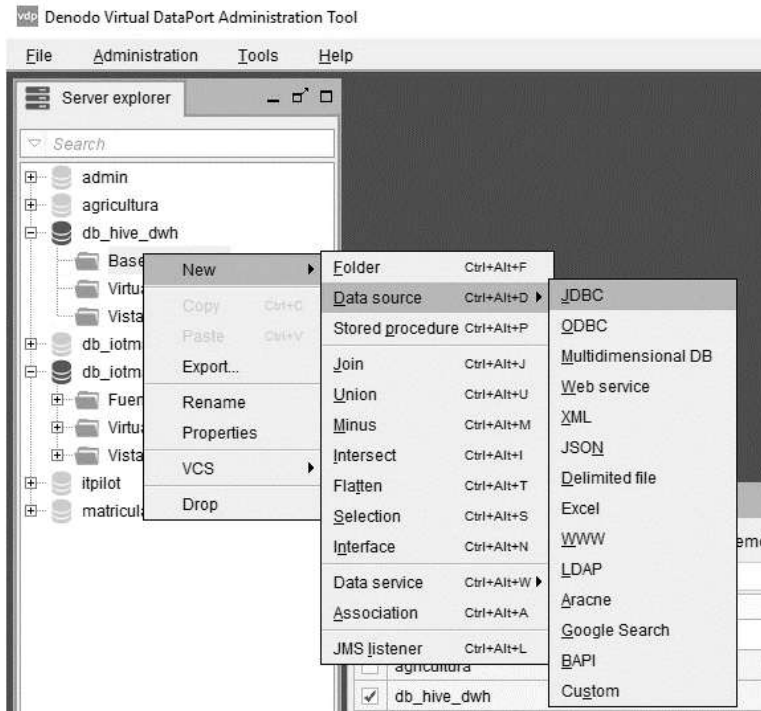
valor_prome...	valor_maximo	valor_minimo	desviacion_e...	varianza	fecha	ruc	empresa	hacienda	parcela	mac	unidad
60.0	60.0	60.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
1.0	1.0	1.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
1.0	1.0	1.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
7.0	7.0	7.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
1.0	1.0	1.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
1.0	1.0	1.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
70.0	70.0	70.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
1.0	1.0	1.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
0.0	0.0	0.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
62.0	62.0	62.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
0.0	0.0	0.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
0.0	0.0	0.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
36.0	36.0	36.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
1.0	1.0	1.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
1.0	1.0	1.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
58.0	58.0	58.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
0.0	0.0	0.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
1.0	1.0	1.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad
66.0	66.0	66.0	0.0	0.0	Wed Aug 24 ...	0704640333...	DIUYAY	Hacienda Je...	SuParcela 2	F1E2D3C4...	Humedad

Connected to database (127.0.0.1:9999)ib_mongodb as user angel

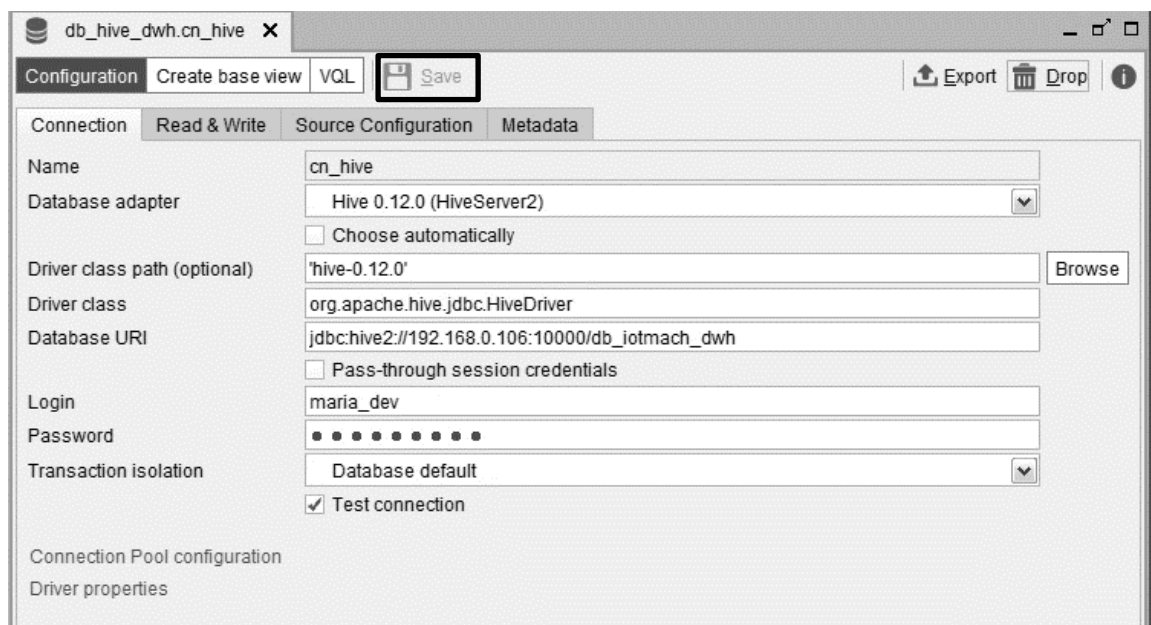
Anexo P

Sentencias de vistas en Denodo Platform

Para crear las vistas necesarias, primero se crea un **Data source JDBC** o fuente de datos de la siguiente forma:



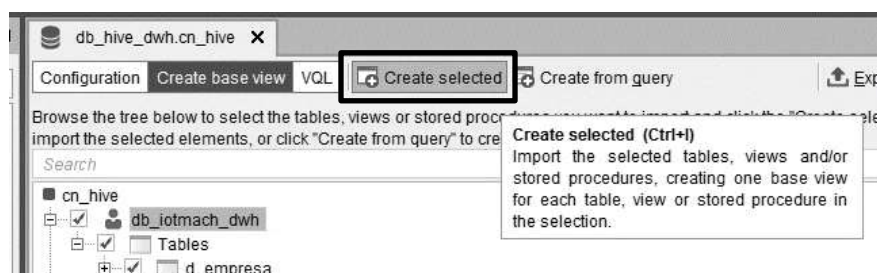
Aquí se llena la información correspondiente a la fuente de datos, que en este caso es la base de datos Hive con la estructura DW creada anteriormente. Y se guarda la conexión con el botón **Save**.



Se selecciona la opción **Create base view** donde se despliega la estructura DW creada en la fuente de datos y se escoge todas las tablas correspondientes



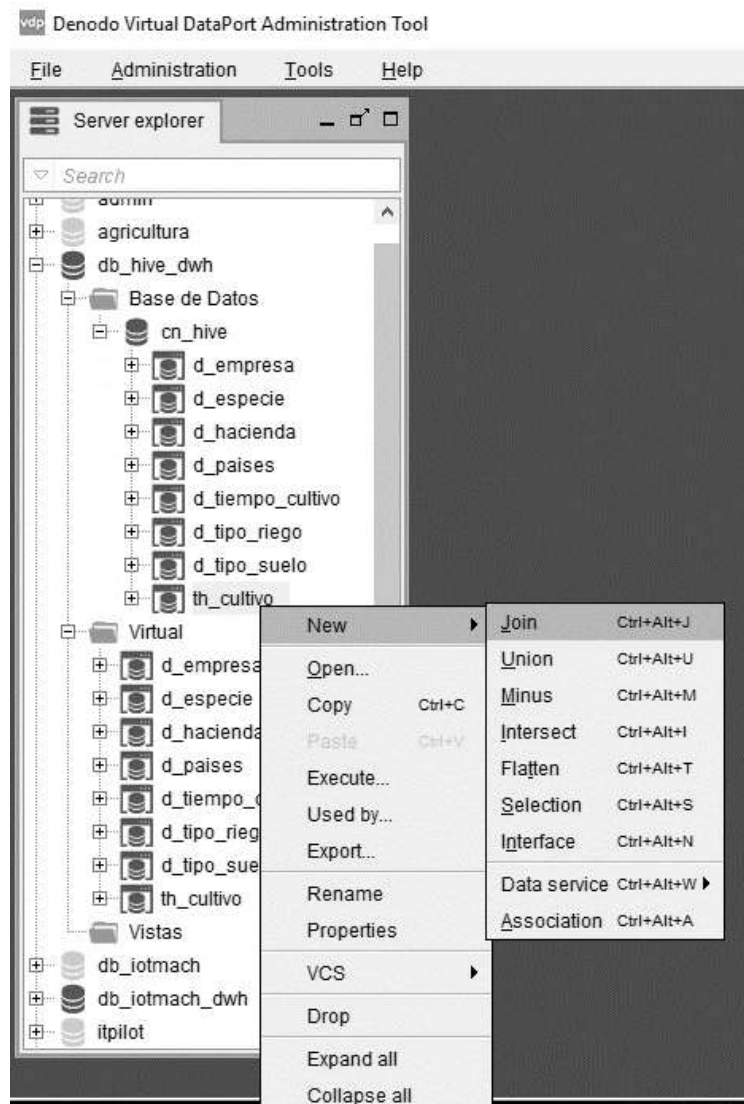
Luego se da click en **Create selected** para crear la virtualización de las tablas seleccionadas.



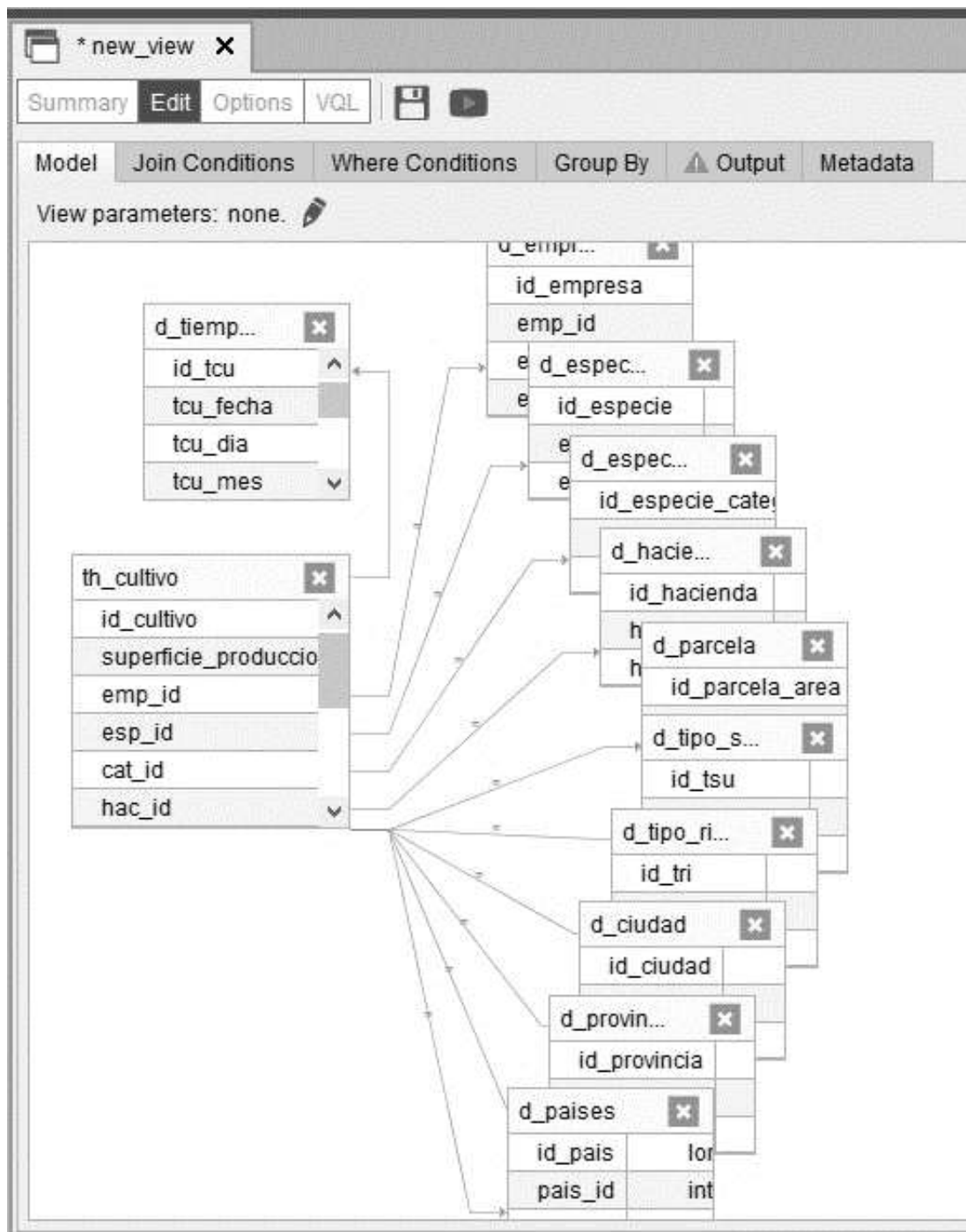
Se espera que el proceso termine.



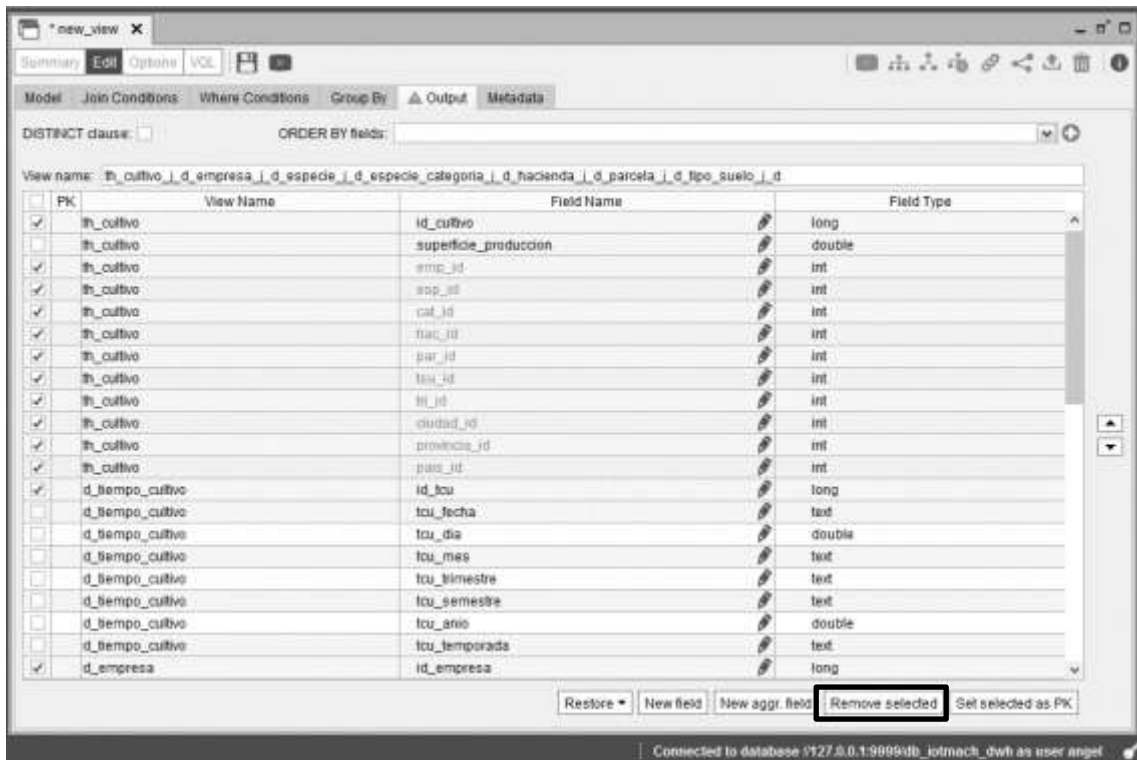
Para la creación de la vista de “Cubo 1: Cultivos”, se da click derecho sobre la tabla de hechos y se selecciona **New Join**.



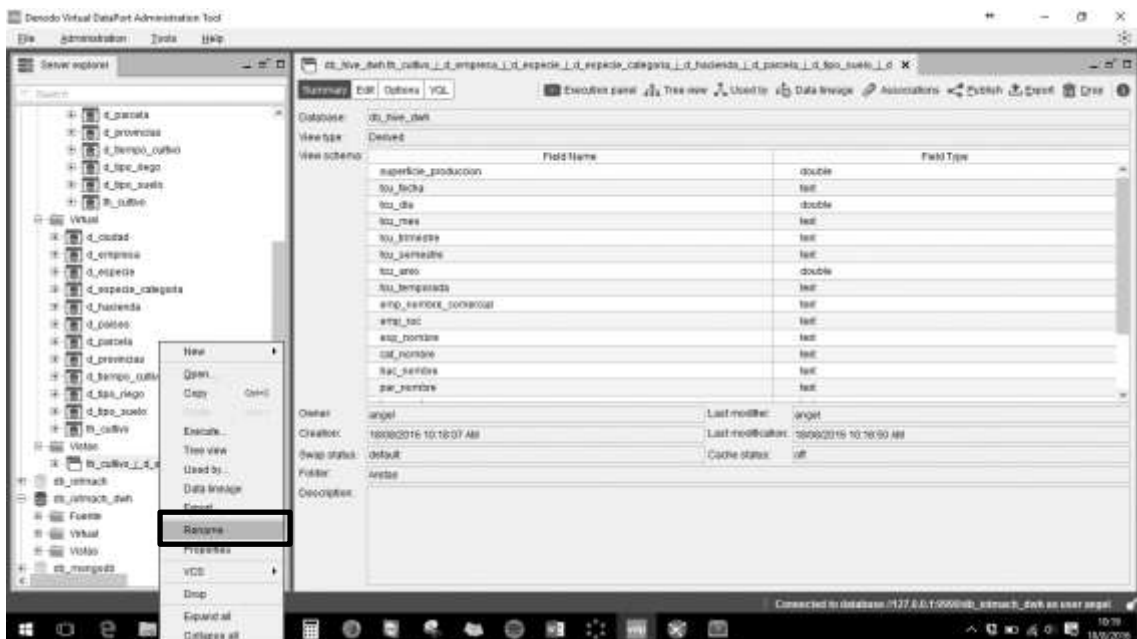
Luego se debe arrastrar todas las tablas necesarias para formar la vista y se realiza la conexión respectiva de acuerdo al modelo creado.



Se debe eliminar los campos no necesarios en la vista en la opción **Output** seleccionando cada una y luego click en **Remove selected**.



Una vez eliminado los campos innecesarios, se renombra la vista dando click derecho en la vista y seleccionar **Reame**, para mayor comodidad al momento de referenciarla.



Luego de renombrarla, se procede a editar la vista en la opción **VQL**, renombrando las tablas mediante la opción **AS** y agregando las opciones necesarias para acomodar la vista.

The screenshot shows a SQL editor window titled '* db_hive_dwh.view_cultivo'. The 'VQL' tab is active. Below the toolbar, there are 'VQL options' including 'Show dependencies' (unchecked) and 'Drop elements before' (checked). The main text area contains the following SQL code:

```

CREATE OR REPLACE VIEW view_cultivo FOLDER = '/vistas' AS
SELECT th_cultivo.superficie_produccion AS superficie_produccion,
d_tiempo_cultivo.tcu_fecha AS fecha,
d_tiempo_cultivo.tcu_dia AS dia,
CASE
    WHEN d_tiempo_cultivo.tcu_mes = '1' THEN 'Enero'
    WHEN d_tiempo_cultivo.tcu_mes = '2' THEN 'Febrero'
    WHEN d_tiempo_cultivo.tcu_mes = '3' THEN 'Marzo'
    WHEN d_tiempo_cultivo.tcu_mes = '4' THEN 'Abril'
    WHEN d_tiempo_cultivo.tcu_mes = '5' THEN 'Mayo'
    WHEN d_tiempo_cultivo.tcu_mes = '6' THEN 'Junio'
    WHEN d_tiempo_cultivo.tcu_mes = '7' THEN 'Julio'
    WHEN d_tiempo_cultivo.tcu_mes = '8' THEN 'Agosto'
    WHEN d_tiempo_cultivo.tcu_mes = '9' THEN 'Septiembre'
    WHEN d_tiempo_cultivo.tcu_mes = '10' THEN 'Octubre'
    WHEN d_tiempo_cultivo.tcu_mes = '11' THEN 'Noviembre'
    WHEN d_tiempo_cultivo.tcu_mes = '12' THEN 'Diciembre'
    ELSE 'Desconocido'
END AS mes,
d_tiempo_cultivo.tcu_trimestre AS trimestre,
d_tiempo_cultivo.tcu_semestre AS semestre,
d_tiempo_cultivo.tcu_anio AS anio,
d_tiempo_cultivo.tcu_temporada AS temporada,
d_empresa.emp_nombre_comercial AS nombre_comercial,
d_empresa.emp_ruc AS ruc,
d_especie.esp_nombre AS especie,
d_especie_categoria.cat_nombre AS categoria,
d_hacienda.hac_nombre AS hacienda,
d_parcela.par_nombre AS parcela,
d_tipo_suelo.tsu_nombre AS tipo_suelo,

```

Vista:

```

CREATE OR REPLACE VIEW dm_cultivo FOLDER = '/vistas' AS
SELECT th_cultivo.superficie_produccion AS superficie_produccion,
d_tiempo_cultivo.tcu_fecha AS fecha,
d_tiempo_cultivo.tcu_dia AS dia,
CASE
    WHEN d_tiempo_cultivo.tcu_mes= '1' THEN 'Enero'
    WHEN d_tiempo_cultivo.tcu_mes= '2' THEN 'Febrero'
    WHEN d_tiempo_cultivo.tcu_mes= '3' THEN 'Marzo'
    WHEN d_tiempo_cultivo.tcu_mes= '4' THEN 'Abril'
    WHEN d_tiempo_cultivo.tcu_mes= '5' THEN 'Mayo'
    WHEN d_tiempo_cultivo.tcu_mes= '6' THEN 'Junio'
    WHEN d_tiempo_cultivo.tcu_mes= '7' THEN 'Julio'
    WHEN d_tiempo_cultivo.tcu_mes= '8' THEN 'Agosto'
    WHEN d_tiempo_cultivo.tcu_mes= '9' THEN 'Septiembre'
    WHEN d_tiempo_cultivo.tcu_mes= '10' THEN 'Octubre'
    WHEN d_tiempo_cultivo.tcu_mes= '11' THEN 'Noviembre'
    WHEN d_tiempo_cultivo.tcu_mes= '12' THEN 'Diciembre'
    ELSE 'Desconocido'
END AS mes,

