

Universidad de las Ciencias Informáticas



Facultad 2

Mercado de Datos para la toma de decisiones en el proceso de enseñanza aprendizaje en la Universidad de las Ciencias Informáticas

**Trabajo de Diploma para optar por el Título de Ingeniero en Ciencias
Informáticas**



Autor:

Armando Paredes Pacheco

Tutor:

Ing. Roanny Lamas López

La Habana, Julio 2016

Año 58 de la Revolución

Declaración de autoría

Declaro ser autor del trabajo de diploma Mercado de Datos para la toma de decisiones en el proceso de enseñanza aprendizaje en la Universidad de las Ciencias Informáticas. Reconozco a la Universidad de las Ciencias Informáticas (UCI), los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los _____ días del mes de julio del año 2016.

Armando Paredes Pacheco

Firma del Autor

Ing. Roanny Lamas López

Firma del Tutor

Agradecimientos

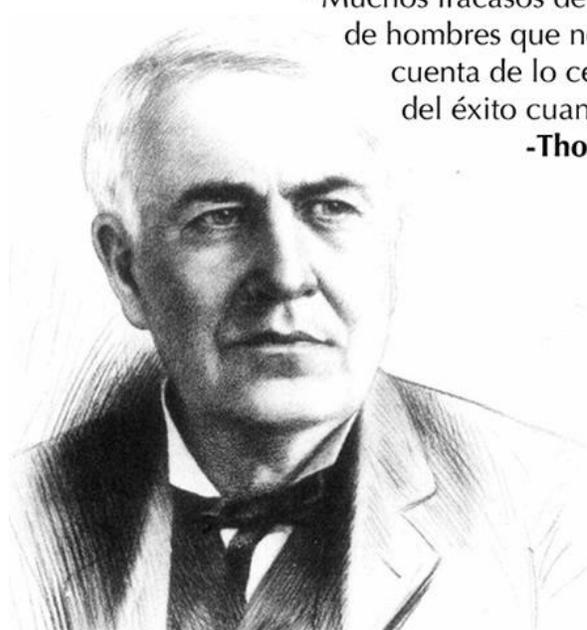
No me perdonaría no agradecer con el corazón a todos los que de cierta forma han puesto un granito de arena para poder lograr esta gran meta que tanto sacrificio, dedicación y trabajo ha costado. Como diría el poeta romano Publio Virgilio Marón:

“Mientras el río corra, los montes hagan sombra y en el cielo haya estrellas, debe durar la memoria del beneficio recibido en la mente del hombre agradecido”.

Quiero agradecer a la hermosa familia que tengo porque de ella he adquirido cualidades tan inquebrantables como mis propios principios. Especiales para mis padres y herma por la especial educación que me dieron, por no perder nunca la paciencia, por impulsarme y ayudarme a entender que sí se podía llegar cuando ya no había esperanzas. Le agradezco a mi novia por haberme acompañado en todos los momentos, me ha ayudado a tomar las mejores decisiones. A mis amigos porque con ellos pase los mejores momentos en la universidad, momentos que nunca se olvidan. A mi amigo incondicional y tutor de este trabajo. Le estaré eternamente agradecido, por dedicar tanto esfuerzo y por invertir tanto de su tiempo para que este trabajo saliera con la mejor y mayor calidad posible. También agradezco a todos los educadores desde mi precolar hasta mi 5to año en la UCI, el profesional que me hice fue también gracias al esfuerzo de todos ustedes, de los cuales aprendí lo bueno y eché a un lado lo malo.

“Muchos fracasos de la vida han sido de hombres que no supieron darse cuenta de lo cerca que estaban del éxito cuando se rindieron”

-Thomas Alva Edison



Dedico este trabajo con todo amor y cariño a las dos personas que lo han sacrificado todo por mí, que han sido mi guía y mi apoyo en todo momento, a ustedes por siempre mis agradecimientos.

Mamá y Papá.

Resumen

El Entorno Virtual de Aprendizaje desarrollado sobre la base del Sistema de Gestión del Aprendizaje MOODLE, en el Centro de Tecnologías para la Formación perteneciente a la Facultad 4 de la Universidad de las Ciencias Informáticas, posibilita la gestión del Proceso de Enseñanza Aprendizaje (PEA) en la institución. En este proceso es crucial contar con información que permita accionar oportunamente sobre las variables que lo afecten, utilizar las oportunidades que ofrece y que son de mayor aceptación y, en general, controlar basados en información, las decisiones que se tomen. El estudio de tendencias, comportamientos en el uso de cursos, recursos y actividades publicados en la plataforma es una de las variantes más utilizadas por profesores y directivos del proceso de formación.

En esta investigación se desarrolla un mercado de datos orientado a los directivos y profesores que gestionan cursos en el Entorno Virtual de Aprendizaje. Se facilita la toma de decisiones al proveer un análisis, con una perspectiva histórica, de los datos almacenados en el período 2013-2015 en el Entorno Virtual de Aprendizaje versión 2.0. Para su desarrollo se emplean metodologías, herramientas y procesos asociados a la construcción de estas aplicaciones. Entre ellas están: el Sistema de Gestión de Bases de Datos PostgreSQL versión 9.4, utilizando el paquete de *software* para inteligencia de negocio Pentaho Open Source Business Intelligence Suite Community Edition en su versión 5.0, para la versión 2.3.4 de MOODLE y, como metodología para guiar el proceso de desarrollo del mercado de datos, fue utilizada HEFESTO en su versión 2.0.

Palabras clave

Inteligencia de negocio, Mercado de datos, MOODLE, sistema de gestión del aprendizaje, toma de decisiones.

Índice de contenido

INTRODUCCIÓN 3

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA DE LA INVESTIGACIÓN..... 9

 1.1 INTELIGENCIA DE NEGOCIO. 9

 1.1.1 Componentes de la inteligencia de negocio. 9

 1.2 ANALÍTICA DE APRENDIZAJE..... 11

 1.3 ALMACENES Y MERCADOS DE DATOS..... 11

 1.3.1 ¿Qué es un almacén de datos? 12

 1.3.2 Ventajas y desventajas de un almacén de datos..... 13

 1.3.3 Mercados de datos..... 14

 1.3.4 Estudio de soluciones similares. 15

 1.3.5 Metodologías para el desarrollo de almacenes de datos..... 16

 1.4 HERRAMIENTAS PARA EL DESARROLLO DE ALMACENES DE DATOS..... 21

 1.4.1 Herramienta de modelado de *software*..... 21

 1.4.2 Sistema Gestor de Bases de Datos..... 24

 1.4.3 Herramienta de Inteligencia de Negocio..... 27

 1.5 CONCLUSIONES 30

CAPÍTULO 2: ANÁLISIS DE LOS REQUERIMIENTOS Y LOS SISTEMAS FUENTES DEL MERCADO DE DATOS..... 32

 2.1 DESCRIPCIÓN DEL NEGOCIO. 32

 2.2 ANÁLISIS DE LOS REQUERIMIENTOS..... 36

 2.2.1 Definición de perspectivas e indicadores..... 37

 2.2.2 Modelo Conceptual. 37

 2.3 DEFINICIÓN DEL ESTADO GENERAL DE LOS SISTEMAS FUENTES..... 38

 2.3.1 Determinación de indicadores..... 38

 2.3.2 Establecimiento de correspondencias..... 39

 2.3.3 Nivel de granularidad. 39

 2.3.4 Modelo Conceptual Ampliado..... 42

 2.4 CONCLUSIONES 43

CAPÍTULO 3: DISEÑO E IMPLEMENTACIÓN DEL MERCADO DE DATOS..... 44

 3.1 MODELO LÓGICO DEL MERCADO DE DATOS. 44

3.1.1	Diseño físico del mercado de datos.....	44
3.1.2	Definición del tipo de modelo lógico del mercado de datos.	45
3.1.3	Identificación de dimensiones y hechos.	46
3.1.4	Realización de uniones entre dimensiones y hechos.	49
3.2	PROCESOS ETL.....	50
3.2.1	Mapeo de datos Fuente-A-Destino.....	51
3.2.2	Carga inicial de los datos y procesos ETL.....	52
3.2.3	Automatización del sistema ETL.	60
3.3	CONCLUSIONES.	61
CAPÍTULO 4: PRUEBAS DE <i>SOFTWARE</i> AL MERCADO DE DATOS.		62
4.1	PRUEBAS UNITARIAS.	62
4.2	PRUEBAS DE INTEGRACIÓN.....	62
4.3	PRUEBAS DE ACEPTACIÓN.	64
4.4	ANÁLISIS DE LOS RESULTADOS DE LAS PRUEBAS.....	68
4.5	CONCLUSIONES.	70
CONCLUSIONES GENERALES		71
RECOMENDACIONES		71
BIBLIOGRAFÍA		72
ANEXOS		75

Introducción

Ciertamente en los últimos años la evolución vertiginosa de las Tecnologías de la Informática y las Comunicaciones (TIC por sus siglas) ha impactado de forma considerable en esferas como el gobierno, la producción, las telecomunicaciones, el deporte, la salud, entre otras, al proveer las herramientas, tecnologías y metodologías que facilitan, agilizan y optimizan su gestión. Precisamente en la educación las TIC dieron un giro radical a las formas y métodos clásicos de educar, introduciendo el llamado E-Learning¹.

Este término es definido por Francisco José García Peñalvo, profesor de la Universidad de Salamanca², como la *“capacitación no presencial que, a través de plataformas tecnológicas, posibilita y flexibiliza el acceso y el tiempo en el proceso de enseñanza-aprendizaje, adecuándolos a las habilidades, necesidades y disponibilidades de cada discente, además de garantizar ambientes de aprendizaje colaborativos mediante el uso de herramientas de comunicación síncrona y asíncrona, potenciando en suma el proceso de gestión basado en competencias”*. (Peñalvo 2005) En Cuba no existe el ambiente ideal para emplear E-Learning debido al grado de informatización de la sociedad. Sin embargo, en la Universidad de las Ciencias Informáticas³ (UCI, por sus siglas en español) dada la convergencia de gran cantidad de tecnología, esta se convierte en un ambiente ideal para aplicar este método.

La UCI, surgida el 23 de septiembre de 2002 con la visión de futuro del Comandante en Jefe Fidel Castro Ruz, es actualmente una de las universidades más jóvenes del país. Creada bajo la concepción de ser un centro de nuevo tipo, de alcance nacional, con tareas concretas en el proyecto de informatización de la sociedad cubana y énfasis en el desarrollo de la industria del *software*. Esta institución de la Educación Superior cubana constituye un centro que forma a sus educandos desde una destacada actividad política, cultural y deportiva, objetivos que se han venido cumpliendo mediante la interrelación de la formación-investigación-producción-extensión universitaria como un proceso único y coherente.

Estas características hacen de la UCI una universidad atípica lo que, unido a la constante evolución de la rama de la Informática, la obligan a hacer uso de las últimas tendencias en los procesos formativos.

¹ En español Aprendizaje Electrónico, aunque suelen utilizarse términos como Aprendizaje en Red, Tele-Formación, Aprendizaje Virtual, Educación Virtual, Formación en Línea y Enseñanza Virtual. (Clark and Mayer 2016)

² Página oficial: <http://www.usal.es/webusal/>.

³ Página oficial: <http://www.uci.cu/>.

Es por ello que el Centro de Tecnologías para la Formación (FORTES) perteneciente a la Facultad 4 desarrolla el Entorno Virtual de Aprendizaje (EVA, por sus siglas en español), el EVA es una plataforma E-Learning construida sobre la base del Sistema de Gestión del Aprendizaje (LMS por sus siglas del inglés: Learning Management System) MOODLE⁴. A pesar de que el Proceso de Enseñanza-Aprendizaje (PEA por sus siglas en español) de la UCI no comprende la modalidad a distancia, se hace uso de esta herramienta con el fin de apoyar la actividad de los profesores y estudiantes.

El EVA es un espacio de apoyo al proceso de formación de la carrera de Ingeniería en Ciencias Informáticas, donde los profesores pueden implementar estrategias de Enseñanza-Aprendizaje complementarias a las clases presenciales, disponiendo los estudiantes de un poderoso medio en el cual pueden obtener, utilizar o compartir recursos educativos. En él se encuentran publicados los cursos de la mayoría de las asignaturas del plan de estudio de la carrera. Como sistema informático almacena constantemente información en su base de datos correspondiente a la interacción de los usuarios con los cursos, actividades y recursos que hayan sido creados. Estos volúmenes de datos pueden ser analizados con el objetivo de tomar decisiones que mejoren el uso del EVA como componente del PEA de la universidad. El área del conocimiento encargada de realizar este tipo de análisis se denomina Inteligencia de Negocio⁵ (IN, por sus siglas en español).

La IN se define como el conjunto de estrategias y herramientas enfocadas a la administración y creación de conocimientos mediante el análisis de datos existentes en una organización o empresa. Este conjunto de herramientas y metodologías tienen en común las siguientes características: accesibilidad a la información, apoyo en la toma de decisiones y orientación al usuario final. Se trata, por tanto, del proceso de análisis de datos de la empresa para poder extraer conocimiento de ellos. (Azua 2007)

En consonancia con lo antes planteado, la aplicación de IN en el área del aprendizaje con el fin de tener un impacto directo sobre los estudiantes, profesores y el proceso de aprendizaje es denominada Analítica de Aprendizaje⁶ (AA, por sus siglas en español). (Gómez-Aguilar et al 2014) Consiste en la medición, recopilación, análisis y presentación de datos sobre los estudiantes, sus contextos y las interacciones que allí se generan, con el fin de comprender el proceso de aprendizaje que se está desarrollando y optimizar los entornos en los que se produce. (García Alcázar 2015) Es obvio que los responsables del PEA, tanto en la enseñanza presencial como en la asistida, requieran de su

⁴ Del inglés: Module Object-Oriented Dynamic Learning Environment. En español: Entorno Modular de Aprendizaje Dinámico Orientado a Objetos. Página oficial: <https://moodle.org/?lang=es>.

⁵ En inglés: Business Intelligence. Comúnmente es utilizada la sigla BI para referirse a este concepto.

⁶ En inglés: Learning Analytics.

retroalimentación para conformar sus cursos, actualizarlos, volverlos atractivos al estudiante y lograr el propósito de instruir y transformar.

Actualmente en la UCI la capacidad de analizar la información gestionada por el EVA se limita a la permitida a través de una serie de reportes existentes en el LMS MOODLE. Dichos reportes se pueden visualizar por curso, persona, fecha y actividad. Según el Manual del profesor de MOODLE, estos son: (A. S. Martínez 2012)

- Informes de actividad del curso: muestran una visión general del número de accesos a los distintos componentes (recursos y actividades) que integran el curso. Queda registrada la fecha del último acceso a cada elemento y permite detectar si hay elementos que no se visitan. Sin embargo, si se necesita saber los datos específicos de algún usuario, con este informe no es posible, se debe acudir al informe de participación en el curso.
- Informes de participación en el curso: permiten filtrar por usuarios y actividades específicas. Posibilitan conocer si el alumno solo ha visto la actividad o si ha participado en ella.
- Informes de actividad del alumno: como su nombre lo indica, informa sobre cuál ha sido la actividad del estudiante en el día actual.
- Informe de la actividad general en todo el curso: informa sobre la actividad del estudiante en todo el curso publicado. Ejemplos de información que se muestran son: participación en foros, recursos vistos, tareas presentadas y cuestionarios realizados con su calificación correspondiente.
- Informes de finalización de actividad: recogen los datos del grado de finalización de una actividad en base a los parámetros que el profesor haya definido cuando incorporó los distintos recursos y actividades a un curso. Es capaz de mostrar el grado de completitud de cada estudiante con la actividad finalizada. Los registros permiten mostrar con detalle todas las actividades de alumnos y profesores dentro del curso. Los profesores pueden conocer a qué páginas han accedido sus estudiantes, las actividades a las que entraron y en las que participaron, desde qué direcciones ip se conectaron y en qué fecha y hora lo hicieron.
- Informe de registros activos: da a conocer la información mencionada para los usuarios que se encuentran conectados.

Estos reportes no posibilitan hacer un análisis de los datos variable en el tiempo ya que no están diseñados para ello, solo se limitan a análisis dentro del tiempo de duración del curso. Por ejemplo: no se conoce la cantidad de acciones realizadas por estudiantes y profesores en el EVA en los últimos cinco años como para medir el comportamiento del uso de la plataforma o qué recurso se ha utilizado

más en asignaturas de la especialidad. Además, debido a que la información se encuentra distribuida en varios reportes no permiten realizar análisis que contemplen a varias variables de conjunto. Tal es el caso de la imposibilidad de analizar la interacción de los usuarios y profesores desde las distintas zonas de la universidad con los recursos de un curso disponible en el EVA. Esta actividad se ve afectada también por el cambio de una versión a otra del LMS MOODLE puesto que la información de la versión antigua solo se guarda una vez para evitar pérdidas, pero, estas salvas, no se encuentra integrada en una única fuente de datos para su explotación.

La situación descrita imposibilita que los decisores fundamentales del PEA de la UCI, dígase profesores y directivos vinculados a la formación académica, se nutran de la información que se puede extraer de los datos generados por la interacción de los usuarios con el EVA. Se dificulta así el estudio de tendencias y comportamientos en el uso de cursos, recursos y actividades publicados en la plataforma. Esto trae como consecuencias que se continúen diseñando y construyendo cursos sin considerar al cliente principal: el estudiante, su forma de aprender, o los métodos y recursos de estudios más empleados para determinado contenido.

Teniendo en cuenta la situación problemática descrita se formula como **problema a resolver**: Se dificulta el análisis de la información gestionada por el EVA imposibilitando la toma de decisiones de los decisores del PEA de la UCI.

Se plantea como **objeto de estudio**: análisis de información en el EVA.

El **objetivo general** de este trabajo es: Desarrollar un mercado de datos para contribuir a la toma de decisiones basadas en el análisis de la información almacenada en el Entorno Virtual de Aprendizaje de la Universidad de las Ciencias Informáticas.

Determinando para la investigación como **campo de acción**: Mercados de datos para el EVA.

Tareas de investigación:

1. Revisión bibliográfica para conformar el estado del arte de la investigación.
2. Selección de las tecnologías, metodología y herramientas a utilizar en el proceso de desarrollo a partir de la revisión de las más utilizadas en la construcción de mercados de datos.
3. Identificar los requisitos del mercado de datos que necesita el EVA.
4. Diseño del mercado de datos que guía la construcción de la solución.
5. Validación mediante la aplicación de pruebas de *software* a la solución propuesta.

Preguntas científicas:

1. ¿Cuáles son las tecnologías, metodologías y herramientas que más se ajustan al desarrollo de la propuesta de solución?
2. ¿Cómo contribuye el desarrollo de un mercado de datos al proceso de toma de decisiones con una perspectiva histórica de los datos?
3. ¿Cómo asegurar la correcta implementación de la propuesta de solución?

Posibles resultados:

1. Documentación asociada que contiene la fundamentación teórica de la investigación y los artefactos generados en cada una de las fases de la metodología de desarrollo seleccionada.
2. Un mercado de datos que contribuya a la toma de decisiones a partir de la información gestionada por el Entorno Virtual de Aprendizaje de la Universidad de las Ciencias Informáticas.
3. Un entorno operable donde el usuario final pueda utilizar el mercado de datos en función de la toma de decisiones para el proceso de enseñanza aprendizaje.

Los **métodos científicos** a utilizar en la investigación estuvieron determinados por el objetivo general y las tareas de investigación previstas.

A nivel teórico serán utilizados los métodos:

Analítico–sintético: para realizar un estudio bibliográfico de la teoría existente alrededor del objeto de estudio, determinar las características que tendrá la propuesta de solución y definir las tecnologías y herramientas a utilizar para el desarrollo de la propuesta de solución.

Inductivo-deductivo: para analizar una muestra representativa de trabajos asociados al desarrollo de mercados de datos para el LMS MOODLE, encontrar las particularidades y elementos en común entre ellos, para posteriormente extrapolar este conocimiento a una propuesta que permita dar solución a la problemática planteada.

Histórico-lógico: para elaborar la fundamentación teórica de la investigación, teniendo en cuenta que se realizó un estudio de lo más relevante en el plano teórico acerca del surgimiento de los mercados de datos.

Modelación: para representar la solución a través de esquemas y diagramas que apoyen la documentación del sistema.

También serán utilizados a nivel empírico los métodos:

Entrevista: para identificar las necesidades de información de los especialistas del Centro de Innovación y Calidad de la Educación y el Centro Nacional de Educación a Distancia.

Observación: para estudiar la interacción de los estudiantes y profesores con el Entorno Virtual de Aprendizaje de la Universidad de las Ciencias Informáticas y observar el flujo de esta interacción dentro del LMS MOODLE.

Para una mejor comprensión de la investigación, se decidió definir una estructura capitular que aporte el grado de organización necesario y facilite la lectura del documento. Los capítulos que lo conforman son los siguientes:

Capítulo 1: Fundamentación teórica de la investigación.

Se exponen los elementos teóricos que sustentan la investigación. Se realiza un estudio de las soluciones similares y se analizan las metodologías, tecnologías y herramientas que se ajustan al desarrollo de la investigación, justificando la selección y utilización de cada una de ellas.

Capítulo 2: Análisis de los requerimientos y los sistemas fuentes del mercado de datos.

Se muestran los artefactos correspondientes a las dos primeras fases de la metodología de desarrollo seleccionada. Se identifican las necesidades de información por parte del cliente, los indicadores y las perspectivas de análisis. Se construyen los modelos conceptuales y los modelos conceptuales ampliados, dos artefactos importantes para los posteriores procesos de diseño e implementación.

Capítulo 3: Diseño e implementación del mercado de datos.

Se muestran los artefactos correspondientes a las dos últimas fases de la metodología de desarrollo seleccionada. Se describe el proceso de diseño e implementación del mercado de datos, obteniéndose el modelo lógico, las tablas de hechos y dimensiones, y las uniones entre estas. Se explican los procesos de Extracción Transformación y Carga, para llevar la información desde la base de datos del Entorno Virtual de Aprendizaje de la Universidad de las Ciencias Informáticas hasta la tabla de hechos del mercado de datos.

Capítulo 4: Pruebas de *software* al mercado de datos.

Se describe el proceso de pruebas realizadas a la propuesta de solución, con el fin de verificar la calidad de los datos almacenados y cumplir las expectativas del cliente. Se documentaron los resultados a cada una de las pruebas realizadas.

Capítulo 1: Fundamentación teórica de la investigación.

Este capítulo está dirigido a plantear los elementos teóricos que sustentan la investigación. Se exponen y relacionan los conceptos que desde el punto de vista teórico permiten un mejor entendimiento de lo planteado en la situación problemática. Se llevará a cabo el estudio de sistemas y soluciones similares que utilizan los almacenes de datos como herramienta para contribuir al proceso de toma de decisiones. También se estudiarán las tecnologías y las herramientas que se emplearán para dar cumplimiento al objetivo general de la investigación, así como las metodologías utilizadas para el desarrollo de almacenes de datos, seleccionando la que más se adecúe a las características del mercado de datos que se desea implementar.

1.1 Inteligencia de negocio.

El término IN caracteriza una amplia variedad de tecnologías, plataformas de *software*, especificaciones de aplicaciones y procesos. El objetivo primario es contribuir a la toma de decisiones que mejoren el desempeño de la empresa, institución u organización y promover su ventaja competitiva en el mercado o sector al que pertenece. A su vez, posibilita tomar mejores decisiones, convertir datos en información y utilizar un método razonable para la gestión empresarial. (Peña 2006) La aplicación de IN trae consigo disímiles beneficios dentro de las cuales se encuentran:

1. Incremento del volumen de clientes.
2. Aumento de la productividad.
3. Crecimiento de las ventas.
4. Reducción de costos.
5. Fidelización y satisfacción de clientes.

1.1.1 Componentes de la inteligencia de negocio.

El concepto IN es lo suficientemente amplio como para detallar los componentes que en él se engloban. En ese sentido se debe considerar desde los entornos operacionales para la captación de datos, pasando por contenedores de estos datos, y acabando con las herramientas que trabajan para generar la información a partir de los datos y presentarla posteriormente a los usuarios finales. (Llombart and Intelligence 2003) A continuación se presentan los cuatro componentes básicos de una solución de IN:

Fuentes no integradas: Pertenecen a este componente los sistemas que se utilizan a nivel transaccional y operativo en la empresa, institución u organización (Ejemplo: Sistemas de Información a Clientes, Sistemas Financieros, Sistemas de Ventas, Sistemas de Producción, Sistemas de Recursos Humanos, Sistemas de Mercadotecnia, Ficheros Microsoft Excel, Bases de Datos Microsoft Access,

Información tabulada no digitalizada), ya que son los que cubren el núcleo de las operaciones tradicionales de captura masiva de datos y servicios básicos de tratamiento de datos.

Integración: En el componente de integración se llevan a cabo cuatro operaciones esenciales. La primera de ellas es la extracción de la información que nutrirá el mercado de datos⁷ (MD, por sus siglas en español) a partir de las fuentes de datos, para posteriormente aplicarle las operaciones de limpieza y transformación, y por último cargar la información en el repositorio de datos. Este proceso comúnmente suele denominarse ETL (Por sus siglas del inglés: Extract, Transform and Load).

Almacenamiento: Es el núcleo principal de la solución, aquí se encuentran las estructuras de datos que soportan el almacenamiento de la información para que pueda ser consultada (Ejemplo: almacenes de datos⁸ (AD, por sus siglas en español) y mercados de datos).

Visualización: Este componente es el encargado de presentar la información al usuario final para su análisis. Los usuarios realizan este análisis por medio de herramientas de Procesamiento Analítico en Línea (OLAP por sus siglas del inglés: OnLine Analytical Processing) y de técnicas de IN que permiten explorar los datos almacenados y obtener conocimiento a partir de ellos.

En la siguiente ilustración se pueden observar los cuatro componentes típicos que forman parte de una solución de IN, así como una vista al interior de cada uno.

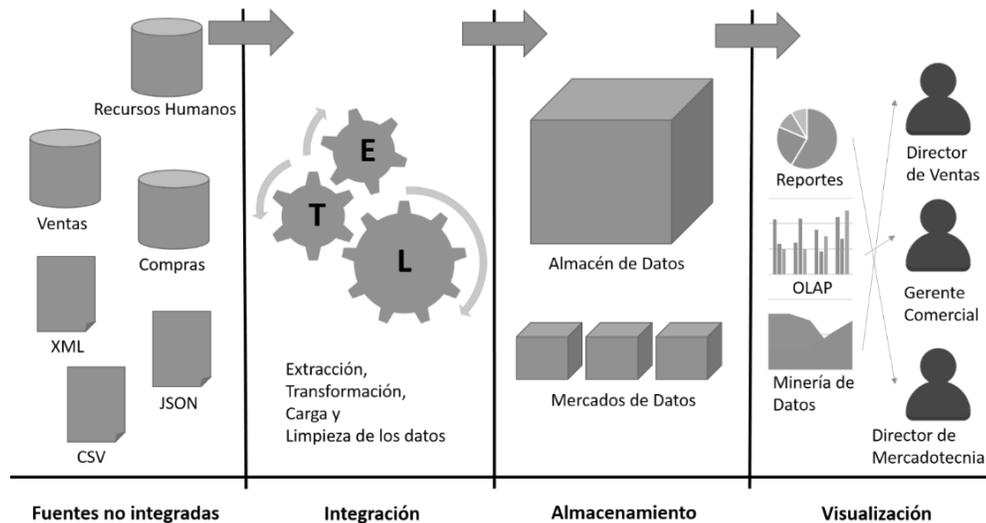


Ilustración 1 Componentes típicos de una solución de IN⁹.

