

**Facultad de Tecnologías Educativas**

**Desarrollo de un Mercado de Datos para el movimiento de estudiantes en el Sistema de Gestión Académica XAUCE AKADEMOS para el MINED**

Trabajo de diploma para optar por el título de   
Ingeniero en Ciencias Informáticas

**Autor:** Joel Alejandro Valdespino Matos

**Tutor(es):**

Ing. Alenys Rivero Nápoles

MSc. Juana Elena Acosta García

La Habana, junio del 2024



**DECLARACIÓN DE AUTORÍA**

El autor del trabajo de diploma con título ***“Desarrollo de un Mercado de Datos para el movimiento de estudiantes en el Sistema de Gestión Académica XAUCE AKADEMOS para el MINED”*** concede a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la investigación, con carácter exclusivo. De forma similar se declara como único autor de su contenido. Para que así conste firma la presente a los \_\_\_ días del mes de \_\_\_\_\_\_\_\_\_\_\_\_\_\_ del año \_\_\_\_\_\_.

|  |
| --- |
| **Joel Alejandro Valdespino Matos** |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Firma del Autor |
| **Ing. Alenys Rivero Nápoles** |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Firma del Tutor |
| **MSc. Juana Elena Acosta García** |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Firma del Tutor |

**DATOS DE CONTACTO**

|  |  |
| --- | --- |
| **Tutor:**  MSc. Juana Elena Acosta García  Universidad de Ciencias Informáticas,  Ciudad de La Habana, Cuba.  **E-mail:** [juana@uci.cu](mailto:juana@uci.cu) | **Tutor:**  Ing. Alenys Rivero Nápoles  Universidad de Ciencias Informáticas,  Ciudad de La Habana, Cuba.  **E-mail:** [alenysrn@uci.cu](mailto:alenysrn@uci.cu) |

**DEDICATORIA**

A todos aquellos que, con su apoyo y aliento, han hecho posible la culminación de este trabajo. Su constante motivación y confianza han sido el motor que me impulsó a alcanzar esta meta.**AGRADECIMIENTOS**

Quiero expresar mi sincero agradecimiento a todos los docentes y profesores que, con su conocimiento y dedicación, contribuyeron a mi formación académica. También agradezco a mis compañeros de estudios, cuyo apoyo y motivación fueron fundamentales en este proceso.

**RESUMEN**

Son numerosos los instrumentos que generan un conjunto cada vez mayor de información. Los centros educativos como organizaciones, necesitan pulir la toma de decisiones relacionada a los procesos que desarrollan, por lo que han reforzado la investigación científica como opción para extender el uso de los datos que gestionan. La presente investigación surge por la necesidad de gestionar toda la información que generan el sistema académico XAUCE AKADEMOS desarrollado en la Universidad de las Ciencias Informáticas utilizado por el Ministerio de Educación, donde se recopilan los datos relacionados con el sistema académico nacional y todas las instituciones que a este pertenecen, con el objetivo de establecer un repositorio de Mercado de Datos que guarde la información de manera estandarizada, centralizada y que esté fácilmente disponible para su consulta y posterior análisis.

Para lograr este propósito, se llevó a cabo una investigación de las tecnologías que hacen posible el almacenamiento de grandes volúmenes de datos, así como la metodología para la creación de almacenes de datos. Posteriormente, se definieron las herramientas necesarias para construir el sistema y se documentó el proceso de análisis, diseño e implementación de los subsistemas de almacenamiento, con el objetivo de garantizar que el producto final satisfaga las necesidades del cliente.

**Palabras clave:** almacén de datos, toma de decisiones, análisis.

**SUMMARY**

There are numerous tools that generate an increasing amount of information. Educational centers, as organizations, need to polish the decision making related to the processes they develop, so they have strengthened scientific research as an option to extend the use of the data they manage. This research arises from the need to manage all the information generated by the academic system XAUCE AKADEMOS, developed at the University of Computer Sciences and used by the Ministry of Education, where data related to the national academic system and all the institutions that belong to it are collected, with the objective of creating a Data Warehouse repository that stores the information in a standardized, centralized and easily accessible way for consultation and subsequent analysis.

In order to achieve this goal, a study was carried out on the technologies that allow the storage of large volumes of data, as well as on the methodology for creating data warehouses. Subsequently, the tools needed to build the system were defined and the process of analysis, design and implementation of the storage subsystems was documented, with the aim of ensuring that the final product meets the client's needs.