```

```

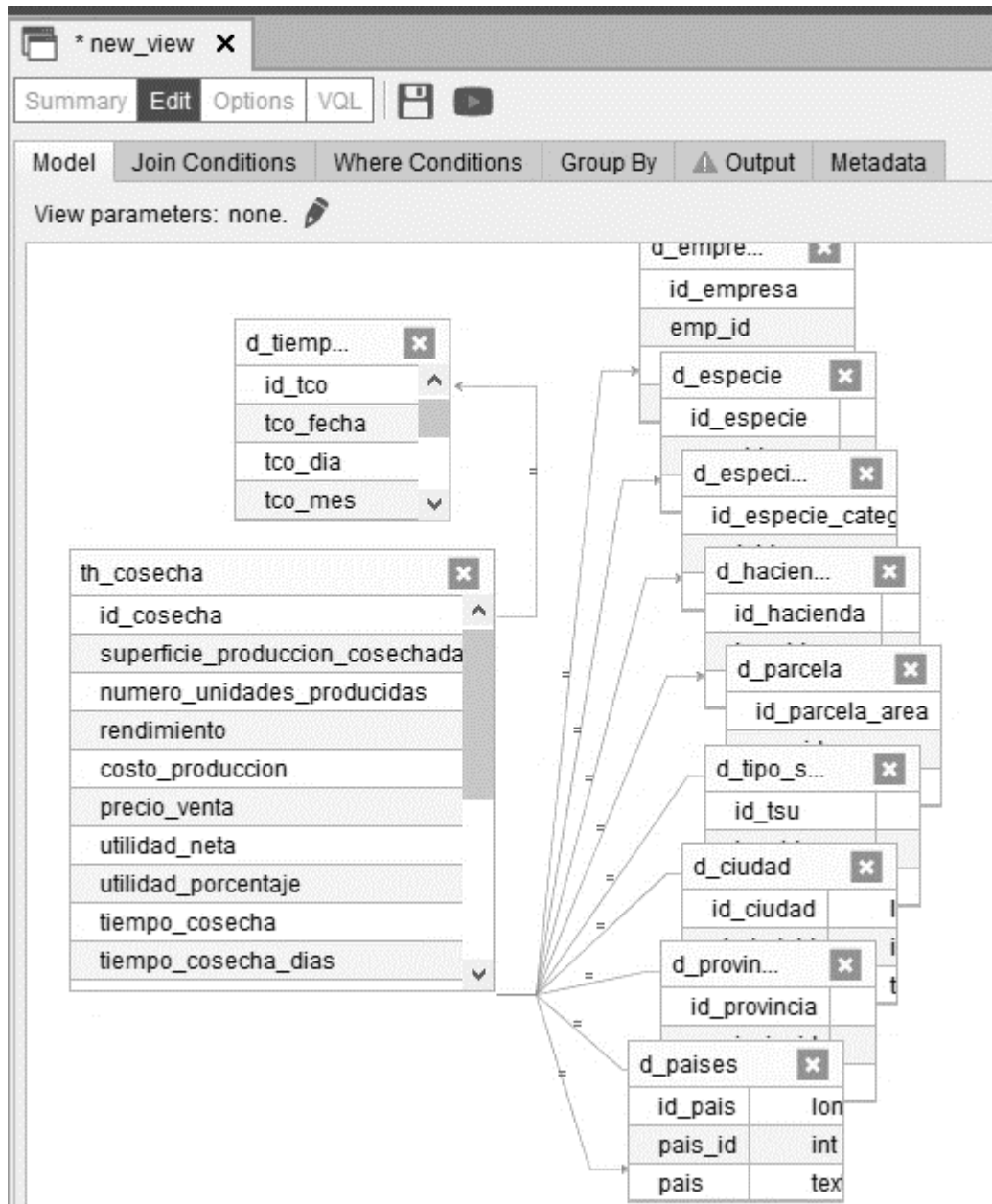
d_tiempo_cultivo.tcu_trimestre AS trimestre,
d_tiempo_cultivo.tcu_semestre AS semestre,
d_tiempo_cultivo.tcu_anio AS anio,
d_tiempo_cultivo.tcu_temporada AS temporada,
d_empresa.emp_nombre_comercial AS empresa,
d_empresa.emp_ruc AS ruc,
d_especie.esp_nombre AS especie,
d_especie_categoria.cat_nombre AS categoria,
d_hacienda.hac_nombre AS hacienda,
d_parcela.par_nombre AS parcela,
d_tipo_suelo.tsu_nombre AS tipo_suelo,
d_tipo_riego.tri_nombre AS tipo_riego,
d_ciudad.ciudad AS ciudad,
d_provincias.provincia AS provincia,
d_paises.pais AS pais
FROM (((((((th_cultivo AS th_cultivo INNER JOIN
d_tiempo_cultivo AS d_tiempo_cultivo ON th_cultivo.id_cultivo = d_tiempo_cultivo.id_tcu ) INNER JOIN
d_empresa AS d_empresa ON th_cultivo.emp_id = d_empresa.emp_id ) INNER JOIN
d_especie AS d_especie ON th_cultivo.esp_id = d_especie.esp_id ) INNER JOIN
d_especie_categoria AS d_especie_categoria ON th_cultivo.cat_id = d_especie_categoria.cat_id ) INNER
JOIN
d_hacienda AS d_hacienda ON th_cultivo.hac_id = d_hacienda.hac_id ) INNER JOIN
d_parcela AS d_parcela ON th_cultivo.par_id = d_parcela.par_id ) INNER JOIN
d_tipo_suelo AS d_tipo_suelo ON th_cultivo.tsu_id = d_tipo_suelo.tsu_id ) INNER JOIN
d_tipo_riego AS d_tipo_riego ON th_cultivo.tri_id = d_tipo_riego.tri_id ) INNER JOIN
d_ciudad AS d_ciudad ON th_cultivo.ciudad_id = d_ciudad.ciudad_id ) INNER JOIN
d_provincias AS d_provincias ON th_cultivo.provincia_id = d_provincias.provincia_id ) INNER JOIN
d_paises AS d_paises ON th_cultivo.pais_id = d_paises.pais_id CONTEXT ('formatted' = 'yes');

```

Cubo 2: Cosechas

Para crear las vistas posteriores se desarrolla el mismo proceso, seleccionando la tabla de hecho y dimensiones respectivas.





new_view X

Summary Edit Options VQL

Model Join Conditions Where Conditions Group By Output Metadata

DISTINCT clause: ORDER BY fields:

View name: th_cosecha_i_d_tiempo_cosecha_i_d_empresa_i_d_especie_i_d_especie_categoria_i_d_hacienda_i_d_parcela

PK	View Name	Field Name	Field Type
<input checked="" type="checkbox"/>	th_cosecha	id_cosecha	long
<input type="checkbox"/>	th_cosecha	superficie_produccion_cosechada	double
<input type="checkbox"/>	th_cosecha	numero_unidades_producidas	double
<input type="checkbox"/>	th_cosecha	rendimiento	double
<input type="checkbox"/>	th_cosecha	costo_produccion	double
<input type="checkbox"/>	th_cosecha	precio_venta	double
<input type="checkbox"/>	th_cosecha	utilidad_neta	double
<input type="checkbox"/>	th_cosecha	utilidad_porcentaje	double
<input type="checkbox"/>	th_cosecha	tiempo_cosecha	text
<input type="checkbox"/>	th_cosecha	tiempo_cosecha_dias	int
<input checked="" type="checkbox"/>	th_cosecha	emp_id	int
<input checked="" type="checkbox"/>	th_cosecha	esp_id	int
<input checked="" type="checkbox"/>	th_cosecha	cat_id	int
<input checked="" type="checkbox"/>	th_cosecha	hac_id	int
<input checked="" type="checkbox"/>	th_cosecha	par_id	int
<input checked="" type="checkbox"/>	th_cosecha	tsu_id	int
<input checked="" type="checkbox"/>	th_cosecha	ciudad_id	int
<input checked="" type="checkbox"/>	th_cosecha	provincia_id	int
<input checked="" type="checkbox"/>	th_cosecha	pais_id	int
<input checked="" type="checkbox"/>	d_tiempo_cosecha	lco_fecha	long
<input type="checkbox"/>	d_tiempo_cosecha	lco_fecha	date

Restore New field New agg. field Remove selected Set selected as PK

new_view X

Summary Edit Options VQL

Model Join Conditions Where Conditions Group By Output Metadata

DISTINCT clause: ORDER BY fields:

View name: th_cosecha_i_d_tiempo_cosecha_i_d_empresa_i_d_especie_i_d_especie_categoria_i_d_hacienda_i_d_parcela

PK	View Name	Field Name	Field Type
<input type="checkbox"/>	th_cosecha	utilidad_neta	double
<input type="checkbox"/>	th_cosecha	utilidad_porcentaje	double
<input type="checkbox"/>	th_cosecha	tiempo_cosecha	text
<input type="checkbox"/>	th_cosecha	tiempo_cosecha_dias	int
<input type="checkbox"/>	d_tiempo_cosecha	lco_fecha	date
<input type="checkbox"/>	d_tiempo_cosecha	lco_dia	double
<input type="checkbox"/>	d_tiempo_cosecha	lco_mes	text
<input type="checkbox"/>	d_tiempo_cosecha	lco_trimestre	text
<input type="checkbox"/>	d_tiempo_cosecha	lco_semestre	text
<input type="checkbox"/>	d_tiempo_cosecha	lco_anio	double
<input type="checkbox"/>	d_tiempo_cosecha	lco_temporada	text
<input type="checkbox"/>	d_empresa	emp_nombre_comercial	text
<input type="checkbox"/>	d_empresa	emp_ruc	text
<input type="checkbox"/>	d_especie	esp_nombre	text
<input type="checkbox"/>	d_especie_categoria	cat_nombre	text
<input type="checkbox"/>	d_hacienda	hac_nombre	text
<input type="checkbox"/>	d_parcela	par_nombre	text
<input type="checkbox"/>	d_spo_suelo	tsu_nombre	text
<input type="checkbox"/>	d_ciudad	ciudad	text
<input type="checkbox"/>	d_provincias	provincia	text
<input type="checkbox"/>	d_paises	pais	text

Restore New field New agg. field Remove selected Set selected as PK

Connected to database //127.0.0.1:9999/db_jotmach_10wh as user angel

```

db_hive_dwh_view_cosecha X
Summary Edit Options SQL
VQL options:
 Show dependencies  Drop elements before creating them  Replace existing elements  Do not replace existing elements

CREATE OR REPLACE VIEW view_cosecha FOLDER = '/vistas' AS
SELECT
th_cosecha.superficie_produccion_cosechada AS superficie_produccion_cosechada,
th_cosecha.numero_unidades_producidas AS numero_unidades_producidas,
th_cosecha.rendimiento AS rendimiento,
th_cosecha.costo_produccion AS costo_produccion,
th_cosecha.precio_venta AS precio_venta,
th_cosecha.utilidad_neta AS utilidad_neta,
th_cosecha.utilidad_porcentaje AS utilidad_porcentaje,
th_cosecha.tiempo_cosecha AS tiempo_cosecha,
th_cosecha.tiempo_cosecha_dias AS tiempo_cosecha_dias,
d_tiempo_cosecha.tco_fecha AS fecha,
d_tiempo_cosecha.tco_dia AS dia,
CASE
WHEN d_tiempo_cosecha.tco_mes = '1' THEN 'Enero'
WHEN d_tiempo_cosecha.tco_mes = '2' THEN 'Febrero'
WHEN d_tiempo_cosecha.tco_mes = '3' THEN 'Marzo'
WHEN d_tiempo_cosecha.tco_mes = '4' THEN 'Abril'
WHEN d_tiempo_cosecha.tco_mes = '5' THEN 'Mayo'
WHEN d_tiempo_cosecha.tco_mes = '6' THEN 'Junio'
WHEN d_tiempo_cosecha.tco_mes = '7' THEN 'Julio'
WHEN d_tiempo_cosecha.tco_mes = '8' THEN 'Agosto'
WHEN d_tiempo_cosecha.tco_mes = '9' THEN 'Septiembre'
WHEN d_tiempo_cosecha.tco_mes = '10' THEN 'Octubre'
WHEN d_tiempo_cosecha.tco_mes = '11' THEN 'Noviembre'
WHEN d_tiempo_cosecha.tco_mes = '12' THEN 'Diciembre'
ELSE 'Desconocido'
END AS mes,
d_tiempo_cosecha.tco_trimestre AS trimestre,
d_tiempo_cosecha.tco_semestre AS semestre,
d_tiempo_cosecha.tco_año AS año,
d_tiempo_cosecha.tco_temporada AS temporada,
d_empresa.emp_nombre_comercial AS nombre_comercial,
d_empresa.emp_ruc AS ruc,
d_especie.esp_nombre AS especie,
d_especie_categoria.cat_nombre AS categoria,
d_hacienda.hac_nombre AS hacienda,
d_parcela.par_nombre AS parcela,
d_tipo_suelo.tsu_nombre AS tipo_suelo,
d_ciudad.ciudad AS ciudad,
d_provincias.provincia AS provincia,
d_paises.pais AS pais
FROM (((((((th_cosecha AS th_cosecha INNER JOIN
d_tiempo_cosecha AS d_tiempo_cosecha ON th_cosecha.id_cosecha = d_tiempo_cosecha.id_tco ) INNER JOIN
d_empresa AS d_empresa ON th_cosecha.emp_id = d_empresa.emp_id ) INNER JOIN
d_especie AS d_especie ON th_cosecha.esp_id = d_especie.esp_id ) INNER JOIN
d_especie_categoria AS d_especie_categoria ON th_cosecha.cat_id = d_especie_categoria.cat_id ) INNER JOIN
d_hacienda AS d_hacienda ON th_cosecha.hac_id = d_hacienda.hac_id ) INNER JOIN
d_parcela AS d_parcela ON th_cosecha.par_id = d_parcela.par_id ) INNER JOIN
d_tipo_suelo AS d_tipo_suelo ON th_cosecha.tsu_id = d_tipo_suelo.tsu_id ) INNER JOIN
d_ciudad AS d_ciudad ON th_cosecha.ciudad_id = d_ciudad.ciudad_id ) INNER JOIN
d_provincias AS d_provincias ON th_cosecha.provincia_id = d_provincias.provincia_id ) INNER JOIN
d_paises AS d_paises ON th_cosecha.pais_id = d_paises.pais_id CONTEXT ('formatted' = 'yes');

```

Vista:

```

CREATE OR REPLACE VIEW dm_cosecha FOLDER = '/vistas' AS
SELECT th_cosecha.superficie_produccion_cosechada AS superficie_produccion_cosechada,
th_cosecha.numero_unidades_producidas AS numero_unidades_producidas,
th_cosecha.rendimiento AS rendimiento,
th_cosecha.costo_produccion AS costo_produccion,
th_cosecha.precio_venta AS precio_venta,
th_cosecha.utilidad_neta AS utilidad_neta,
th_cosecha.utilidad_porcentaje AS utilidad_porcentaje,
th_cosecha.tiempo_cosecha AS tiempo_cosecha,
th_cosecha.tiempo_cosecha_dias AS tiempo_cosecha_dias,
d_tiempo_cosecha.tco_fecha AS fecha,
d_tiempo_cosecha.tco_dia AS dia,
CASE
WHEN d_tiempo_cosecha.tco_mes= '1' THEN 'Enero'
WHEN d_tiempo_cosecha.tco_mes= '2' THEN 'Febrero'
WHEN d_tiempo_cosecha.tco_mes= '3' THEN 'Marzo'
WHEN d_tiempo_cosecha.tco_mes= '4' THEN 'Abril'
WHEN d_tiempo_cosecha.tco_mes= '5' THEN 'Mayo'
WHEN d_tiempo_cosecha.tco_mes= '6' THEN 'Junio'

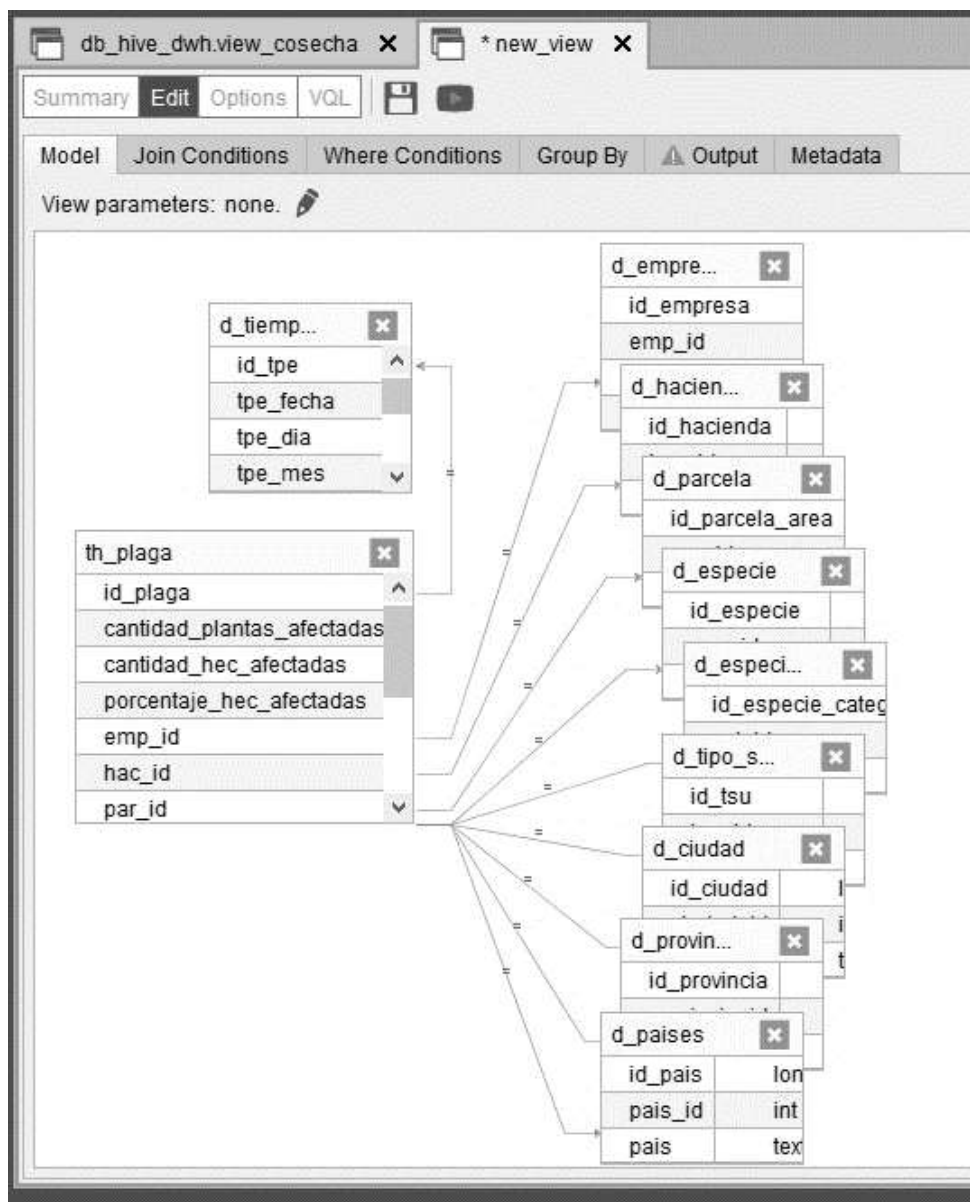
```

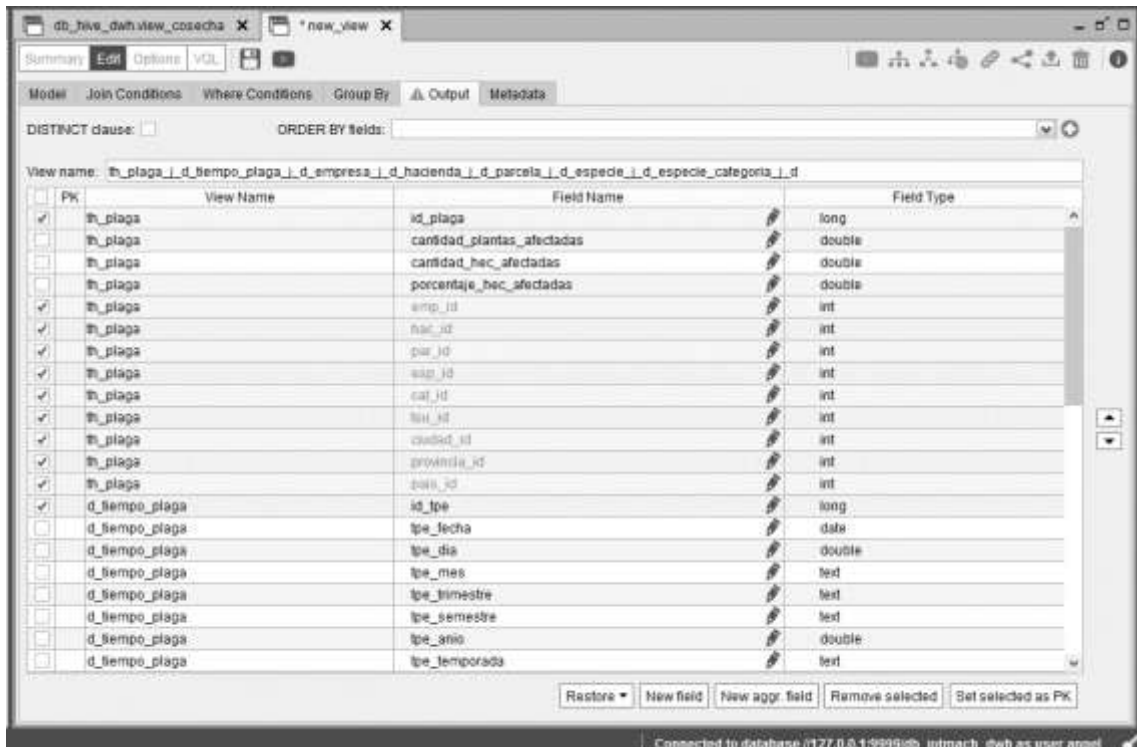
```

        WHEN d_tiempo_cosecha.tco_mes= '7' THEN 'Julio'
        WHEN d_tiempo_cosecha.tco_mes= '8' THEN 'Agosto'
        WHEN d_tiempo_cosecha.tco_mes= '9' THEN 'Septiembre'
        WHEN d_tiempo_cosecha.tco_mes= '10' THEN 'Octubre'
        WHEN d_tiempo_cosecha.tco_mes= '11' THEN 'Noviembre'
        WHEN d_tiempo_cosecha.tco_mes= '12' THEN 'Diciembre'
        ELSE 'Desconocido'
    END AS mes,
    d_tiempo_cosecha.tco_trimestre AS trimestre,
    d_tiempo_cosecha.tco_semestre AS semestre,
    d_tiempo_cosecha.tco_anio AS anio,
    d_tiempo_cosecha.tco_temporada AS temporada,
    d_empresa.emp_nombre_comercial AS empresa,
    d_empresa.emp_ruc AS ruc,
    d_especie.esp_nombre AS especie,
    d_especie_categoria.cat_nombre AS categoria,
    d_hacienda.hac_nombre AS hacienda,
    d_parcela.par_nombre AS parcela,
    d_tipo_suelo.tsu_nombre AS tipo_suelo,
    d_ciudad.ciudad AS ciudad,
    d_provincias.provincia AS provincia,
    d_paises.pais AS pais
FROM (((((((th_cosecha AS th_cosecha INNER JOIN
d_tiempo_cosecha AS d_tiempo_cosecha ON th_cosecha.id_cosecha = d_tiempo_cosecha.id_tco ) INNER JOIN
d_empresa AS d_empresa ON th_cosecha.emp_id = d_empresa.emp_id ) INNER JOIN
d_especie AS d_especie ON th_cosecha.esp_id = d_especie.esp_id ) INNER JOIN
d_especie_categoria AS d_especie_categoria ON th_cosecha.cat_id = d_especie_categoria.cat_id ) INNER
JOIN
d_hacienda AS d_hacienda ON th_cosecha.hac_id = d_hacienda.hac_id ) INNER JOIN
d_parcela AS d_parcela ON th_cosecha.par_id = d_parcela.par_id ) INNER JOIN
d_tipo_suelo AS d_tipo_suelo ON th_cosecha.tsu_id = d_tipo_suelo.tsu_id ) INNER JOIN
d_ciudad AS d_ciudad ON th_cosecha.ciudad_id = d_ciudad.ciudad_id ) INNER JOIN
d_provincias AS d_provincias ON th_cosecha.provincia_id = d_provincias.provincia_id ) INNER JOIN
d_paises AS d_paises ON th_cosecha.pais_id = d_paises.pais_id CONTEXT ('formatted' = 'yes');