⁷ En inglés: Data Marts. Comúnmente es utilizada la sigla DM para referirse a este concepto.

⁸ En inglés: Data Warehouse. Comúnmente es utilizada la sigla DW para referirse a este concepto.

⁹ Fuente: Elaboración propia.

1.2 Analítica de aprendizaje.

La AA como proceso tiene bases similares a la IN, solo que la información que será objeto de análisis y los usuarios finales a los cuales está dirigida es mucho más reducido. En este caso las fuentes de información se centran en los datos asociados al PEA y los análisis sobre estos están orientados a los decisores dentro del proceso, dentro de los cuales figuran los Profesores, Jefes de Grado, Directores de Centros Educativos.

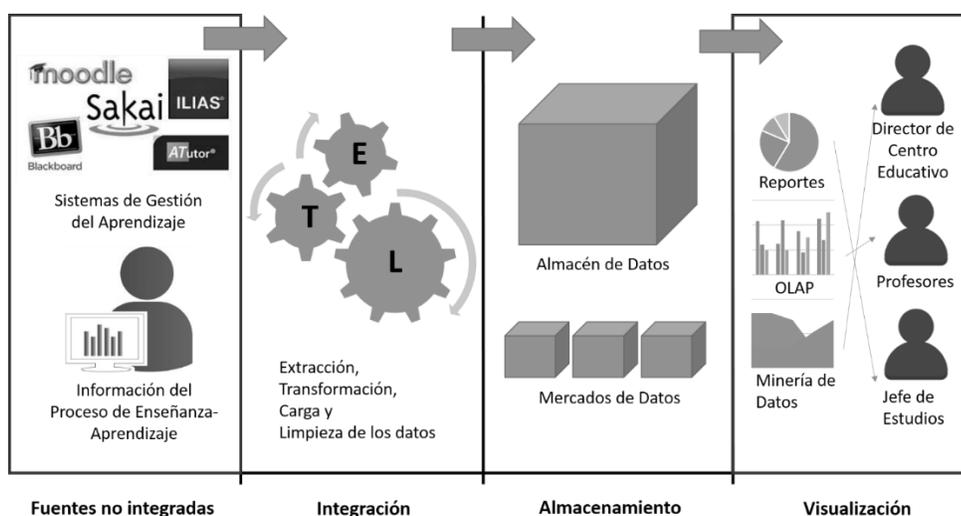


Ilustración 2 Solución de AA centrada en los componentes típicos de una solución de IN¹⁰.

Como elemento en común entre IN y AA se presenta la tecnología de almacenes de datos como un componente imprescindible para almacenar la información proveniente de las diversas fuentes de datos y que será objeto de estudios. Si bien el uso de almacenes de datos no es obligatorio, por la posibilidad de almacenar los datos en otras estructuras como bases de datos relacionales y las bases de datos NOSQL (Del inglés: Not Only SQL), estos se muestran como una alternativa fiable debido a la necesidad de realizar análisis variable en el tiempo con un carácter multidimensional de la información.

1.3 Almacenes y mercados de datos.

Un AD proporciona un ambiente para que las organizaciones hagan un mejor uso de la información que está siendo administrada por diversas aplicaciones operacionales. Posibilita la extracción de datos de sistemas operacionales y fuentes externas, la integración y homogenización de los datos, provee información que ha sido transformada y resumida, para que ayude en el proceso de toma de decisiones estratégicas y tácticas.

¹⁰ Fuente: Elaboración propia.

1.3.1 ¿Qué es un almacén de datos?

Al definir conceptualmente la terminología AD es preciso referenciar a personalidades en esta área del conocimiento, tales son los siguientes casos:

Bill Inmon¹¹: *“Un Data Warehouse es una colección de datos orientados a temas, integrados, no volátiles y variante en el tiempo, organizados para soportar necesidades empresariales”*. (Inmon 2005)

La definición dada por Inmon destaca cuatro aspectos claves:

1. Orientados por temas: Está organizado de acuerdo a los temas más importantes para la organización, facilitando que todos los datos referentes a un proceso, evento u objeto de la realidad se relacionen entre sí.
2. Integrados: Los datos provienen de diversas fuentes, por lo general heterogéneas, las cuales confluyen en el contexto de la organización.
3. No volátiles: Los datos no sufren procesos de actualización, ni eliminación, una vez almacenados solo pueden ser consultados.
4. Variables en el tiempo: Los datos se almacenan con una referencia temporal, lo cual posibilita la realización de análisis históricos.

Ralph Kimball¹²: *“Un Data Warehouse es una copia de las transacciones de datos específicamente estructurada para la consulta y el análisis”*. (Kimball and Ross 2011)

Kimball también lo definió como *“la unión de todos los Data Marts de una entidad”*. Siendo estos un pequeño almacén de datos centrado en un tema o un área de negocio específica dentro de una organización.

Barry Devlin¹³: *“Un Data Warehouse simplemente es un almacén de datos único, completo y consistente, obtenido de una variedad de fuentes y puesto a disposición de los usuarios terminales de tal manera que ellos puedan entenderlo y usarlo en un contexto empresarial”*. (Devlin 1997)

Estas definiciones están centradas fundamentalmente en los datos como elemento central, destacando su carácter variable en el tiempo para posibilitar la toma de decisiones empresariales. En la siguiente ilustración se resumen las características de los almacenes de datos que guiarán la investigación.

¹¹ Información sobre Bill Inmon: <http://www.inmoncif.com/about/>.

¹² Información sobre Ralph Kimball: <http://www.kimballgroup.com/about-kimball-group/>.

¹³ Información sobre Barry Devlin: <http://smartdatacollective.com/users/barry-devlin>.



Ilustración 3 Definición de AD¹⁴.

1.3.2 Ventajas y desventajas de un almacén de datos.

A continuación, se muestra un conjunto de ventajas y desventajas que trae consigo el uso de almacenes de datos en una institución.

Ventajas:

1. Transforma datos orientados a las aplicaciones en información orientada a la toma de decisiones.
2. Integra y consolida diferentes fuentes de datos y departamentos empresariales, que anteriormente formaban islas, en una única plataforma sólida y centralizada.
3. Provee la capacidad de analizar y explotar las diferentes áreas de trabajo y de realizar un análisis inmediato de las mismas.
4. Permite reaccionar rápidamente a los cambios del mercado.
5. Aumenta la competitividad en el mercado.
6. Elimina la producción y el procesamiento de datos que no son utilizados ni necesarios, producto de aplicaciones mal diseñadas o ya no utilizadas.
7. Mejora la entrega de información, es decir, información completa, correcta, consistente, oportuna y accesible; información que los usuarios necesitan, en el momento adecuado y en el formato apropiado.
8. Logra un impacto positivo sobre los procesos empresariales.
9. Aumento de la competitividad de los encargados de tomar decisiones.
10. Los usuarios pueden acceder directamente a la información en línea, ya que esta gira en torno al negocio.

¹⁴ Fuente: Disponible en <http://www.businessintelligence.info/docs/hefesto-v2.pdf>.

11. Permite la toma de decisiones estratégicas y tácticas.

Desventajas:

1. Requiere una gran inversión, debido a que su correcta construcción no es tarea sencilla y consume muchos recursos, además, su misma implementación implica desde la adquisición de herramientas de consulta y análisis, hasta la capacitación de los usuarios.
2. Existe resistencia al cambio por parte de los usuarios.
3. Los beneficios del AD son apreciados en el mediano y largo plazo.
4. No todos los usuarios confiarán en el AD en una primera instancia, pero sí lo harán una vez que comprueben su efectividad y ventajas. Además, su correcta utilización surge de la propia experiencia.
5. Infravaloración de los recursos necesarios para la captura, carga y almacenamiento de los datos.
6. Infravaloración del esfuerzo necesario para su diseño y creación.
7. Incremento continuo de los requerimientos del usuario.

1.3.3 Mercados de datos.

Un mercado de datos (MD, por sus siglas del español) es la implementación de un AD con alcance restringido a un área funcional, problema en particular, departamento, tema o grupo de necesidades. Muchos AD comienzan siendo mercados de datos, para, entre otros motivos, minimizar riesgos y producir una primera entrega en tiempos razonables. Pero una vez que estos se han implementado exitosamente, su alcance se irá ampliando paulatinamente.

Dentro de las ventajas de aplicar un MD a un negocio, se encuentran las siguientes:

1. Son simples de implementar.
2. Ofrecen mayor rapidez de consulta.
3. Conllevan poco tiempo de construcción y puesta en marcha.
4. Permiten manejar información confidencial.
5. Reflejan rápidamente sus beneficios y cualidades.
6. Reducen la demanda del depósito de datos.

Debido a la complejidad y el alcance de la propuesta de solución es que se decide la construcción de un MD, el cual deberá crecer en el tiempo al nutrirse de la información de nuevos procesos que sean objeto de interés; en el marco de la investigación solo será objeto de estudio la información relacionada al proceso de Interacción Usuario-EVA. Aunque la propuesta de solución consista en un MD, tanto la

metodología como las herramientas seleccionadas están enfocadas a los AD, ya que los primeros son una versión reducida de los segundos.

1.3.4 Estudio de soluciones similares.

En Cuba desde hace algunos años se vienen desarrollando investigaciones y aplicaciones basadas en el uso de los almacenes de datos. El creciente desarrollo de la gestión del conocimiento en las empresas y en las instituciones cubanas ha propiciado que los especialistas iniciaran la introducción de herramientas como estas, para elevar sus niveles de eficiencia desde el punto de vista organizativo, de control, analítico, económico y para dar solución a los problemas que se presentan con el análisis de grandes volúmenes de información. El sector de la educación nacional no se encuentra exento de esta situación, específicamente en el área del E-Learning han sido creadas varias soluciones.

A continuación, se presentan varias soluciones nacionales que comprenden a los almacenes y mercados de datos como herramienta para la toma de decisiones en el área del aprendizaje:

Ámbito Nacional:

El AD de la Plataforma Educativa ZERA desarrollada en el centro FORTES, permite estructurar y analizar la información relacionada con el uso de los recursos publicados. (Vitier Urquizu 2014)

El AD para el Entorno Virtual de Aprendizaje constituye una solución fundamentalmente enfocada a la integración en una única fuente de datos que contendrá la información de la interacción de los usuarios con la plataforma, mostrando un análisis básico y sin la concepción de un entorno operable para los usuarios finales. (Castellón Martín. 2016)

Mercado de Datos para la toma de decisiones a partir de la información de los estudiantes generada por el Sistema de Gestión Universitaria y el Entorno Virtual de Enseñanza Aprendizaje de la Universidad de las Ciencias Informáticas. Permite el análisis de los resultados académicos de los estudiantes, a partir de las evaluaciones en las actividades realizadas en el EVA y teniendo en cuenta además la información registrada en el Sistema de Gestión Universitaria (Marrero Pérez and Ramos Pajón 2015)

En la arena internacional la variedad de soluciones es más amplia debido a la brecha tecnológica existente entre nuestro país y el mundo desarrollado, fundamentalmente causada por las condiciones del extraterritorial bloqueo económico y financiero impuesto por el gobierno de los Estados Unidos de América a nuestra isla.

A continuación, se presentan varias soluciones internacionales que comprenden a los almacenes de datos como herramienta para la toma de decisiones en el área del aprendizaje:

Ámbito internacional:

El AD para una herramienta de seguimiento de usuarios de la Universidad Católica de Loja, el cual permite estructurar y conocer información sobre la interacción de los usuarios sobre la plataforma Moodle. (Narváez and Miguel 2010)

El AD de la plataforma Desire2Learn en la universidad de Wisconsin, para investigar el potencial y las capacidades del sistema de éxito estudiantil. (Moore 2013)

Conclusiones del análisis de las soluciones similares en el ámbito nacional e internacional:

Del estudio de las soluciones similares se puede arribar a que todas constituyen soluciones construidas a la medida del problema que resuelven. Dieron una perspectiva acerca de la construcción de los almacenes y mercados de datos, el levantamiento de los indicadores y la selección de las perspectivas de análisis. Mostraron una visión acerca de la información gestionada por los LMS, que se debe tener en cuenta para tomar decisiones (Interacción de los usuarios con actividades, cursos y recursos), así como posibles indicadores (Ejemplo: Número de visitas, Duración de las visitas) y perspectivas de análisis (Usuario, Curso, Fecha).

1.3.5 Metodologías para el desarrollo de almacenes de datos.

Al transcurrir los primeros años de la década de los 70 el proceso de desarrollo de *software* a nivel mundial se encontraba envuelto en una crisis, desatada por el empleo de malas prácticas. Esta situación provocó que comenzaran a surgir estándares y metodologías que estructuran un desarrollo de *software* más organizado y con mejor calidad. Surge así una nueva disciplina, la Ingeniería de *Software*, que según unos de sus padres Roger Pressman, es una disciplina que integra el proceso, los métodos, y las herramientas para el desarrollo de *software* de computadora. (Rogers 2005) Precisamente una de las buenas prácticas que promueve esta disciplina es el uso de metodologías para guiar el proceso desarrollo de *software*. “*Las metodologías imponen un proceso disciplinado sobre el desarrollo de software con el fin de hacerlo más predecible y eficiente.*” (González 1982); a continuación se muestra el estudio para seleccionar la más adecuada a la investigación, analizando dos metodologías dirigidas por los requisitos de usuario y por la descripción de las bases de datos operacionales, denominadas HEFESTO y Kimball.

Metodología HEFESTO.

Esta metodología propone que la construcción e implementación de un AD puede adaptarse muy bien a cualquier ciclo de vida de desarrollo de *software*, con la salvedad de que, para algunas fases en particular, las acciones que se han de realizar serán muy diferentes. Lo que se debe tener muy en

cuenta es no entrar en la utilización de metodologías que requieran fases extensas de reunión de requerimientos y análisis, fases de desarrollo monolítico que conlleve demasiado tiempo y fases de despliegue muy largas. Lo que se busca, es entregar una primera implementación que satisfaga una parte de las necesidades, para demostrar las ventajas del AD y motivar a los usuarios. La metodología HEFESTO, puede ser incluida en cualquier ciclo de vida que cumpla con la condición antes declarada.

Con HEFESTO se comienza recolectando las necesidades de información de los usuarios y se obtienen las preguntas claves del negocio. Luego, se deben identificar los indicadores resultantes de los interrogativos y sus respectivas perspectivas de análisis, mediante las cuales se construye el modelo conceptual de datos del AD. Después, se analizan los sistemas de Procesamiento Transaccional en Línea (OLTP por sus siglas del inglés: OnLine Transaction Processing) para determinar cómo construir los indicadores, señalar las correspondencias con los datos fuentes y para seleccionar los campos de estudio de cada perspectiva.

Una vez hecho esto, se pasa a la construcción del modelo lógico del AD, donde se define el tipo de esquema que se implementará. Seguidamente, se confeccionan las tablas de dimensiones y las tablas de hechos, para luego efectuar sus respectivas uniones. Por último, utilizando procesos de ETL, se definen políticas y estrategias para la carga inicial del AD y su respectiva actualización. (Bernabeu 2009)

El proceso anteriormente descrito puede observarse en la siguiente ilustración, donde figuran las fases y las actividades que se realizan en cada una de ellas.



Ilustración 4 Fases de la metodología HEFESTO¹⁵.

¹⁵ Fuente: Disponible en <http://www.businessintelligence.info/mercado/metodologia-hefesto-v2.html>.

Metodología Kimball

La metodología se basa en lo que Ralph Kimball denomina Ciclo de Vida Dimensional del Negocio¹⁶. (Kimball 2008) Este ciclo de vida del proyecto de AD, está basado en cuatro principios básicos:

1. Centrarse en el negocio: Hay que concentrarse en la identificación de los requerimientos del negocio y su valor asociado, y usar estos esfuerzos para desarrollar relaciones sólidas con el negocio, agudizando el análisis del mismo y la competencia consultiva de los implementadores.
2. Construir una infraestructura de información adecuada: Diseñar una base de información única, integrada, fácil de usar, de alto rendimiento donde se reflejará la amplia gama de requerimientos de negocio identificados en la empresa.
3. Realizar entregas en incrementos significativos: Crear el AD en incrementos entregables en plazos de 6 a 12 meses. Hay que usar el valor de negocio de cada elemento identificado para determinar el orden de aplicación de los incrementos. En esto la metodología se parece a las metodologías ágiles de construcción de *software*.
4. Ofrecer la solución completa: Proporcionar todos los elementos necesarios para entregar valor a los usuarios de negocios. Esto significa tener un almacén de datos sólido, bien diseñado, con calidad probada, y accesible.

La metodología se divide en cuatro fases donde se realizan un grupo de actividades:

Fase I – Requerimientos y gestión del proyecto

1. Definición del proyecto.
2. Gestión y planeación del proyecto.
3. Gestión y planeación del programa.
4. Definición de requisitos del negocio.

Fase II – Arquitectura técnica AD/IN

1. Diseño de la arquitectura.
2. Selección de productos.
3. Gestión de metadatos.
4. Implementación de medidas tácticas de seguridad.
5. Desarrollo del plan estratégico de seguridad.
6. Desarrollo del plan de infraestructura.

¹⁶ En inglés: Business Dimensional Lifecycle.

7. Instalación de productos.

Fase III – Diseño e implementación

1. Diseño del modelo de datos dimensional.
2. Diseño físico de la base de datos.
3. Implementación física de la base de datos.
4. Diseño del subsistema ETL.
5. Desarrollo del subsistema de ETL.
6. Diseño del subsistema de IN.
7. Desarrollo del subsistema de IN.

Fase IV – Implantación y operaciones

1. Pruebas de pre-implantación.
2. Pruebas de datos y procesos.
3. Optimización del rendimiento.
4. Implantación del sistema.
5. Capacitación y transferencia tecnológica.
6. Operaciones de mantenimiento.
7. Operación de soporte.

El ciclo de vida de la metodología permite organizar el desarrollo en tres caminos o rutas (Ver la siguiente ilustración).

1. Pista de Tecnología (Camino Superior): Implica tareas relacionadas con *software* específico. Se seleccionan las herramientas que se van a utilizar en la solución.
2. Pista de Datos (Camino del medio): Se diseña e implementa el modelo dimensional, y el subsistema ETL para cargar el AD.
3. Pista de Aplicaciones de Inteligencia de Negocio (Camino Inferior): En esta ruta se diseñan y desarrollan las aplicaciones para la visualización de información a los usuarios finales.

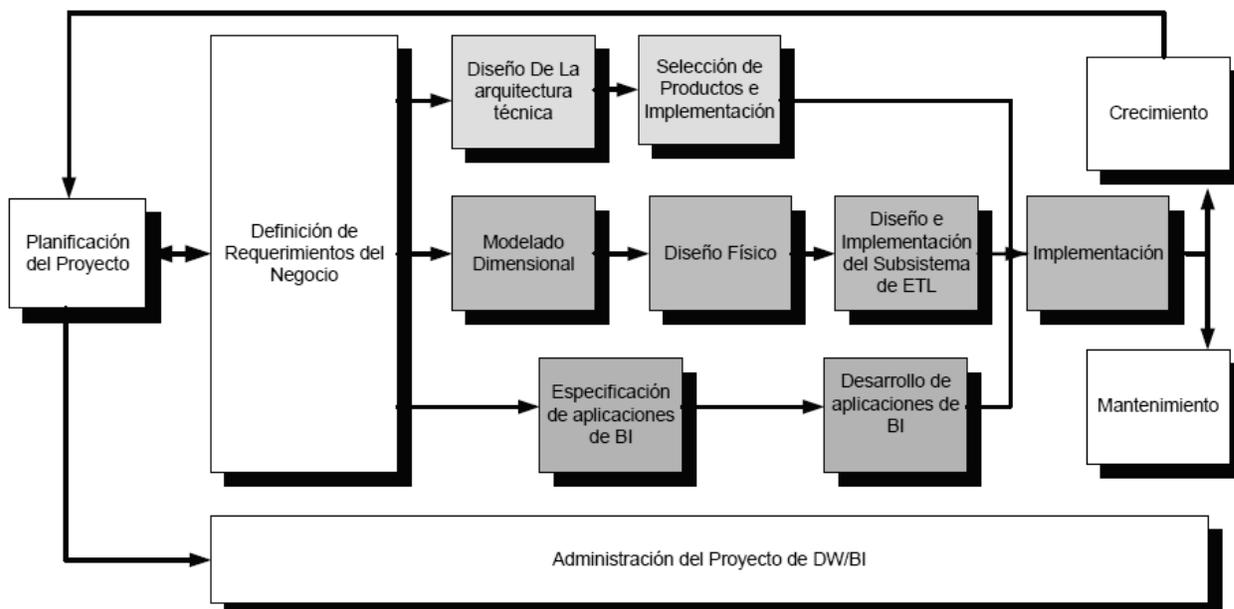


Ilustración 5 Fases de la metodología Kimball¹⁷.

Fundamentación de la metodología seleccionada

Debido a que el MD debe desarrollarse en un corto período de tiempo, se necesita de una metodología sencilla y ágil, pero a su vez madura, que garantice la integración de la información que actualmente se maneja en el EVA.

Por tales razones se escoge la metodología HEFESTO 2.0, ya que permite la construcción de un MD de forma sencilla, ordenada e intuitiva y está fundamentada en una amplia investigación, comparación de metodologías existentes y experiencias en procesos de confección de almacenes de datos. Además, esta metodología brinda las siguientes ventajas:

1. Los objetivos y resultados esperados en cada fase se distinguen fácilmente y son sencillos de comprender.
2. Se basa en los requerimientos del usuario, por lo cual su estructura es capaz de adaptarse con facilidad y rapidez ante los cambios en el negocio.
3. Reduce la resistencia al cambio, ya que involucra al usuario final en cada etapa para que tome decisiones respecto al comportamiento y funciones del AD.
4. Utiliza modelos conceptuales y lógicos, los cuales son sencillos de interpretar y analizar.
5. Es independiente del tipo de ciclo de vida que se emplee para contener la metodología.

¹⁷ Fuente: Disponible en http://docplayer.es/774287-Universidad-nacional-mayor-de-san-marcos.html#show_full_text.

6. Es independiente de las herramientas que se utilicen para su implementación.
7. Es independiente de las estructuras físicas que contengan el AD y de su respectiva distribución.
8. Cuando se culmina con una fase, los resultados obtenidos se convierten en el punto de partida para llevar a cabo el paso siguiente.

1.4 Herramientas para el desarrollo de almacenes de datos.

La industria del software evoluciona constantemente y esto genera la necesidad de utilizar herramientas para el desarrollo de soluciones por parte de los especialistas del sector. Producto de la constante evolución surgen continuamente nuevas herramientas que superan a otras ya existentes y que facilitan aún más el trabajo. Es por ello que se hace necesario realizar un estudio para seleccionar la herramienta de modelado, el servidor de base de datos y la herramienta de IN a utilizar en el desarrollo de la propuesta de solución.

1.4.1 Herramienta de modelado de *software*.

Las herramientas CASE (Por sus siglas del inglés: *Computer Aided Software Engineering*, en español se utiliza el término: Ingeniería de *Software* Asistida por Ordenador) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de *software* reduciendo el costo del desarrollo en términos de tiempo y dinero. Estas herramientas sirven de ayuda en todos los aspectos del ciclo de vida de desarrollo del *software*, en tareas como el diseño del proyecto, cálculo de costos, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación y detección de errores. A continuación, se presenta el estudio realizado para seleccionar la herramienta de modelado de *software* más adecuada al desarrollo del MD.

IBM Rational Rose



Ilustración 6 Logo de IBM Rational Rose¹⁸.

IBM Rational Rose es una herramienta de modelado visual para ayudar en el análisis y diseño de *software* orientado a objetos. Es un *software* propietario de la empresa multinacional estadounidense IBM (Por sus siglas del inglés *International Business Machines*). Constituye una opción para un ambiente de modelado que soporte la generación de código a partir de modelos en Ada, ANSI C++, C++, CORBA, Java™/J2EE™, Visual C++ y Visual Basic. Como todos los demás productos de esta categoría, IBM Rational Rose proporciona un lenguaje común de modelado para el equipo, que facilita

¹⁸ Fuente: Disponible en <http://kurser.iha.dk/eit/tivdm1/materiale.html>.

el desarrollo de *software* con calidad en menores períodos de tiempo. (Quatrani 2003) Otras características adicionales son:

1. Soporte para ingeniería forward y/o inversa para algunos de los conceptos más comunes de Java 1.5.
2. La generación de código Ada, ANSI C++, C++, CORBA, Java y Visual Basic, con capacidad de sincronización Modelo-Código configurables.
3. Capacidad de análisis de calidad de código.
4. Modelado UML (Por sus siglas en español: Lenguaje unificado de modelado) para trabajar en diseños de base de datos, con capacidad de representar la integración de los datos y los requerimientos de aplicación a través de diseños lógicos y físicos.
5. Publicación web y generación de informes para optimizar la comunicación dentro del equipo.

Visual Paradigm for UML



Ilustración 7 Logo de Visual Paradigm for UML¹⁹.

Visual Paradigm for UML es una herramienta CASE que soporta el ciclo de vida completo del desarrollo de *software*: análisis y diseño orientados a objetos, implementación y pruebas. Es un *software* propietario del Hong Kong Institute of Vocational Education²⁰, aunque posee una versión libre para la comunidad denominada Visual Paradigm for UML Community Edition (VP-UML CE por sus siglas en español) para uso no comercial. Ayuda a una rápida construcción de

aplicaciones de calidad. Permite construir diagramas de diversos tipos, código inverso, generar código desde diagramas y generar documentación. Esta herramienta CASE para UML también proporciona abundantes tutoriales, demostraciones interactivas y ejemplos de proyectos UML. (Cabrera González and Pompa Torres 2013)

Entre sus características fundamentales se encuentran:

1. **Multiplataforma:** soportada en plataforma Java para sistemas operativos Windows, Linux y Mac OS.
2. **Interoperabilidad:** intercambia diagramas UML y modelos con otras herramientas. Soporta la importación y exportación a formatos XML (Por sus siglas del inglés: *XML Metadata Interchange*) y XML (Por sus siglas del inglés: *eXtensible Markup Language*, en español se utiliza el término:

¹⁹ Fuente: Disponible en <http://www.macupdate.com/app/mac/16945/visual-paradigm-for-uml>.

²⁰ Página oficial: http://tyict2.vtc.edu.hk/final/facilities_ootc.php.