Keywords: data warehouse, decision making, analysis.

**INDICE**

[INTRODUCCION 11](#_Toc168784627)

[CAPITULO 1: Fundamentos teóricos relacionados con los Mercados de datos. 15](#_Toc168784628)

[1.1 Mercado de Datos 15](#_Toc168784629)

[1.1.1 Características de los Mercados de Datos 15](#_Toc168784630)

[1.1.2 Arquitectura del mercado de datos 16](#_Toc168784631)

[1.1.3 Modelo multidimensional de los Mercados de Datos 17](#_Toc168784632)

[1.2 Estado del arte 17](#_Toc168784633)

[1.2.1 Metodologías para el diseño e implementación de un MD 19](#_Toc168784634)

[Selección de la metodología. 21](#_Toc168784635)

[1.3 Herramientas y tecnologías utilizadas 22](#_Toc168784636)

[1.3.1 Herramienta para el modelado 23](#_Toc168784637)

[1.3.2 Lenguaje de programación 23](#_Toc168784638)

[1.3.3 Pentaho Data Integration (PDI) 23](#_Toc168784639)

[1.3.4 Visual Studio Code 24](#_Toc168784640)

[1.3.5 Sistema Gestor de Base de Datos 24](#_Toc168784641)

[Conclusiones del capítulo 25](#_Toc168784642)

[CAPÍTULO 2: Analisis, diseño e implementacion del Almacen de Datos 26](#_Toc168784643)

[2.1 Análisis de requerimientos 26](#_Toc168784644)

[2.2 Análisis de los OLTP 28](#_Toc168784645)

[2.2.1 Hechos e indicadores 28](#_Toc168784646)

[2.2.2 Mapeo 29](#_Toc168784647)

[2.2.3 Granularidad 30](#_Toc168784648)

[2.2.4 Modelo conceptual ampliado 32](#_Toc168784649)

[2.3 Modelo lógico del MD 32](#_Toc168784650)

[2.3.1 Tipología 33](#_Toc168784651)

[2.3.2 Tablas de Dimensiones 33](#_Toc168784652)

[2.3.3 Tablas de Hechos 35](#_Toc168784653)

[2.3.4 Uniones 36](#_Toc168784654)

[2.4 Integración de datos 37](#_Toc168784655)

[2.4.1 Carga inicial 37](#_Toc168784656)

[2.4.2 Actualización 39](#_Toc168784657)

[Conclusiones del capítulo 40](#_Toc168784658)

[CAPÍTULO 3: Prueba del AD para el Sistema de Gestión Académica XAUCE AKADEMOS. 41](#_Toc168784659)

[3.1 Estándares de codificación 41](#_Toc168784660)

[3.2 Plan de pruebas. 41](#_Toc168784661)

[3.3 Conclusiones del capítulo 42](#_Toc168784662)

**INDICE DE TABLAS**

[**Tabla 1** *Análisis de sistemas homólogos* 19](#_Toc169060476)

[**Tabla 2** *Indicadores y perspectivas* 28](#_Toc169060477)

[**Tabla 3** *Hechos e indicadores* 30](#_Toc169060478)

**INDICE DE FIGURAS**

[**Figura *1*** *Arquitectura general de un almacén de datos* 17](#_Toc169060376)

[**Figura 2** *Fases Metodología Hefesto* 21](#_Toc169060377)

[**Figura 3** *Ciclo de vida de la Metodología de Desarrolllo para Proyectos de Almacenes de Datos* 22](#_Toc169060378)

[**Figura 4** *Modelo conceptual.* 28](#_Toc169060379)

[**Figura 5** *Modelo conceptual ampliado* 32](#_Toc169060380)

[**Figura 6** *Esquema en estrella.* 33](#_Toc169060381)

[**Figura 7** *Dimensión: dim\_motivo* 34](#_Toc169060382)

[**Figura 8** *Dimensión: dim\_tiempo* 34](#_Toc169060383)

[**Figura 9** *Diseño de la tabla de hechos* 35](#_Toc169060384)

[**Figura 10** *Uniones* 35](#_Toc169060385)

# INTRODUCCION

La sorprendente cantidad de datos que se genera a diario por numerosas aplicaciones informáticas convierten su gestión eficiente en un desafío esencial. No obstante, la abundancia de información plantea la cuestión fundamental de, cómo organizar, almacenar y garantizar un acceso rápido y eficiente cuando más se necesita (Carrillo, 2013).