```

Cubo 3: Plagas y enfermedades





```

CREATE OR REPLACE VIEW view_plagas FOLDER = '/vistas' AS
SELECT
th_plaga.cantidad_plantas_afectadas AS cantidad_plantas_afectadas,
th_plaga.cantidad_hec_afectadas AS cantidad_hec_afectadas,
th_plaga.porcentaje_hec_afectadas AS porcentaje_hec_afectadas,
d_tiempo_plaga.tpe_fecha AS fecha,
d_tiempo_plaga.tpe_dia AS dia,
CASE
    WHEN d_tiempo_plaga.tpe_mes = '1' THEN 'Enero'
    WHEN d_tiempo_plaga.tpe_mes = '2' THEN 'Febrero'
    WHEN d_tiempo_plaga.tpe_mes = '3' THEN 'Marzo'
    WHEN d_tiempo_plaga.tpe_mes = '4' THEN 'Abril'
    WHEN d_tiempo_plaga.tpe_mes = '5' THEN 'Mayo'
    WHEN d_tiempo_plaga.tpe_mes = '6' THEN 'Junio'
    WHEN d_tiempo_plaga.tpe_mes = '7' THEN 'Julio'
    WHEN d_tiempo_plaga.tpe_mes = '8' THEN 'Agosto'
    WHEN d_tiempo_plaga.tpe_mes = '9' THEN 'Septiembre'
    WHEN d_tiempo_plaga.tpe_mes = '10' THEN 'Octubre'
    WHEN d_tiempo_plaga.tpe_mes = '11' THEN 'Noviembre'
    WHEN d_tiempo_plaga.tpe_mes = '12' THEN 'Diciembre'
    ELSE 'Desconocido'
END AS mes,
d_tiempo_plaga.tpe_trimestre AS trimestre,
d_tiempo_plaga.tpe_semestre AS semestre,
d_tiempo_plaga.tpe_anio AS anio,
d_tiempo_plaga.tpe_temporada AS temporada,
d_empresa.emp_nombre_comercial AS nombre_comercial,
d_empresa.emp_ruc AS ruc,
d_hacienda.hac_nombre AS hacienda,
d_parcela.par_nombre AS parcela,

```

```

ELSE 'Desconocido'
END AS mes,
d_tiempo_plaga.tpe_trimestre AS trimestre,
d_tiempo_plaga.tpe_semestre AS semestre,
d_tiempo_plaga.tpe_anio AS anio,
d_tiempo_plaga.tpe_temporada AS temporada,
d_empresa.emp_nombre_comercial AS nombre_comercial,
d_empresa.emp_ruc AS ruc,
d_hacienda.hac_nombre AS hacienda,
d_parcela.par_nombre AS parcela,
d_especie.esp_nombre AS especie,
d_especie_categoria.cat_nombre AS categoria,
d_tipo_suelo.tsu_nombre AS tipo_suelo,
d_ciudad.ciudad AS ciudad,
d_provincias.provincia AS provincia,
d_paises.pais AS pais
FROM
((((((((th_plaga AS th_plaga INNER JOIN
d_tiempo_plaga AS d_tiempo_plaga ON th_plaga.id_plaga = d_tiempo_plaga.id_tpe ) INNER JOIN
d_empresa AS d_empresa ON th_plaga.emp_id = d_empresa.emp_id ) INNER JOIN
d_hacienda AS d_hacienda ON th_plaga.hac_id = d_hacienda.hac_id ) INNER JOIN
d_parcela AS d_parcela ON th_plaga.par_id = d_parcela.par_id ) INNER JOIN
d_especie AS d_especie ON th_plaga.esp_id = d_especie.esp_id ) INNER JOIN
d_especie_categoria AS d_especie_categoria ON th_plaga.cat_id = d_especie_categoria.cat_id ) INNER JOIN
d_tipo_suelo AS d_tipo_suelo ON th_plaga.tsu_id = d_tipo_suelo.tsu_id ) INNER JOIN
d_ciudad AS d_ciudad ON th_plaga.ciudad_id = d_ciudad.ciudad_id ) INNER JOIN
d_provincias AS d_provincias ON th_plaga.provincia_id = d_provincias.provincia_id ) INNER JOIN
d_paises AS d_paises ON th_plaga.pais_id = d_paises.pais_id CONTEXT ('formatted' = 'yes');

```

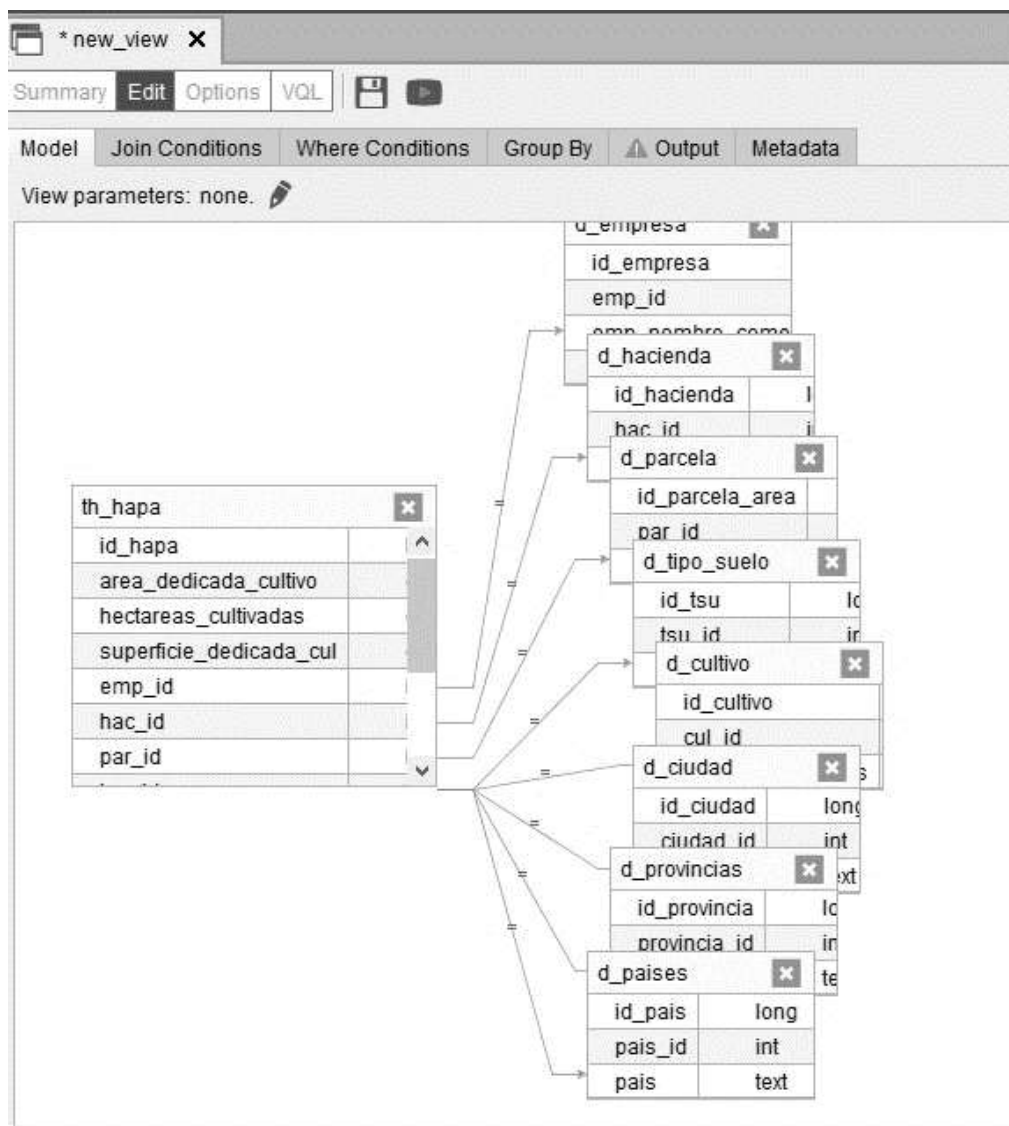
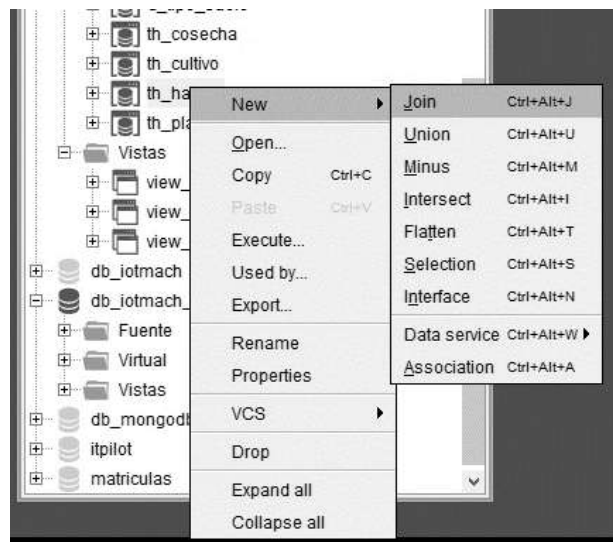
Vista:

```

CREATE OR REPLACE VIEW dm_plaga FOLDER = '/vistas' AS
SELECT th_plaga.cantidad_plantas_afectadas AS cantidad_plantas_afectadas,
th_plaga.cantidad_hec_afectadas AS cantidad_hec_afectadas,
th_plaga.porcentaje_hec_afectadas AS porcentaje_hec_afectadas,
d_tiempo_plaga.tpe_fecha AS fecha,
d_tiempo_plaga.tpe_dia AS dia,
CASE
    WHEN d_tiempo_plaga.tpe_mes= '1' THEN 'Enero'
    WHEN d_tiempo_plaga.tpe_mes= '2' THEN 'Febrero'
    WHEN d_tiempo_plaga.tpe_mes= '3' THEN 'Marzo'
    WHEN d_tiempo_plaga.tpe_mes= '4' THEN 'Abril'
    WHEN d_tiempo_plaga.tpe_mes= '5' THEN 'Mayo'
    WHEN d_tiempo_plaga.tpe_mes= '6' THEN 'Junio'
    WHEN d_tiempo_plaga.tpe_mes= '7' THEN 'Julio'
    WHEN d_tiempo_plaga.tpe_mes= '8' THEN 'Agosto'
    WHEN d_tiempo_plaga.tpe_mes= '9' THEN 'Septiembre'
    WHEN d_tiempo_plaga.tpe_mes= '10' THEN 'Octubre'
    WHEN d_tiempo_plaga.tpe_mes= '11' THEN 'Noviembre'
    WHEN d_tiempo_plaga.tpe_mes= '12' THEN 'Diciembre'
    ELSE 'Mes Desconocido'
END AS mes,
d_tiempo_plaga.tpe_trimestre AS trimestre,
d_tiempo_plaga.tpe_semestre AS semestre,
d_tiempo_plaga.tpe_anio AS anio,
d_tiempo_plaga.tpe_temporada AS temporada,
d_empresa.emp_nombre_comercial AS empresa,
d_empresa.emp_ruc AS ruc,
d_hacienda.hac_nombre AS hacienda,
d_parcela.par_nombre AS parcela,
d_especie.esp_nombre AS especie,
d_especie_categoria.cat_nombre AS categoria,
d_tipo_suelo.tsu_nombre AS tipo_suelo,
d_ciudad.ciudad AS ciudad,
d_provincias.provincia AS provincia,
d_paises.pais AS pais
FROM (((((((th_plaga AS th_plaga INNER JOIN
d_tiempo_plaga AS d_tiempo_plaga ON th_plaga.id_plaga = d_tiempo_plaga.id_tpe ) INNER JOIN
d_empresa AS d_empresa ON th_plaga.emp_id = d_empresa.emp_id ) INNER JOIN
d_hacienda AS d_hacienda ON th_plaga.hac_id = d_hacienda.hac_id ) INNER JOIN
d_parcela AS d_parcela ON th_plaga.par_id = d_parcela.par_id ) INNER JOIN
d_especie AS d_especie ON th_plaga.esp_id = d_especie.esp_id ) INNER JOIN
d_especie_categoria AS d_especie_categoria ON th_plaga.cat_id = d_especie_categoria.cat_id ) INNER
JOIN
d_tipo_suelo AS d_tipo_suelo ON th_plaga.tsu_id = d_tipo_suelo.tsu_id ) INNER JOIN
d_ciudad AS d_ciudad ON th_plaga.ciudad_id = d_ciudad.ciudad_id ) INNER JOIN
d_provincias AS d_provincias ON th_plaga.provincia_id = d_provincias.provincia_id ) INNER JOIN
d_paises AS d_paises ON th_plaga.pais_id = d_paises.pais_id CONTEXT ('formatted' = 'yes');

```

Cubo 4: Haciendas y parcelas



new_view X

Summary Edit Options VQL

Model Join Conditions Where Conditions Group By Output Metadata

DISTINCT clause: ORDER BY fields:

View name: th_hapa | d_empresa | d_hacienda | d_parcela | d_tipo_suelo | d_cultivo | d_ciudad | d_provincias |

PK	View Name	Field Name	Field Type
<input checked="" type="checkbox"/>	th_hapa	id_hapa	long
<input type="checkbox"/>	th_hapa	area_dedicada_cultivo	double
<input type="checkbox"/>	th_hapa	hectareas_cultivadas	double
<input type="checkbox"/>	th_hapa	superficie_dedicada_cul	double
<input checked="" type="checkbox"/>	th_hapa	emp_id	int
<input checked="" type="checkbox"/>	th_hapa	hac_id	int
<input checked="" type="checkbox"/>	th_hapa	par_id	int
<input checked="" type="checkbox"/>	th_hapa	tau_id	int
<input checked="" type="checkbox"/>	th_hapa	cul_id	int
<input checked="" type="checkbox"/>	th_hapa	ciudad_id	int
<input checked="" type="checkbox"/>	th_hapa	provincia_id	int
<input checked="" type="checkbox"/>	th_hapa	pais_id	int
<input checked="" type="checkbox"/>	d_empresa	id_empresa	long
<input checked="" type="checkbox"/>	d_empresa	emp_id	int
<input type="checkbox"/>	d_empresa	emp_nombre_comercial	text
<input type="checkbox"/>	d_empresa	emp_ruc	text
<input checked="" type="checkbox"/>	d_hacienda	id_hacienda	long
<input checked="" type="checkbox"/>	d_hacienda	hac_id	int
<input type="checkbox"/>	d_hacienda	hac_nombre	text
<input checked="" type="checkbox"/>	d_parcela	id_parcela_area	long
<input checked="" type="checkbox"/>	d_parcela	par_id	int

Restore New field New aggr. field Remove selected Set selected as PK

Connected to database #127.0.0.1:9999/db_kitmach_dwh as user angel

db_hive_dwh.view_hacienda_parcela X

Summary Edit Options VQL

VQL options: Show dependencies Drop elements before creating them Replace existing elements Do not replace existing elements

```

CREATE OR REPLACE VIEW view_hacienda_parcela FOLDER = '/vistas' AS
SELECT
th_hapa.area_dedicada_cultivo AS area_dedicada_cultivo,
th_hapa.hectareas_cultivadas AS hectareas_cultivadas,
th_hapa.superficie_dedicada_cul AS superficie_dedicada_cul,
d_empresa.emp_nombre_comercial AS nombre_comercial,
d_empresa.emp_ruc AS ruc,
d_hacienda.hac_nombre AS hacienda,
d_parcela.par_nombre AS parcela,
d_tipo_suelo.tau_nombre AS tipo_suelo,
d_cultivo.numero_hectareas AS numero_hectareas,
d_ciudad.ciudad AS ciudad,
d_provincias.provincia AS provincia,
d_paises.pais AS pais
FROM
(((((((th_hapa AS th_hapa INNER JOIN
d_empresa AS d_empresa ON th_hapa.emp_id = d_empresa.emp_id )
INNER JOIN d_hacienda AS d_hacienda ON th_hapa.hac_id = d_hacienda.hac_id )
INNER JOIN d_parcela AS d_parcela ON th_hapa.par_id = d_parcela.par_id )
INNER JOIN d_tipo_suelo AS d_tipo_suelo ON th_hapa.tau_id = d_tipo_suelo.tau_id )
INNER JOIN d_cultivo AS d_cultivo ON th_hapa.cul_id = d_cultivo.cul_id )
INNER JOIN d_ciudad AS d_ciudad ON th_hapa.ciudad_id = d_ciudad.ciudad_id )
INNER JOIN d_provincias AS d_provincias ON th_hapa.provincia_id = d_provincias.provincia_id )
INNER JOIN d_paises AS d_paises ON th_hapa.pais_id = d_paises.pais_id ;

ALTER VIEW view_hacienda_parcela
LAYOUT {th_hapa = [50, 230, 320, 266], d_empresa = [480, 10, 200, 106], d_hacienda = [500, 120, 200, 86], d_parcela = [520, 210, 200, 86],

```

Word wrap

Connected to database #127.0.0.1:9999/db_kitmach_dwh as user angel

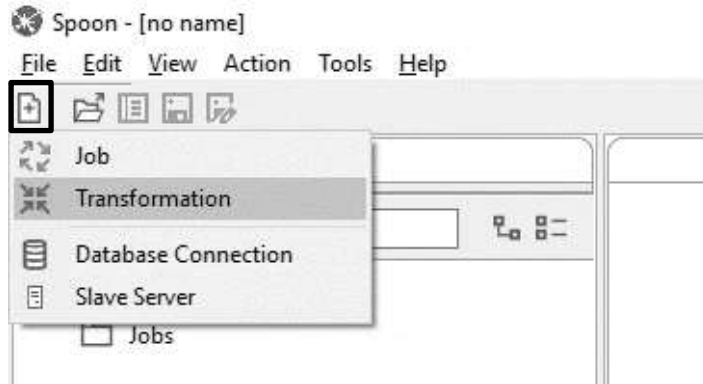
Vista:

```
CREATE OR REPLACE VIEW dm_hapa FOLDER = '/vistas' AS
SELECT th_hapa.area_dedicada_cultivo AS area_dedicada_cultivo,
th_hapa.hectareas_cultivadas AS hectareas_cultivadas,
th_hapa.superficie_dedicada_cul AS superficie_dedicada_culivo,
d_empresa.emp_nombre_comercial AS empresa,
d_empresa.emp_ruc AS ruc,
d_hacienda.hac_nombre AS hacienda,
d_parcela.par_nombre AS par_nombre, d_tipo_suelo.tsu_nombre AS tipo_suelo,
d_cultivo.numero_hectareas AS numero_hectareas, d_ciudad.ciudad AS ciudad,
d_provincias.provincia AS provincia, d_paises.pais AS pais
FROM ((((((th_hapa AS th_hapa INNER JOIN
d_empresa AS d_empresa ON th_hapa.emp_id = d_empresa.emp_id ) INNER JOIN
d_hacienda AS d_hacienda ON th_hapa.hac_id = d_hacienda.hac_id ) INNER JOIN
d_parcela AS d_parcela ON th_hapa.par_id = d_parcela.par_id ) INNER JOIN
d_tipo_suelo AS d_tipo_suelo ON th_hapa.tsu_id = d_tipo_suelo.tsu_id ) INNER JOIN
d_cultivo AS d_cultivo ON th_hapa.cul_id = d_cultivo.cul_id ) INNER JOIN
d_ciudad AS d_ciudad ON th_hapa.ciudad_id = d_ciudad.ciudad_id ) INNER JOIN
d_provincias AS d_provincias ON th_hapa.provincia_id = d_provincias.provincia_id ) INNER JOIN
d_paises AS d_paises ON th_hapa.pais_id = d_paises.pais_id CONTEXT ('formatted' = 'yes');
```

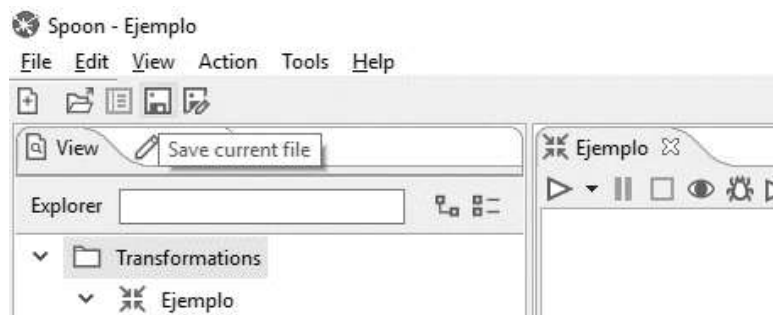
Anexo Q

Crear procesos ETL en la herramienta Spoon Pentaho data integration

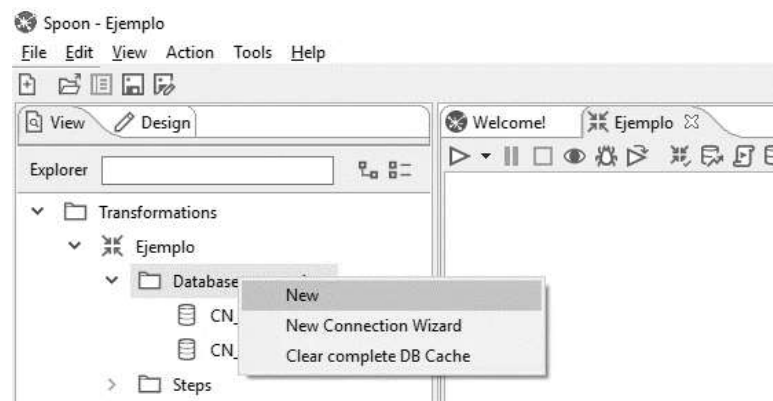
Al abrir la herramienta Spoon Pentaho Data Integration, En la opción **Nuevo** se selecciona **Transformation**.