- lenguaje de marcas Extensible) y archivos Microsoft Excel. Permite importar proyectos de IBM Rational Rose y la integración con Microsoft Office Visio.
3. **Modelado de requisitos:** captura de requisitos mediante diagramas de requisitos, modelado de casos de uso y análisis textual.
 4. **Colaboración de equipo:** realiza el modelado simultáneamente con el Visual Paradigm Team Work Server y Subversión.
 5. **Generación de documentación:** comparte y genera documentación de diagramas y diseños en formatos PDF, HTML, Microsoft Word.
 6. **Editor de detalles de casos de uso:** entorno para la especificación de detalles de casos de usos, incluyendo la especificación del modelo general y las descripciones de los casos de uso.
 7. **Ingeniería de código:** permite la generación de código e ingeniería inversa para los lenguajes: Java, C, C++, PHP, XML, Python, C#, VB .Net, Flash, ActionScript, Delphi y Perl.
 8. **Modelado de procesos de negocio:** visualiza, comprende y mejora los procesos de negocio con la herramienta para procesos de negocio.
 9. **Integración con entornos de desarrollo:** apoyo al ciclo de vida completo de desarrollo de *software* en IDE como: *Eclipse*, *Microsoft Visual Studio*, *NetBeans*, *Sun ONE*, *Oracle JDeveloper*, *Jbuilder* y otros. Modelado de bases de datos: generación de bases de datos y conversiones de diagramas entidad relación a tablas de bases de datos, además de mapeos de objetos y relaciones. (Ledesma Rodríguez et al. 2011)

Fundamentación de la herramienta seleccionada.

Después de realizar un análisis de las herramientas de modelado de *software*, se optó por VP-UML CE en su versión 8.0, ya que esta herramienta en la versión para la comunidad puede ser utilizada sin necesidad de comprar alguna licencia para su uso. En la UCI es utilizada con fines académicos, lo cual facilita la creación de habilidades en el uso de esta herramienta y la preferencia de la misma como herramienta de modelado en proyectos de *software*. Soporta el lenguaje UML (Por sus siglas del inglés Unified Modeling Language). Permite la generación automática de documentación. Proporciona asistencia a los analistas, ingenieros de *software* y desarrolladores, durante todas las etapas del ciclo de vida de desarrollo de un *software*. Posee características para el trabajo con bases de datos, las cuales ofrecen grandes ventajas para el desarrollo del MD, ya que permite elaborar el Diagrama Entidad-Relación (DER por sus siglas en español) y generar automáticamente la base de datos física.

1.4.2 Sistema Gestor de Bases de Datos.

El concepto de Sistema Gestor de Bases de Datos (SGBD por sus siglas en español), se define en el Glosario IEEE²¹ de Ingeniería de *Software* de la siguiente forma: Un sistema informático compuesto por hardware, *software* o ambos, que proporciona una técnica sistemática para la creación, el almacenamiento, el procesamiento y la consulta de la información almacenada en bases de datos. Un SGBD actúa como un intermediario entre las aplicaciones y los datos, o bien entre los datos y la base de datos. (Geraci et al. 1991)

Dentro de los SGBD más conocidos y potentes del mercado se encuentran: MySQL, Oracle Database, Microsoft SQL Server y PostgreSQL.

MySQL™



Ilustración 8 Logo de MySQL™²².

SGBD relacional, de propósito general, multiusuario, rápido, multihilo y robusto, liberado bajo una licencia dual, con una versión propietaria y una comunitaria liberada bajo la licencia GNU GPL (Del inglés *GNU General Public License*).

Ofrece un conjunto de funcionalidades como la portabilidad (se ejecuta en varias plataformas UNIX, Windows y MacOS X), la velocidad (usa técnicas de indexado eficiente, tablas temporales en memoria y algoritmos de optimización del operador JOIN), la escalabilidad (producto de su modularidad y flexibilidad en la configuración), la facilidad de uso (no se necesitan conocimientos extras para instalarlo y configurarlo), el modelo de seguridad bien distribuido (se pueden restringir los permisos de usuarios desde una base de datos completa hasta una columna) y el acceso desde otros lenguajes y sistemas (tiene bibliotecas y APIs (Del inglés *Application Programming Interface*) para la conexión a él desde Java, C/C++, Perl, PHP y TCL). (Vázquez Ortiz and Piñero Pérez 2013) Entre las características esenciales se encuentran: (Vázquez Ortiz and Piñero Pérez 2013) (Oracle Corporation and/or its affiliates 2016)

1. Probado con un amplio número de compiladores.

²¹ Del inglés: Institute of Electrical and Electronics Engineers. En español: Instituto de Ingeniería Eléctrica y Electrónica.

²² Fuente: Disponible en <https://www.mysql.com/about/legal/logos.html>.

2. Las funciones SQL están implementadas usando una librería altamente optimizada, para que sean tan rápidas como sea posible. Normalmente no hay reserva de memoria tras toda la inicialización para consultas.
3. El código MySQL™ se prueba con Purify (un detector de memoria perdida comercial) así como con Valgrind, una herramienta GPL.
4. Ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está cifrado al conectarse a un servidor. Soporte a grandes bases de datos.
5. Los clientes pueden conectarse al servidor MySQL™ usando sockets TCP/IP en cualquier plataforma. En sistemas Windows los clientes se pueden conectar usando named pipes²³ y en sistemas Unix pueden usar ficheros socket Unix.
6. MySQL™ server tiene soporte para comandos SQL para chequear, optimizar, y reparar tablas. Estos comandos están disponibles a través de la línea de comandos y el cliente mysqlcheck.
7. Al ser comprado por una compañía norteamericana, desde Cuba no se puede acceder al código fuente de la versión comunitaria que aún mantienen.

Oracle® Database



Ilustración 9 Logo de Oracle® Database²⁴.

Es básicamente una herramienta Cliente/Servidor para la gestión de bases de datos. Consume gran cantidad de recursos en el servidor, lo que trae consigo que se necesite de buenos recursos de hardware. Cuenta con herramientas necesarias para su explotación, así como con herramientas de programación básicas de Oracle®. Entre sus

principales características se encuentra su seguridad, garantizando la autenticidad apropiada de los usuarios y la privacidad e integridad de la información. Posee lectura de multiversión, proporcionándoles a los usuarios respuestas consistentes. (Heurtel 2009)

²³ Método de comunicación entre procesos.

²⁴ Fuente: Disponible en

<https://raw.githubusercontent.com/ServiceStack/Assets/master/img/livedemos/techstacks/oracle-logo.png>.

Oracle® es configurable en ambientes de OLTP, procesamiento paralelo, clúster y también es una buena solución para implantar almacenes de datos. Se mantiene actualmente como una solución privativa, lo que conlleva a pagar altos costos para adquirir la licencia de uso. (Corey and Abbey 1996)

PostgreSQL



PostgreSQL

Ilustración 10 Logo de PostgreSQL²⁵.

Es un sistema de gestión de bases de datos objeto-relacional. Clasifica como *software* libre por su distribución bajo la licencia BSD (Del inglés Berkeley *Software* Distribution), por lo que su código fuente y ficheros binarios pueden ser utilizados, modificados, redistribuidos y hasta incluidos en *software* con carácter no libre. Utiliza un modelo Cliente/Servidor y utiliza multiprocesos en vez de multihilos para garantizar que el sistema sea estable. Las características técnicas que posee lo hacen uno de los gestores más potentes y robustos del mercado. Su desarrollo comenzó hace más de 16 años, y durante este tiempo, estabilidad, potencia, robustez, facilidad de administración e implementación de estándares han sido las características que más se han tenido en cuenta durante su desarrollo. (R. Martínez 2016) Funciona muy bien con grandes cantidades de datos y una alta concurrencia de usuarios accediendo al mismo tiempo al sistema. Dentro de sus características más relevantes se pueden destacar: (R. Martínez 2016)

1. Es multiplataforma puesto que está disponible para plataformas Linux, Unix y Windows.
2. Posee APIs para programar en distintos lenguajes dentro de los que destacan C/C++, Java, .Net, Perl, Python, Ruby, Tcl, ODBC, PHP, Lisp, Scheme y Qt.
3. Es una base de datos 100% ACID (Del inglés Atomicity, Consistency, Isolation and Durability).
4. Permite realizar copias de seguridad en caliente.
5. Permite múltiples métodos de autenticación y acceso encriptado vía SSL (Del inglés Secure Sockets Layer).
6. Permite replicación sincrónica y asincrónica.
7. Permite escribir procedimientos almacenados en distintos lenguajes como PL/pgSQL, PL/Perl, PL/Python y PL/Tcl.
8. Posee numerosos tipos de datos y ofrece la posibilidad de definir nuevos.
9. Soporta el almacenamiento de objetos binarios grandes como gráficos, videos y sonido.
10. El tamaño de la base de datos solo depende del tamaño del sistema de almacenamiento con que se cuente.

²⁵ Fuente: Disponible en <http://www.ubuntu-pomoc.org/kategorie/aplikacje-instalacja/>.

Fundamentación del SGBD seleccionado

Se decide utilizar PostgreSQL en su versión 9.4 pues es un SGBD objeto-relacional, distribuido bajo licencia BSD (Ello nos exime del pago por su uso), multiplataforma, bien documentado y con su código fuente disponible libremente. Constituye uno de los SGBD de código abierto más potente del mercado, con una comunidad de desarrollo sólida, independiente y establecida; permitiendo el soporte, futuro desarrollo y continuidad. Además, al ser PostgreSQL un sistema de código abierto, adoptarlo contribuiría a incrementar la soberanía tecnológica cubana.

1.4.3 Herramienta de Inteligencia de Negocio²⁶.

En la actualidad se han desarrollado diversas herramientas con el fin de dar un acercamiento a la automatización del diseño, construcción, implementación y mantenimiento de los almacenes de datos. Existen compañías que han marcado hitos en este sentido y que trabajan en la producción de este tipo de soluciones. A continuación, se mencionan algunas de las principales y se describen las principales características de sus principales productos.

Oracle® Business Intelligence Suite



Ilustración 11 Logo de Oracle® Business Intelligence²⁷.

El gigante Oracle, es una de las principales compañías de este tipo, su principal ventaja es su arquitectura multiplataforma en el desarrollo de sus aplicaciones. Con su producto Oracle Business Intelligence, puso al mercado una poderosa herramienta que aborda todo el espectro de requisitos de IN. La suite cuenta con diferentes componentes, algunos de los cuales se describen a continuación:

(Russell and Cohn 2012)

1. Oracle Business Intelligence Server: Es un servidor de IN, diseñado para ser altamente dimensionable, posibilita tener disponible la suite a la audiencia más grande posible. Permite mantener un entorno integrado y el almacenamiento centralizado en el servidor, posibilitando la realización de estudios de mercados e IN, y brinda la posibilidad de exportar los resultados a hojas de cálculos.
2. Oracle Business Intelligence Dashboards: En el ambiente de Oracle Business Intelligence Dashboards, el usuario final trabaja con informaciones vivas, permite realizar mapas y gráficos

²⁶ Si bien la solución que se presenta en la investigación es una solución de AA y no una solución de IN, cabe destacar que como se pretende desarrollar un MD, las herramientas de IN muestran mayor madurez para estas tareas ya que los AD son un componente esencial de la IN.

²⁷ Fuente: Disponible en <http://www.axline.com/product-and-services/oracle-bi-applications/>.

- en una pura arquitectura web. El usuario tiene capacidad completa para navegar, modificar, e interactuar con estos resultados.
3. Oracle Business Intelligence Answers: Permite a los usuarios interactuar con una vista lógica de la información, completamente ajenos de la complejidad de las estructuras de datos. Posibilita la visualización de la información mediante gráficos.
 4. Oracle Delivers: Es una herramienta de alertas en tiempo real activadas bajo unos parámetros predeterminados. Estas alertas se pueden recibir a través de diferentes canales y formatos dependiendo de las preferencias del usuario. Están orientadas a generar acciones o medidas correctoras de forma automática.

Oracle Business Intelligence ofrece una serie de elementos independientes que están diseñados para proporcionar una extensa y completa visibilidad de los datos de negocio a amplias audiencias de usuarios, pudiendo acceder de manera autónoma vía web a aquellas informaciones relevantes que les interesan.

IBM® Cognos® Business Intelligence Suite

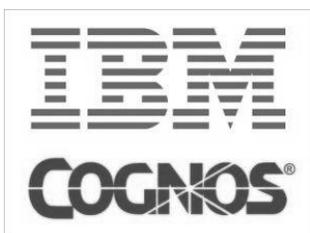


Ilustración 12 Logo de IBM® Cognos® Business Intelligence²⁸.

IBM Cognos Business Intelligence es una solución de IN que proporciona una gama completa de capacidades de este tema como son: creación de informes, análisis, indicadores, cuadros de mando, planificación, elaboración de presupuestos y previsión, además de integración de datos, en una única arquitectura probada. Los principales componentes que conforman la solución se describen a continuación: (Adkison 2013)

IBM Cognos Business Intelligence Reporting: Esta herramienta permite el acceso a una completa lista de tipos de informes de autoservicio, se adapta a cualquier fuente de datos, y opera desde una única capa de metadatos. Posibilita ofrecer reportes en múltiples idiomas.

IBM Cognos Business Intelligence Dashboard: Permite realizar Cuadros de Mando del negocio, que ofrecen información compleja de forma rápida. Muestra la información de los diversos sistemas y datos corporativos utilizando indicadores, mapas y gráficos.

IBM Cognos Business Intelligence Analysis: Posibilita la exploración guiada y el análisis de información relacionada con todas las dimensiones del negocio, con independencia de donde se encuentren

²⁸ Fuente: Disponible en <http://www.itpalooza.com/sponsors/>.

almacenados los datos. Analiza y genera informes a partir de fuentes OLAP y fuentes de datos relacionales basadas en dimensiones.

Pentaho™ Open Source Business Intelligence™ Suite Community Edition



Ilustración 13 Logo de Pentaho™ Open Source Business Intelligence™²⁹.

Pentaho Open Source Business Intelligence es una herramienta de código abierto perteneciente a la familia de aplicaciones de IN, la misma ofrece una gama completa de funcionalidades de IN incluyendo las capacidades de consultas y presentación de informes, análisis interactivo, Cuadros de Mando, ETL, Minería de Datos y una plataforma que la han convertido en la suite de código abierto más

popular del mundo. La suite incluye: (Bouman and Dongen 2010)

1. Pentaho Reporting: Permite a las empresas acceder fácilmente a la información, darle formato y distribuirla a sus empleados, clientes y socios. Posibilita flexibilidad en la implementación de sus reportes.
2. Para obtener la funcionalidad de procesamiento OLAP la suite utiliza otras dos aplicaciones: el servidor OLAP Mondrian, que combinado con Jpivot, permiten realizar consultas a almacenes de datos, que los resultados sean presentados mediante un navegador de modo que el usuario pueda realizar las actividades típicas de navegación. Mondrian utiliza MDX³⁰ como lenguaje de consulta, que fue un lenguaje propuesto por Microsoft. Funciona sobre las bases de datos estándar del mercado, dentro de las que se encuentran: Oracle, DB2, SQL-Server, MySQL y PostgreSQL, lo cual habilita y facilita el desarrollo del negocio basado en la plataforma Pentaho.
3. Pentaho Schema Workbench: Es una herramienta para el desarrollo y prueba de cubos OLAP. La definición del XML no es extremadamente compleja, pero en la práctica resulta engorroso recordar cada uno de los elementos junto a sus atributos y sub-elementos tal y como se encuentran en el almacén. Con esta aplicación, se puede configurar una conexión con el modelo físico, para luego elaborar el esquema lógico de manera simple y efectiva. Para ello la herramienta ofrece un editor de esquemas con la fuente de datos subyacente para su validación.
4. Pentaho Business Intelligence Server: Permite que puedan ejecutarse los informes y aplicaciones, se puede usar como base para construir un sistema propio de IN con Pentaho.

²⁹ Fuente: Disponible en <http://www.mytechlogy.com/IT-blogs/8253/top-5-open-source-business-intelligence-tools/>.

³⁰ Multi-Dimensional eXpresions, más conocida por sus siglas MDX.

5. Pentaho Report Designer: Es una herramienta para generar todo tipo de documentos empresariales e informes de formato complejo, posee una interfaz unificada que permite un acceso rápido a los objetos de reportes de uso común, maximizando el área de diseño y una buena flexibilidad para la visualización de la información. Permite acceder a datos de múltiples fuentes y mostrarlos en cualquier parte del reporte.
6. Pentaho Designer Studio: Es una herramienta que incluye una colección de editores, visores y módulos de administración integrados, que proporcionan un ambiente gráfico para construir y probar documentos.
7. Plug-in de Pentaho para CMD: Es un API escrita en Java que sabe cómo manejar las solicitudes web para obtener un tablero de instrumentos y agregar funcionalidades personalizadas.

De manera general no existen grandes diferencias entre las propuestas código abierto y las propietarias. Todas éstas tienen en común que a partir de una recuperación de una base de datos ofrezcan la posibilidad de la realización de cálculos adicionales, manipulación de los datos y la presentación de la información ante los usuarios en formatos, ya sean predefinidos o especificados dinámicamente.

Fundamentación de la plataforma seleccionada.

De los proveedores de *software* para IN analizados solo Pentaho saca al mercado una versión gratis de sus herramientas, que sin embargo no cuenta con todas las funcionalidades que posee la versión comercial. Este pequeño escenario ilustra lo que pasa actualmente en el mercado de este tipo de *software*, donde solo un restringido grupo de empresas desarrollan esta tecnología para ofertarla gratis o con un costo aceptable, el resto solo oferta sus productos a quienes puedan pagarlos. Esto provoca que solo las empresas poderosas puedan retroalimentarse de sus datos para trazar estrategias que los mantengan liderando el mercado y que las pequeñas y medianas empresas sean cada vez un eslabón más bajo en la cadena económica para poco a poco ser adquiridas por las grandes transnacionales que van monopolizando los precios del mercado e incluso incrementando el desempleo. Se selecciona por lo anteriormente planteado Pentaho™ Open Source Business Intelligence™ Suite Community Edition en su versión 5.0.

1.5 Conclusiones

A partir del estudio realizado para dar solución a la problemática planteada, se arribó a las siguientes conclusiones:

1. Se debe trabajar en un proyecto de AA para el cual será necesario la construcción de un MD que integre la información gestionada por el EVA y que es producto de la interacción de los usuarios con los cursos, actividades y recursos.
2. Fue seleccionada como metodología para guiar el proceso de desarrollo del MD a HEFESTO 2.0, ya que permite la construcción de un MD de forma sencilla, ordenada e intuitiva. Además, los objetivos y resultados esperados en cada fase se distinguen fácil y son sencillos de comprender.
3. Se seleccionó como herramienta de modelado VP-UML CE 8.0 por las facilidades que oferta para el trabajo con bases de datos, especialmente la generación automática de bases de datos a partir de un DER; además de la familiarización que se tiene con la misma y que posee licencia libre de costo para uso no comercial.
4. Se seleccionó el SGBD PostgreSQL 9.4 por ser *software* libre, de código abierto y multiplataforma, lo cual va en consonancia con las políticas de soberanía tecnológica del país; además de mostrar buen rendimiento ante el manejo de grandes volúmenes de datos.
5. Se escogió como herramienta de IN a Pentaho™ Open Source Business Intelligence™ Suite Community Edition 5.0 por su alta completitud como herramienta de este tipo y ser el líder en el sector del *software* libre.

Capítulo 2: Análisis de los requerimientos y los sistemas fuentes del mercado de datos.

El presente capítulo se dedica a las dos primeras fases de la metodología seleccionada, “Análisis de requerimientos” y “Análisis de los OLTP”, generándose los artefactos correspondientes. Se describe el negocio, a fin de lograr un mejor entendimiento de las necesidades del cliente. Se identifican las necesidades de información por parte del cliente, los indicadores y las perspectivas de análisis. Por último, se construyen los modelos conceptuales y los modelos conceptuales ampliados, dos artefactos muy importantes para los posteriores procesos de diseño e implementación.

2.1 Descripción del negocio.

EL EVA constituye un espacio de apoyo al proceso de formación de la carrera de Ingeniería en Ciencias Informáticas de la UCI, donde los profesores pueden implementar estrategias de Enseñanza-Aprendizaje complementarias a las clases presenciales, disponiendo los estudiantes de un poderoso medio en el cual pueden obtener, utilizar o compartir recursos educativos.

Como parte de los procesos que se llevan a cabo dentro del EVA cabe destacar que:

1. Los actores fundamentales dentro de los procesos son los profesores y los estudiantes.
2. Los profesores diseñan los cursos que se encuentran disponibles en la plataforma.
3. Los cursos contienen recursos, actividades y bloques.
4. Las actividades son un conjunto de procesos que conllevan una evaluación del estudiante.
5. Los recursos son los materiales que se publican (Ejemplos: documentos, enlaces a páginas web, videos, audios e imágenes).
6. Los bloques son cuadros informativos sobre actividades, procesos o alertas.
7. Los cursos se dividen en módulos.
8. Existen actividades de evaluación, aprendizaje y comunicación.
9. Las actividades de comunicación son: mensajería interna, blog, foro y chat.
10. Las actividades de evaluación son: lección, consulta, encuesta, módulo de encuesta, taller y cuestionario.
11. Las actividades de aprendizaje son: foro, glosario, taller, blog, wiki, consulta, base de datos, lección y tarea.
12. Los profesores mantienen retroalimentación estrecha con los estudiantes a través de las actividades que provee la plataforma.
13. Los estudiantes interactúan con los cursos y a su vez con los recursos, actividades y bloques de estos.
14. Los cursos se agrupan por categorías.

15. Los profesores y estudiantes son usuarios con roles diferentes dentro de la plataforma.
16. Los usuarios poseen perfiles personalizables (Ejemplos: editar información, cambio de clave de acceso).
17. Los cursos son personalizables (Ejemplos: editar ajustes, reiniciar, calificaciones).
18. La interacción de los usuarios con la plataforma se registra en la base de datos.
19. Por cada interacción de un usuario se almacena en base de datos la fecha de acceso, dirección ip, nombre del curso, nombre del módulo, nombre de la acción que realizó y una breve descripción de la operación realizada.

El siguiente diagrama resume los elementos antes descritos, a través de los principales conceptos presentes en el LMS MOODLE, las relaciones entre estos y la cardinalidad de dichas relaciones.

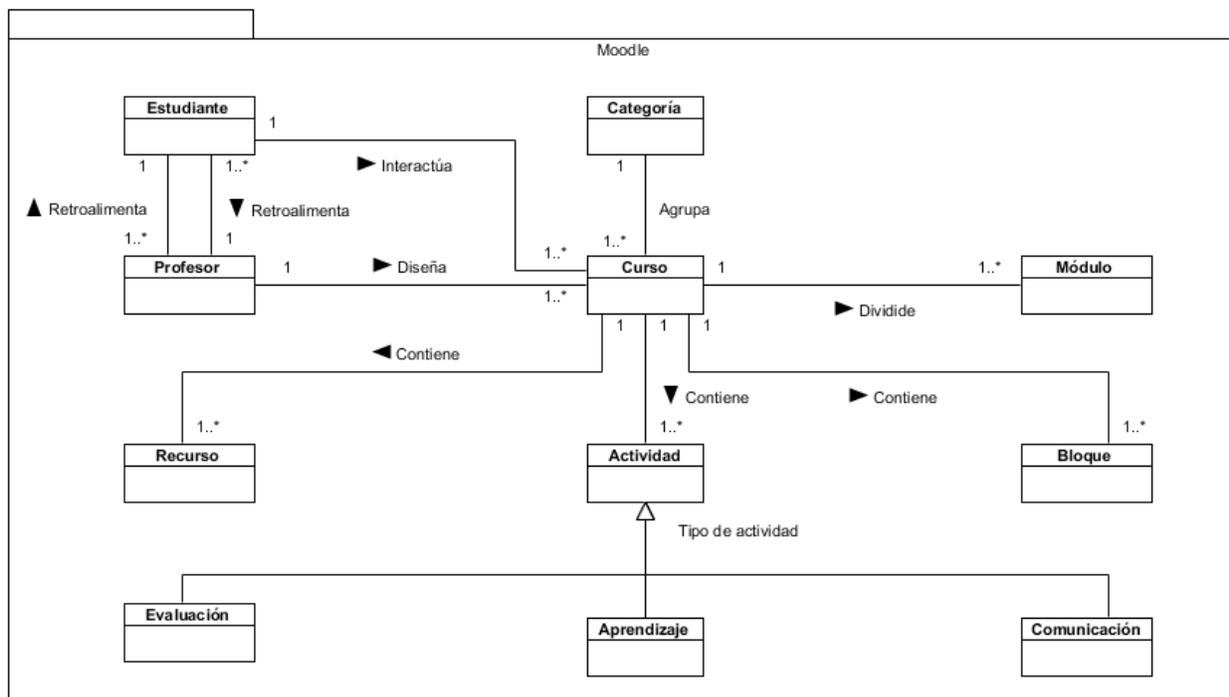


Ilustración 14 Principales conceptos de MOODLE.

Para registrar las interacciones de los usuarios con el EVA, existen en la base de datos de la plataforma un grupo de tablas destinadas a ello. A continuación, se muestra la descripción de las principales tablas relacionadas con el registro de la actividad de los usuarios.

Tabla mdl_course

Esta tabla almacena información relacionada a los cursos que han sido creados en la plataforma. La información que brinda esta tabla es útil para tener un registro de información en base a los cursos, categorías y fecha en que se inicia el curso.

Tabla 1 Descripción de la tabla mdl_course.

Campos	Descripción
category	Identificador de la categoría a la que pertenece el curso.
fullname	Nombre completo del curso.
summary	Pequeña descripción del curso.
format	Especifica si el formato del curso es por semanas o temas de estudio.
startdate	Fecha en que se inicia el curso.

Tabla mdl_course_categories

Contiene información acerca de las distintas categorías existentes en la plataforma. Esta tabla presenta un campo que revela el número de cursos que están asociados a esa categoría, pudiendo de esta manera determinarse un número de cursos por categoría.

Tabla 2 Descripción de la tabla mdl_course_categories.

Campos	Descripción
name	Nombre de la categoría.
description	Descripción de la categoría.
coursecount	Número de cursos que pertenecen a la categoría.
visible	El valor 1 indica que está visible y el valor 0 indica lo contrario.
timemodified	Fecha en la que se ha modificado la categoría.

Tabla mdl_log

Registra información de las distintas acciones que ha realizado el usuario sobre la plataforma.

Tabla 3 Descripción de la tabla mdl_log.

Campos	Descripción
--------	-------------

time	Fecha y hora en la que se realiza una acción.
userid	Identificador del usuario.
ip	Dirección IP desde donde se realiza la acción.
course	Identificador del curso en donde tuvo efecto la interacción del usuario.
module	Módulo sobre el cual se realiza la acción.
action	Acción realizada (Ejemplos: view, view discussion, login, logout).
url	URL del módulo o recurso sobre el cual se realiza la acción.

Tabla mdl_modules

Almacena información acerca de los distintos módulos de la plataforma que están disponibles a nivel general. La tabla indica si los módulos están habilitados o inhabilitados.

Tabla 4 Descripción de la tabla mdl_modules.

Campos	Descripción
name	Nombre del módulo.
version	Versión del módulo.
visible	Si el valor es 1, el módulo está disponible, caso contrario, si tiene un valor de 0 no lo estará.

Tabla mdl_role

Guarda información acerca de todos los roles definidos para los usuarios. Estos roles de usuarios son utilizados para definir los privilegios de acceso sobre los distintos módulos y partes de la plataforma.

Tabla 5 Descripción de la tabla mdl_role.