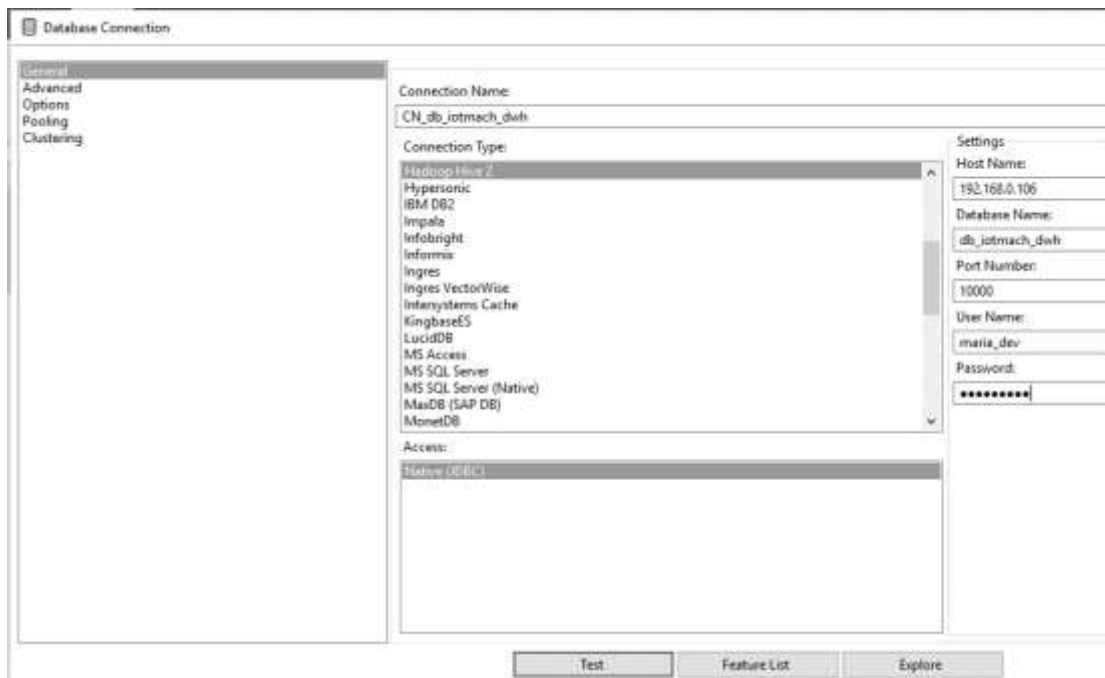
Una vez creada la transformación, se le da click en **Save current file** y se le da un nombre.



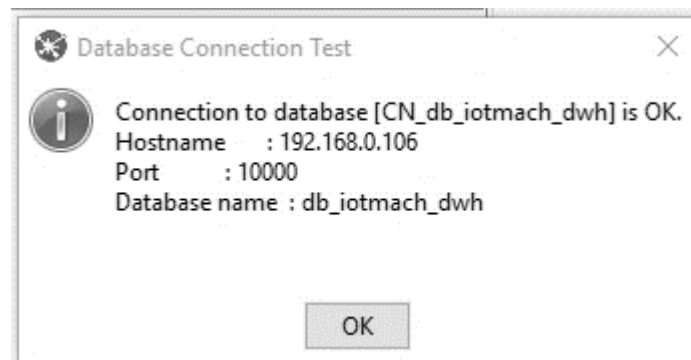
Una vez guardada la transformación, se crea la conexión a la base de datos, tanto para datos de entrada como para datos de salida. Click derecho en **Database Connections** luego **New**.



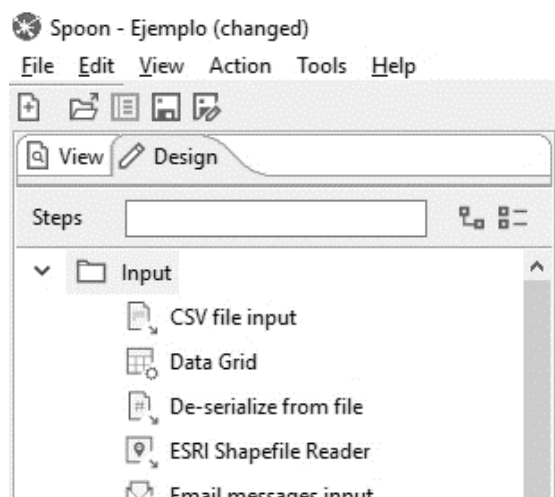
Aquí se llenan los datos correspondientes a la base de datos como el ejemplo.



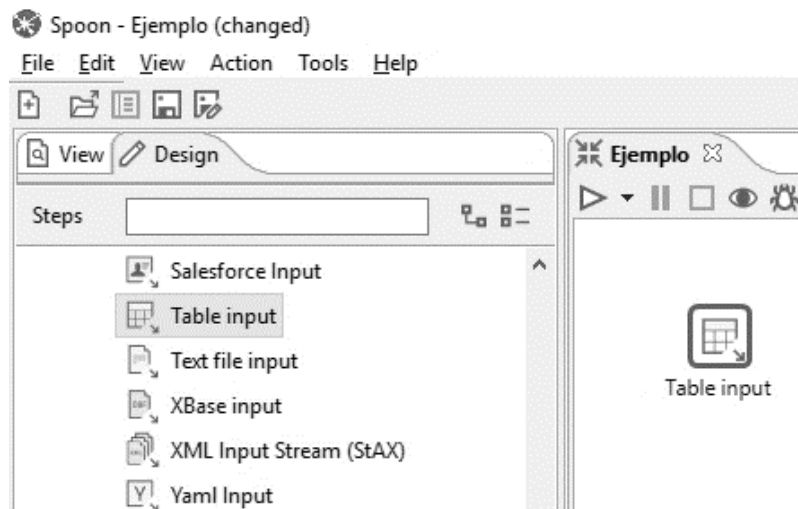
Una vez ingresados los datos, se da click en **Test** y así se verifica que la conexión sea correcta.



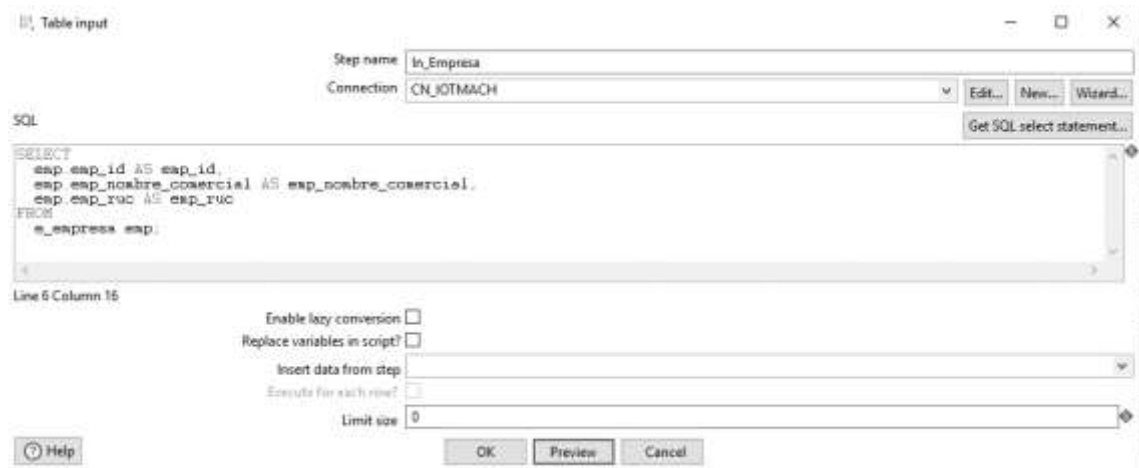
Luego de agregar las conexiones, se crea los datos de entrada, para esto se despliega la opción **Input** donde se encuentran varias formas para obtener los datos de entrada.



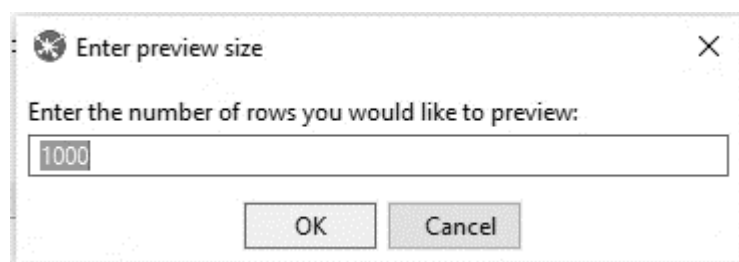
Se selecciona la opción **Tabla input** y la ubicamos en el área de trabajo arrastrándola.



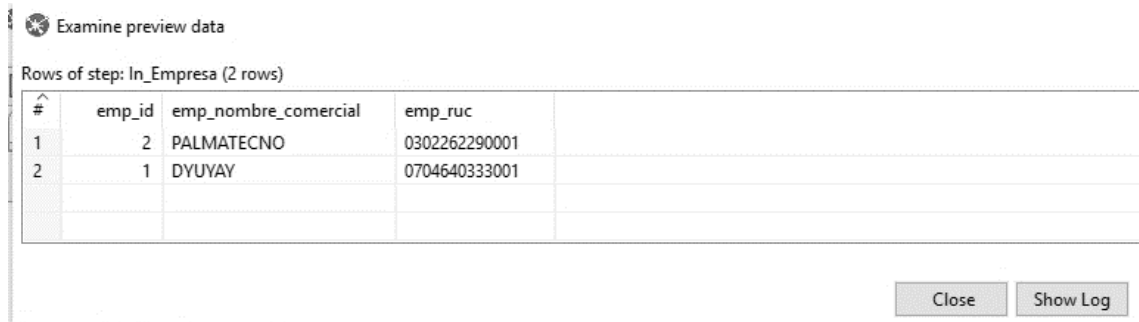
Se le da doble click para configurar la opción, donde se pone un nombre en **Setup name**, se selecciona la conexión correspondiente en **Connection** o si no está creada aún, se puede crear una nueva dando click en **Nex**. En **SQL** se escribe la sentencia que obtendrá los datos desde la base de datos de entrada, con la complejidad necesaria para obtenerlos. Luego click en **Preview** para pre visualizar los datos que se están obteniendo mediante la consulta sql.



Se escribe la cantidad de filas a mostrar, ya que, si es una tabla muy extensa, para pruebas se puede escribir una cantidad prudente y ahorrar tiempo, click en **Ok**.



Si los datos mostrados son los correctos, y no se presentan mensajes de error, click en **Close** y se da por finalizada la configuración del Input.



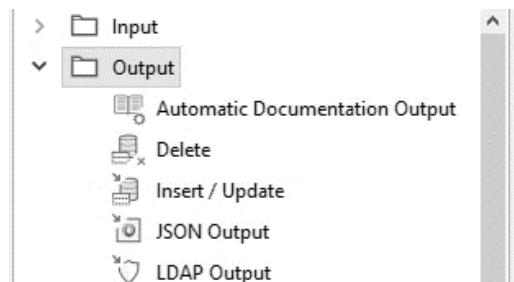
Examine preview data

Rows of step: In_Empresa (2 rows)

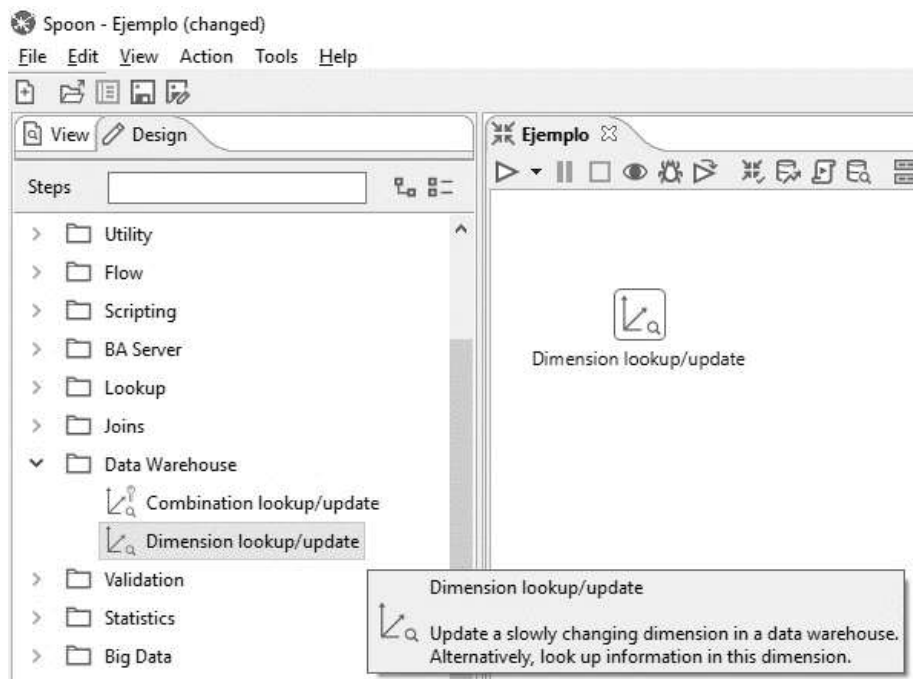
#	emp_id	emp_nombre_comercial	emp_ruc
1	2	PALMATECNO	0302262290001
2	1	DYUYAY	0704640333001

Close Show Log

Ahora se configura la salida de los datos, se despliega la opción **Output**, y encontraremos varias opciones de salida de datos.



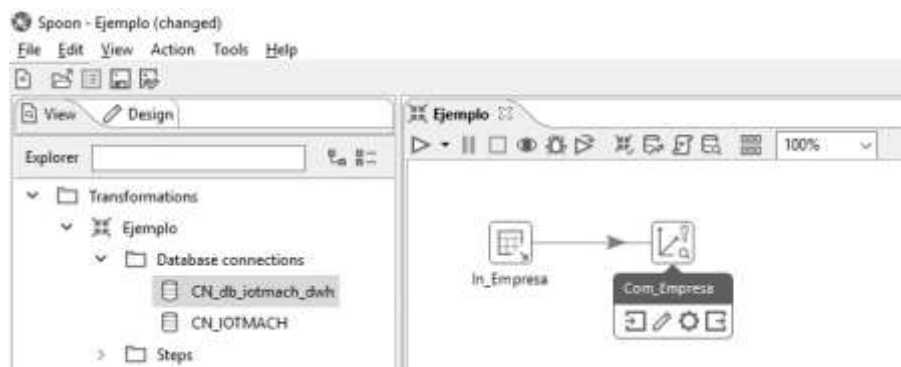
En este caso se despliega la opción **Data Warehouse** y se selecciona la opción **Dimension lookup/update** ya que crearemos una estructura DW, se arrastra al área de trabajo y se da doble click en la opción escogida para acceder a las configuraciones de salida de datos.



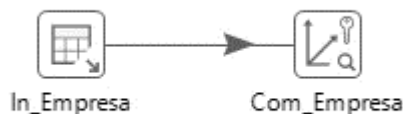
Antes de entrar a las configuraciones de **Dimension lookup/update**, se debe enlazar la entrada correspondiente para así obtener los datos transformados para cargarlos, para esto, se selecciona la opción **connector** como se muestra en el ejemplo:



Y se lo dirige hacia la opción de salida de datos correspondiente a los datos cargados.

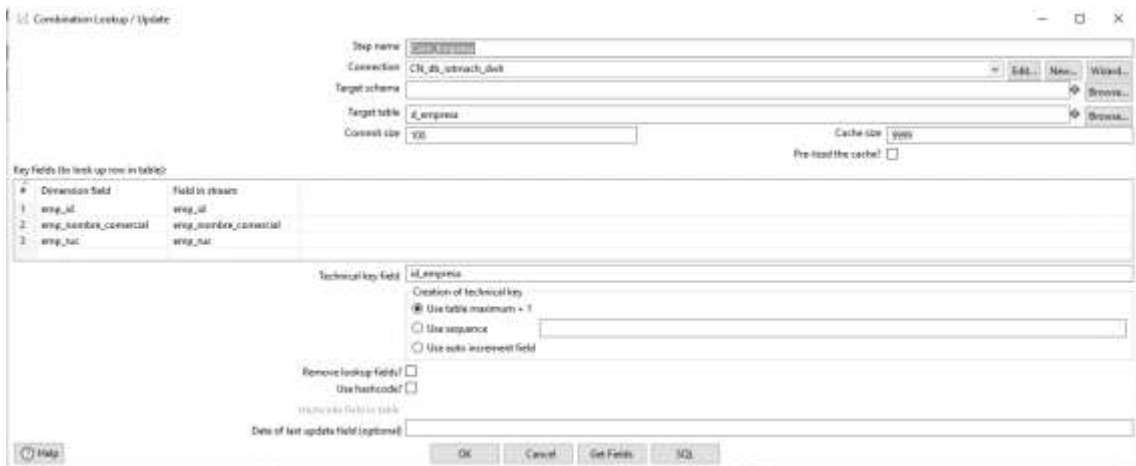


Quedando de la siguiente forma:

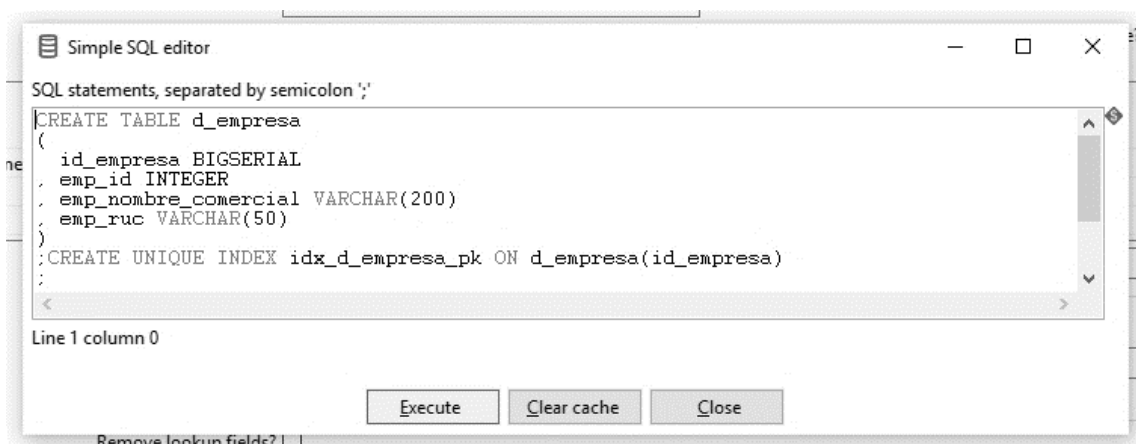


Luego de esto se accede a las configuraciones de **Dimension lookup/update**, donde en **Step name** se escribe el nombre, se selecciona la conexión correspondiente a la salida de datos en la opción **Connection**, se puede crear el nombre de la tabla ubicando el nombre en **Target table** o simplemente se selecciona la estructura creada en la base de datos en **Target schema**, dando click en **Browse**. Una vez llenando los datos se da click en **Get fields** para llenar los campos de dentro de las configuraciones y poder cambiar el nombre si es necesario, luego en **Technical key field** creamos un nombre que será la clave de la tabla.

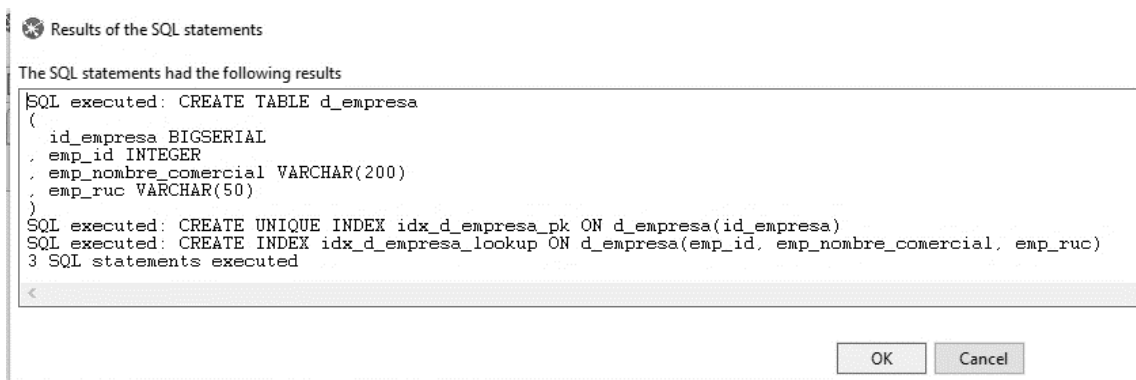




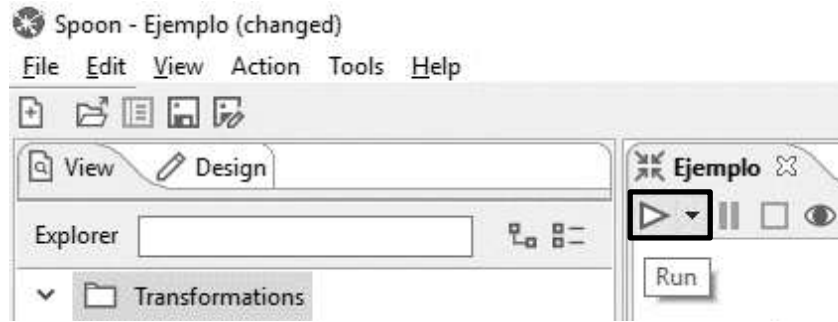
Se da click en **SQL** para crear el esquema de la tabla, y click en **Execute** para ejecutar la sentencia.



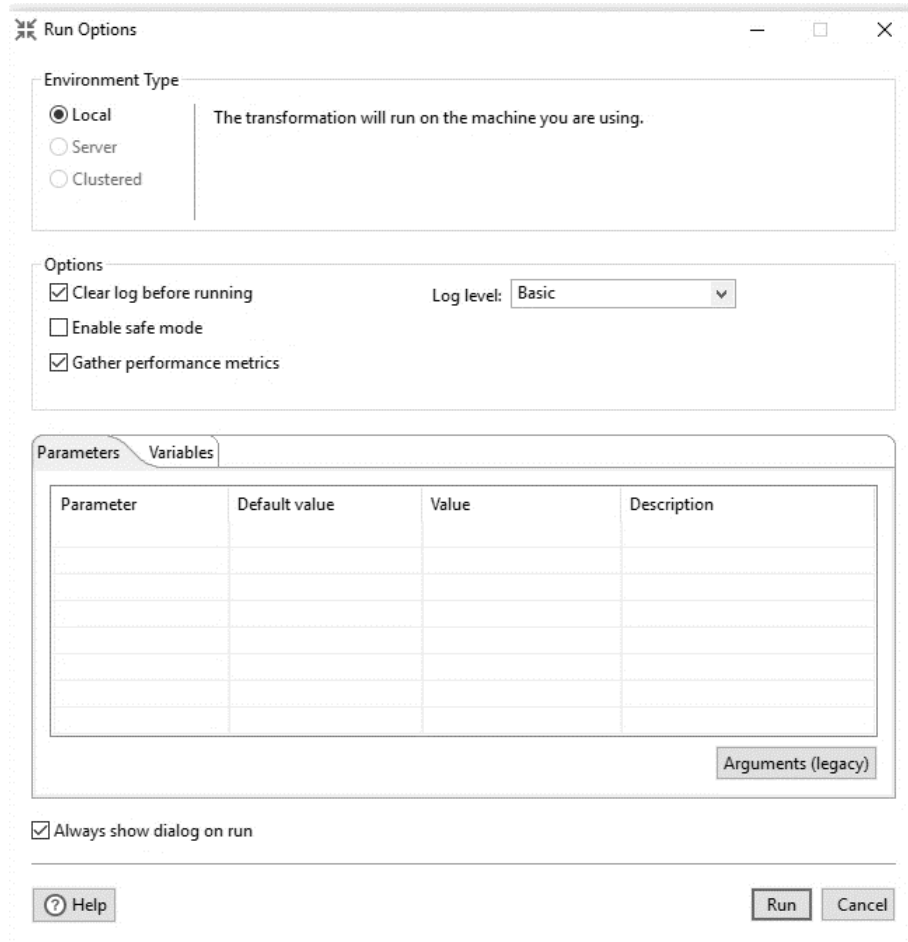
Una vez creada la tabla sin errores, click en **Ok** para dar por finalizada la configuración de **Dimension lookup/update** como opción de salida de datos.



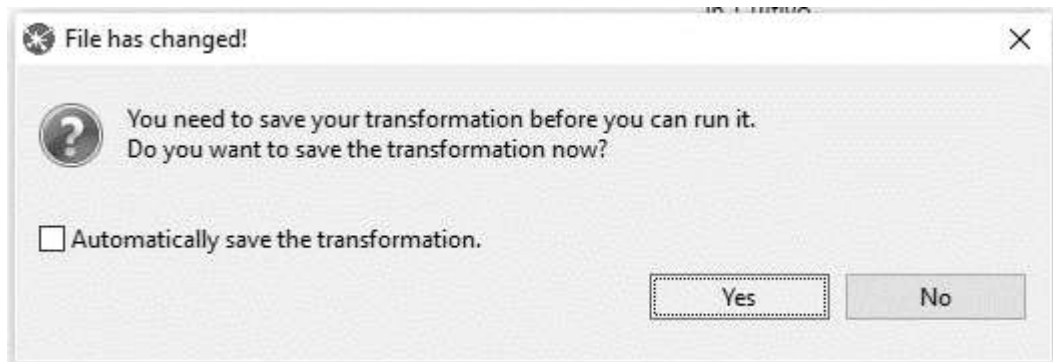
Por último, se ejecuta la transformación para realizar el proceso ETL, se debe ir a la opción **Run**.



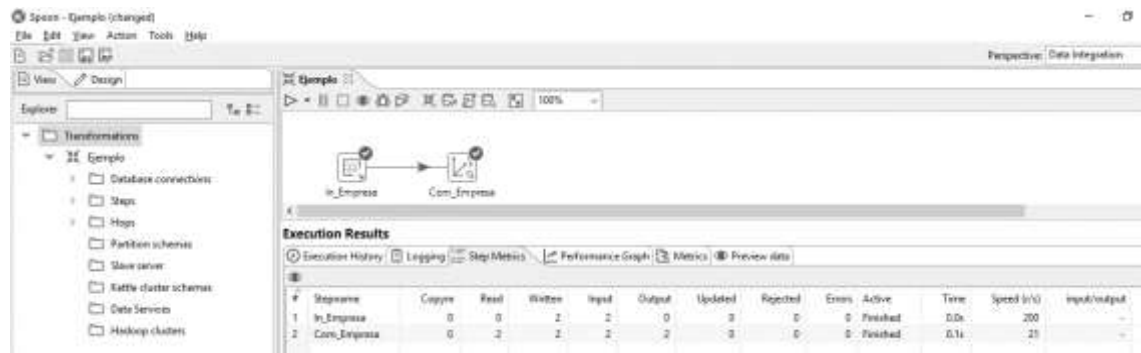
Se deja las opciones por defecto y click en **Run**.



Se debe guardar el modelo, se da click en **Yes** para continuar.



Se puede verificar el éxito de la carga de los datos con el visto sobre las opciones de entrada y salida, y verificamos el **Step Metrics** que muestra los datos leídos y escritos.



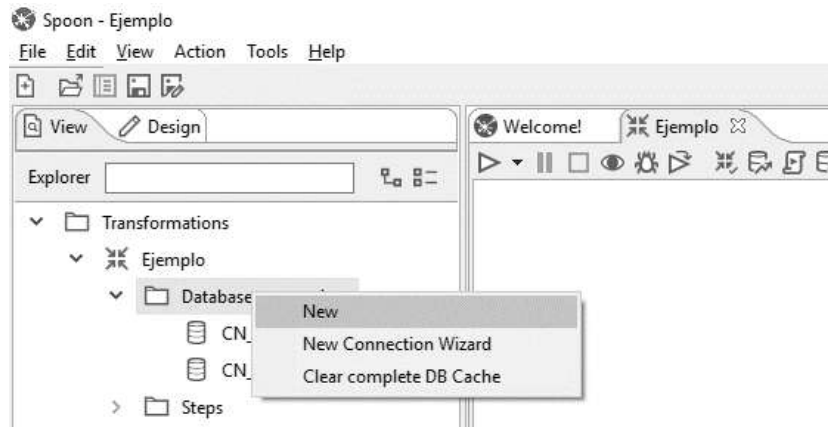
The screenshot shows the Apache Spark IDE interface. The main workspace displays a job graph with two steps: 'In_Empresa' and 'Com_Empresa'. The 'Step Metrics' tab is selected, showing the following execution results table:

#	Stepname	Copied	Read	Written	Input	Output	Updated	Rejected	Errors	Active	Time	Speed (r/s)	Input/output
1	In_Empresa	0	0	2	2	0	0	0	0	Finished	0.0s	200	-
2	Com_Empresa	0	2	2	2	2	0	0	0	Finished	0.1s	25	-

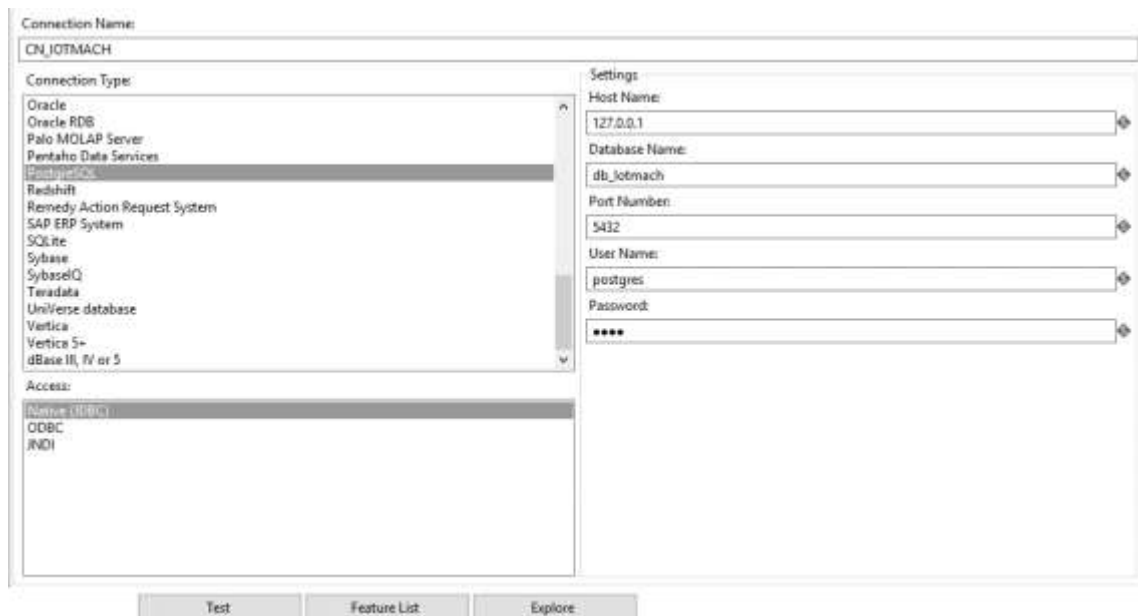
Anexo R

Conexión Spoon Data Integration con Postgresql

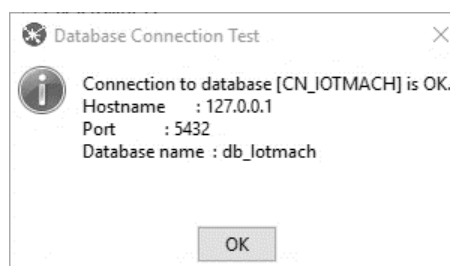
Para agregar una nueva conexión en Spoon, se da click derecho sobre **Databases connections** y luego click en **New**.



Aquí se llenan los datos correspondientes a la base de datos como el ejemplo:



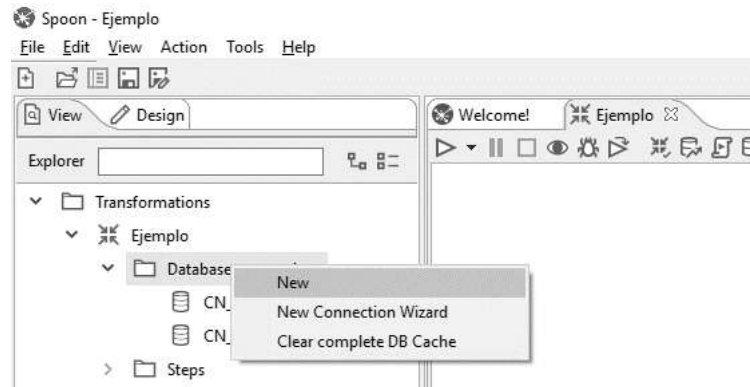
Una vez ingresados los datos, se da click en **Test** y así se verifica que la conexión sea correcta.



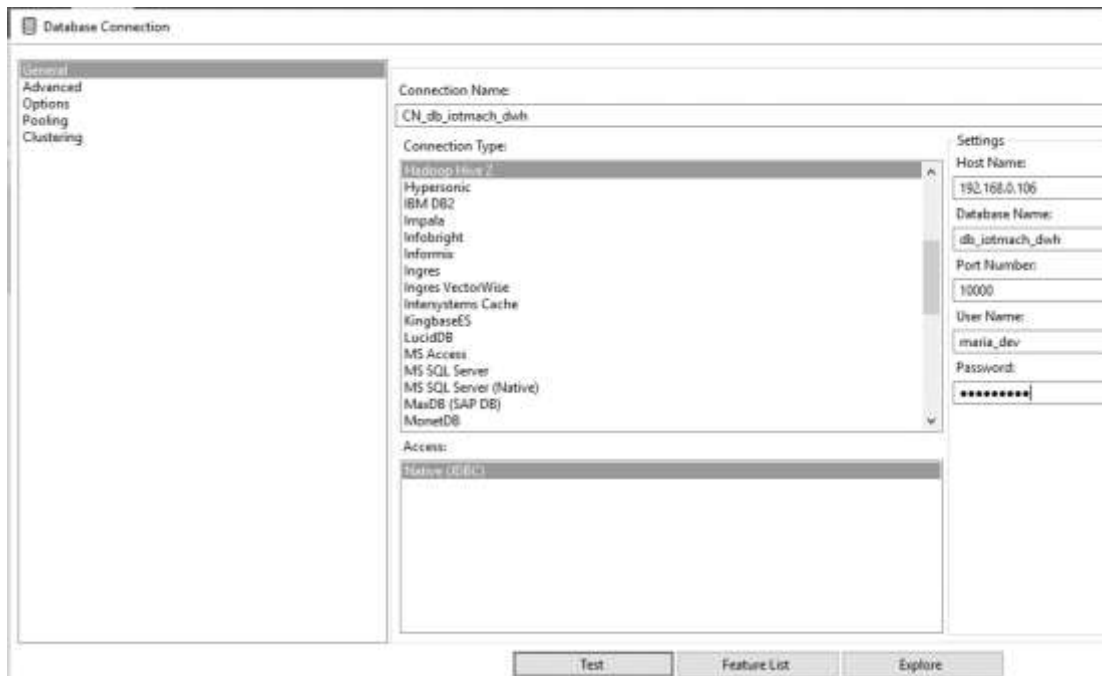
Anexo S

Conexión Spoon Data Integration con Hive

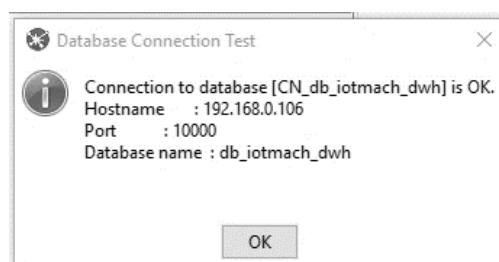
Para agregar una nueva conexión en Spoon, se da click derecho sobre **Databases connections** y luego click en **New**.



Aquí se llenan los datos correspondientes a la base de datos como el ejemplo.



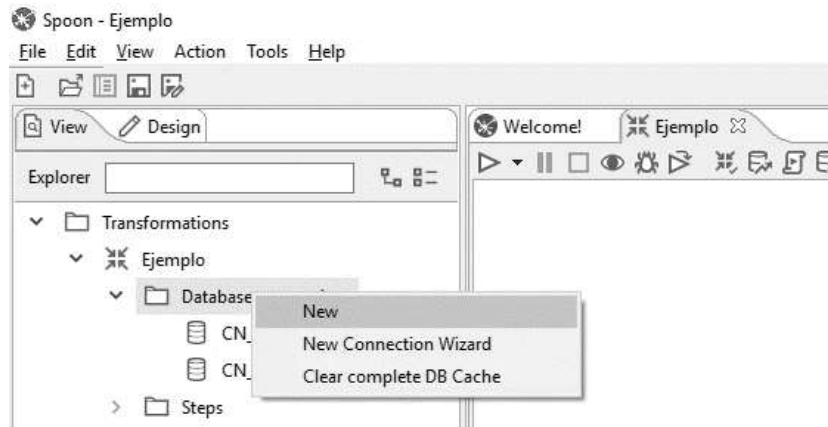
Una vez ingresados los datos, se da click en **Test** y así se verifica que la conexión sea correcta.



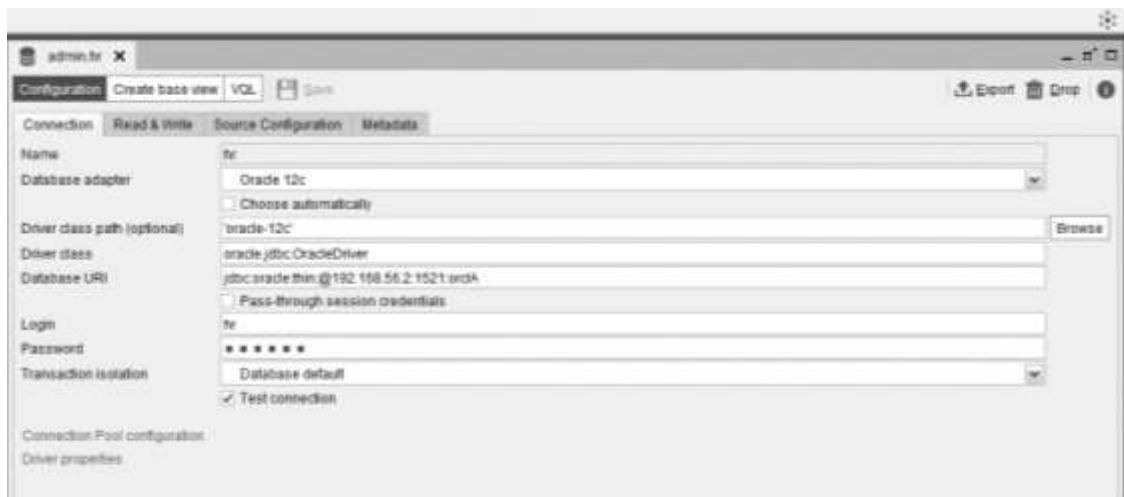
Anexo T

Conexión Spoon Data Integration con Oracle

Para agregar una nueva conexión en Spoon, se da click derecho sobre **Databases connections** y luego click en **New**.



Aquí se llenan los datos correspondientes a la base de datos como el ejemplo:

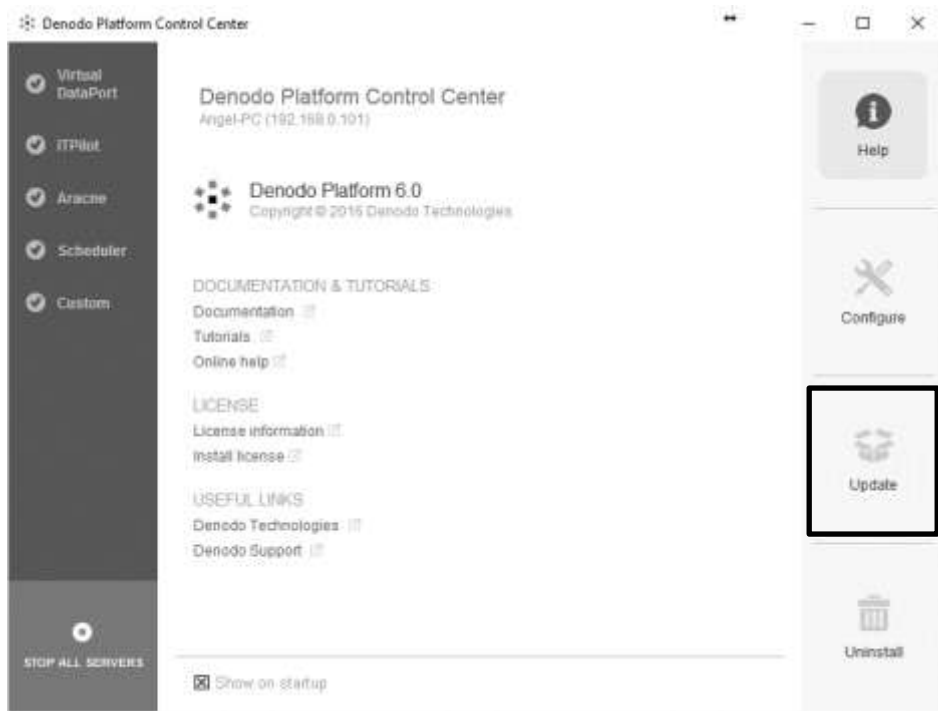


Una vez ingresados los datos, se da click en **Test** y así se verifica que la conexión sea correcta.