Campos	Descripción
name	Nombre que identifica al rol (Ejemplos: Administrador, estudiante, profesor).
shortname	Nombre corto para el rol (nickname).
description	Breve descripción acerca del rol definido.

Tabla mdl_user

Registra datos acerca de los usuarios, como: el nombre, apellido, cédula, información de contacto, la dirección de su domicilio, ciudad y país donde se encuentra, y algunos otros datos más.

Tabla 6 Descripción de la tabla mdl_user.

Campos	Descripción
username	Nombre de usuario para el ingreso a Moodle.
idnumber	Número del documento nacional de identidad del usuario.
firstname	Nombres.
lastname	Apellidos.
address	Dirección.
city	Ciudad.
country	País.
lastaccess	Último acceso.
lastip	Última dirección IP.

2.2 Análisis de los requerimientos.

El análisis de los requerimientos es la etapa más importante en el proceso de desarrollo de un MD. El mismo debe ser un proceso riguroso puesto que de él depende la realización de un producto acorde a las necesidades reales del cliente. Para hacer el levantamiento de los requisitos se realizaron entrevistas no estructuradas³¹, en las que se determinaron las necesidades de información que presentaba el cliente, de las cuales resultaron las siguientes:

1. Cantidad de acciones por usuario en un período de tiempo.
2. Cantidad de acciones por rol en un período de tiempo.
3. Cantidad de acciones por usuarios en un curso.
4. Cantidad de acciones por usuario según el tipo de acción.
5. Cantidad de acciones por tipo de acción en un período de tiempo.
6. Cantidad de acciones por usuarios en un módulo.

³¹ La entrevista no estructurada o libre es aquella en la que se trabaja con preguntas abiertas, sin un orden preestablecido, adquiriendo características de conversación. Esta técnica consiste en realizar preguntas de acuerdo a las respuestas que vayan surgiendo durante la entrevista.

- 7. Cantidad de acciones por módulo en un período de tiempo.
- 8. Cantidad de acciones por usuario desde una dirección ip.
- 9. Cantidad de acciones por usuario a una URL.
- 10. Cantidad de acciones por usuario y curso a una URL.

2.2.1 Definición de perspectivas e indicadores.

Luego de analizar las necesidades informacionales del cliente se definen los indicadores y las perspectivas. Los indicadores son los valores numéricos que representan concretamente los requerimientos antes mencionados y las perspectivas son los objetos mediante los cuales se desea analizar cada uno de estos indicadores, ver la siguiente tabla.

Tabla 7 Definición de perspectivas e indicadores.

Indicador	Perspectivas
Cantidad de acciones	Usuario
	Rol
	Curso
	Módulo
	Acción
	URL
	Dirección IP
	Fecha

2.2.2 Modelo Conceptual.

Luego de determinar los indicadores y las perspectivas se procede a realizar los modelos conceptuales, permitiendo tener una mejor percepción del alcance del MD. La siguiente ilustración muestra el modelo conceptual³² del proceso Interacción Usuario-EVA:

³² Un modelo conceptual es una descripción de alto nivel de la estructura de la base de datos, en la cual la información es representada a través de objetos, relaciones y atributos.

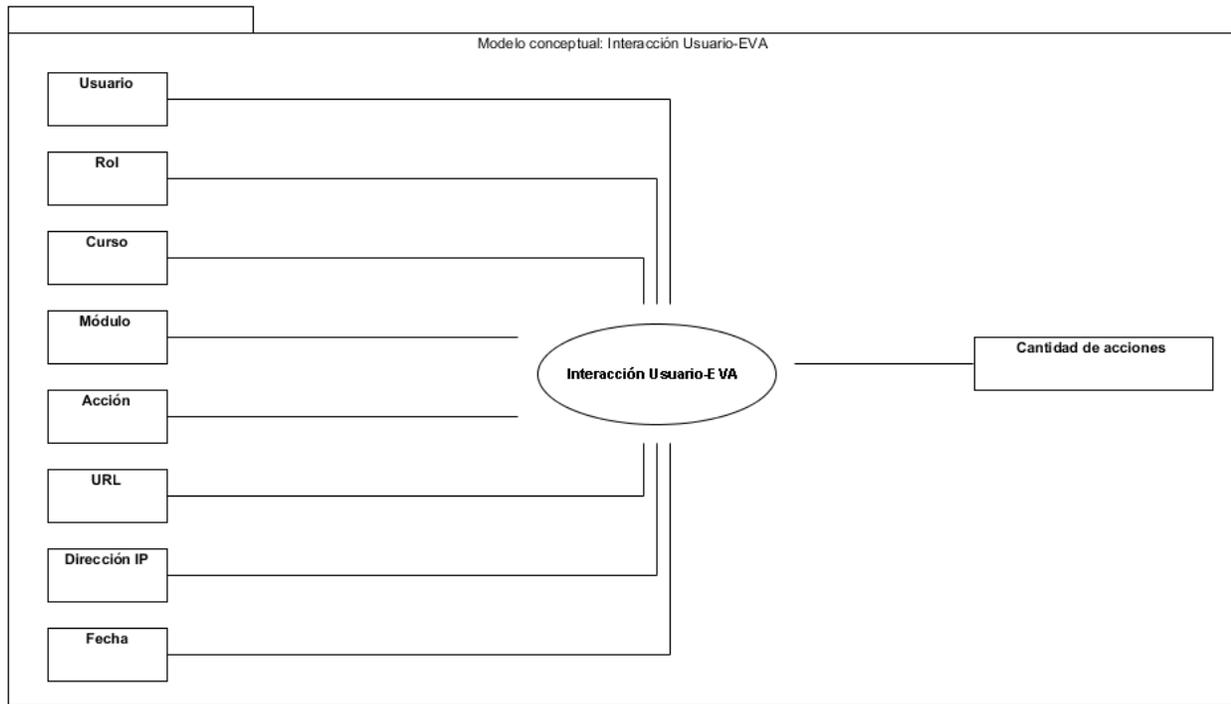


Ilustración 15 Modelo conceptual del proceso Interacción Usuario-EVA.

2.3 Definición del estado general de los sistemas fuentes.

En la presente investigación la fuente de datos consiste en la base de datos del EVA en su versión 2.0, el cual está construido sobre la base del LMS MOODLE en su versión 2.3.4. La misma está disponible para el SGBD PostgreSQL en su versión 9.4. Esta base de datos recibe constantemente nueva información debido al acceso de los estudiantes y profesores a la plataforma.

2.3.1 Determinación de indicadores.

Luego de definidos los indicadores se determina la forma de calcular cada uno de los ellos, ya sea mediante una fórmula definida o mediante una función de sumarización. En la siguiente tabla se muestran los indicadores del proceso antes expuesto en el epígrafe 2.2.2, ver la siguiente tabla.

Tabla 8 Especificación de indicadores.

Proceso	Indicadores	Denominación	Función de sumarización
Interacción Usuario-EVA	Cantidad de acciones	Expresa la sumatoria de cualquier operación registrada	SUM

		por el LMS MOODLE en la tabla mdl_log, las cuales se pueden obtener de cualquier combinación entre las perspectivas definidas.	
--	--	--	--

2.3.2 Establecimiento de correspondencias.

En este paso se procede a establecer las correspondencias entre los elementos del modelo conceptual y los objetos de las fuentes de datos, para garantizar que cada uno de estos elementos sean encontrados en la fuente de la cual se va a extraer la información pertinente, ver la siguiente tabla.

Tabla 9 Correspondencias de datos.

Proceso	Perspectiva	Tabla fuente
Interacción Usuario-EVA	Usuario	mdl_user
	Rol	mdl_role
	Curso	mdl_course
	Módulo	mdl_modules, mdl_log
	Acción	mdl_log
	URL	mdl_log
	Dirección IP	mdl_log
	Fecha	mdl_log

2.3.3 Nivel de granularidad.

Luego de establecer las relaciones de las perspectivas con los OLTP, es necesario detallar qué atributos se analizarán de cada una de ellas, puesto que no es necesario analizar toda la información que se brinda de estos objetos según los requerimientos de información antes definidos, ver la siguiente tabla.

Tabla 10 Nivel de granularidad de cada perspectiva.

Proceso	Perspectiva	Detalle	Descripción
Interacción Usuario-EVA	Usuario	Usuario	Se refiere a la denominación que recibe un usuario para tener acceso a la plataforma. (Ejemplo: apacheco)
	Usuario	Nombre y apellidos	Se refiere al nombre y los apellidos de un usuario de la plataforma. (Ejemplo: Armando Paredes Pacheco)
	Rol	Nombre corto	Se refiere a la denominación abreviada de un rol que puede ser asignado a un usuario en la plataforma. (Ejemplo: principal)
	Rol	Nombre completo	Se refiere al nombre completo de un rol que puede ser asignado a un usuario en la plataforma. (Ejemplo: Profesor principal)
	Curso	Nombre corto	Se refiere a la denominación abreviada de un curso que se encuentra montado en el EVA. (Ejemplo: CEIR)
	Curso	Nombre completo	Se refiere al nombre completo de un curso que se encuentra montado en el EVA. (Ejemplo: Configuración de Equipamiento de Interconexión de Redes)
	Módulo	Módulo	Se refiere a la denominación que reciben los módulos del LMS MOODLE. (Ejemplo: course)
	Acción	Acción	Se refiere a la denominación que reciben las acciones que pueden

			realizar los usuarios al interactuar con el LMS MOODLE. (Ejemplo: login)
	URL	URL	Se refiere a las URL de la plataforma que han sido visitadas por los usuarios. (Ejemplo: view.php?id=522&course=1)
	URL	Denominación	Se refiere a la denominación del objeto relacionado con la URL visitada (curso, recurso, actividad, ...). (Ejemplo: La URL view.php?id=1278 pertenece al recurso Weka 3.7.10)
	Dirección IP	Dirección IP	Se refiere a las direcciones ip desde las cuales acceden los usuarios a la plataforma. (Ejemplo: 10.8.93.250)
	Dirección IP	Zona IP	Se refiere a las zonas de la universidad desde las cuales acceden los usuarios a la plataforma y que son de interés analizar. (Ejemplo: Residencia UCI, Docente 1, Docente 2, Docente 3, Docente 4, Docente 5, Docente 6)
	Fecha	Año	Se refiere al año de la fecha en que se registra la acción.
		Semestre	Se refiere al semestre al que pertenece la fecha en que se registra la acción.
		Trimestre	Se refiere al trimestre al que pertenece la fecha en que se registra la acción.

		Mes	Se refiere al mes de la fecha en que se registra la acción.
		Semana	Se refiere a la semana del año a la que pertenece la fecha en que se registra la acción.
		Día	Se refiere al día de la fecha en que se registra la acción.
		Hora	Se refiere a la hora de la fecha en que se realiza la acción.

2.3.4 Modelo Conceptual Ampliado.

Una vez detallados los campos a analizar de cada perspectiva se puede ampliar el modelo conceptual adicionando los atributos correspondientes a cada perspectiva como se muestra en el modelo conceptual ampliado del proceso Interacción Usuario-EVA.

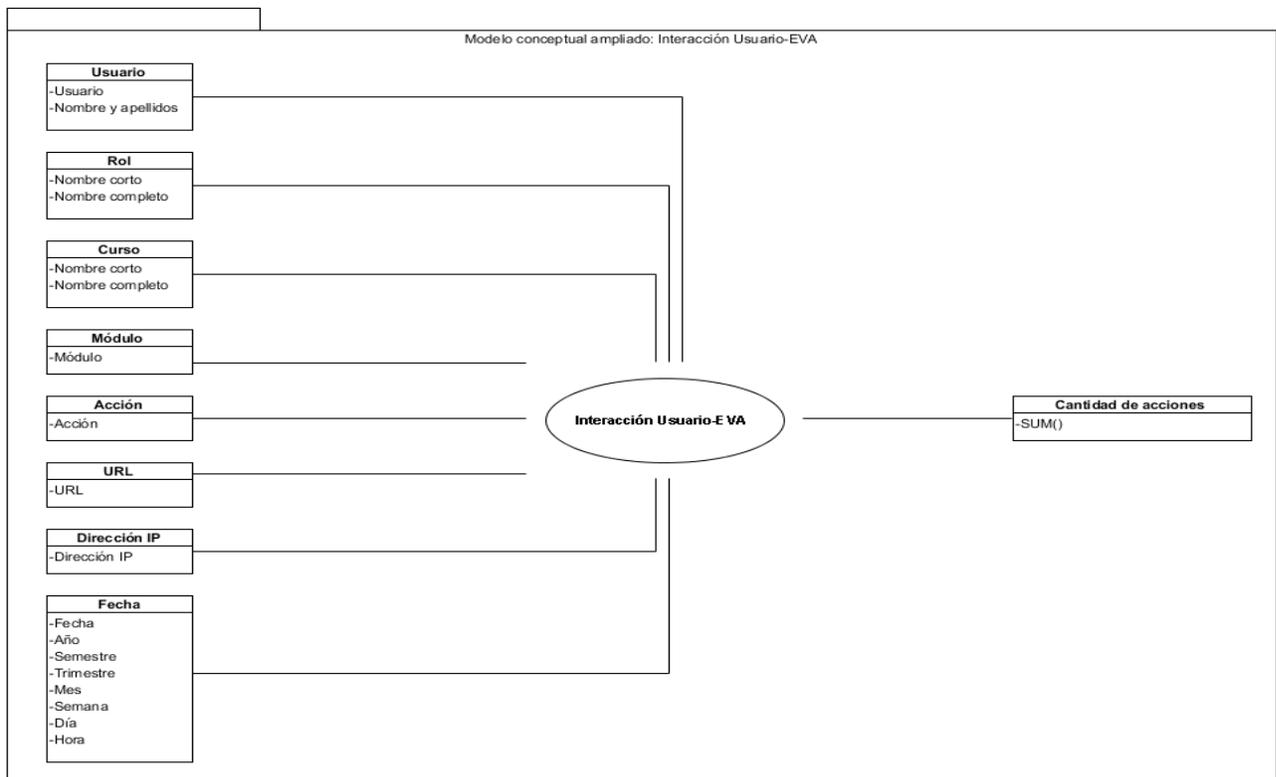


Ilustración 16 Modelo conceptual ampliado del proceso Interacción Usuario-EVA.

2.4 Conclusiones

Fue definido el listado de los requerimientos con el objetivo de desarrollar el MD cumpliendo con las especificaciones del cliente.

Fueron definidas las perspectivas de análisis y los indicadores, lo cual permitió la elaboración del modelo conceptual y modelo conceptual ampliado, esenciales para construir el modelo lógico del MD.

Fueron establecidas las correspondencias entre los elementos del modelo conceptual y la fuente de datos con el objetivo de determinar el lugar de origen de la información que se cargará en el MD.

Se desarrollaron las fases de Análisis de requerimientos y Análisis de los OLTP correspondientes a la metodología HEFESTO, generando los artefactos correspondientes para elaborar el modelo lógico del MD y los posteriores procesos de integración de datos.

Capítulo 3: Diseño e implementación del mercado de datos.

El presente capítulo se dedica a las dos primeras fases de la metodología seleccionada, “Modelo lógico del MD” e “Integración de datos”, generándose los artefactos correspondientes. Se define el modelo lógico por el cual se rige el MD, se identifican y diseñan las tablas dimensiones y hechos del MD, realizando también las uniones entre estas. Por último, se realiza un mapeo lógico de los datos de la base de datos fuente y el destino; se implementan y automatizan los procesos ETL para realizar la carga inicial de los datos en el MD.

3.1 Modelo lógico del mercado de datos.

3.1.1 Diseño físico del mercado de datos

Un sistema OLAP es una proyección multidimensional redundante de una relación. Al computar todas las consultas del tipo “agrupar por”, realiza una agregación de sus resultados en un espacio N-dimensional para responder consultas.(Piedrabuena and Vázquez 2005) Un ejemplo clásico es en el análisis de ventas de determinada empresa, donde se pueden agrupar por producto, región y año, para responder preguntas sobre el comportamiento de los montos totales de las ventas en dependencia de estos factores.

OLAP proporciona varias características a los usuarios que realizan análisis, como por ejemplo:

1. Provee análisis multidimensional dinámico, permitiendo a los usuarios finales realizar actividades analíticas y de navegación sobre los datos, que incluyen cálculo de dimensiones, análisis en periodos de tiempo, visualización de subconjuntos de datos, subir o bajar niveles, comparaciones de varias dimensiones en el área de visualización, etc.
2. Está basado en una modalidad cliente/servidor multiusuario, que ofrece respuestas rápidas, independientemente del tamaño y la complejidad de la base de datos.

Para el procesamiento analítico en línea existen actualmente tres modelos, MOLAP (Por sus siglas del inglés: Multidimensional On Line Analytic Processing), ROLAP (Por sus siglas del inglés: Relational On Line Analytic Processing) y HOLAP (Por sus siglas del inglés: Hybrid On Line Analytic Processing). El proceso de análisis es realizado de la misma forma pero en uno y otro caso varía la metodología de almacenamiento. Esta forma de almacenamiento es crítica para garantizar la velocidad de recuperación de la información y el procesamiento de los datos en general. En la presente investigación dada la selección de un SGBD relacional para implementar el MD, se utiliza el modelo ROLAP.

La premisa de esta arquitectura es que los análisis OLAP se soportan mejor contra bases de datos relacionales, donde estas se encargan del manejo, acceso y obtención de los datos. Por otra parte, el motor ROLAP se integra con niveles de presentación, a través de los cuales los usuarios realizan los análisis multidimensionales, que posteriormente son traducidos a consultas SQL para ser ejecutadas en las bases de datos relacionales. Los resultados obtenidos se relacionan mediante tablas cruzadas y conjuntos multidimensionales para devolver los resultados a los usuarios finales. (Ramón Cueto and Díaz García 2009)

3.1.2 Definición del tipo de modelo lógico del mercado de datos.

La definición del tipo de modelo lógico es determinante para realizar la posterior implementación del MD, puesto que puede influir considerablemente en los tiempos de respuesta al usuario final. Se toma como base el modelo conceptual creado en el capítulo 2. Debido a que existe un solo proceso (Interacción Usuario-EVA), no existen jerarquías entre las perspectivas y el indicador determinado puede ser analizado por todas las perspectivas, se decide utilizar un esquema en estrella. El esquema en estrella, consta de una tabla de hechos³³ central y de varias tablas de dimensiones³⁴ relacionadas a esta, a través de sus respectivas claves.

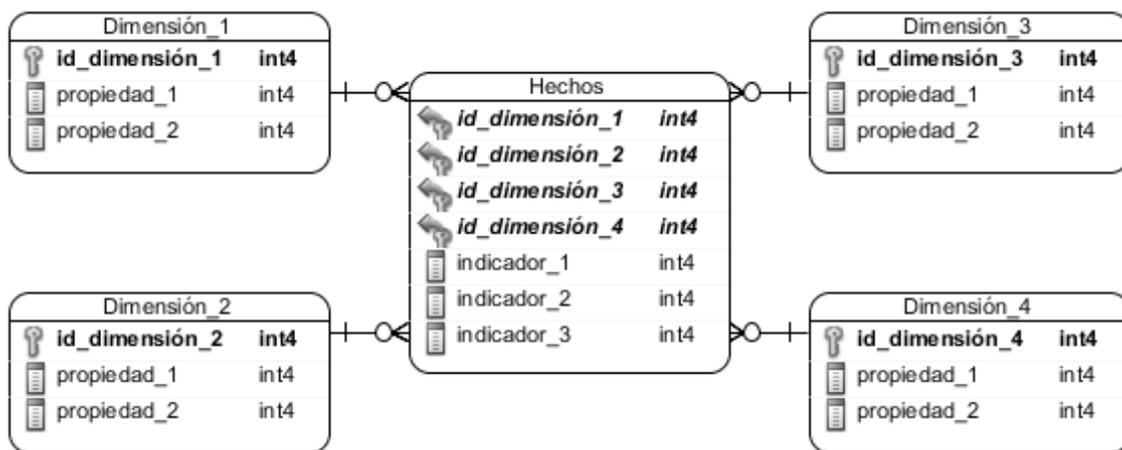


Ilustración 17 Esquema en estrella³⁵.

³³ Es la tabla central de un esquema dimensional, contiene los valores de las medidas o indicadores del negocio. Cada medida se toma mediante la intersección de las dimensiones que la definen, dichas dimensiones estarán relacionadas con la tabla de hechos. Contienen datos cuantitativos.

³⁴ Definen como están los datos organizados lógicamente y proveen el medio para analizar el contexto del negocio. Contienen datos cualitativos.

³⁵ Fuente: Elaboración propia.

A continuación se presentan características relevantes de este tipo de esquema: (Bernabeu 2009)

1. Posee los mejores tiempos de respuesta.
2. Su diseño es fácilmente modificable.
3. Totalmente desnormalizado.
4. Existe paralelismo entre su diseño y la forma en que los usuarios visualizan y manipulan los datos.
5. Simplifica el análisis.
6. Facilita la interacción con herramientas de consulta y análisis.

3.1.3 Identificación de dimensiones y hechos.

Una vez definido el tipo de esquema a utilizar, se identifican y diseñan las tablas de dimensiones y hechos que formarán parte del MD, se toma como referencia el artefacto modelo conceptual ampliado creado en el capítulo 2. A su vez, se hacen corresponder las perspectivas de análisis con las tablas de dimensiones, el proceso en cuestión con las tablas de hechos y los indicadores pasan a formar parte de una tabla de hechos en dependencia proceso al cual se encuentran relacionados. A continuación, se presenta una descripción de las tablas identificadas y diseñadas.

Dimensión Curso

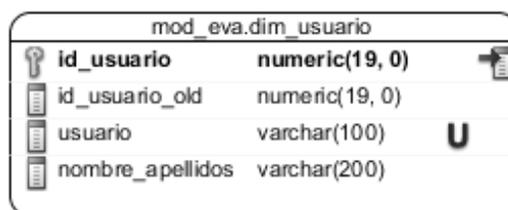
La dimensión Curso está representada en el modelo de datos del MD por la tabla dim_curso, la misma está compuesta por cuatro atributos. El primero (id_curso) funciona como identificador de la tabla, es de tipo numérico con precisión 19, escala 0 y autoincremental. El segundo (id_curso_old) constituye el identificador del curso en la tabla fuente, es de tipo numérico con precisión 19 y escala 0. El tercero (nombre_corto) representa la denominación abreviada del curso, es de tipo cadena de texto con longitud 255 y su valor es único. El cuarto (nombre_completo) representa el nombre completo del curso, es de tipo cadena de texto con longitud 254. La tabla posee un índice sobre el atributo identificador para optimizar las búsquedas sobre este.

mod_eva.dim_curso		
	id_curso	numeric(19, 0)
	id_curso_old	numeric(19, 0)
	nombre_corto	varchar(255) U
	nombre_completo	varchar(254)

Ilustración 18 Tabla dimensión dim_curso.

Dimensión Usuario

La dimensión Usuario está representada en el modelo de datos del MD por la tabla dim_usuario, la misma está compuesta por cuatro atributos. El primero (id_usuario) funciona como identificador de la tabla, es de tipo numérico con precisión 19, escala 0 y autoincremental. El segundo (id_usuario_old) constituye el identificador del usuario en la tabla fuente, es de tipo numérico con precisión 19 y escala 0. El tercero (usuario) representa la denominación de los usuarios que acceden a la plataforma, es de tipo cadena de texto con longitud 100, su valor es único y es utilizado para realizar búsquedas sobre esta dimensión. El cuarto (nombre_apellidos) representa el nombre y los apellidos de los usuarios, es de tipo cadena de texto con longitud 200. La tabla posee un índice sobre el atributo identificador para optimizar las recuperaciones sobre este.



mod_eva.dim_usuario	
id_usuario	numeric(19, 0)
id_usuario_old	numeric(19, 0)
usuario	varchar(100)
nombre_apellidos	varchar(200)

Ilustración 19 Tabla dimensión dim_usuario.

Dimensión Tiempo

La dimensión Tiempo está representada en el modelo de datos del MD por la tabla dim_tiempo, la misma está compuesta por 14 atributos. El atributo id_tiempo funciona como identificador de la tabla, es de tipo numérico con longitud 19 y precisión 0. Los atributos: fecha de tipo fecha, año de tipo numérico con longitud 4 y precisión 0, semestre de tipo cadena de texto con longitud 20, trimestre de tipo cadena de texto con longitud 20, mes de tipo cadena de texto con longitud 20, semana de tipo cadena de texto con longitud 20, día de tipo cadena de texto con longitud 20 y hora de tipo numérico con longitud 19 y precisión 0, hacen referencia a la fecha de registro, año, semestre, trimestre, mes, semana del año, día del mes y hora en la que se genera el log. Los atributos numero_semestre, numero_trimestre, numero_mes, numero_semana y numero_dia se utilizan para establecer un orden entre los semestres (1,2), los trimestres (1,2,3,4), los meses (1 al 12), las semanas del año (1 al 53) y los días del mes (1 al 31). La tabla posee un índice sobre el atributo identificador para optimizar las búsquedas sobre este.

mod_eva.dim_tiempo	
 id_tiempo	numeric(19, 0)
 fecha	date
 anno	numeric(4, 0)
 numero_semestre	numeric(1, 0)
 semestre	varchar(20)
 numero_trimestre	numeric(1, 0)
 trimestre	varchar(20)
 numero_mes	numeric(2, 0)
 mes	varchar(20)
 numero_semana	numeric(2, 0)
 semana	varchar(20)
 numero_dia	numeric(2, 0)
 dia	varchar(20)
 hora	numeric(2, 0)

Ilustración 20 Tabla dimensión dim_tiempo.

Hecho Interacción Usuario-EVA

El hecho Interacción Usuario-EVA está representado en el modelo de datos del MD por la tabla cub_interaccion, la misma está compuesta por nueve atributos. Los atributos id_usuario, id_rol, id_curso, id_modulo, id_accion, id_url, id_direccionip e id_tiempo conforman la llave primaria de la tabla, son de tipo numérico con precisión 19 y escala 0, cada combinación de estos representa una acción de un usuario con un rol a un módulo de un curso, accesible mediante una url y realizada desde una dirección ip en un instante de tiempo. A su vez los atributos id_usuario, id_rol, id_curso, id_modulo, id_accion, id_url, id_direccionip e id_tiempo constituyen llaves foráneas provenientes de cada una de las tablas dimensiones y traspasadas debido a la existencia de una relación con cardinalidad de uno a muchos entre las dimensiones y el hecho. El atributo cantidad_acciones representa al indicador cantidad de acciones, es de tipo numérico con precisión 19 y escala 0. La tabla posee un índice sobre la llave primaria compuesta para optimizar las búsquedas sobre esta.

mod_eva.cub_interaccion		
	<i>id_usuario</i>	<i>numeric(19, 0)</i>
	<i>id_rol</i>	<i>numeric(19, 0)</i>
	<i>id_curso</i>	<i>numeric(19, 0)</i>
	<i>id_modulo</i>	<i>numeric(19, 0)</i>
	<i>id_accion</i>	<i>numeric(19, 0)</i>
	<i>id_url</i>	<i>numeric(19, 0)</i>
	<i>id_direccionip</i>	<i>numeric(19, 0)</i>
	<i>id_tiempo</i>	<i>numeric(19, 0)</i>
	cantidad_acciones	numeric(19, 0)

Ilustración 21 Tabla hecho cub_interaccion.