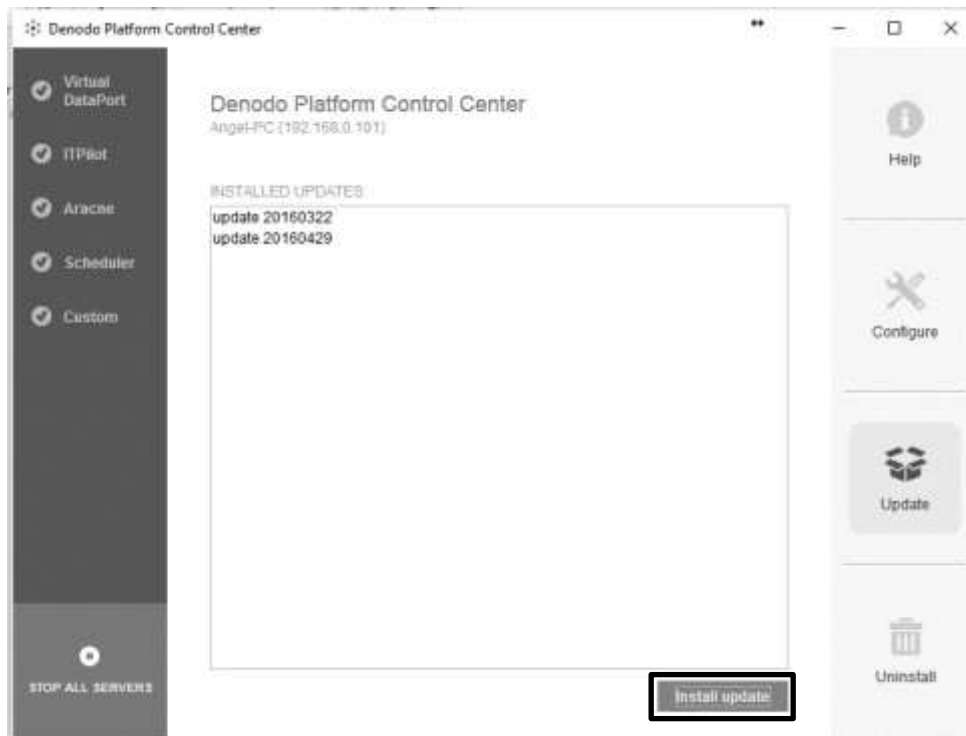
Anexo U

Actualizar Denodo Platform

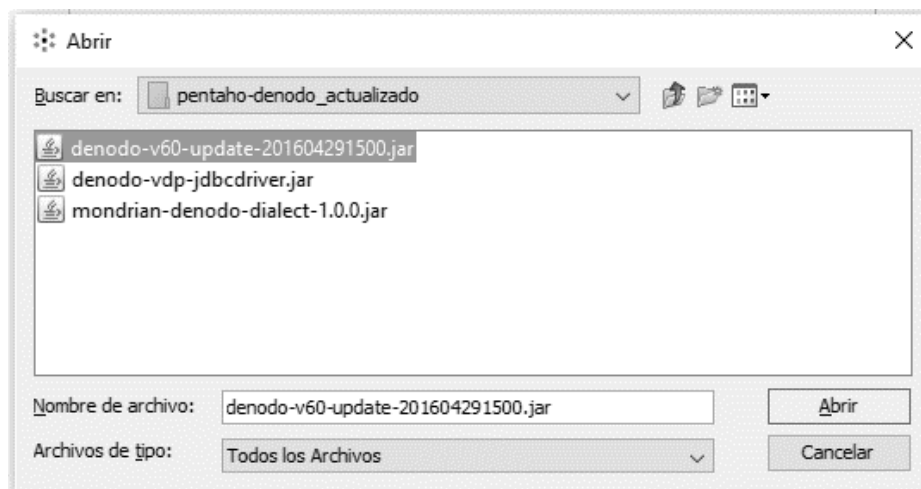
Abrimos Denodo Platform, ejecutándolo como Administrador, seleccionamos **Update**.



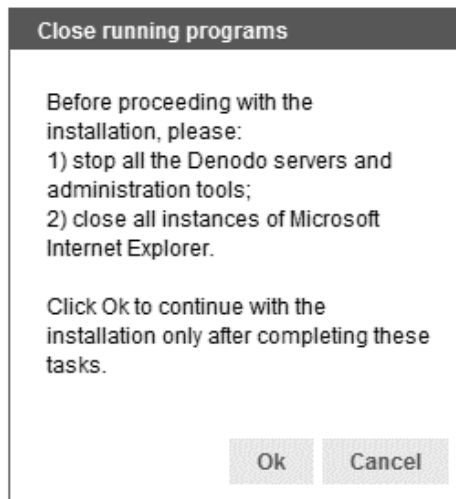
Damos click en **Install Update**, nos aparecerá la ventana para seleccionar la actualización.



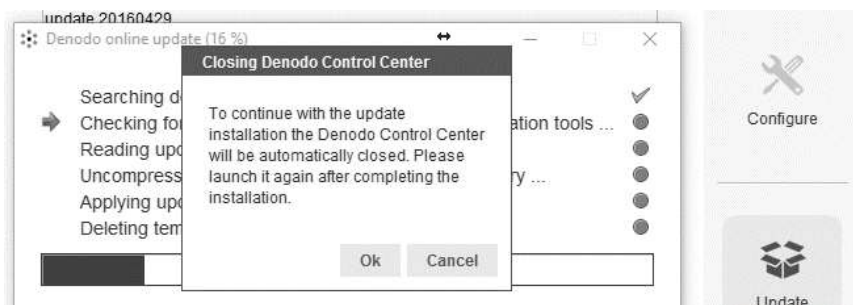
Buscamos en nuestro sistema de archivos, la actualización de Denodo y seleccionamos **Abrir**.



Nos aseguramos que los programas en la lista estén cerrados y damos click en **Ok**.

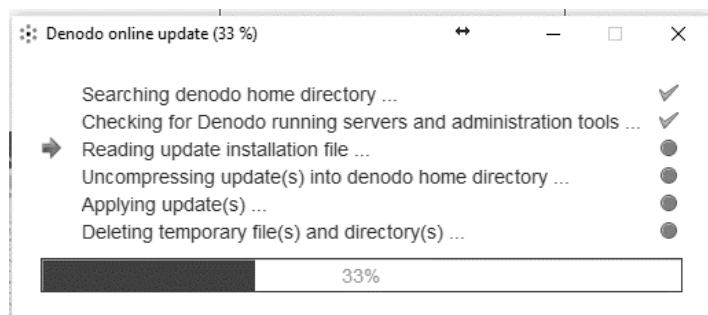


Nos indica que Denodo se cerrará automáticamente para la actualización, click en **Ok**.

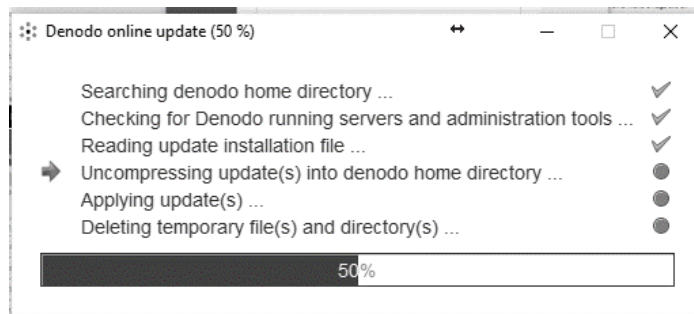


Vemos el proceso de actualización.

Actualización 33%



Actualización 50%



Actualización 1/5

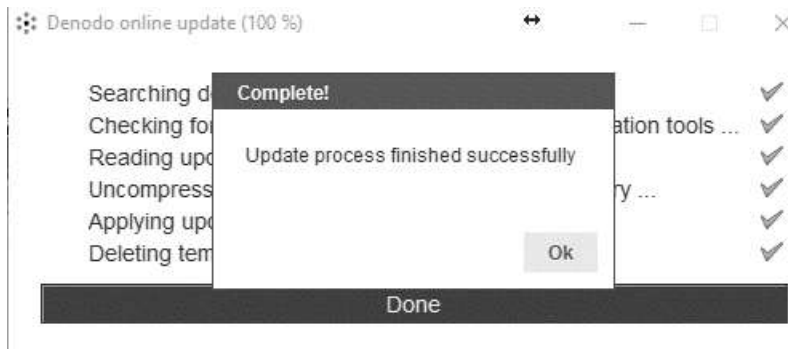


Actualización 5/5



Una vez actualizado, nos muestra el mensaje, y damos click en **Ok**.

Actualización Finalizada

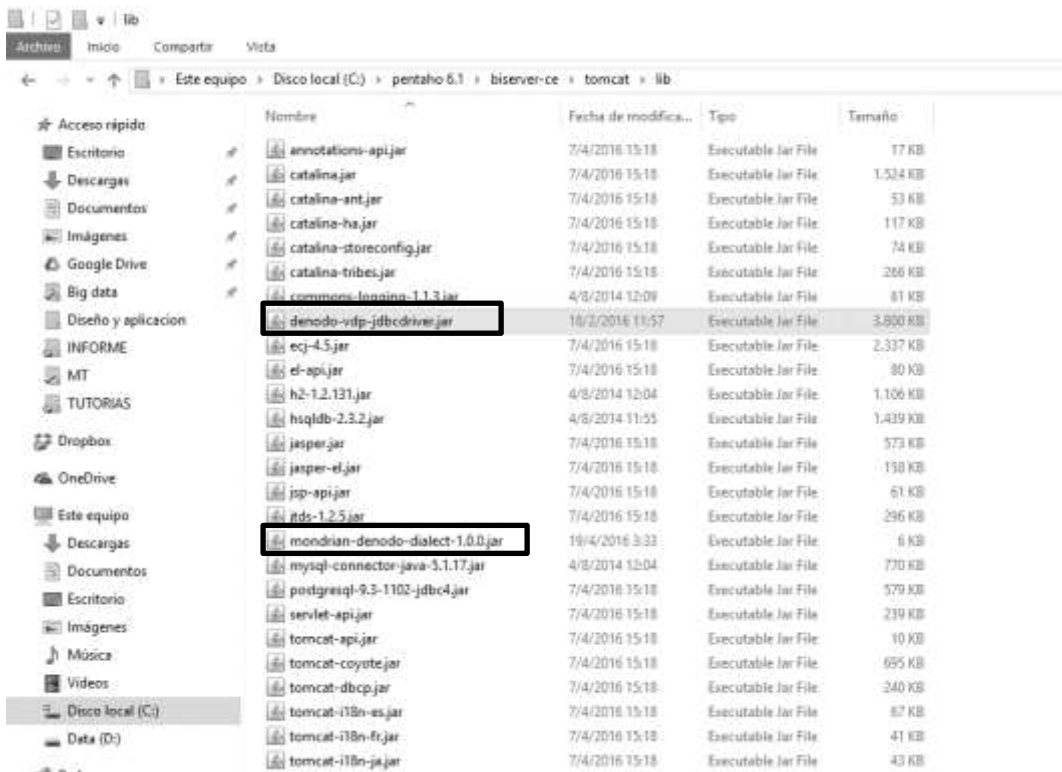


Ubicar dos librerías dentro de la carpeta de Pentaho

Driver para Pentaho

- denodo-vdp-jdbcdriver.jar
- mondrian-denodo-dialect-1.0.0.jar

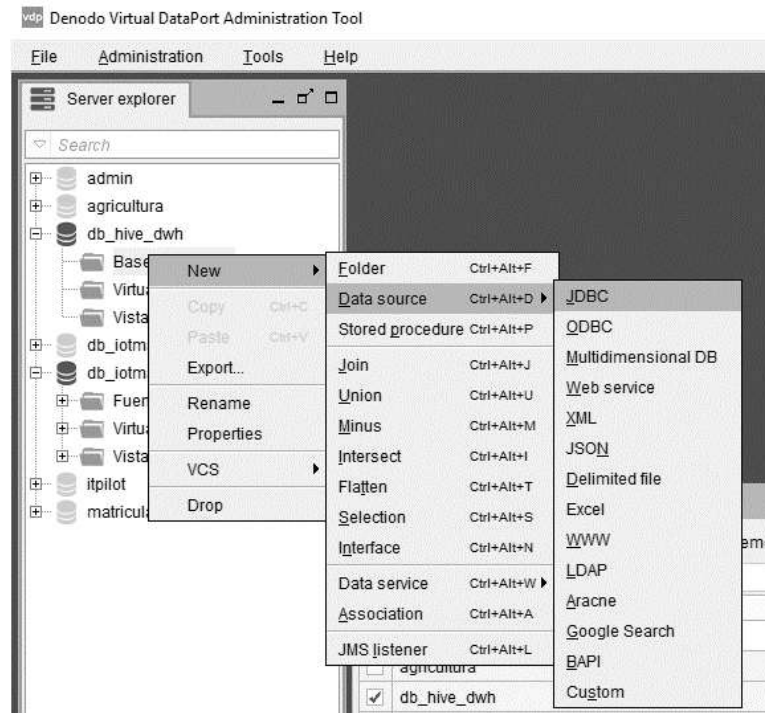
Esto debemos hacerlo donde está ubicado Pentaho, ejemplo: "C:\pentaho 6.1\biserver-ce\tomcat\lib"



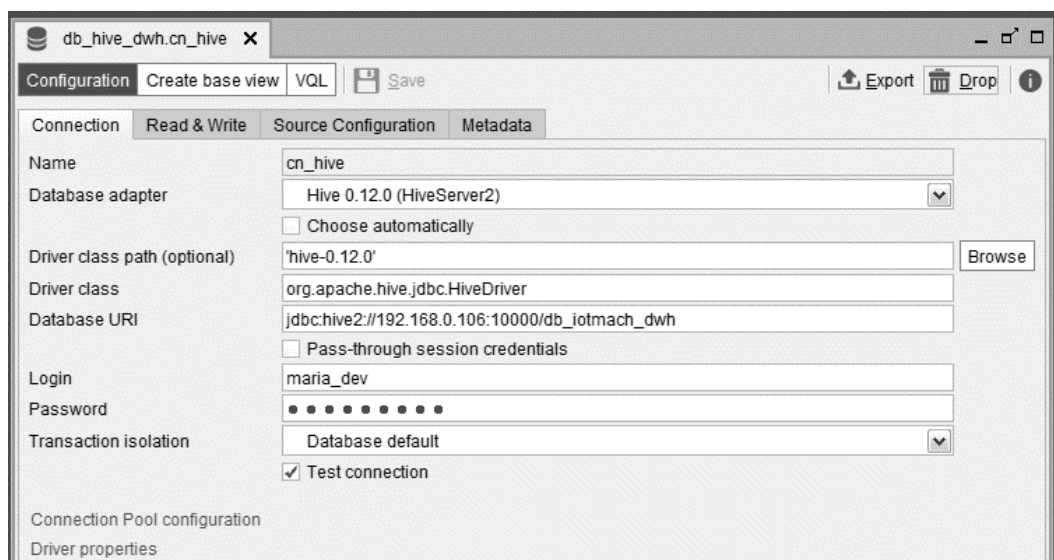
Anexo V

Conexión Denodo con Hive

Para crear la conexión de Denodo con Hive, se hace click derecho en la carpeta donde vamos a ubicar la fuente, clienk en **Data source JDBC** o fuente de datos de la siguiente forma:



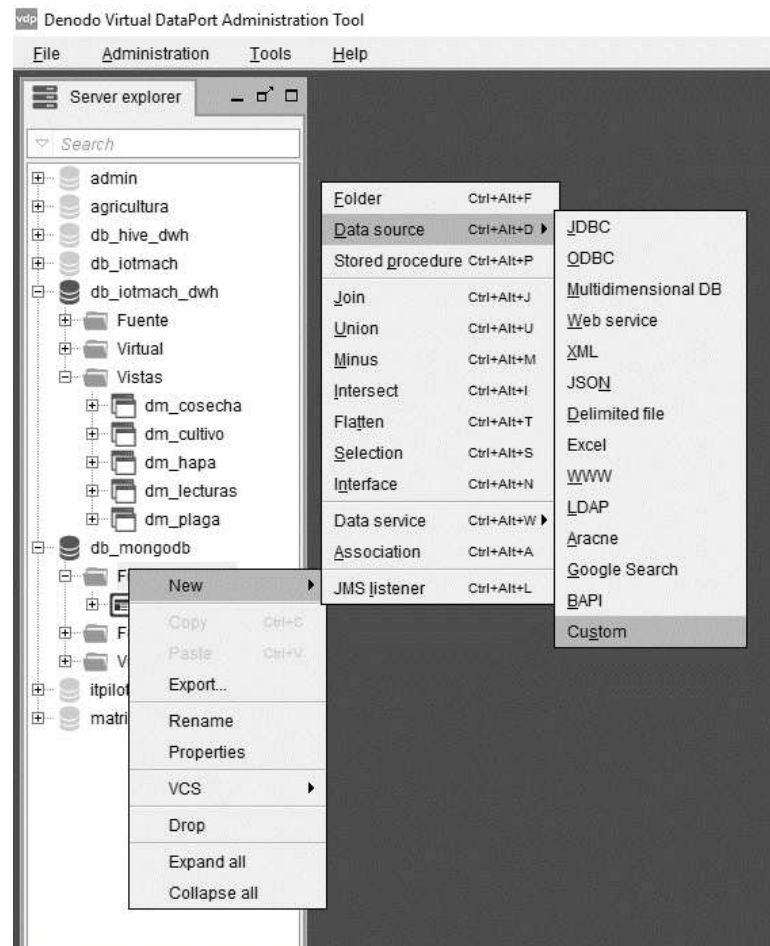
Aquí se llena la información correspondiente a la fuente de datos, que en este caso es la base de datos Hive con la estructura DW creada anteriormente. Y se guarda la conexión con el botón **Save**.



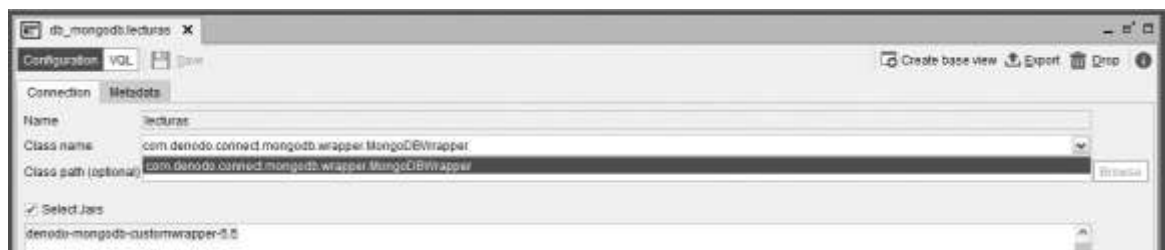
Anexo W

Conexión Denodo con MongoDB

Para crear la conexión de Denodo con MongoDB, se hace click derecho en la carpeta donde vamos a ubicar la fuente, clic en **Data source Custom** o fuente de datos de la siguiente forma:



Se selecciona el **Wrapper** correspondiente para la conexión con **MongoDB**.



Se guarda con la opción **Save**, y luego la opción **Create base view** para continuar.



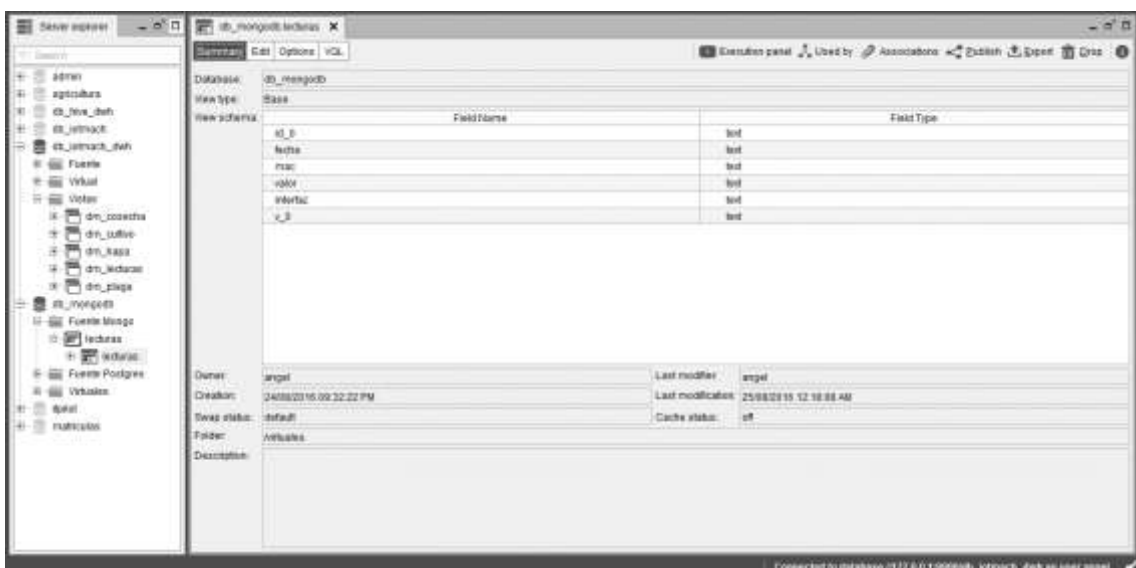
Aquí se llena la información correspondiente a la fuente de datos, que en este caso es la base de datos MongoDB. Y se guarda la conexión con el botón **Ok**.

Edit Wrapper Parameter values

Enter values for the following wrapper parameters:

Host	127.0.0.1
Port	27017
User	
Password	
Database	iotmach
Collection	despaquetados
Connection String	
Fields	
Introspection query	

Luego ponemos visualizar los datos para posteriormente virtualizar y combinar con datos de PostgreSQL.



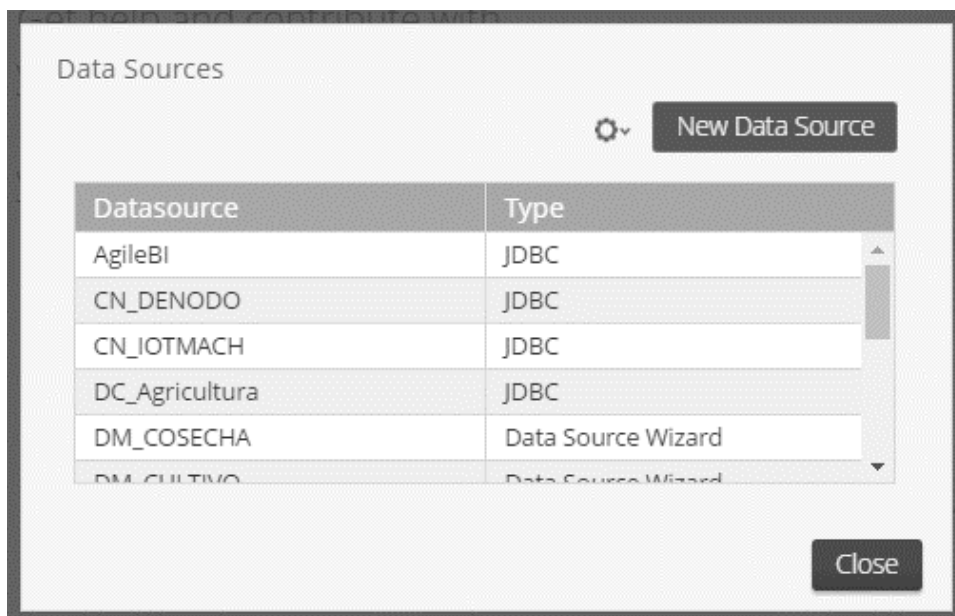
Anexo X

Conexión Pentaho con Denodo

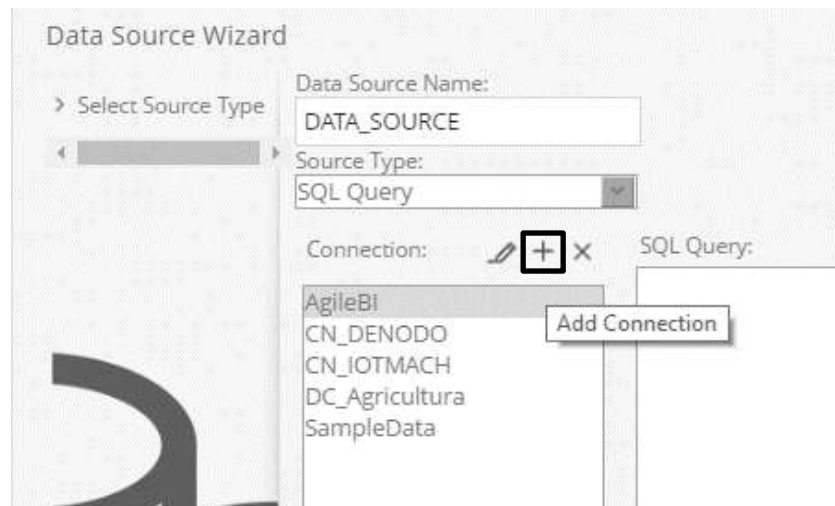
Para crear la conexión de Pentaho con Denodo, se debe seleccionar la opción **Archivo Gestionar fuente de datos**.



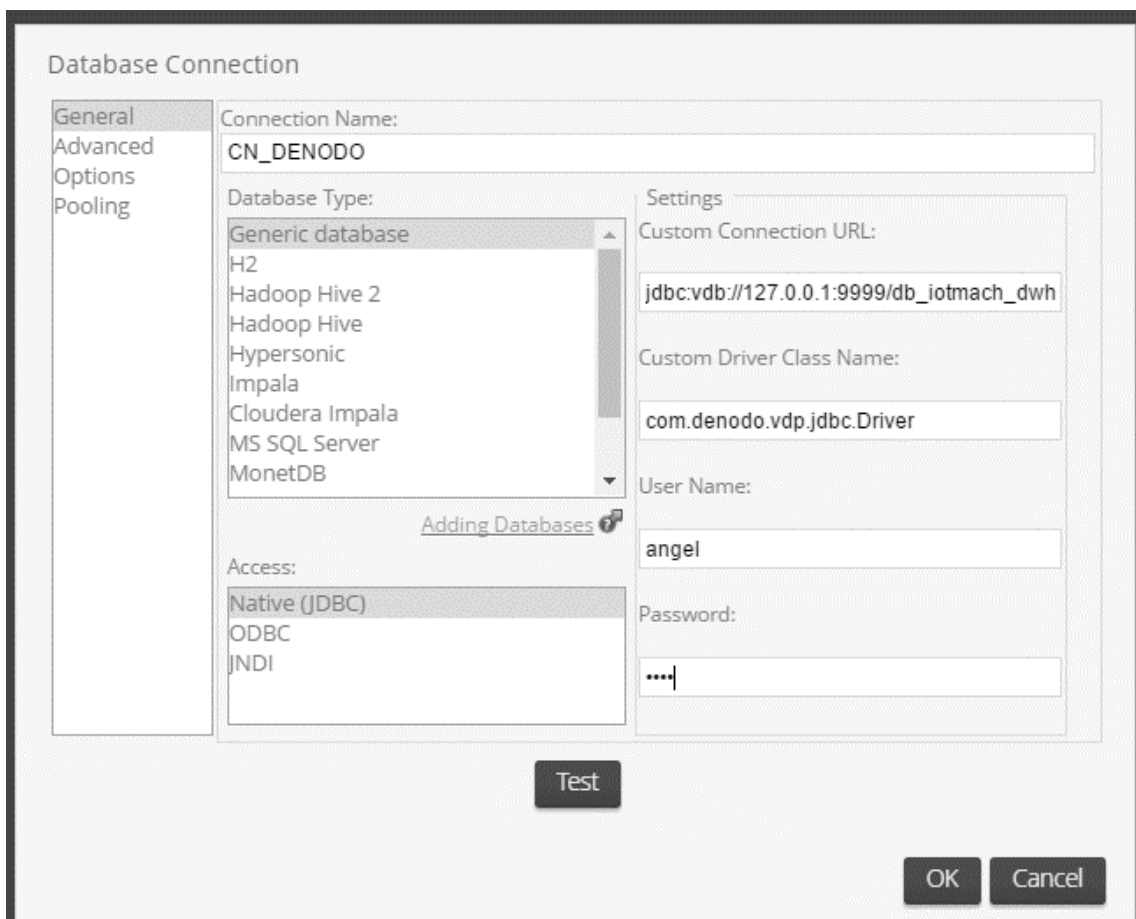
Luego se selecciona **New Data Source** para agregar la conexión.



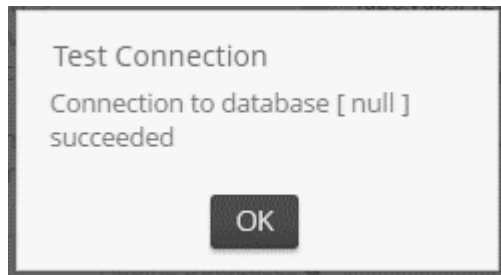
Una vez dentro de **Data Source Wizard**, se crea un nombre para la fuente de datos, luego en **Coennection**, se selecciona el botón + como en el ejemplo:



Y se agregan los datos correspondientes a la conexión con Denodo, seleccionando **Generic database** en **Database Type**. Una vez llenos los datos del ejemplo, se da click en **Test**.



De esta forma se comprueba que la conexión es exitosa.

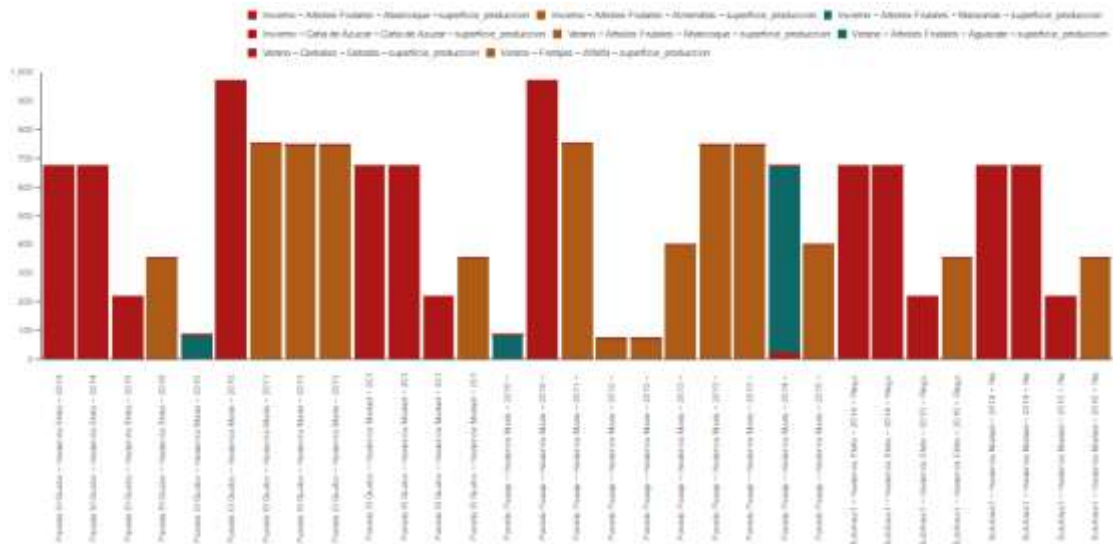


Anexo Y

Pruebas de Reportes mediante herramientas Pentaho

DATA MART 1: CULTIVOS

- 1) ¿Cuál es la superficie de producción (número de hectáreas de cultivo) por especie, categoría de especie, empresa, día, mes, año, semestre, trimestre, temporada (estaciones del año), hacienda, parcela, tipo de suelo, tipo de riego y sector (país, provincia, cantón)?



```

WITH
SET [~COLUMNS_temporada_temporada] AS
{[temporada].[temporada].Members}
SET [~COLUMNS_categoria_categoria] AS
{[categoria].[categoria].Members}
SET [~COLUMNS_especie_especie] AS
{[especie].[especie].Members}
SET [~ROWS_parcela_parcela] AS
{[parcela].[parcela].Members}
SET [~ROWS_hacienda_hacienda] AS
{[hacienda].[hacienda].Members}
SET [~ROWS_anio_anio] AS
{[anio].[anio].Members}
SET [~ROWS_tipo_riego_tipo_riego] AS
{[tipo_riego].[tipo_riego].Members}
SET [~ROWS_tipo_suelo_tipo_suelo] AS
{[tipo_suelo].[tipo_suelo].Members}
SELECT
NON EMPTY CrossJoin(NonEmptyCrossJoin([~COLUMNS_temporada_temporada],
NonEmptyCrossJoin([~COLUMNS_categoria_categoria],
[~COLUMNS_especie_especie])), {[Measures].[superficie_produccion]})
ON COLUMNS,
NON EMPTY NonEmptyCrossJoin([~ROWS_parcela_parcela],
NonEmptyCrossJoin([~ROWS_hacienda_hacienda],
NonEmptyCrossJoin([~ROWS_anio_anio],
NonEmptyCrossJoin([~ROWS_tipo_riego_tipo_riego],
[~ROWS_tipo_suelo_tipo_suelo]))) ON ROWS
FROM [DS_IOTMACH_DWH]

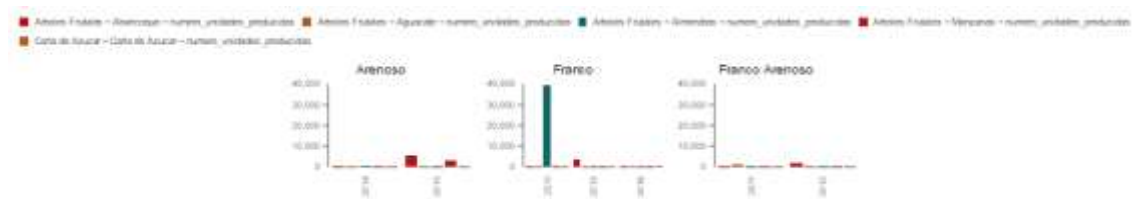
```

DATA MART 2: COSECHA

- 2) ¿Cuántas unidades (toneladas, cajas, quintales, etc.) se producen por especie, categoría de especie, empresa, día, mes, año, semestre, trimestre, temporada (estaciones del año), sector (país, provincia, cantón) y tipo de suelo?

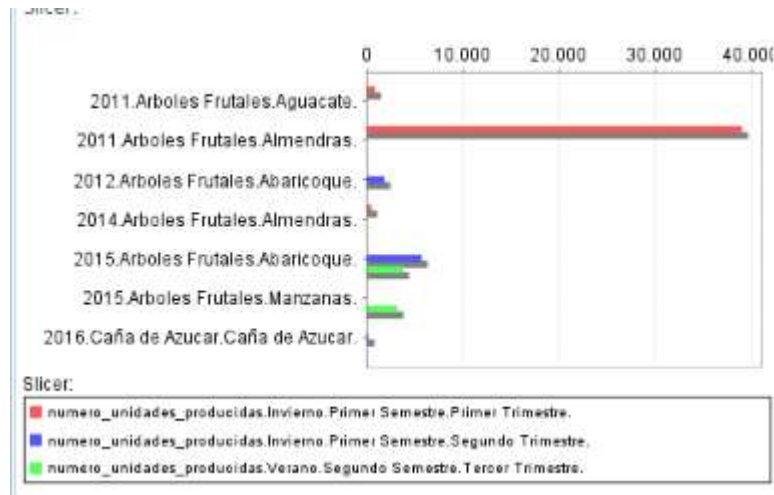
Conocer las unidades de producción de una determinada empresa en un determinado año, en un sector específico y en un tipo de suelos ayuda en la toma de decisiones, sobre todo a la hora de cultivar una determinada especie, y esta consulta busca facilitar esta información a las organizaciones o usuarios.

Como indicador principal se tiene al número de unidades producidas por empresa; y las dimensiones que se deben considerar son especie, categoría especie, empresa, año, semestre, sector, tipo suelo.

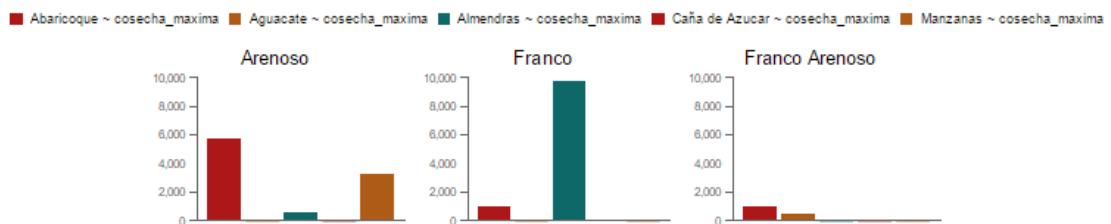


```
WITH
SET [~COLUMNS_categoria_categoria] AS
  {[categoria].[categoria].Members}
SET [~COLUMNS_especie_especie] AS
  {[especie].[especie].Members}
SET [~ROWS_tipo_suelo_tipo_suelo] AS
  {[tipo_suelo].[tipo_suelo].Members}
SET [~ROWS_anio_anio] AS
  {[anio].[anio].Members}
SELECT
NON EMPTY CrossJoin(NonEmptyCrossJoin([~COLUMNS_categoria_categoria],
[~COLUMNS_especie_especie]),
{[Measures].[numero_unidades_producidas]}) ON COLUMNS,
NON EMPTY NonEmptyCrossJoin([~ROWS_tipo_suelo_tipo_suelo],
[~ROWS_anio_anio]) ON ROWS
FROM [DS_DM2]
```

- 3) ¿Cuál es el rendimiento agrícola (unidades producidas o toneladas / superficie de producción), por especie, categoría de especie, empresa, hacienda, parcela, año, semestre, trimestre, temporada (estaciones del año) y sector (país, provincia, cantón)?



4) ¿Cuál es la especie de mayor producción y en qué tipo de suelo?



```

WITH
SET [~COLUMNS] AS
  {[especie].[especie].Members}
SET [~ROWS] AS
  {[tipo_suelo].[tipo_suelo].Members}
SELECT
NON EMPTY CrossJoin([~COLUMNS], {[Measures].[cosecha_maxima]}) ON COLUMNS,
NON EMPTY [~ROWS] ON ROWS
FROM [DS_DM2]

```

5) ¿Cuál es el costo de producción, precio de venta, utilidad neta y utilidad en porcentaje por especie, categoría de especie, empresa, hacienda, parcela, año, semestre, trimestre, temporada (estaciones del año) y sector (país, provincia, cantón)?

■ 2011 ~ Arboles Frutales ~ Aguacate
 ■ 2011 ~ Arboles Frutales ~ Almenoras
 ■ 2012 ~ Arboles Frutales ~ Abaricoque
 ■ 2014 ~ Arboles Frutales ~ Almenoras
■ 2015 ~ Arboles Frutales ~ Abaricoque
 ■ 2015 ~ Arboles Frutales ~ Manzanas
 ■ 2016 ~ Caña de Azucar ~ Caña de Azucar

Abril ~ Primer Semestre ~ Segundo Trimestre ~ costo_produccion



Abril ~ Primer Semestre ~ Segundo Trimestre ~ precio_venta



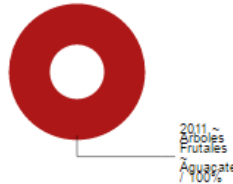
Abril ~ Primer Semestre ~ Segundo Trimestre ~ utilidad_neta



Enero ~ Primer Semestre ~ Primer Trimestre ~ costo_produccion



Enero ~ Primer Semestre ~ Primer Trimestre ~ precio_venta



Enero ~ Primer Semestre ~ Primer Trimestre ~ utilidad_neta



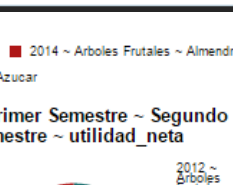
Febrero ~ Primer Semestre ~ Primer Trimestre ~ costo_produccion



Febrero ~ Primer Semestre ~ Primer Trimestre ~ precio_venta

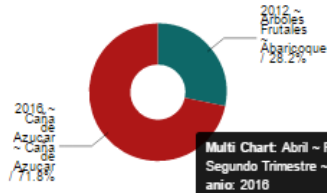


Febrero ~ Primer Semestre ~ Primer Trimestre ~ utilidad_neta



■ 2011 ~ Arboles Frutales ~ Aguacate
 ■ 2011 ~ Arboles Frutales ~ Almenoras
 ■ 2012 ~ Arboles Frutales ~ Abaricoque
 ■ 2014 ~ Arboles Frutales ~ Almenoras
■ 2015 ~ Arboles Frutales ~ Abaricoque
 ■ 2015 ~ Arboles Frutales ~ Manzanas
 ■ 2016 ~ Caña de Azucar ~ Caña de Azucar

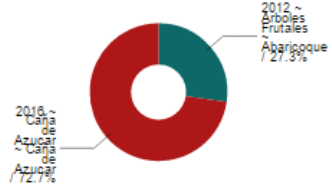
Abril ~ Primer Semestre ~ Segundo Trimestre ~ costo_produccion



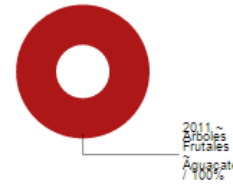
Abril ~ Primer Semestre ~ Segundo Trimestre ~ precio_venta



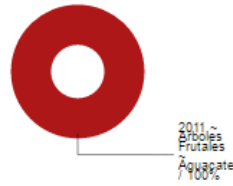
Abril ~ Primer Semestre ~ Segundo Trimestre ~ utilidad_neta



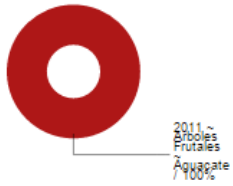
Enero ~ Primer Semestre ~ Primer Trimestre ~ costo_p



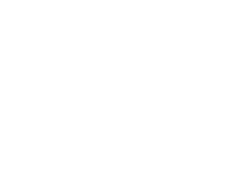
Enero ~ Primer Semestre ~ Primer Trimestre ~ precio_venta



Enero ~ Primer Semestre ~ Primer Trimestre ~ utilidad_neta



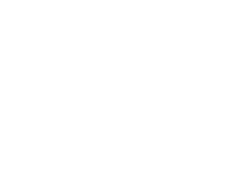
Febrero ~ Primer Semestre ~ Primer Trimestre ~ costo_produccion



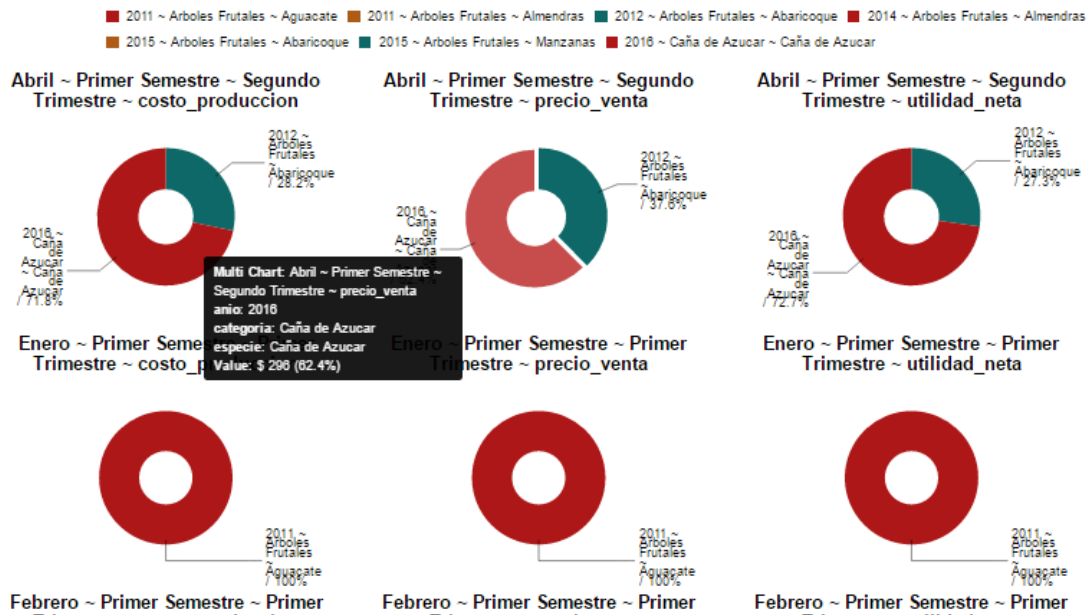
Febrero ~ Primer Semestre ~ Primer Trimestre ~ precio_venta



Febrero ~ Primer Semestre ~ Primer Trimestre ~ utilidad_neta



Multi Chart: Abril ~ Primer Semestre ~ Segundo Trimestre ~ precio_venta
 año: 2016
 categoria: Caña de Azucar
 especie: Caña de Azucar
 Value: \$ 290 (62.4%)



```

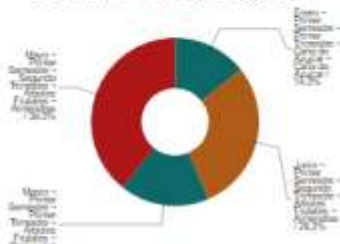
WITH
SET [~COLUMNS] AS
  Hierarchize( {[mes].[mes].Members}, {[mes].[Semestre].Members},
  {[mes].[Trimestre].Members} )
SET [~ROWS_anio_anio] AS
  {[anio].[anio].Members}
SET [~ROWS_categoria_categoria] AS
  Hierarchize( {[categoria].[categoria].Members},
  {[categoria].[especie].Members} )
SELECT
NON EMPTY CrossJoin( [~COLUMNS], {[Measures].[costo_produccion],
[Measures].[precio_venta], [Measures].[utilidad_neta]} ) ON COLUMNS,
NON EMPTY NonEmptyCrossJoin( [~ROWS_anio_anio],
[~ROWS_categoria_categoria] ) ON ROWS
FROM [DS_DM2]
  
```

DATA MART 3: PLAGAS Y ENFERMEDADES

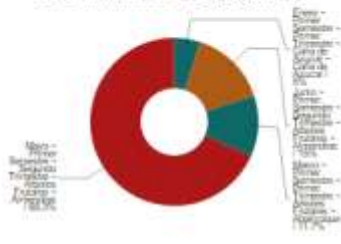
6) ¿Cuántas plantas y hectáreas afectadas por plagas o enfermedades se tienen por especie, categoría de especie, empresa, día, mes, año, semestre, trimestre, temporada (estaciones del año), sector (país, provincia, cantón) y tipo de suelo?