En este epígrafe solo se muestra la descripción de las tablas de dimensiones Curso, Usuario y Tiempo, además se describe la tabla de hechos Interacción Usuario-EVA. Las restantes descripciones de tablas se encuentran en los anexos de la investigación, en la sección con igual nombre que este epígrafe.

3.1.4 Realización de uniones entre dimensiones y hechos.

Una vez identificadas y diseñadas las tablas de dimensiones y hechos es necesario relacionarlas. Esta tarea se lleva a cabo con el fin de dar por concluido el modelo de datos del MD, donde cada hecho se relaciona con las dimensiones por las cuales serán analizados sus respectivos indicadores. La siguiente ilustración muestra el DER de la propuesta de solución, donde se pueden observar estos aspectos.

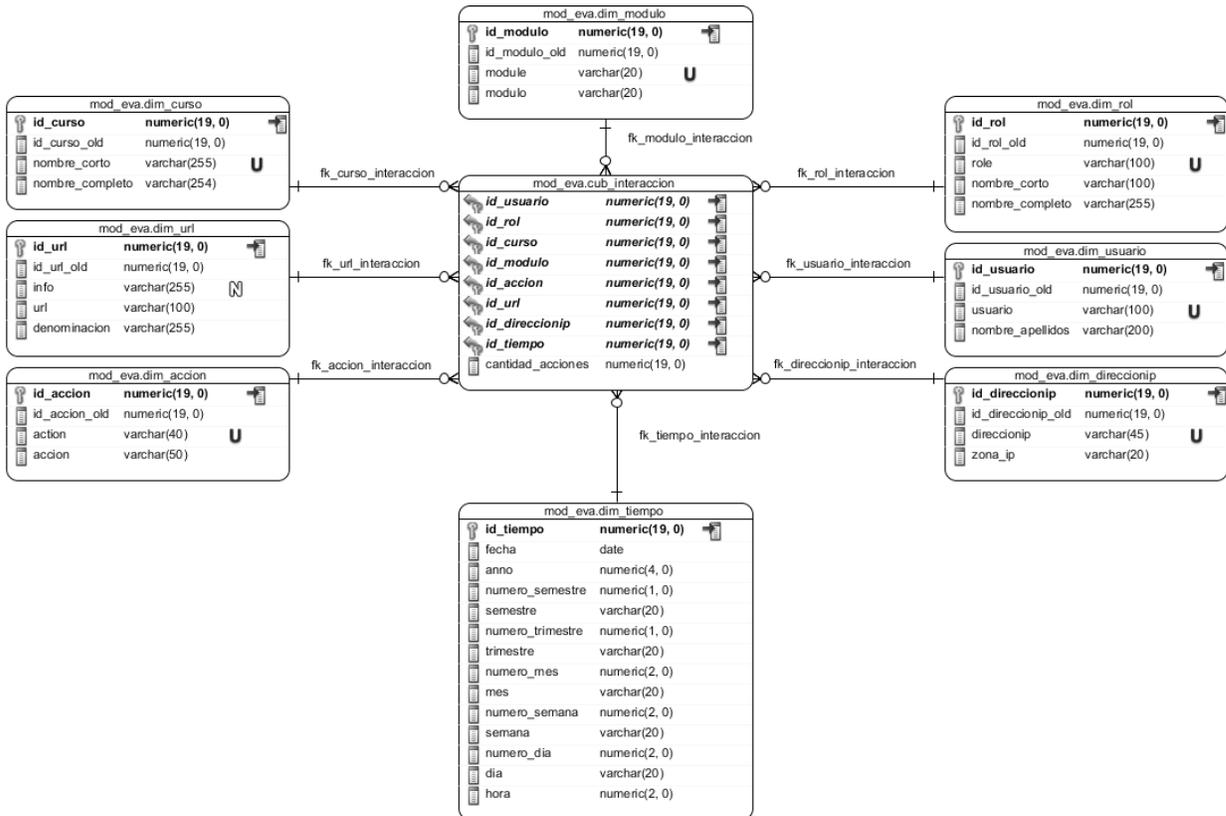


Ilustración 22 Diagrama Entidad-Relación del MD de la propuesta de solución.

Concluido el modelo de datos del MD se procede a implementar el mismo en el SGBD seleccionado, PostgreSQL en su versión 9.4. Esta tarea se realiza por medio de la funcionalidad Generación de Código de Base de Datos que posee la herramienta CASE seleccionada, VP-UML CE en su versión 8.0. Esta funcionalidad genera a partir de un DER una base de datos, en un destino especificado por medio de una conexión configurada en la herramienta.

3.2 Procesos ETL.

Posterior a la implantación del MD en el SGBD PostgreSQL, se procede a mapear los campos de las tablas del mismo, con los respectivos campos de la base de datos del EVA. Esta actividad se lleva a cabo con el objetivo de conocer el origen y el destino de la información que será utilizada para poblar el MD y así evitar la pérdida de información. Seguidamente se realiza la implementación y automatización de los procesos ETL para realizar la carga inicial de los datos.

3.2.1 Mapeo de datos Fuente-A-Destino.

Al realizar el mapeo lógico de los datos se indica tanto para la base de datos fuente como para el destino el nombre de la tabla, nombre del campo y el tipo de datos de donde se extrae y donde se inserta la información, evidenciando de esta forma algunas de las transformaciones realizadas.

Tabla 11. Mapeo lógico de los datos necesarios para cargar el MD de la propuesta de solución.

Fuente de datos			Destino		
Denominación de la fuente de datos: Base de Datos del EVA.			Denominación del destino de los datos: MD de la propuesta de solución.		
Tabla	Campo	Tipo de dato	Tabla	Campo	Tipo de dato
			dim_curso	id_curso	numeric(19,0)
mdl_course	id	bigserial	dim_curso	id_curso_old	numeric(19,0)
mdl_course	fullname	character varying(255)	dim_curso	nombre_corto	character varying(255)
mdl_course	shortname	character varying(255)	dim_curso	nombre_completo	character varying(255)
mdl_log	time	bigint	dim_tiempo	fecha	date
			dim_tiempo	anno	numeric(4,0)
			dim_tiempo	numero_semestre	numeric(1,0)
			dim_tiempo	semestre	varchar(20)
			dim_tiempo	numero_trimestre	numeric(1,0)
			dim_tiempo	trimestre	varchar(20)
			dim_tiempo	numero_mes	numeric(2,0)

			dim_tiempo	mes	varchar(20)
			dim_tiempo	numero_semana	numeric(2,0)
			dim_tiempo	semana	varchar(20)
			dim_tiempo	numero_dia	numeric(2,0)
			dim_tiempo	dia	varchar(20)
			dim_tiempo	hora	numeric(2,0)
			dim_usuario	id_usuario	numeric(19,0)
mdl_user	id	bigserial	dim_usuario	id_usuario_ol d	numeric(19,0)
mdl_user	username	character varying(100)	dim_usuario	usuario	character varying(100)
mdl_user	firstname	character varying(100)	dim_usuario	nombre_apell idos	character varying(200)
mdl_user	lastname	character varying(100)	dim_usuario	nombre_apell idos	character varying(200)

En este epígrafe solo se muestra el mapeo de las dimensiones Curso, Tiempo y Usuario con la base de datos fuente. Los restantes mapeos se encuentran en los anexos de la investigación, en la sección con igual nombre que este epígrafe.

3.2.2 Carga inicial de los datos y procesos ETL.

En este epígrafe se describen las transformaciones y trabajos realizados para llevar a cabo los procesos ETL y realizar la carga inicial del MD. Para la construcción de los procesos ETL fueron usados un conjunto de componentes de la herramienta Pentaho Data Integration en su versión 5.0.1, perteneciente a la suite Pentaho Open Source Business Intelligence, los cuales se describen a continuación.

Tabla 12 Componentes utilizados para la construcción de los procesos ETL.

 <p>Entrada tabla: Lee información de una base de datos, usando una conexión y consultas SQL.</p>	 <p>Valor JavaScript modificado: Permite la ejecución de código JavaScript, utilizando los campos del flujo como parámetros y creando nuevos campos de salida en caso de ser necesario.</p>
 <p>Insertar / Actualizar: Inserta o actualiza filas en una tabla determinada.</p>	 <p>Añadir constantes: Añade una o más constantes a las filas de entrada.</p>
 <p>Generar filas: Produce una cantidad de filas, por defecto vacías, pero con la opción de contener una cantidad de campos estáticos.</p>	 <p>Selecciona / Renombra valores: Selecciona o elimina campos de una fila. Opcionalmente, se puede establecer la meta-información del campo: tipo, longitud y precisión.</p>
 <p>Ordenar filas: Ordena filas basándose en campos específicos, incluyendo sub-ordenamientos, en orden ascendente o descendente.</p>	 <p>Iniciar: Marca el inicio de la ejecución de un trabajo.</p>
 <p>Filas únicas: Elimina valores duplicados del flujo de entrada.</p>	 <p>Transformación: Ejecuta la transformación seleccionada.</p>
 <p>Concatenar campos: Concatena campos del flujo y los devuelve en uno nuevo.</p>	 <p>Satisfactorio: Marca el fin satisfactorio de la ejecución de un trabajo.</p>
 <p>Procedimiento de base de datos: Ejecuta un procedimiento almacenado en base de datos.</p>	 <p>Verificar conexión a base de datos: Verifica si se encuentra disponible la base de datos seleccionada.</p>
 <p>Búsqueda en base de datos: Obtiene tuplas de una tabla de base de datos a partir de determinados campos de búsqueda.</p>	 <p>Trabajo: Ejecuta el trabajo seleccionado.</p>

 <p>Consulta a base de datos: Ejecuta una consulta de base de datos utilizando valores del flujo como parámetros.</p>	 <p>Escribir en fichero Log: Escribe un mensaje en un fichero log.</p>
 <p>Si el valor del campo es nulo: Cambia el valor de un campo del flujo por una constante, si el valor de este es nulo.</p>	 <p>Mostrar mensaje de información: Muestra un cuadro de diálogo con información al usuario.</p>
 <p>Filtrar filas: Permite filtrar un flujo de datos aplicando fórmulas condicionales sencillas.</p>	 <p>Transformación simulada: Este componente no hace nada, en ocasiones suele usarse para enviar a él las filas que se desechan luego de aplicar un filtro.</p>
 <p>Mapeo de valores: Establece una concersión, de valores de un campo a otro valor.</p>	 <p>Operaciones con cadenas de texto: Permite realizar operaciones sencillas sobre campos de tipo cadena de texto .</p>
 <p>Reemplazar en cadena de texto: Reemplaza todas las ocurrencias de una cadena de texto en otra, por un valor establecido.</p>	

Transformación dim_curso

Esta transformación es la encargada de extraer los cursos que se encuentran montados en el EVA. Son extraídos de la tabla mdl_course y almacenados en la tabla dim_curso del MD. Los datos extraídos de la fuente son filtrados para eliminar valores nulos, además se reemplazan caracteres extraños existentes para lograr una mayor legibilidad de los datos al presentarlos al usuario final.

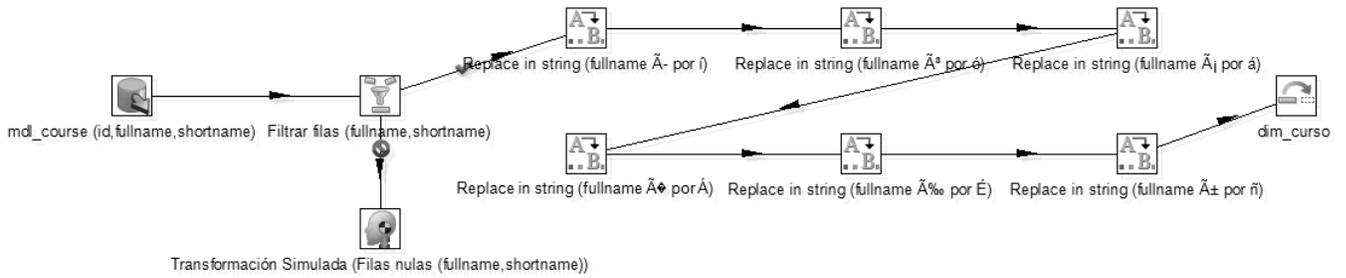


Ilustración 23 Transformación dim_curso.

Transformación dim_usuario

Esta transformación es la encargada de extraer los usuarios que tienen acceso al EVA. Estos son extraídos de la tabla mdl_user y almacenados en la tabla dim_usuario del MD. Durante el proceso es necesario concatenar el nombre y los apellidos del usuario puesto que se guardan en un único campo en el MD. Los datos extraídos de la fuente son filtrados para eliminar valores nulos, además se reemplazan caracteres extraños existentes para lograr una mayor legibilidad de los datos al presentarlos al usuario final.

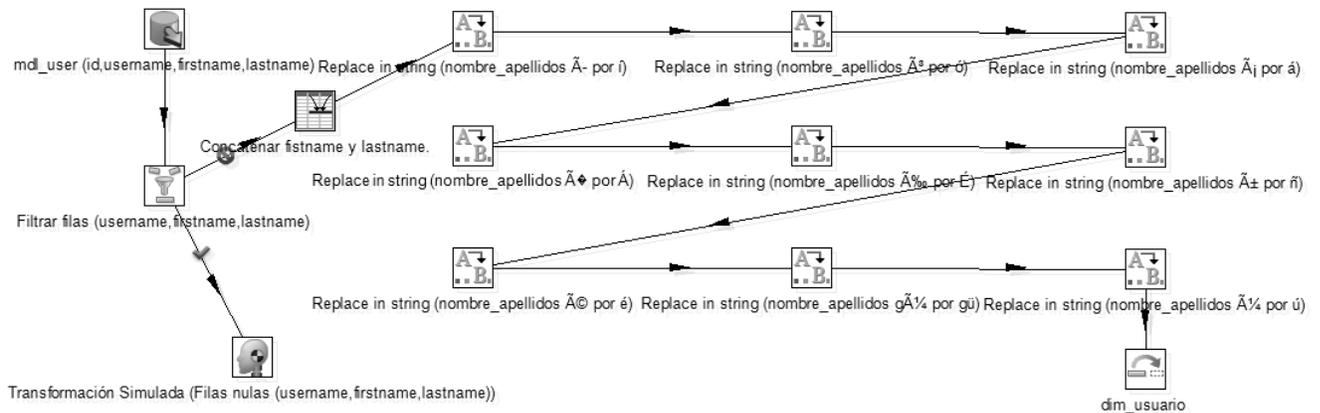


Ilustración 24 Transformación dim_usuario.

Transformación dim_tiempo

Esta transformación es la encargada de realizar la carga de los datos de la dimensión tiempo, para cada combinación de fecha, año, semestre, trimestre, mes, semana, día y hora se genera una tupla. Se necesita como parámetro la fecha de inicio y la fecha de fin del período a utilizar para realizar la carga inicial. Durante la ejecución de la transformación se hace uso de tres procedimientos almacenados que son los encargados de generar las tuplas que serán insertadas en la tabla dim_tiempo del MD. Estos procedimientos son fun_guardar_rango_fechas, fun_guardar_fecha,

fun_obtener_semestre; el primero recibe como parámetro los extremos de un período de tiempo y genera todas las fechas pertenecientes al mismo, el segundo recibe como parámetro una fecha y la almacena en base de datos y el tercero dado un trimestre devuelve el semestre al cual pertenece (fun_guardar_rango_fechas → fun_guardar_fecha → fun_obtener_semestre).



Ilustración 25 Transformación dim_tiempo.

```
CREATE OR REPLACE FUNCTION mod_eva.fun_guardar_rango_fechas(parametro_fecha_inicio TIMESTAMP
WITHOUT TIME ZONE, parametro_fecha_fin TIMESTAMP WITHOUT TIME ZONE)
RETURNS VOID AS
$BODY$
    DECLARE
        variable_fecha_inicio TIMESTAMP;
        variable_hora NUMERIC;
        variable_dia NUMERIC;
    BEGIN
        variable_fecha_inicio = parametro_fecha_inicio;
        variable_hora = 0;
        WHILE (variable_fecha_inicio <= parametro_fecha_fin) LOOP
            WHILE (variable_hora <= 23) LOOP
                variable_fecha_inicio = (((variable_fecha_inicio::DATE)::TEXT) || ' ' ||
                variable_hora::TEXT || ':00:00')::TIMESTAMP;
                EXECUTE mod_eva.fun_guardar_fecha(variable_fecha_inicio);
                IF (variable_hora = 23) THEN
                    variable_fecha_inicio = (((variable_fecha_inicio::DATE + 1)::TEXT) ||
                    ' ' || '00:00:00')::TIMESTAMP;
                END IF;
                variable_hora = variable_hora + 1;
            END LOOP;
            variable_hora = 0;
        END LOOP;
    END;
$BODY$
LANGUAGE plpgsql;
```

Ilustración 26 Función fun_guardar_rango_fechas.

La función fun_obtener_semestre utilizada por la función fun_guardar_fecha, que devuelve el número de semestre al que pertenece un trimestre determinado del año, se encuentra formando parte de los anexos de la investigación en la sección con igual nombre que este epígrafe.

```

CREATE OR REPLACE FUNCTION mod_eva.fun_guardar_fecha(parametro_fecha TIMESTAMP WITHOUT TIME
ZONE)
RETURNS VOID AS
$BODY$
    DECLARE
        array_dias_semana VARCHAR[10];
        array_meses_anno VARCHAR[15];
        nombre_semestre VARCHAR;
        nombre_trimestre VARCHAR;
        nombre_mes_anno VARCHAR;
        nombre_semana VARCHAR;
        nombre_dia_mes VARCHAR;
        nombre_dia_semana VARCHAR;

        numero_anno NUMERIC;
        numero_semestre NUMERIC;
        numero_trimestre NUMERIC;
        numero_mes_anno NUMERIC;
        numero_semana NUMERIC;
        numero_dia_mes NUMERIC;
        numero_dia_semana NUMERIC;
        numero_hora NUMERIC;

        contador NUMERIC;
    BEGIN
        array_dias_semana = ARRAY['Lunes', 'Martes', 'Miércoles', 'Jueves', 'Viernes',
        'Sábado', 'Domingo'];
        array_meses_anno = ARRAY['Enero', 'Febrero', 'Marzo', 'Abril', 'Mayo', 'Junio',
        'Julio', 'Agosto', 'Septiembre', 'Octubre', 'Noviembre', 'Diciembre'];

        SELECT date_part('YEAR', parametro_fecha) INTO numero_anno;
        SELECT date_part('QUARTER', parametro_fecha) INTO numero_trimestre;
        SELECT mod_eva.fun_obtener_semestre(numero_trimestre) INTO numero_semestre;

        SELECT date_part('MONTH', parametro_fecha) INTO numero_mes_anno;
        SELECT date_part('WEEK', parametro_fecha) INTO numero_semana;
        SELECT date_part('DAY', parametro_fecha) INTO numero_dia_mes;
        SELECT date_part('ISODOW', parametro_fecha) INTO numero_dia_semana;
        SELECT date_part('HOUR', parametro_fecha) INTO numero_hora;

        nombre_semestre = 'Semestre ' || numero_semestre;
        nombre_trimestre = 'Trimestre ' || numero_trimestre;
        nombre_mes_anno = array_meses_anno[numero_mes_anno];
        nombre_semana = 'Semana ' || numero_semana;
        nombre_dia_semana = array_dias_semana[numero_dia_semana];
        nombre_dia_mes = nombre_dia_semana || ' ' || numero_dia_mes;

        SELECT COUNT(tiempo. id_tiempo)
        FROM mod_eva.dim_tiempo AS tiempo
        WHERE tiempo.anno = numero_anno AND tiempo.numero_dia = numero_dia_mes AND
        tiempo.numero_mes = numero_mes_anno AND tiempo.hora = numero_hora INTO contador;

        IF (contador = 0) THEN
            INSERT INTO mod_eva.dim_tiempo (id_tiempo, fecha, anno, numero_semestre,
            semestre, numero_trimestre, trimestre, numero_mes, mes, numero_semana,
            semana, numero_dia, dia, hora)
            VALUES ( nextval('mod_eva.sec_tiempo_seq'), parametro_fecha, numero_anno,
            numero_semestre, nombre_semestre, numero_trimestre, nombre_trimestre,
            numero_mes_anno, nombre_mes_anno, numero_semana, nombre_semana,
            numero_dia_mes, nombre_dia_mes, numero_hora);
        END IF;
    END;
$BODY$
LANGUAGE plpgsql;
    
```

Ilustración 27 Función fun_guardar_fecha.

Transformación cub_interaccion

Esta transformación es la encargada de extraer los eventos de interacción de los usuarios con el EVA. Son extraídos de la tabla mdl_log y almacenados en la tabla cub_interacción. Las acciones llevadas a cabo para el llenado de la tabla hecho son:

1. Cargar todos los eventos de interacción de los usuarios con el EVA; por cada evento se extrae de la tabla mdl_log los siguientes campos id, time, userid, ip, course, module, action, url e info.
2. Buscar el valor del campo action en la tabla dim_accion y devolver el valor del campo id_accion correspondiente.
3. Buscar el valor del campo ip en la tabla dim_direccionip y devolver el valor del campo id_direccionip correspondiente.
4. Buscar el valor del campo url en la tabla dim_url y devolver el valor del campo id_url correspondiente.
5. Buscar el rol asociado al usuario en el curso que se genera el log, para ello es necesario usar los campos userid y course, posteriormente se devuelve el valor del campo id_rol correspondiente.
6. Buscar el valor del campo userid en la tabla dim_usuario y devolver el valor del campo id_usuario correspondiente.
7. Buscar el valor del campo course en la tabla dim_curso y devolver el valor del campo id_curso correspondiente.
8. Buscar el valor del campo time en la tabla dim_tiempo y devolver el valor del campo id_tiempo correspondiente.
9. Adicionar a cada tupla un campo denominado cantidad_acciones con un valor constante igual a 1, esto significa que cada tupla aporta un valor de 1 al indicador cantidad de acciones.
10. Al realizarse las búsquedas y no encontrar resultado siempre se devuelve el valor 0 por defecto para el atributo identificador, esto significa que el valor es desconocido.
11. Las acciones de búsqueda se realizan para cada evento interacción cargado.
12. Filtrar las filas que viajan en el flujo de datos para eliminar las que poseen valores nulos (valor 0 para el atributo identificador).
13. Agrupar por aquellas combinaciones de identificadores iguales y almacenar en el campo cantidad_acciones la cantidad de ocurrencias iguales.
14. Insertar cada combinación de los campos id_interaccion, id_usuario, id_curso, id_accion, id_direccionip, id_url, id_rol, id_modulo, id_tiempo y cantidad_acciones en la tabla cub_interaccion.

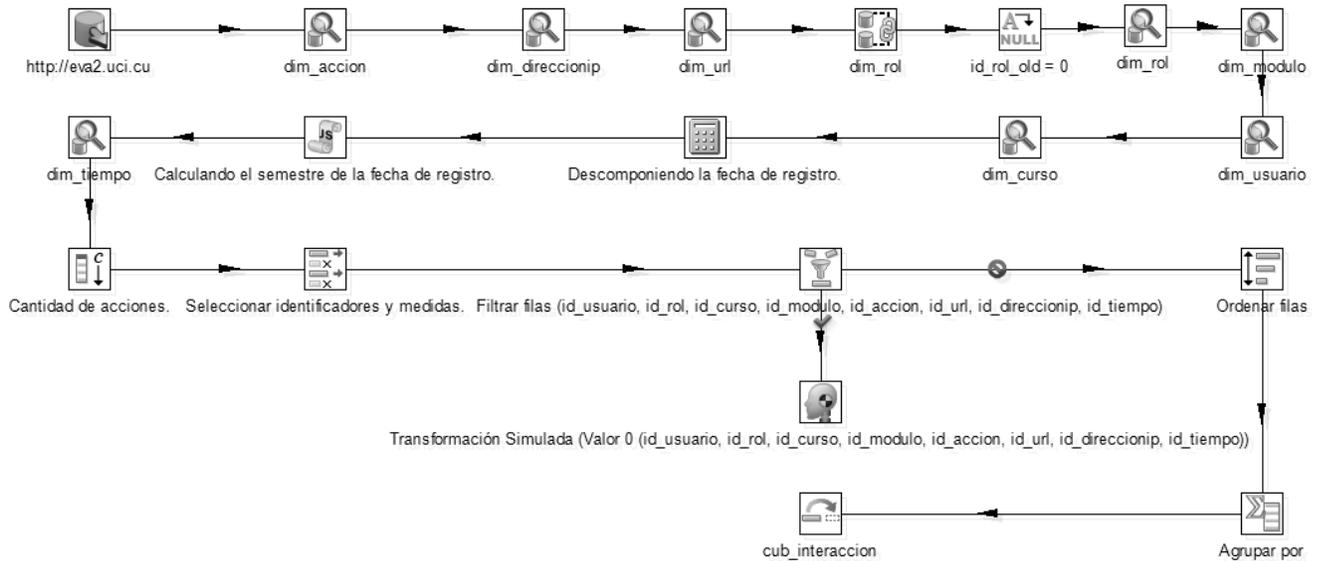


Ilustración 28 Transformación cub_interaccion.

Trabajo trab_dimensiones

Este trabajo es el encargado de ejecutar las transformaciones construidas para llenar las tablas dimensiones del MD. Las transformaciones son ejecutadas en el siguiente orden: dim_accion, dim_curso, dim_direccionip, dim_url, dim_modulo, dim_rol, dim_usuario y dim_tiempo.

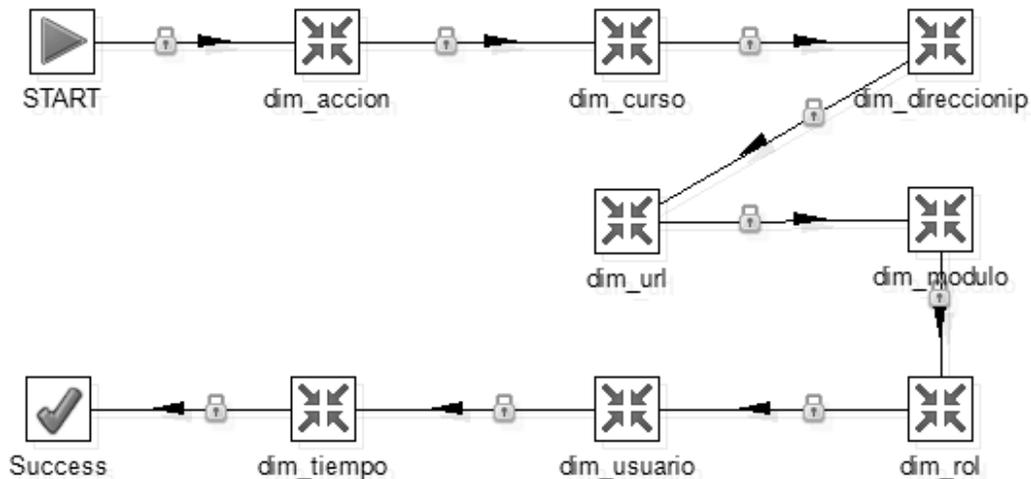


Ilustración 29 Trabajo trab_dimensiones.

Trabajo trab_cubos

Este trabajo es el encargado de ejecutar la transformación construida para llenar la tabla hecho.