- Agosto - Segundo Semestre - Tercer Trimestre - Arboles Frutales - Aguacate
- Diciembre - Segundo Semestre - Cuarto Trimestre - Arboles Frutales - Albaricoque
- Enero - Primer Semestre - Primer Trimestre - Caña de Azúcar - Caña de Azúcar
- Julio - Segundo Semestre - Tercer Trimestre - Arboles Frutales - Aguacate
- Junio - Primer Semestre - Segundo Trimestre - Arboles Frutales - Almendras
- Mayo - Primer Semestre - Primer Trimestre - Arboles Frutales - Albaricoque
- Mayo - Primer Semestre - Segundo Trimestre - Arboles Frutales - Almendras
- Noviembre - Segundo Semestre - Cuarto Trimestre - Arboles Frutales - Almendras
- Octubre - Segundo Semestre - Cuarto Trimestre - Arboles Frutales - Manzana
- Septiembre - Segundo Semestre - Tercer Trimestre - Arboles Frutales - Albaricoque

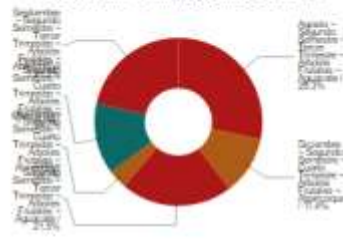
2012 ~ Invierno ~ cantidad_plantas_afectadas



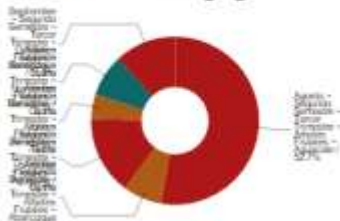
2012 ~ Invierno ~ cantidad_hec_afectadas



2012 ~ Verano ~ cantidad_plantas_afectadas

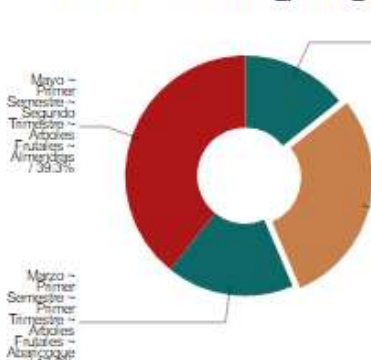


2012 ~ Verano ~ cantidad_hec_afectadas

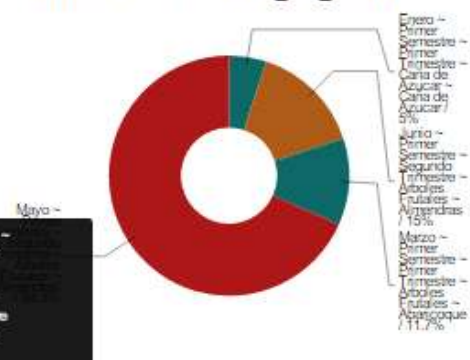


- Agosto - Segundo Semestre - Tercer Trimestre - Arboles Frutales - Aguacate
- Diciembre - Segundo Semestre - Cuarto Trimestre - Arboles Frutales - Albaricoque
- Julio - Segundo Semestre - Tercer Trimestre - Arboles Frutales - Aguacate
- Junio - Primer Semestre - Segundo Trimestre - Arboles Frutales - Almendras
- Mayo - Primer Semestre - Segundo Trimestre - Arboles Frutales - Almendras
- Noviembre - Segundo Semestre - Cuarto Trimestre - Arboles Frutales - Almendras
- Septiembre - Segundo Semestre - Tercer Trimestre - Arboles Frutales - Albaricoque

2012 ~ Invierno ~ cantidad_plantas_afectadas



2012 ~ Invierno ~ cantidad_hec_afectadas



Multi Chart: 2012 ~ Invierno ~ cantidad_plantas_afectadas
 mes: Junio
 semestre: Primer Semestre
 trimestre: Segundo Trimestre
 categoria: Arboles Frutales
 especie: Almendras
 Value: 1748 (29.2%)

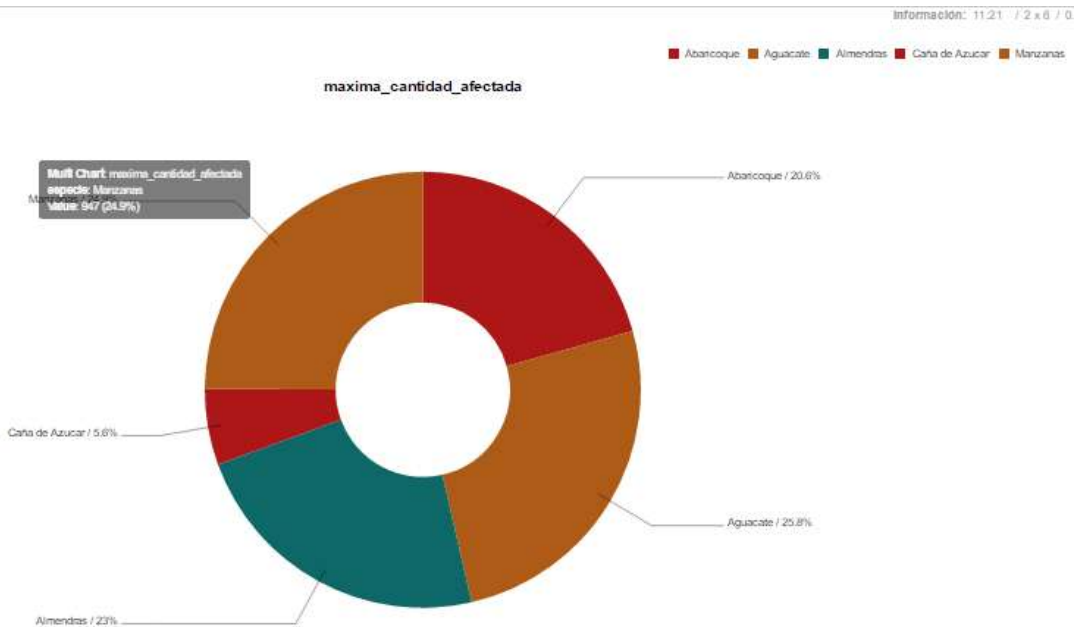
2012 ~ Verano ~ cantidad_hec_afec



WITH

```
SET [~COLUMNS] AS
    Hierarchize({[anio].[anio].Members},
    {[anio].[temporara].Members})
SET [~ROWS_mes_mes] AS
    Hierarchize({[mes].[mes].Members},
    {[mes].[semestre].Members}, {[mes].[trimestre].Members})
SET [~ROWS_categoria_categoria] AS
    Hierarchize({[categoria].[categoria].Members},
    {[categoria].[especie].Members})
SELECT
    NON EMPTY CrossJoin([~COLUMNS],
    {[Measures].[cantidad_plantas_afectadas],
    [Measures].[cantidad_hec_afectadas]}) ON COLUMNS,
    NON EMPTY NonEmptyCrossJoin([~ROWS_mes_mes],
    [~ROWS_categoria_categoria]) ON ROWS
FROM [DS_DM3]
    WHERE CrossJoin({[pais].[pais].Members},
    CrossJoin({[provincia].[provincia].Members},
    {[ciudad].[ciudad].Members}))
```

7) ¿Cuál es la especie de mayor afectación de plagas o enfermedades?

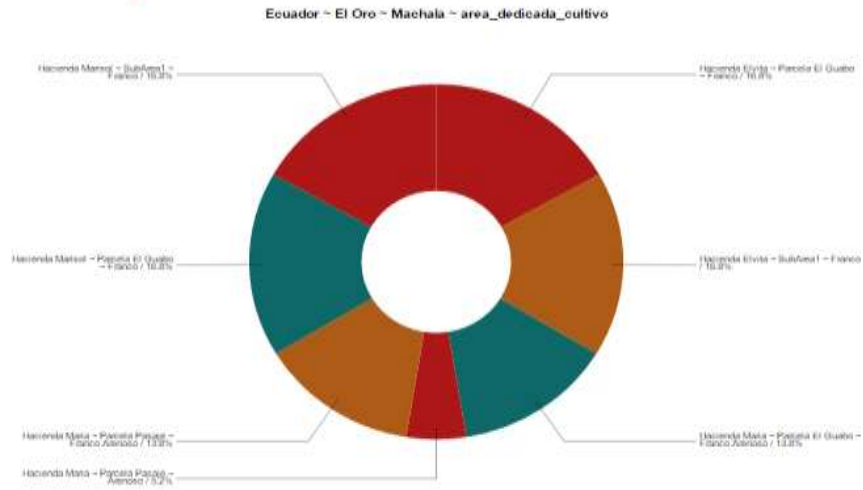


```
WITH
SET [~ROWS] AS
    {[especie].[especie].Members}
SELECT
    NON EMPTY {[Measures].[maxima_cantidad_afectada]} ON COLUMNS,
    NON EMPTY [~ROWS] ON ROWS
FROM [DS_DM3]
```

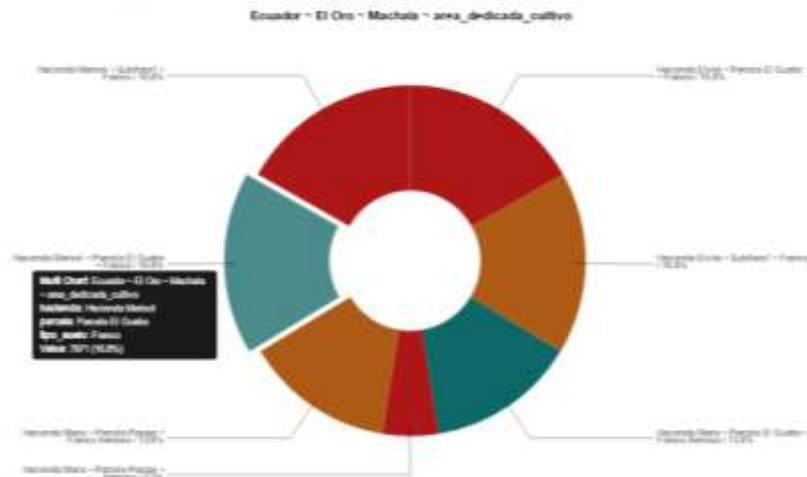
DATA MART 4: HACIENDAS Y PARCELAS

8) ¿Cuál es el área neta (número de hectáreas), que se dedica a cultivos agrícolas, por empresa, hacienda, parcela, sector (país, provincia, cantón), tipo de suelo?

■ Hacienda Elvira - Parcela El Guabro - Fierro ■ Hacienda Elvira - SubAraí - Fierro ■ Hacienda Mata - Parcela El Guabro - Fierro Avenzo ■ Hacienda Mata - Parcela Pajape - Avenzo ■ Hacienda Mata - Parcela Pajape - Fierro Avenzo
 ■ Hacienda Mansal - Parcela El Guabro - Fierro ■ Hacienda Mansal - SubAraí - Fierro



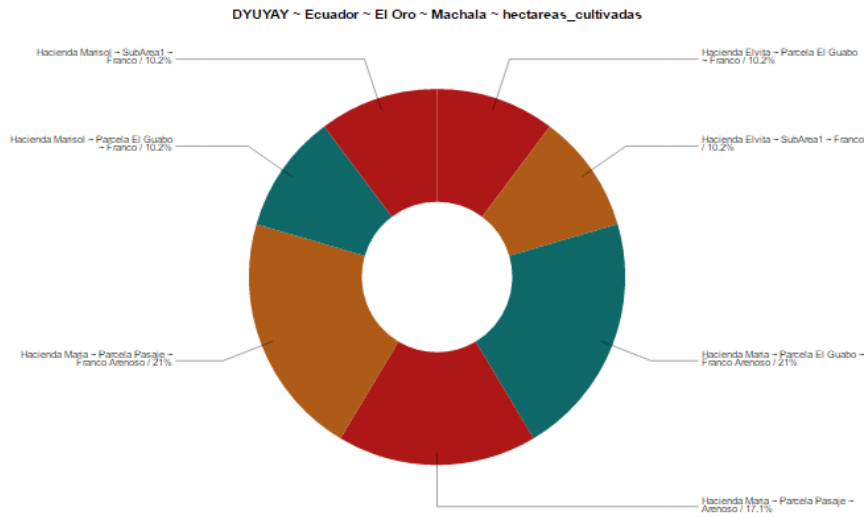
■ Hacienda Elvira - Parcela El Guabro - Fierro ■ Hacienda Elvira - SubAraí - Fierro ■ Hacienda Mata - Parcela El Guabro - Fierro Avenzo ■ Hacienda Mata - Parcela Pajape - Avenzo ■ Hacienda Mata - Parcela Pajape - Fierro Avenzo
 ■ Hacienda Mansal - Parcela El Guabro - Fierro ■ Hacienda Mansal - SubAraí - Fierro



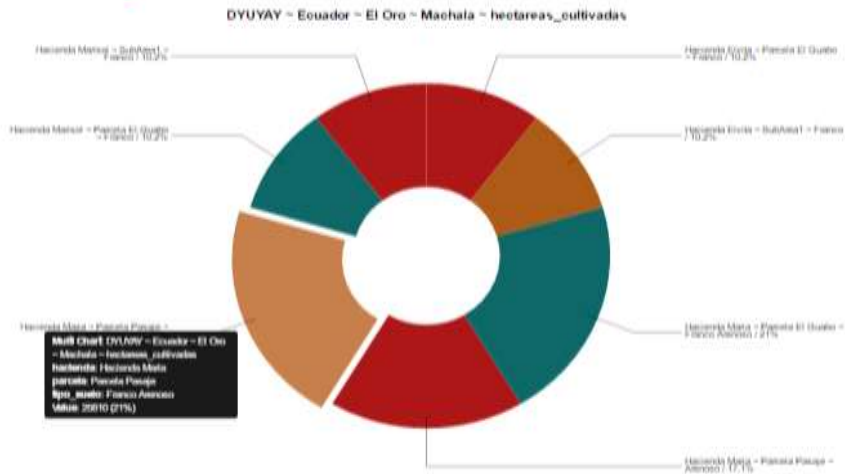
```
WITH
SET [~COLUMNS] AS
  Hierarchize({[pais].[pais].Members},
  {[pais].[provincia].Members}, {[pais].[ciudad].Members})
SET [~ROWS_hacienda_hacienda] AS
  Hierarchize({[hacienda].[hacienda].Members},
  {[hacienda].[parcela].Members})
SET [~ROWS_tipo_suelo_tipo_suelo] AS
  {[tipo_suelo].[tipo_suelo].Members}
SELECT
NON EMPTY CrossJoin([~COLUMNS], {[Measures].[area_dedicada_cultivo]})
ON COLUMNS,
NON EMPTY NonEmptyCrossJoin([~ROWS_hacienda_hacienda],
[~ROWS_tipo_suelo_tipo_suelo]) ON ROWS
FROM [DS_DM4]
```

9) ¿Cuántas hectáreas se cultivan por empresa, hacienda, parcela, sector (país, provincia, cantón) y tipo de suelo?

■ Hacienda Elvita - Parcela El Guabo - Franco ■ Hacienda Elvita - SubArea1 - Franco ■ Hacienda Maria - Parcela El Guabo - Franco Arenoso ■ Hacienda Maria - Parcela Pasaje - Arenoso ■ Hacienda Maria - Parcela Pasaje - Franco Arenoso
 ■ Hacienda Mansol - Parcela El Guabo - Franco ■ Hacienda Mansol - SubArea1 - Franco



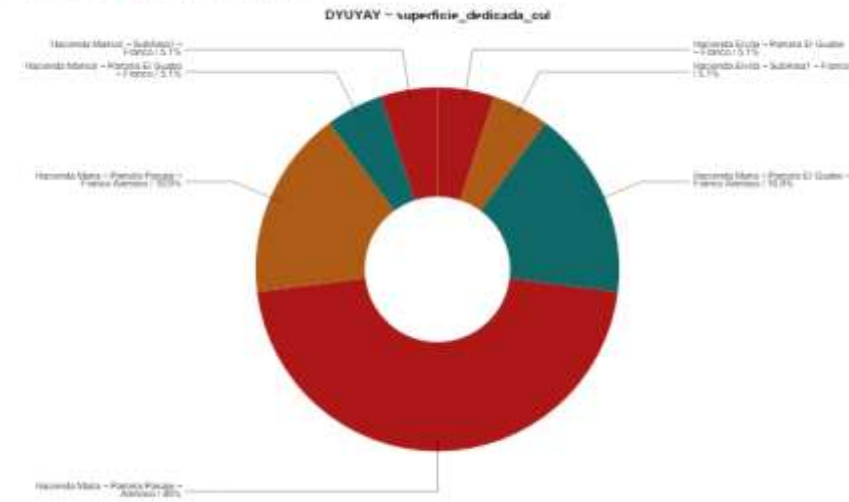
■ Hacienda Elvita - Parcela El Guabo - Franco ■ Hacienda Elvita - SubArea1 - Franco ■ Hacienda Maria - Parcela El Guabo - Franco Arenoso ■ Hacienda Maria - Parcela Pasaje - Arenoso ■ Hacienda Maria - Parcela Pasaje - Franco Arenoso
 ■ Hacienda Mansol - Parcela El Guabo - Franco ■ Hacienda Mansol - SubArea1 - Franco



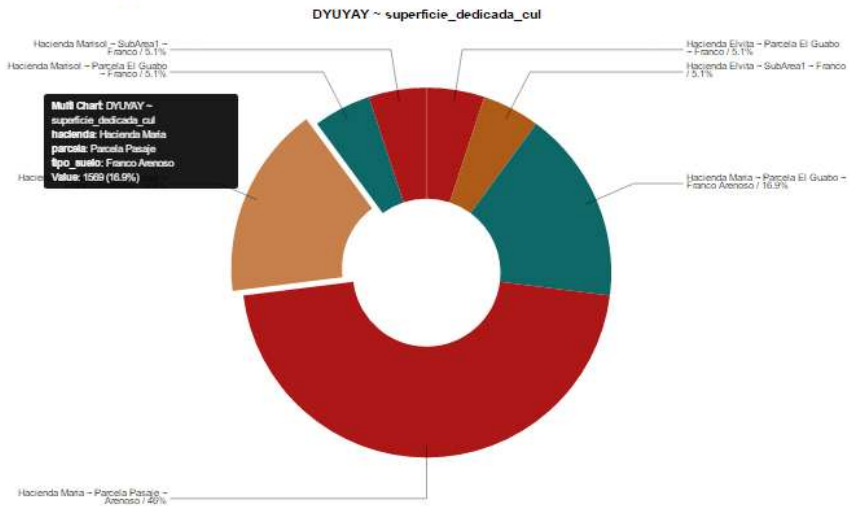
```
WITH
SET [~COLUMNS_empresa_empresa] AS
    {[empresa].[empresa].Members}
SET [~COLUMNS_pais_pais] AS
    Hierarchize({[pais].[pais].Members},
    {[pais].[provincia].Members}, {[pais].[ciudad].Members})
SET [~ROWS_hacienda_hacienda] AS
    Hierarchize({[hacienda].[hacienda].Members},
    {[hacienda].[parcela].Members})
SET [~ROWS_tipo_suelo_tipo_suelo] AS
    {[tipo_suelo].[tipo_suelo].Members}
SELECT
NON EMPTY CrossJoin(NonEmptyCrossJoin([~COLUMNS_empresa_empresa],
[~COLUMNS_pais_pais]), {[Measures].[hectareas_cultivadas]}) ON
COLUMNS,
NON EMPTY NonEmptyCrossJoin([~ROWS_hacienda_hacienda],
[~ROWS_tipo_suelo_tipo_suelo]) ON ROWS
FROM [DS_DM4]
```

10) ¿Cuál es el porcentaje de la superficie dedicada a cultivo por empresa, hacienda, parcela, sector (país, provincia, cantón) y tipo de suelo?

■ Hacienda Elvita - Parcela El Guabo - Franco ■ Hacienda Elvita - SubArea1 - Franco ■ Hacienda Mata - Parcela El Guabo - Franco Arenoso ■ Hacienda Mata - Parcela Pasaje - Arenoso ■ Hacienda Mata - Parcela Pasaje - Franco Arenoso
 ■ Hacienda Marisol - Parcela El Guabo - Franco ■ Hacienda Marisol - SubArea1 - Franco



■ Hacienda Elvita - Parcela El Guabo - Franco ■ Hacienda Elvita - SubArea1 - Franco ■ Hacienda Mata - Parcela El Guabo - Franco Arenoso ■ Hacienda Mata - Parcela Pasaje - Arenoso ■ Hacienda Mata - Parcela Pasaje - Franco Arenoso
 ■ Hacienda Marisol - Parcela El Guabo - Franco ■ Hacienda Marisol - SubArea1 - Franco



```
WITH
SET [~COLUMNS] AS
    {[empresa].[empresa].Members}
SET [~ROWS_hacienda_hacienda] AS
    Hierarchize({[hacienda].[hacienda].Members},
    {[hacienda].[parcela].Members})
SET [~ROWS_tipo_suelo_tipo_suelo] AS
    {[tipo_suelo].[tipo_suelo].Members}
SELECT
NON          EMPTY          CrossJoin([~COLUMNS],
[Measures].[superficie_dedicada_cul]) ON COLUMNS,
NON          EMPTY          NonEmptyCrossJoin([~ROWS_hacienda_hacienda],
[~ROWS_tipo_suelo_tipo_suelo]) ON ROWS
FROM [DS_DM4]
```