Ilustración 30 Trabajo trab_cubos.

Trabajo trab_almacen

Este trabajo es el encargado de ejecutar los trabajos diseñados para llenar las dimensiones y la tabla hecho, en este preciso orden. Las acciones llevadas a cabo para el llenado del MD son:

1. Verificar que existe conexión con la base de datos del sistema fuente, en este caso el EVA.
2. Verificar que existe conexión con el MD de la propuesta de solución.
3. Ejecutar el trabajo trab_dimensiones, para el llenado de las tablas dimensiones.
4. Ejecutar el trabajo trab_cubos, para el llenado de la tabla hecho.
5. En caso de ocurrir algún error el mismo es guardado en un fichero log y se le notifica al usuario mediante un mensaje del sistema, concluyendo de forma fallida el proceso.
6. Al finalizar la ejecución se le muestra al usuario un mensaje indicando que ha sido satisfactorio.

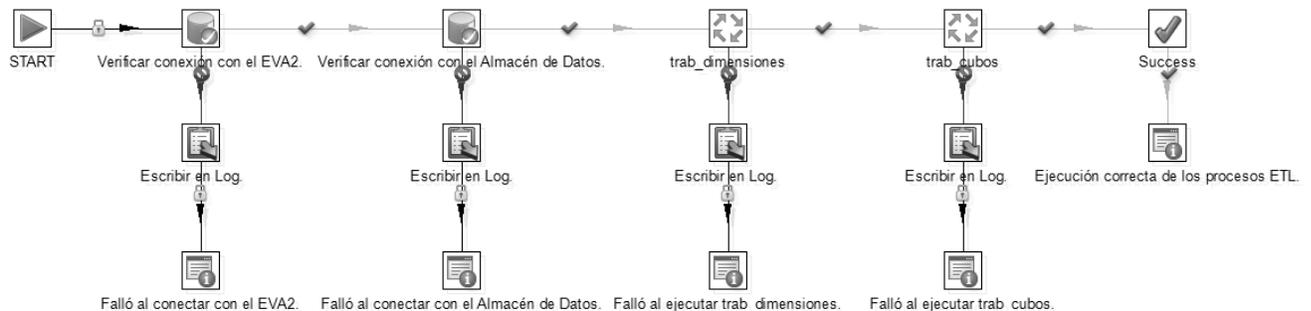


Ilustración 31 Trabajo trab_almacen.

En este epígrafe solo se muestra la descripción de las transformaciones dim_curso, dim_usuario, dim_tiempo y cub_interaccion, además se muestran las descripciones de los trabajos trab_dimensiones, trab_cubos y trab_almacen. Las restantes descripciones se encuentran en los anexos de la investigación, en la sección con igual nombre que este epígrafe.

3.2.3 Automatización del sistema ETL.

Para realizar la carga inicial de los datos en el MD, fue necesaria la implementación de ocho transformaciones para las dimensiones, una transformación para el llenado de la tabla hecho y tres trabajos que serán los encargados de ejecutar de forma automática las transformaciones. Es necesario

además, que en la estación de trabajo en la cual sea desplegada la propuesta de solución se configuren tareas automatizadas con el programador de tareas del sistema operativo, para ejecutar los procesos ETL con cierta frecuencia e incorporar al MD la nueva información generada producto de la interacción de los usuarios con el EVA.

3.3 Conclusiones.

Fue definido el esquema en estrella como el tipo de modelo lógico idóneo, posibilitando realizar un diseño que muestre los mejores tiempos de respuesta al usuario.

Fueron identificadas, diseñadas y establecidas las relaciones entre las tablas dimensiones y hechos, lo cual permitió realizar la implementación del MD en el SGBD seleccionado.

Fue realizado el mapeo lógico, el cual permitió conocer el nombre de la tabla, nombre del campo y el tipo de dato de la información que viaja desde la base de datos fuente hasta el AD.

Se desarrollaron las fases Modelo lógico del MD e Integración de datos correspondientes a la metodología HEFESTO, generando los artefactos correspondientes.

Se implementaron y automatizaron los procesos ETL para realizar la carga inicial del AD.

Capítulo 4: Pruebas de *software* al mercado de datos.

Las pruebas de *software* realizadas al MD de la propuesta de solución verifican que el producto cumple con los requerimientos del cliente y cuenta con la calidad proyectada. Las pruebas realizadas son unitarias, de integración y de aceptación con el cliente. Cada una de ellas se describen en los epígrafes siguientes.

4.1 Pruebas unitarias.

Las pruebas unitarias se enfocan en un programa o un componente que desempeña una función específica, que puede ser probada y que se asegura que funcione tal y como lo define la especificación del programa. Los programadores siempre prueban el código durante el desarrollo, por lo que las pruebas unitarias son realizadas solamente después de que el programador considera que el componente está libre de errores. (Brizeula Figueredo et al. 2013)

Una vez culminado el desarrollo se aplicaron pruebas unitarias a los componentes de integración de datos, almacenamiento y visualización de la información. Durante la validación del componente de integración se detectaron elementos duplicados en la dimensión Módulo. Para solucionar este problema se agregaron a la transformación los componentes Ordenar filas y Filas únicas, los cuales permitieron ordenar los valores de entrada y eliminar los que tenían más de una ocurrencia. Una vez cargados los datos se verificó además, que durante la ejecución de la transformación no se perdiera ninguno.

Durante la validación del componente de visualización se detectaron inicialmente que las cantidades de acciones que se mostraban no coincidían con los que existían en la fuente. Los datos ausentes se debían a la existencia de valores nulos en campos de las tuplas y a la existencia de fechas de registro fuera del rango de fechas que comprendía la dimensión Tiempo. El problema quedó solucionado al verificar que la diferencia existente coincidía con la cantidad de tuplas con valores nulos y aumentando el rango de fechas de la dimensión Tiempo desde 01/01/2013 hasta 31/12/2015 para solucionar el problema de las fechas fuera de rango. Finalmente se comprobó que las cantidades de acciones visualizadas eran correctas, por lo que no había pérdida de la información.

4.2 Pruebas de integración.

Las pruebas de integración tienen como objetivo identificar errores introducidos por la combinación de programas o componentes probados unitariamente, además, verificar que las especificaciones de diseño sean alcanzadas. Componentes individuales son combinados con otros componentes para asegurar que la comunicación, enlaces y los datos compartidos ocurran apropiadamente. No son

verdaderamente pruebas de sistema porque los componentes no están implementados en el ambiente operativo. (Brizeula Figueredo et al. 2013)

Para la ejecución de las pruebas de integración fueron realizadas consultas a la medida de los resultados que se querían obtener de las tablas fuentes, las tablas dimensiones de análisis y la tabla hecho. Esta actividad se realiza con el objetivo de verificar la correcta integración entre todos los componentes a partir de que la información extraída, transformada, almacenada y visualizada no se pierda al pasar de un componente a otro. La diferencia existente (una tupla) entre la base de datos del EVA y el MD es precisamente que a cada tabla dimensión se le agregó una tupla con identificador 0 y valor Desconocido, para evitar que fallen los procesos ETL cuando existan valores nulos.

Tabla 13 Prueba de integración a la dimensión Curso.

	Base de datos EVA	Mercado de datos	Herramienta de análisis
Consulta	SELECT COUNT(DISTINCT shortname) FROM public.mdl_course WHERE shortname IS NOT NULL;	SELECT COUNT(DISTINCT nombre_corto) FROM mod_eva.dim_curso;	SELECT NON EMPTY {Hierarchize ({[Fecha.Fechas].[Fechas]}) ON COLUMNS, NON EMPTY {Hierarchize ({[Curso.Cursos].[Curso].Members})} ON ROWS FROM [Interacción Usuario-EVA]
Resultado	50 / 0.213 milisegundos	50 / 0.197 milisegundos	50 / 1.03 segundos

Tabla 14 Prueba de integración a la dimensión Usuario.

	Base de datos EVA	Mercado de datos	Herramienta de análisis
Consulta	SELECT COUNT(DISTINCT username) FROM public.mdl_user WHERE username IS NOT NULL;	SELECT COUNT(DISTINCT usuario) FROM mod_eva.dim_usuario;	SELECT NON EMPTY {Hierarchize ({[Fecha.Fechas].[Fechas]}) ON COLUMNS, NON EMPTY {Hierarchize ({[Usuario.Usuarios].[Usuario].Members})} ON ROWS FROM [Interacción Usuario-EVA]
Resultado	3969 / 3.433 milisegundos	3969 / 19.768 milisegundos	3969 / 0.58 segundos

Tabla 15 Prueba de integración al hecho Interacción Usuario-EVA.

	Base de datos EVA	Mercado de datos	Herramienta de análisis
Consulta	<pre>SELECT COUNT(*) FROM mdl_log WHERE mdl_log.id IS NOT NULL AND mdl_log."time" IS NOT NULL AND mdl_log.userid IS NOT NULL AND mdl_log.ip IS NOT NULL AND mdl_log.course IS NOT NULL AND mdl_log.module IS NOT NULL AND mdl_log.action IS NOT NULL AND mdl_log.url IS NOT NULL;</pre>	<pre>SELECT COUNT(*) FROM mod_eva.cub_interacc ion;</pre>	<pre>SELECT NON EMPTY {Hierarchize ({[Fecha.Fechas].[Fech as]})} ON COLUMNS, NON EMPTY {Hierarchize ({[Curso.Cursos].[Curs o].Members})} ON ROWS FROM [Interacción Usuario-EVA]</pre>
Resultado	662978 / 184.062 milisegundos	662978 / 157.434 milisegundos	662.978 / 0.94 segundos

En este epígrafe solo se muestran los resultados de las pruebas de integración realizadas a las dimensiones Curso y Usuario, además se presenta el resultado de dicha prueba a la tabla de hechos Interacción Usuario-EVA. Los restantes resultados se encuentran en los anexos de la investigación, en la sección con igual nombre que este epígrafe.

4.3 Pruebas de aceptación.

Las pruebas de aceptación son realizadas principalmente por los usuarios con el apoyo del equipo del proyecto. El propósito es confirmar que el sistema está terminado, que desarrolla puntualmente las necesidades de la organización y que es aceptado por los usuarios finales. (Brizuela Figueredo et al. 2013)

Para la realización de las pruebas de aceptación se hizo necesario configurar un entorno operable para explotar la capacidad de análisis del MD y verificar que cumple con las necesidades del cliente. Para ello primeramente se hizo necesario crear un fichero XML con la configuración necesaria para que el servidor OLAP Mondrian, el cual es parte de la suite Pentaho Open Source Business Intelligence, interpretara el MD que fue implementado en un SGBD Relacional como una base de datos

multidimensional. Para la creación del XML que contiene el cubo de información³⁶ denominado Interacción Usuario-EVA, fue utilizada la herramienta Pentaho Schema Workbench en su versión 3.8.

```

<Schema name="DwhEva">
  <Dimension type="StandardDimension" visible="true" highCardinality="false" name="Usuario">
  <Dimension type="StandardDimension" visible="true" highCardinality="false" name="Rol">
  <Dimension type="StandardDimension" visible="true" highCardinality="false" name="Curso">
  <Dimension type="StandardDimension" visible="true" highCardinality="false" name="Módulo">
  <Dimension type="StandardDimension" visible="true" highCardinality="false" name="Acción">
  <Dimension type="StandardDimension" visible="true" name="URL">
  <Dimension type="StandardDimension" visible="true" highCardinality="false" name="Dirección IP">
  <Dimension type="TimeDimension" visible="true" highCardinality="false" name="Fecha">
  <Cube name="Interacción Usuario-EVA" visible="true" cache="true" enabled="true">
</Schema>
    
```

Ilustración 32 XML Mondrian del MD de la propuesta de solución.

Posteriormente se hizo necesario configurar la conexión al mercado de datos en la herramienta de análisis SAIKU en su versión 2.5, la cual en forma de complemento se puede integrar con la suite Pentaho Open Source Business Intelligence.

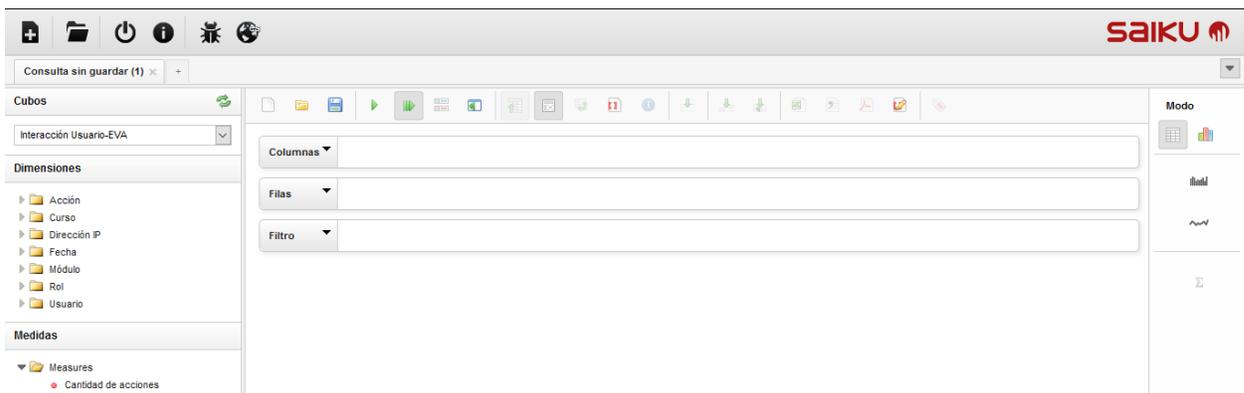


Ilustración 33 Herramienta de análisis SAIKU.

A continuación se muestra una serie de análisis gráficos y tabulares configurados en SAIKU que tienen cierta capacidad de análisis a partir de la información generada por el EVA y que es producto de la interacción de los usuarios con la plataforma.

³⁶ Un cubo de información o cubo OLAP, es una estructura de datos que supera las limitaciones de las bases de datos relacionales y proporciona un análisis rápido de los datos. Pueden mostrar y sumar grandes cantidades de datos, a la vez que proporcionan a los usuarios acceso mediante búsqueda a los puntos de datos. De este modo, los datos se pueden resumir o reorganizar según sea necesario, para procesar la variedad más amplia de preguntas pertinentes al área de interés de un usuario. (Microsoft Corporation 2016)

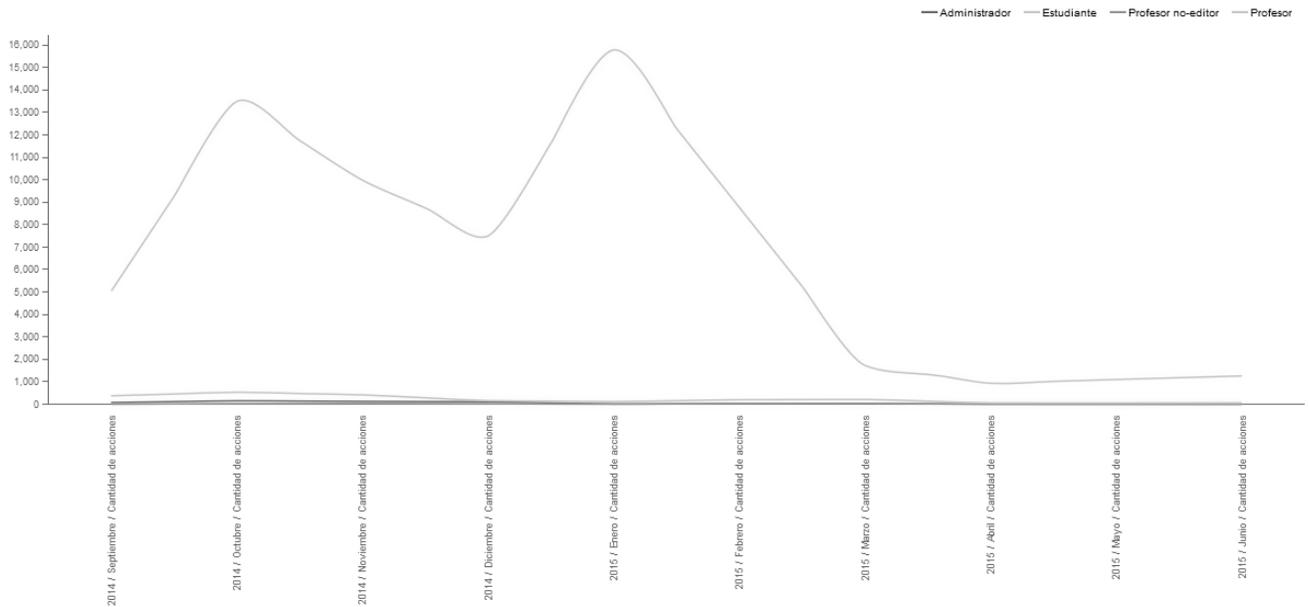


Ilustración 34 Cantidad de acciones por Rol, Año y Mes en el período lectivo 2014-2015 en el EVA.

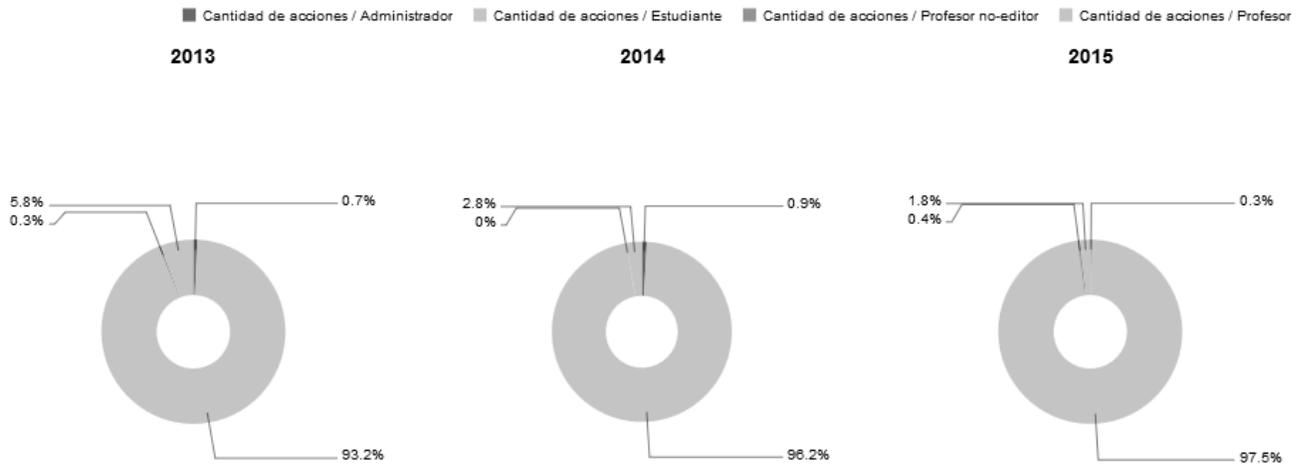


Ilustración 35 Cantidad de acciones por Rol y Año en el período lectivo 2014-2015 en el EVA.

■ Cantidad de acciones / Administrador ■ Cantidad de acciones / Estudiante ■ Cantidad de acciones / Profesor no-editor ■ Cantidad de acciones / Profesor

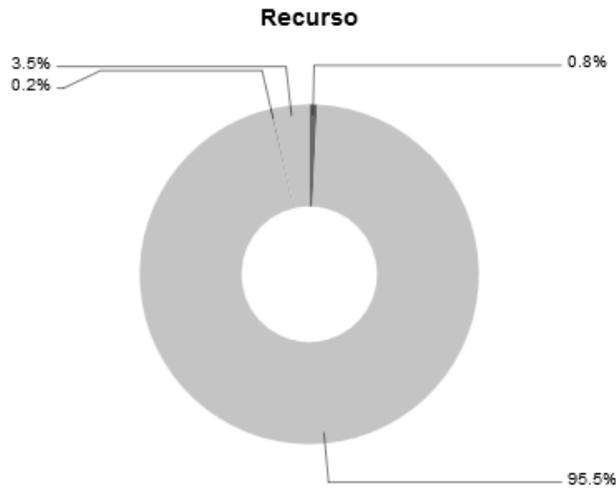


Ilustración 36 Cantidad de acciones por Rol y Módulo en el EVA (En porciento).

Curso	URL	Denominación del objeto	Cantidad de acciones
IA2	view.php?id=1257	Conferencia 7 Introducción al KDD	2.102
	view.php?id=371	Conferencia 1 Introducción a los Sistemas Basados en el Conocimiento	1.917
	view.php?id=607	Conferencia 2 Formas de Representación del Conocimiento	1.817
	view.php?id=1278	Weka 3.7.10	1.800
	view.php?id=893	Conferencia 4 Tratamiento de Incertidumbre	1.681
	view.php?id=1015	Conferencia 5 Lógica difusa	1.658
	view.php?id=1298	Conferencia 8 Árboles de decisión	1.580
	view.php?id=746	Conferencia 3 Sistemas basados en reglas	1.549
	view.php?id=1523	Conferencia 10 Introducción a las Redes Neuronales Artificiales	1.545
	view.php?id=1103	Conferencia 6 Sistemas basados en Casos	1.515
	view.php?id=1540	Conferencia 11 Modelos de RNA	1.505
	view.php?id=1587	Guía de ejercicios	1.462
	view.php?id=1496	Conferencia 9 Algoritmos de clustering	1.450

Ilustración 37 Cantidad de acciones por Curso y URL en el período lectivo 2014-2015 en el EVA.

Usuario	Descripción	Año	Mes	Curso	Acción	Cantidad de acciones
Alfamas	Estudiante	2013	Septiembre	IA2	Ver	22
			Octubre	IA2	Ver	2
			Noviembre	IA2	Ver	18
			Diciembre	IA2	Ver	3
		2014	Enero	IA2	Ver	46
			Febrero	IA2	Ver	14

Ilustración 38 Cantidad de acciones por Usuario, Rol, Año, Mes, Curso y Acción en el período lectivo 2014-2015 en el EVA.

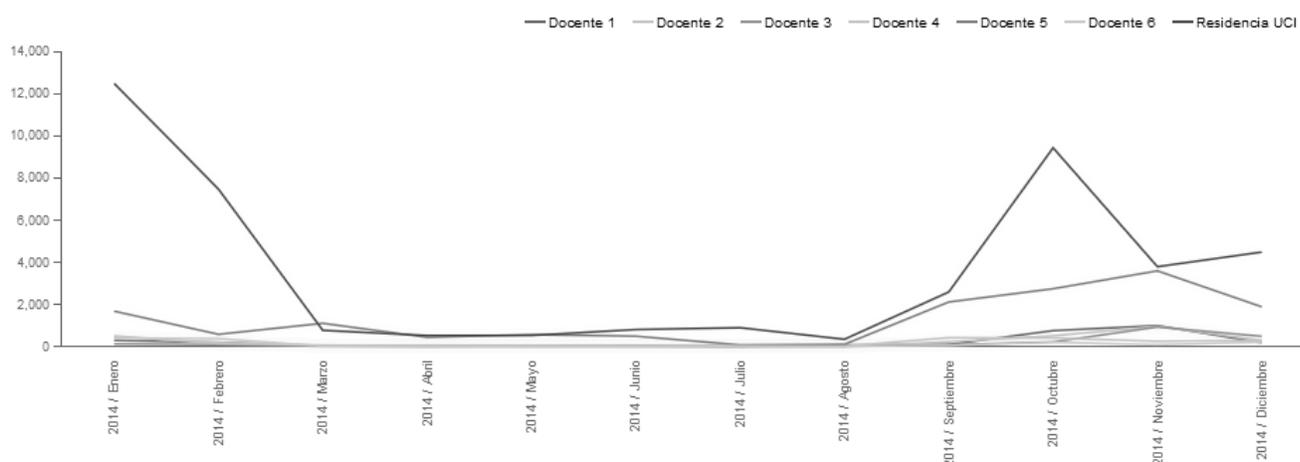


Ilustración 39 Cantidad de acciones por Zona IP, Año y Mes, en el año 2014 en el EVA.

En los anexos de la investigación, en la sección con igual nombre que este epígrafe, se encuentran otros gráficos que brindan capacidad de análisis sobre la interacción de los usuarios de la universidad con el EVA.

4.4 Análisis de los resultados de las pruebas.

A continuación, se muestran las cantidades de no conformidades detectadas durante la ejecución del proceso de pruebas, para el cual se realizaron 4 iteraciones.

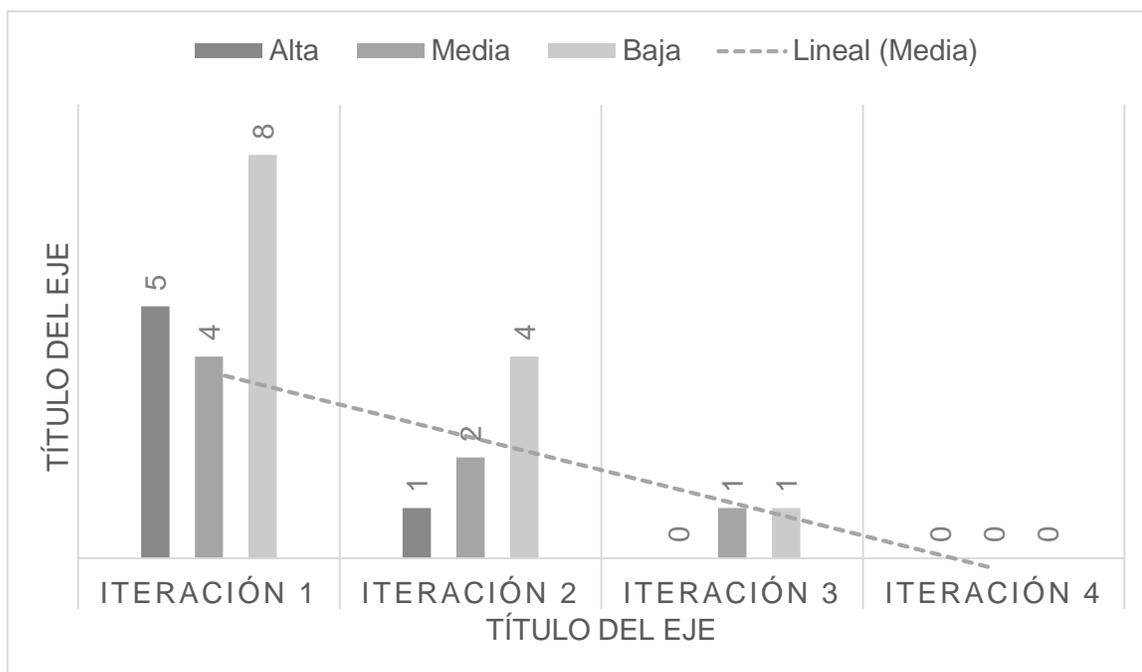


Ilustración 40 Cantidades de no conformidades detectadas por iteración.

La siguiente tabla muestra los tipos de no conformidades detectadas al MD de la propuesta de solución, agrupadas de acuerdo al nivel de complejidad.

Tabla 16 Tipos de no conformidades detectadas por nivel de complejidad.

No conformidades de complejidad Alta	No conformidades de complejidad Media	No conformidades de complejidad Baja
<ul style="list-style-type: none"> -No presentación de dimensiones en la herramienta de análisis. -Selección errónea de la función de agregación prevista para la sumarización de los datos en el XML Mondrian. -No presentación de niveles en dimensiones de análisis. -Errores en llaves foráneas y primarias. 	<ul style="list-style-type: none"> -Longitud de campos de tipo cadena de texto ínfimas. -Longitud de campos de tipo numérico ínfimas. -Valores duplicados en dimensiones de análisis. -Valores nulos en las dimensiones de análisis y en la tabla de hechos. 	<ul style="list-style-type: none"> -Errores ortográficos en la presentación de contenidos en la herramienta de análisis. -Errores de formato en la presentación de contenidos en la herramienta de análisis. -Errores ortográficos en la documentación. -Errores de formato en la documentación.

-Errores en el XML de Mondrian para la visualización del MD.		
--	--	--

Al concluir el proceso de pruebas de *software* al MD, todas las no conformidades fueron resueltas, lo cual valida en cierto grado la calidad de la propuesta de solución, logrando una mayor satisfacción por parte del cliente.

4.5 Conclusiones.

La realización de las pruebas unitarias a los componentes de integración, almacenamiento y visualización permitió verificar su correcto funcionamiento.

Las pruebas de integración permitieron constatar una correcta integración entre los componentes de integración, almacenamiento y visualización al no existir pérdida de información.

Las pruebas de aceptación permitieron al cliente verificar la completitud y calidad de la propuesta de solución.

Durante el proceso de pruebas de *software* al MD de la propuesta de solución fueron detectadas seis no conformidades de complejidad alta, siete de complejidad media y 13 de complejidad baja, al finalizar todas las no conformidades fueron resueltas.

Conclusiones generales

La culminación de la presente investigación permitió arribar a las siguientes conclusiones:

1. Al investigar el objeto de estudio, los métodos científicos empleados posibilitaron identificar los conceptos y teorías que sustentan la investigación determinando construir una solución de AA mediante el desarrollo de un MD, utilizando las herramientas VP-UML CE, PostgreSQL y Pentaho Open Source Business Intelligence.
2. Se realizó un análisis sobre las metodologías de desarrollo de almacenes de datos donde fue seleccionada HEFESTO como la más apropiada, la cual permitió cumplir con el objetivo general de la investigación y satisfacer de esta forma las necesidades del cliente.
3. Se realizó un estudio de diferentes soluciones existentes que utilizan los almacenes de datos como solución tanto en el sector empresarial como en el E-Learning, siendo soluciones parciales o totales construidas a la medida del problema por el cual surgieron.
4. Se diseñó e implementó un MD para la toma de decisiones a partir de la información gestionada por el EVA de la UCI, enfocada a mejorar el uso del EVA como componente del PEA de la institución.
5. La validación del MD de la propuesta de solución por medio de pruebas unitarias, de integración y de aceptación demostró la eficacia del proceso de desarrollo llevado a cabo, la satisfacción del cliente y además, comprobó el correcto funcionamiento de la solución propuesta.

Recomendaciones

Tomando como base la investigación realizada y el análisis de los resultados obtenidos se recomienda:

1. Poner a disposición de la comunidad internacional del LMS MOODLE los resultados obtenidos en esta investigación, con tal de contribuir al proceso de toma de decisiones sobre el uso por parte de los usuarios de esta plataforma.
2. Incluir otros aspectos de interés, los cuales requieran ser analizados con carácter variable en el tiempo y así conocer su comportamiento histórico.

Bibliografía

- Adkison, Dustin. 2013. IBM Cognos Business Intelligence. Packt Publishing Ltd. https://books.google.com/books?hl=es&lr=lang_es|lang_en&id=z5w62yBw9_YC&oi=fnd&pg=PT11&dq=IBM+Cognos+Business+Intelligence+&ots=Lu_Fg6mbg9&sig=yxJKze_XQ97utnYwDDCVUD91Nrk.
- Azua, Sabin. 2007. "La responsabilidad social, corporativa como refuerzo de la estrategia y la competitividad empresarial." Revista Escuela de Administración de Negocios. <http://redalyc.org/articulo.oa?id=20611495007>.
- Bernabeu, R. D. 2009. HEFESTO, Metodología Propia Para La Construcción de Un DataWarehouse. Córdoba-Argentina.
- Bouman, Roland, and Jos van Dongen. 2010. Pentaho Solutions: Business Intelligence and Data Warehousing with Pentaho and MySQL. John Wiley & Sons.
- Brizeula Figueredo, Clara, Clara Elena Brizeula Figueredo, Claudia García Suárez del Villar, and Julio Cesar Brito Rodríguez. 2013. "Mercado de Datos Para El Área de Planeación de La Empresa de Mantenimiento de Grupos Electrónicos Fuel-Oil." Serie Científica-Universidad de Las Ciencias Informáticas 6 (4). <http://publicaciones.uci.cu/index.php/SC/article/view/1165>.
- Cabrera González, Lianet, and Enrique Roberto Pompa Torres. 2013. "Extensión de Visual Paradigm for UML para el Desarrollo Dirigido por Modelos de aplicaciones de gestión de información.," May. http://repositorio_institucional.uci.cu/jspui/handle/ident/JCE-2012-F323-P164-Ponencia-2777.
- Castellón Martín., Ovidio José. 2016. "Almacén de datos para el Entorno Virtual de Aprendizaje.," January. http://repositorio_institucional.uci.cu/jspui/handle/ident/9028.
- Clark, Ruth C., and Richard E. Mayer. 2016. E-Learning and the Science of Instruction: Proven Guidelines for Consumers and Designers of Multimedia Learning. John Wiley & Sons. https://books.google.es/books?hl=es&lr=&id=v1uzCgAAQBAJ&oi=fnd&pg=PR17&dq=DEFINE+E-LEARNING&ots=TLzKhKbLhi&sig=Ofw-GZhtVi_058PXt9_LwKizU_I.
- Corey, Michael J., and Michael Abbey. 1996. Oracle Data Warehousing. Berkeley, CA, USA: Osborne/McGraw-Hill.
- Devlin, Barry. 1997. Data Warehouse: From Architecture to Implementation. Addison-Wesley.
- García Alcázar, Laura. 2015. "Explotación de analíticas de aprendizaje en entornos educativos online," July. <https://repositorio.uam.es/handle/10486/668918>.
- Geraci, Anne, Freny Katki, Louise McMonegal, Bennett Meyer, John Lane, Paul Wilson, Jane Radatz, Mary Yee, Hugh Porteous, and Fredrick Springsteel. 1991. IEEE Standard Computer Dictionary:

- Compilation of IEEE Standard Computer Glossaries. IEEE Press.
<http://dl.acm.org/citation.cfm?id=574566>.
- Gómez-Aguilar, Diego-Alonso, Francisco-José García-Peñalvo, and Roberto Therón. 2014. "Análítica visual en E-Learning." *El Profesional de la Información* 23 (3): 236–45. doi:10.3145/epi.2014.may.03.
- González, D. J. 1982. "La Motivación." UNA ORIENTACIÓN PARA SU ESTUDIO. Editorial Científico-Técnica. Ciudad de La Habana.
<http://www.profesores.elo.utfsm.cl/~agv/elo329/1s06/lectures/Motivacion.pdf>.
- Heurtel, Olivier. 2009. Oracle 11g: Administración. Ediciones ENI.
- Inmon, W. H. 2005. Building the Data Warehouse. John Wiley & Sons.
- Kimball, Ralph. 2008. The Data Warehouse Lifecycle Toolkit. John Wiley & Sons.
https://books.google.com/books?hl=es&lr=&id=XaUV6r2Xy0IC&oi=fnd&pg=PP1&dq=The+Data+warehouse+Lifecycle+Toolkit+&ots=H-u2JRXnxG&sig=32ezSxdySJ_Ai-M_9-VoC49YoOs.
- Kimball, Ralph, and Margy Ross. 2011. The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling. John Wiley & Sons.
- Ledesma Rodríguez, Yenisleydis, Yunier Boza Roget, Armando Robert Lobo, Sergio García De La Puente, and Marleysi López Duque. 2011. "Extensión de La Herramienta Visual Paradigm for UML Para El Soporte Al Desarrollo Dirigido Por Modelos Con Ext JS." http://repositorio_institucional.uci.cu//jspui/handle/ident/TD_04358_11.
- Llombart, A., and Business Intelligence. 2003. "BI: Inteligencia Aplicada Al Negocio." DAA Contenidos Digitales, CMS-Spain. com[En Línea]. Madrid, Disponible En: [Http://www. Eldiarioexterior. Com/conocimiento/docs](Http://www.Eldiarioexterior.Com/conocimiento/docs).
http://www.eldiarioexterior.com/anterior/conocimiento/docs/BI_Inteligencia_aplicada_al_negocio.pdf.
- Marrero Pérez, Lisvet, and Pedro Pablo Ramos Pajón. 2015. "Mercado de Datos para la toma de decisiones a partir de la información de los estudiantes generada por el Sistema de Gestión Universitaria y el Entorno Virtual de Enseñanza Aprendizaje de la Universidad de las Ciencias Informáticas." La Habana, Cuba: Universidad de las Ciencias Informáticas.
- Martínez, Antonio Saorín. 2012. "MOODLE 2.0-MANUAL DEL PROFESOR." Con URL: [Http://www. Inap. es/alfresco/d/d/workspace/SpacesStore/22597ae0-96ea-4d89-833b-6a11b6d902a6/manual_profesor. Pdf](Http://www.Inap.es/alfresco/d/d/workspace/SpacesStore/22597ae0-96ea-4d89-833b-6a11b6d902a6/manual_profesor.Pdf) [recuperado El 04/03/2016].
http://www.inap.es/alfresco/d/d/workspace/SpacesStore/22597ae0-96ea-4d89-833b-6a11b6d902a6/manual_profesor.pdf.

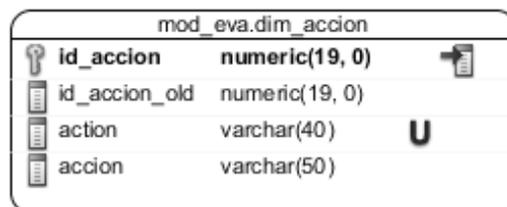
- Martínez, Rafael. 2016. "Sobre PostgreSQL | Wwv.postgresql.org.es." Accessed April 12. http://www.postgresql.org/es/sobre_postgresql.
- Microsoft Corporation. 2016. "Información sobre los cubos OLAP." Microsoft System Center. Accessed May 29. [https://technet.microsoft.com/es-es/library/hh916543\(v=sc.12\).aspx](https://technet.microsoft.com/es-es/library/hh916543(v=sc.12).aspx).
- Moore, Michael. 2013. "Desire2Learn Insights - Achievement Reporting (Outcomes Assessment)." October 10. <http://es.slideshare.net/D2LBarry/desire2learn-insights-achievement-reporting-outcomes-assessment>.
- Narváez, Maldonado, and José Miguel. 2010. "Sistemas operativos en moodle: módulo de seguimiento de usuarios." <http://dspace.utpl.edu.ec/handle/123456789/1839>.
- Oracle Corporation and/or its affiliates. 2016. "MySQL :: MySQL 5.7 Reference Manual." Accessed April 11. <http://dev.mysql.com/doc/refman/5.7/en/>.
- Peña, Alejandro. 2006. "Inteligencia de Negocios: Una Propuesta Para Su Desarrollo En Las Organizaciones." Instituto Politécnico Nacional, Dirección de Publicaciones, México.
- Peñalvo, Francisco José García. 2005. "Estado actual de los sistemas E-Learning." Teoría de la Educación. Educación y Cultura en la Sociedad de la Información. <http://www.redalyc.org/articulo.oa?id=201021055001>.
- Petkovic, Dusan. 2008. MICROSOFT SQL SERVER 2008 A BEGINNER'S GUIDE 4/E: A Beginner's Guide. McGraw Hill Professional.
- Piedrabuena, Federico, and Gustavo Vázquez. 2005. "Relevamiento :diseño físico de sistemas OLAP." <https://www.colibri.udelar.edu.uy/handle/123456789/3518>.
- Quatrani, Terry. 2003. Visual Modeling with Rational Rose 2002 and UML. Addison-Wesley Professional.
- Ramón Cueto, A., and J. Díaz García. 2009. "Implementación de Un Data Warehouse Para El Control Del Recurso Humano de La Salud [tesis]." Universidad de Las Ciencias Informáticas (UCI).
- Rogers, Pressman. 2005. "Ingeniería de Software Un Enfoque Práctico." Editorial McGraw-Hill, Madrid.
- Russell, Jesse, and Ronald Cohn. 2012. Oracle Business Intelligence Suite Enterprise Edition. Book on Demand.
- University, MicroStrategy. 2013. Implementing MicroStrategy. MicroStrategy Inc.
- Vázquez Ortiz, Yudisney, and Pedro Yobanis Piñero Pérez. 2013. "Estrategia para la obtención de un gestor de bases de datos cubano.," October. http://repositorio_institucional.uci.cu//jspui/handle/ident/7996.
- Vitier Urquizu, Ramel. 2014. "Almacén de datos operacional para contribuir a la toma de decisiones basado en el uso de los recursos de la Plataforma Educativa ZERA," October. http://repositorio_institucional.uci.cu//jspui/handle/ident/8155.

Anexos

Anexo # 1 Identificación de dimensiones y hechos.

Dimensión Acción

La dimensión Acción está representada en el modelo de datos del MD por la tabla dim_accion, la misma está compuesta por cuatro atributos. El primero (id_accion) funciona como identificador de la tabla, es de tipo numérico con precisión 19, escala 0 y autoincremental. El segundo (id_accion_old) constituye el identificador de la acción en la tabla fuente, es de tipo numérico con precisión 19 y escala 0. El tercero (action) representa la denominación de la acción en idioma inglés (Idioma en el que se encuentra la base de datos de MOODLE), es de tipo cadena de texto con longitud 40, su valor es único y es utilizado para realizar búsquedas sobre esta dimensión. El cuarto constituye la denominación de la acción en idioma español, es de tipo cadena de texto con longitud 50. La tabla posee un índice sobre el atributo identificador para optimizar las recuperaciones sobre este.



mod_eva.dim_accion			
	id_accion	numeric(19, 0)	
	id_accion_old	numeric(19, 0)	
	action	varchar(40)	U
	accion	varchar(50)	

Ilustración 41 Tabla dimensión dim_accion.

Dimensión Dirección IP

La dimensión Dirección IP está representada en el modelo de datos del MD por la tabla dim_direcciónip, la misma está compuesta por cuatro atributos. El primero (id_direccionip) funciona como identificador de la tabla, es de tipo numérico con precisión 19, escala 0 y autoincremental. El segundo (id_direccionip_old) constituye el identificador de la dirección ip en la tabla fuente, es de tipo numérico con precisión 19 y escala 0. El tercero (direccionip) representa las direcciones ip desde las cuales acceden los usuarios al EVA, es de tipo cadena de texto con longitud 45, su valor es único y es utilizado para realizar búsquedas sobre esta dimensión. El cuarto (zona_ip) constituye la zona de la UCI a la cual corresponde dicha dirección ip, es de tipo cadena de texto con longitud 20. La tabla posee un índice sobre el atributo identificador para optimizar las recuperaciones sobre este.

mod_eva.dim_direccionip			
	id_direccionip	numeric(19, 0)	
	id_direccionip_old	numeric(19, 0)	
	direccionip	varchar(45)	U
	zona_ip	varchar(20)	

Ilustración 42 Tabla dimensión dim_direccionip.

Dimensión Módulo

La dimensión Módulo está representada en el modelo de datos del MD por la tabla dim_modulo, la misma está compuesta por cuatro atributos. El primero (id_modulo) funciona como identificador de la tabla, es de tipo numérico con precisión 19, escala 0 y autoincremental. El segundo (id_modulo_old) constituye el identificador del módulo en la tabla fuente, es de tipo numérico con precisión 19 y escala 0. El tercero (module) constituye la denominación del módulo en idioma inglés (Idioma en el que se encuentra la base de datos de MOODLE), es de tipo cadena de texto con longitud 20, su valor es único y es utilizado para realizar búsquedas sobre esta dimensión. El cuarto (modulo) representa la denominación que reciben los módulos del LMS MOODLE en idioma español, es de tipo cadena de texto con longitud 20. La tabla posee un índice sobre el atributo identificador para optimizar las recuperaciones sobre este.

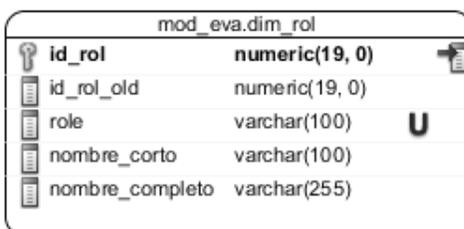
mod_eva.dim_modulo			
	id_modulo	numeric(19, 0)	
	id_modulo_old	numeric(19, 0)	
	module	varchar(20)	U
	modulo	varchar(20)	

Ilustración 43 Tabla dimensión dim_modulo.

Dimensión Rol

La dimensión Rol está representada en el modelo de datos del MD por la tabla dim_rol, la misma está compuesta por cinco atributos. El primero (id_rol) funciona como identificador de la tabla, es de tipo numérico con precisión 19, escala 0 y autoincremental. El segundo (id_rol_old) constituye el identificador del rol en la tabla fuente, es de tipo numérico con precisión 19 y escala 0. El tercero (role) representa la denominación del rol en idioma inglés (Idioma en el que se encuentra la base de datos de MOODLE), es de tipo cadena de texto con longitud 100, su valor es único y es utilizado para realizar búsquedas sobre esta dimensión. El cuarto (nombre_corto) representa la denominación abreviada del rol en idioma español, es de tipo cadena de texto con longitud 100. El quinto (nombre_completo)

constituye el nombre completo del rol, es de tipo cadena de texto con longitud 255. La tabla posee un índice sobre el atributo identificador para optimizar las recuperaciones sobre este.

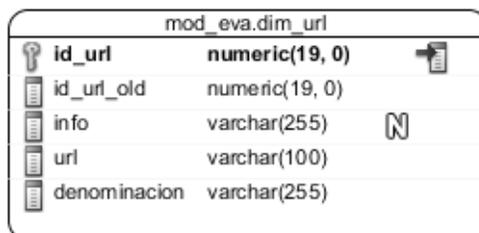


mod_eva.dim_rol	
id_rol	numeric(19, 0)
id_rol_old	numeric(19, 0)
role	varchar(100)
nombre_corto	varchar(100)
nombre_completo	varchar(255)

Ilustración 44 Tabla dimensión dim_rol.

Dimensión URL

La dimensión URL está representada en el modelo de datos del MD por la tabla dim_url, la misma está compuesta por cinco atributos. El primero (id_url) funciona como identificador de la tabla, es de tipo numérico con precisión 19, escala 0 y autoincremental. El segundo (id_url_old) constituye el identificador de la URL en la tabla fuente, es de tipo numérico con precisión 19 y escala 0. El tercero (info) representa la información asociada al log que contiene la URL, es de tipo cadena de texto con longitud 255. El cuarto (url) representa las URL de la plataforma visitadas por los usuarios, es de tipo cadena de texto con longitud 100. Los campos url e info son utilizados para realizar búsquedas sobre esta dimensión. El quinto (denominacion) constituye la denominación del objeto relacionado con la URL (curso, recurso, actividad, ...), es de tipo cadena de texto con longitud 255. La tabla posee un índice sobre el atributo identificador para optimizar las recuperaciones sobre este.



mod_eva.dim_url	
id_url	numeric(19, 0)
id_url_old	numeric(19, 0)
info	varchar(255)
url	varchar(100)
denominacion	varchar(255)

Ilustración 45 Tabla dimensión dim_url.

Anexo # 2 Mapeo de datos Fuente-A-Destino

Tabla 17. Mapeo lógico de los datos necesarios para cargar el MD de la propuesta de solución.

Fuente de datos	Destino

Denominación de la fuente de datos: Base de Datos del EVA.			Denominación del destino de los datos: MD de la propuesta de solución.		
Tabla	Campo	Tipo de dato	Tabla	Campo	Tipo de dato
			dim_accion	id_accion	numeric (19,0)
mdl_log	id	bigserial	dim_accion	id_accion_old	numeric (19,0)
mdl_log	action	character varying(40)	dim_accion	action	character varying(40)
			dim_accion	accion	character varying(50)
			dim_direccioni p	id_direccionip	numeric(19,0)
mdl_course	id	bigserial	dim_direccioni p	id_direccionip _old	numeric(19,0)
mdl_log	ip	character varying(45)	dim_direccioni p	direccionip	character varying(45)
			dim_direccioni p	zona_ip	character varying(20)
			dim_modulo	id_modulo	numeric(19,0)
mdl_modules	id	bigserial	dim_modulo	id_modulo_ol d	numeric(19,0)
mdl_log	id	bigserial	dim_modulo	id_modulo_ol d	numeric(19,0)
mdl_log	module	character varying(20)	dim_modulo	module	character varying(20)
mdl_modules	name	character varying(20)	dim_modulo	module	character varying(20)

			dim_modulo	modulo	character varying(20)
			dim_rol	id_rol	numeric(19,0)
mdl_role	id	bigserial	dim_rol	id_rol_old	numeric(19,0)
mdl_role	shortname	character varying(100)	dim_rol	role	character varying(100)
			dim_rol	nombre_cort o	character varying(100)
mdl_role	name	character varying(255)	dim_rol	nombre_com pleto	character varying(255)
			dim_url	id_url	numeric(19,0)
mdl_log	id	bigserial	dim_url	id_url_old	numeric(19,0)
mdl_log	info	character varying(255)	dim_url	info	character varying(255)
mdl_log	url	character varying(100)	dim_url	url	character varying(100)
mdl_resource	name	character varying(255)	dim_url	denominacio n	character varying(255)
mdl_assign	name	character varying(255)	dim_url	denominacio n	character varying(255)
mdl_book	name	character varying(255)	dim_url	denominacio n	character varying(255)
mdl_course	fullname	character varying(254)	dim_url	denominacio n	character varying(255)
mdl_data	name	character varying(255)	dim_url	denominacio n	character varying(255)

mdl_forum_discussions	name	character varying(255)	dim_url	denominacion	character varying(255)
mdl_folder	name	character varying(255)	dim_url	denominacion	character varying(255)
mdl_forum	name	character varying(255)	dim_url	denominacion	character varying(255)
mdl_glossary_entries	concept	character varying(255)	dim_url	denominacion	character varying(255)
mdl_imsdp	name	character varying(255)	dim_url	denominacion	character varying(255)
mdl_label	name	character varying(255)	dim_url	denominacion	character varying(255)
mdl_lesson_pages	title	character varying(255)	dim_url	denominacion	character varying(255)
mdl_lti	name	character varying(255)	dim_url	denominacion	character varying(255)
mdl_message	subject	character varying(255)	dim_url	denominacion	character varying(255)
mdl_page	name	character varying(255)	dim_url	denominacion	character varying(255)
mdl_quiz	name	character varying(255)	dim_url	denominacion	character varying(255)
mdl_scorm	name	character varying(255)	dim_url	denominacion	character varying(255)
mdl_survey	name	character varying(255)	dim_url	denominacion	character varying(255)

mdl_url	name	character varying(255)	dim_url	denominacio n	character varying(255)
mdl_user	firstname, lastname	character varying(200)	dim_url	denominacio n	character varying(255)
mdl_wiki	name	character varying(255)	dim_url	denominacio n	character varying(255)
mdl_workshop	name	character varying(255)	dim_url	denominacio n	character varying(255)

Anexo # 3 Carga inicial de los datos y procesos ETL.

Transformación dim_accion

Esta transformación es la encargada de extraer las distintas denominaciones que utiliza MOODLE para las acciones que pueden generarse sobre el sistema, las mismas son extraídas de la tabla mdl_log y almacenadas en la tabla dim_accion del MD. Los datos extraídos de la fuente son filtrados para eliminar los valores duplicados y nulos, además se realiza la traducción de estos valores al idioma español y se mantiene solo el primero de los caracteres de cada denominación en mayúscula para lograr una mejor presentación visual de los datos.

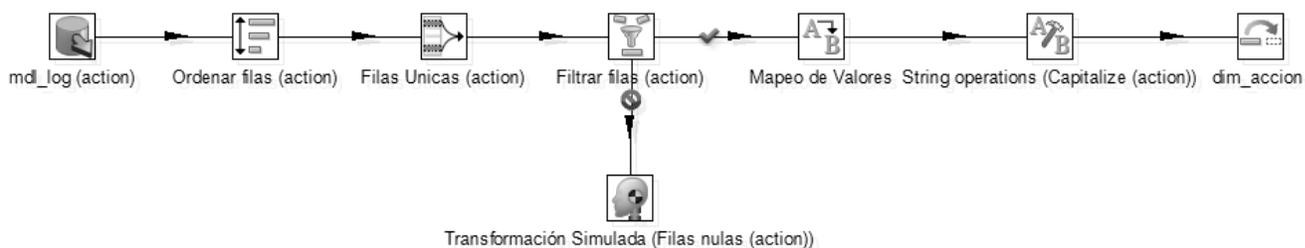


Ilustración 46 Transformación dim_accion.

Transformación dim_direccionip

Esta transformación es la encargada de extraer las distintas direcciones ip desde las cuales se accede al EVA, las mismas son extraídas de la tabla mdl_log y almacenadas en la tabla dim_direccionip del MD. Los datos extraídos de la fuente son filtrados para eliminar valores duplicados y nulos, además por cada dirección ip es determinada su zona ip (Residencia UCI, Docente 1, Docente 2, Docente 3, Docente 4, Docente 5, Docente 6).

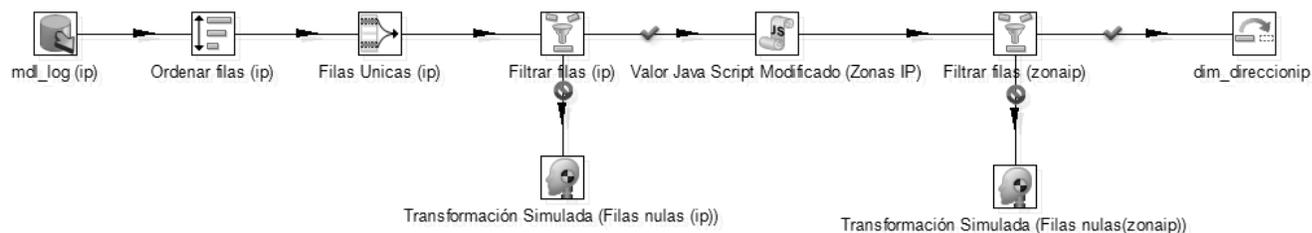


Ilustración 47 Transformación dim_direccionip.

Transformación dim_modulo

Esta transformación es la encargada de extraer los módulos con que cuenta MOODLE, los mismos son extraídos de las tablas mdl_modules y mdl_log, y almacenados en la tabla dim_modulo del MD. Los datos extraídos de la fuente son filtrados para eliminar valores nulos y duplicados, además se realiza la traducción de estos valores al idioma español y se mantiene solo el primero de los caracteres de cada denominación en mayúscula para lograr una mejor presentación visual de los datos.

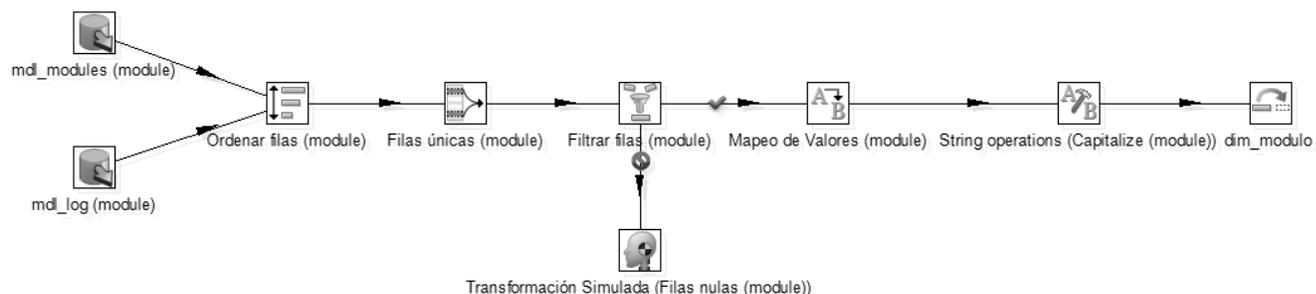


Ilustración 48 Transformación dim_modulo.

Transformación dim_rol

Esta transformación es la encargada de extraer los roles que utiliza MOODLE para permitir el acceso de los usuarios a las distintas funcionalidades con que cuenta la plataforma, los mismos son extraídos de la tabla mdl_role y almacenados en la tabla dim_rol del MD. Los datos extraídos de la fuente son filtrados para eliminar los valores nulos, además se realiza la traducción de estos valores al idioma español.

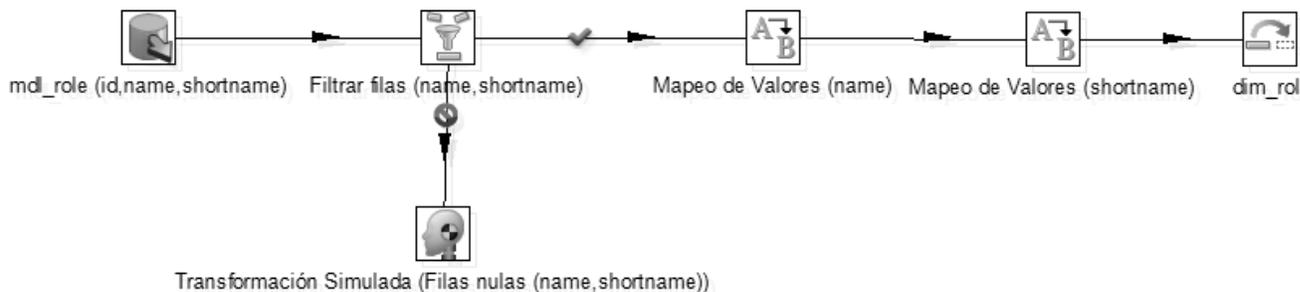
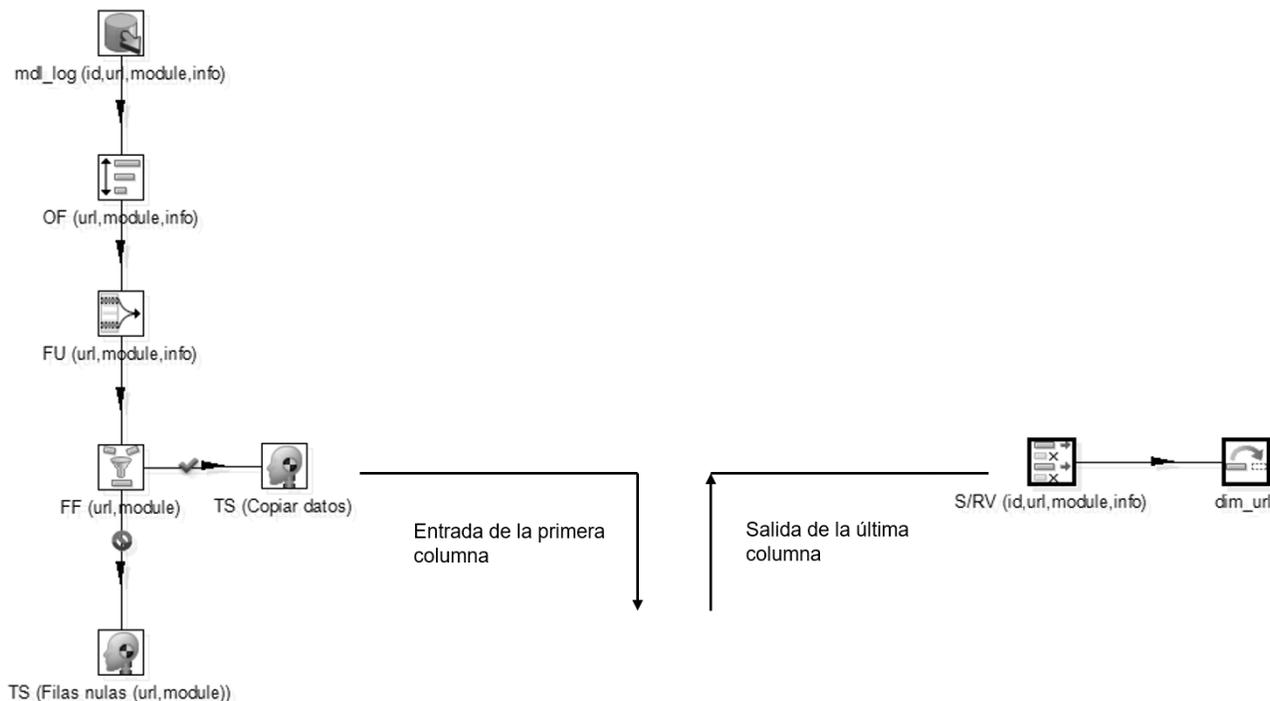
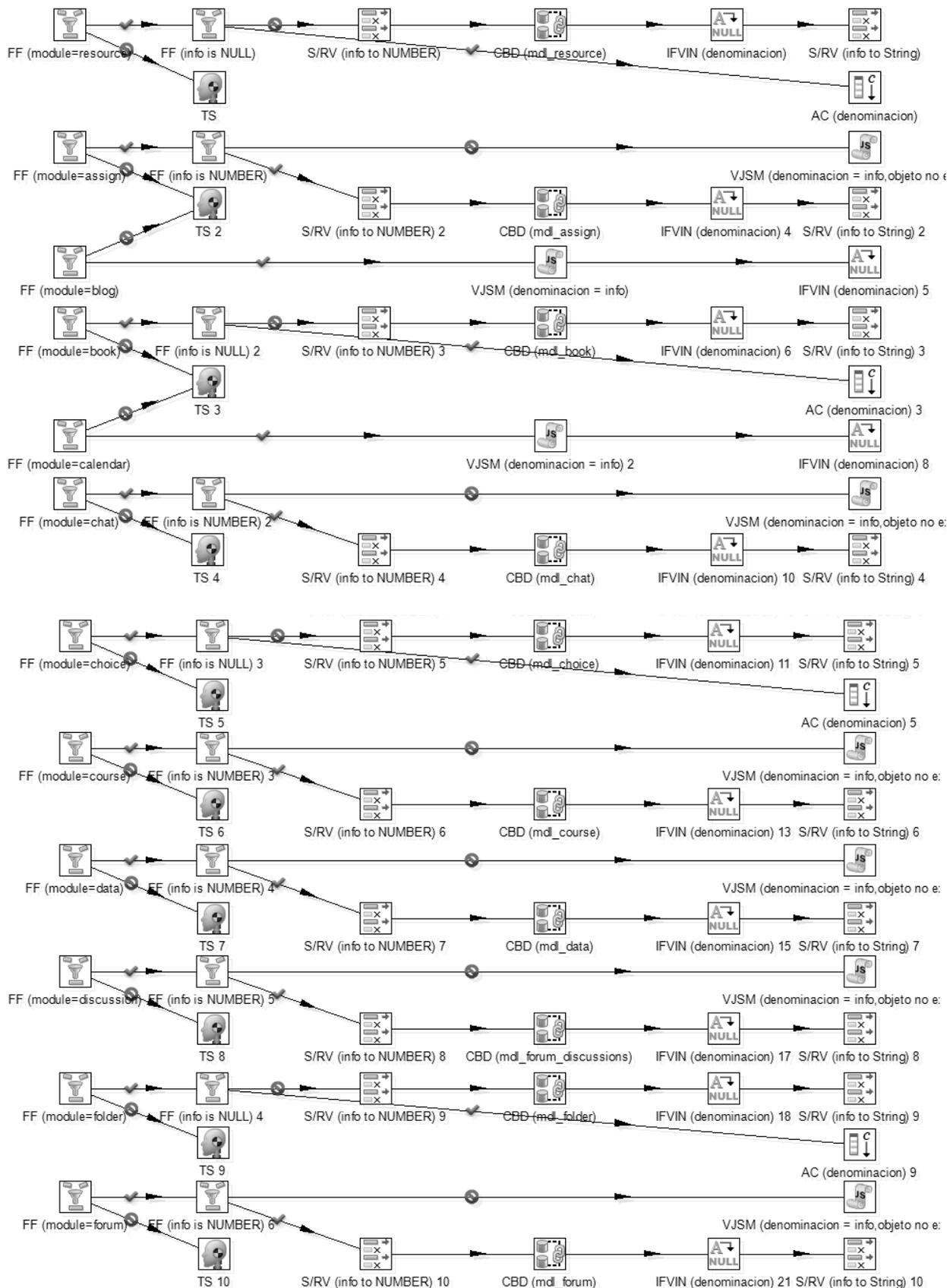


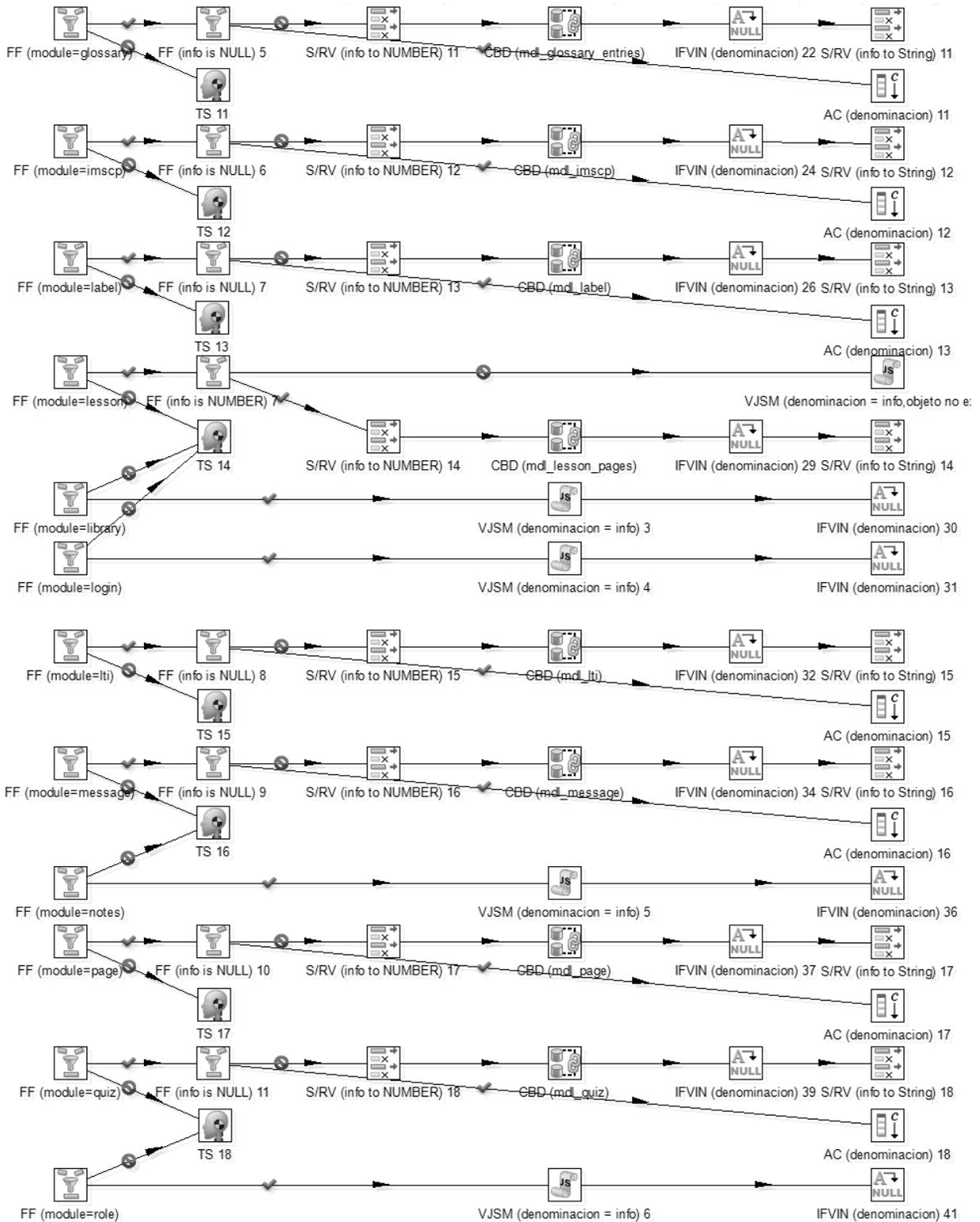
Ilustración 49 Transformación dim_rol.

Transformación dim_url

Esta transformación es la encargada de extraer las denominaciones de las distintas url a las que han accedido los usuarios del EVA, las mismas son extraídas de la tabla mdl_log y almacenadas en la tabla dim_url del MD. Los datos extraídos de la fuente son filtrados para eliminar valores nulos y duplicados, además se realiza la búsqueda de las denominaciones de los objetos relacionados con las URL visitadas, estas búsquedas se realizan en dependencia del módulo de MOODLE que haya generado el log, todo ello con el fin de conocer que curso, recurso o actividad fue visitado.







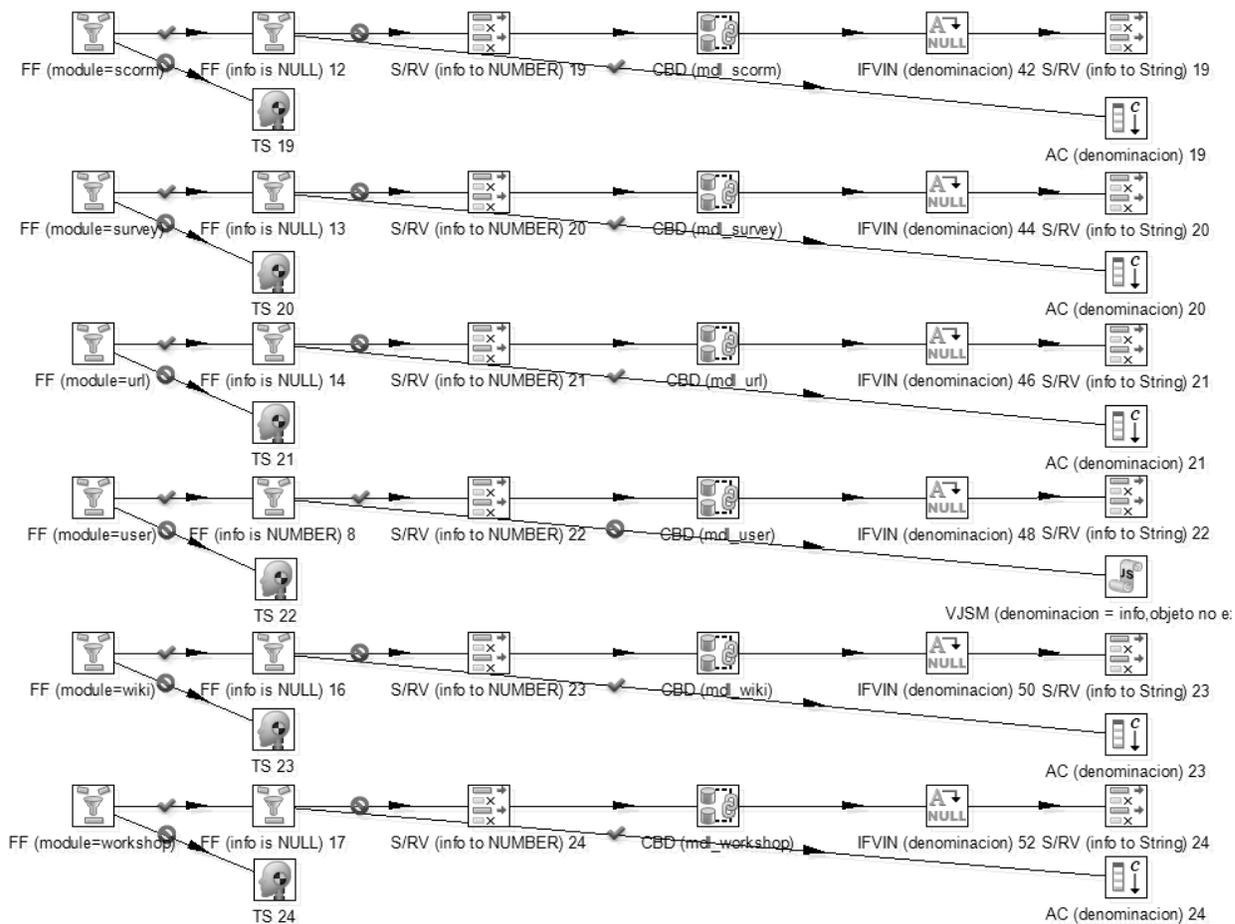


Ilustración 50 Transformación dim_url.

Función fun_obtener_semestre

```

CREATE OR REPLACE FUNCTION mod_eva.fun_obtener_semestre(parametro_trimestre NUMERIC)
RETURNS NUMERIC AS
  $BODY$
  DECLARE
  BEGIN
    IF (parametro_trimestre = 1 OR parametro_trimestre = 2) THEN
      RETURN 1;
    ELSE
      RETURN 2;
    END IF;
  END;
  $BODY$
LANGUAGE plpgsql;
    
```

Ilustración 51 Función fun_obtener_semestre.

Anexo # 4 Pruebas de integración.

Tabla 18 Prueba de integración a la dimensión Acción.

	Base de datos EVA	Mercado de datos	Herramienta de análisis
Consulta	SELECT COUNT(DISTINCT action) FROM public.mdl_log WHERE action IS NOT NULL;	SELECT COUNT(DISTINCT accion) FROM mod_eva.dim_accion;	SELECT NON EMPTY {Hierarchize ({[Fecha.Fechas].[Fechas]}) ON COLUMNS, NON EMPTY {Hierarchize ({[Acción.Acciones].[Acción].Members})} ON ROWS FROM [Interacción Usuario-EVA]
Resultado	110 / 11644.137 milisegundos	110 / 0.433 milisegundos	110 / 0.92 segundos

Tabla 19 Prueba de integración a la dimensión Rol.

	Base de datos EVA	Mercado de datos	Herramienta de análisis
Consulta	SELECT COUNT(DISTINCT shortname) FROM public.mdl_role WHERE shortname IS NOT NULL;	SELECT COUNT(DISTINCT nombre_corto) FROM mod_eva.dim_rol;	SELECT {Hierarchize ({[Fecha.Fechas].[Fechas]}) ON COLUMNS, {Hierarchize ({[Rol.Roles].[Rol].Members})} ON ROWS FROM [Interacción Usuario-EVA]
Resultado	9 / 0.082 milisegundos	9 / 0.075 milisegundos	9 / 0.06 segundos

Tabla 20 Prueba de integración a la dimensión Módulo.

	Base de datos EVA	Mercado de datos	Herramienta de análisis
Consulta	SELECT COUNT(*) FROM (SELECT DISTINCT name FROM public.mdl_modules WHERE name IS NOT NULL UNION SELECT DISTINCT module FROM public.mdl_log WHERE module IS NOT NULL) AS modulos;	SELECT COUNT(DISTINCT modulo) FROM mod_eva.dim_modulo;	SELECT NON EMPTY {Hierarchize ({[Fecha.Fechas].[Fechas]}) ON COLUMNS, NON EMPTY {Hierarchize ({[Curso.Cursos].[Curso].Members})} ON ROWS FROM [Interacción Usuario-EVA]

Resultado	32 / 362.930 milisegundos	32 / 0.113 milisegundos	32 / 0.04 segundos
------------------	-------------------------------------	--------------------------------	---------------------------

Tabla 21 Prueba de integración a la dimensión Dirección IP.

	Base de datos EVA	Mercado de datos	Herramienta de análisis
Consulta	SELECT COUNT(DISTINCT ip) FROM public.mdl_log WHERE ip IS NOT NULL;	SELECT COUNT(DISTINCT direccionip) FROM mod_eva.dim_direcciones;	SELECT NON EMPTY {Hierarchize ({[Fecha.Fechas].[Fechas]})} ON COLUMNS, NON EMPTY {Hierarchize ({[Dirección IP.Direcciones IP].[Dirección IP].Members})} ON ROWS FROM [Interacción Usuario-EVA]
Resultado	12715 / 15308.707 milisegundos	12715 / 105.803 milisegundos	12715 / 1.95 segundos

Tabla 22 Prueba de integración a la dimensión URL.

	Base de datos EVA	Mercado de datos	Herramienta de análisis
Consulta	SELECT COUNT(DISTINCT url) FROM public.mdl_log WHERE url IS NOT NULL;	SELECT COUNT(DISTINCT url) FROM mod_eva.dim_url;	SELECT NON EMPTY {Hierarchize ({[Fecha.Fechas].[Fechas]})} ON COLUMNS, NON EMPTY {Hierarchize ({[URL.URL's].[URL].Members})} ON ROWS FROM [Interacción Usuario-EVA]
Resultado	17713 / 28330.327 milisegundos	17713 / 212.251 milisegundos	17713 / 0.25 segundos

Anexo # 5 Pruebas de aceptación.

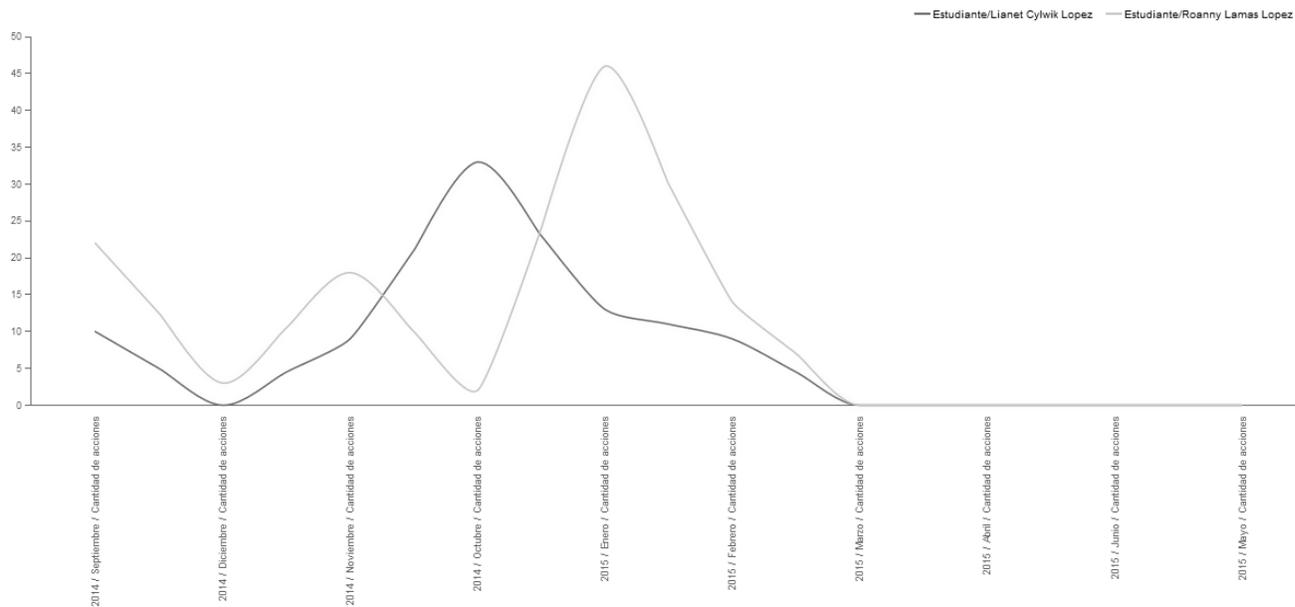


Ilustración 52 Cantidad de acciones por Usuario, Año y Mes para los usuarios "Roanny Lamas López" y "Lianet Cylwik López" en el período lectivo 2014-2015 en el EVA.

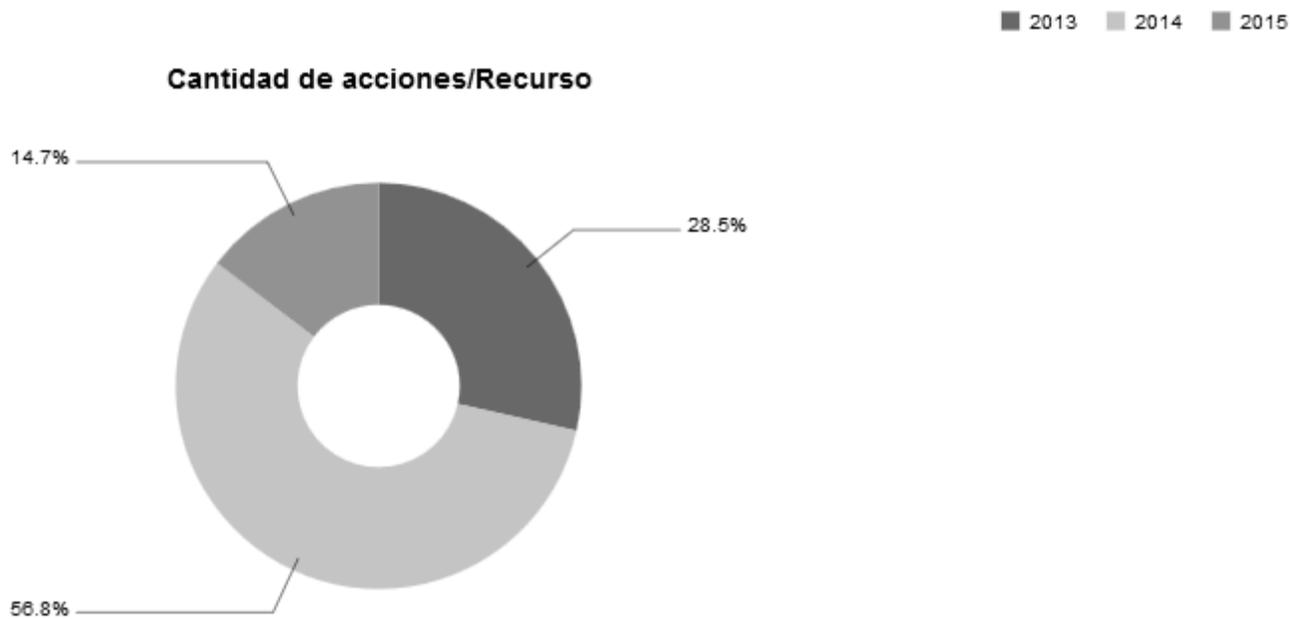


Ilustración 53 Cantidad de acciones por Módulo y Año en el período lectivo 2013-2015 en el EVA (En porciento).