Disímiles sectores son los que llevan a cabo procedimientos de informatización de sus procesos de negocio y estos llegan a formar parte de su actuar diario, siendo de vital importancia realizar reportes y análisis estadísticos con determinada periodicidad. El sector educacional, que no queda atrás en adoptar las Tecnologías de la Información y las Comunicaciones (TIC) en este creciente progreso tecnológico, implementando nuevos recursos que ponen de manifiesto la necesidad de reconceptualizar los procesos y modelos tradicionales de enseñanza, aprendizaje y gestión académica. Para ayudar al proceso docente se hace necesario la utilización de sistemas informáticos (*¿Qué es la ciencia de datos?*, 2023).

Como alternativa para el proceso de gestión académica se han desarrollado sistemas que controlan y organizan las actividades del proceso docente, lo que permite a los encargados tener un servicio automatizado de los elementos que intervienen en la labor académica, mitigar los riesgos que impone el dinamismo del proceso enfrentando de manera natural los cambios requeridos, adaptarse a nuevas condiciones, y brindar un papel activo a todos los involucrados en el proceso (Infante, 2019).

Cuba se proyecta a un futuro innovador, donde instituciones como la Universidad de Ciencias Informáticas (UCI) se encuentran inmersas en el desarrollo de sistemas para informatizar el proceso de gestión académica. La UCI, además de ser un centro de estudio, presenta una infraestructura productiva con una red de centros de desarrollo de software. Dentro de esta red de centros se encuentra el Centro de Tecnologías para la Formación (FORTES) donde se desarrolla un Sistema de Gestión Académica cuyo nombre es XAUCE-AKADEMOS como alternativa para el Ministerio de Educación (MINED).

Desde la implementación de XAUCE AKADEMOS para el MINED, se ha establecido una base de datos que centraliza los datos de todas las estructuras educativas del país, tanto de trabajadores, estudiantes y otra información relevante.

Dentro de los datos contemplados el movimiento de estudiantes, pero realizado de forma manual

A medida que la cantidad de datos almacenados ha crecido considerablemente, se han incorporado varios módulos para la generación de informes y análisis estadísticos. No obstante, la complejidad actual en la estructura de almacenamiento, así como las fórmulas y ecuaciones sofisticadas utilizadas han provocado que la generación de informes sea un proceso engorroso. Esto, se debe a que para el cálculo de algunos indicadores es necesario realizar consultas complejas que incluye la unión de varias tablas.

Teniendo en cuenta la situación problemática descrita se plantea como **problema a resolver:** ¿Cómo almacenar información estadística sobre el movimiento de estudiantes en el Sistema de Gestión Académica XAUCE AKADEMOS para el MINED?

Se plantea como **objeto de estudio:** almacenamiento de información estadística en sistemas informáticos.

Se establece como **campo de acción:** almacenamiento de información estadística del movimiento de estudiantes y como **objetivo general**, se persigue: Construir un mercado de datos para almacenar información estadística sobre el movimiento de estudiantes en el Sistema de Gestión Académica XAUCE AKADEMOS para el MINED.

Para dar cumplimiento el objetivo general planteado, se definen los siguientes objetivos específicos de la investigación:

1. Analizar del marco teórico-metodológico para el desarrollo de Mercados de Datos.
2. Analizar los sistemas de almacenamiento de datos utilizados por XAUCE AKADEMOS.
3. Establecer los requisitos para el desarrollo del Mercado de Datos.
4. Diseñar un Mercado de Datos para almacenar información estadística sobre XAUCE.
5. Implementar un Mercado de Datos para almacenar información del movimiento estudiantil en XAUCE.
6. Realizar las pruebas pertinentes a la solución propuesta.

A nivel teórico serán utilizados los métodos científicos:

**Histórico-lógico**

El método histórico-lógico se empleó para elaborar la fundamentación teórica de la investigación, describiendo el surgimiento y la evolución de los mercados de datos. Se analizaron los factores históricos y tecnológicos que han influido en su desarrollo, lo que permitió contextualizar la creación del mercado de datos para el sistema XAUCE AKADEMOS.

**Analítico–sintético**

El método analítico-sintético se aplicó para desglosar y examinar los distintos componentes del sistema XAUCE AKADEMOS mediante el análisis detallado de los diferentes aspectos del mismo, tales como la estructura de datos y los procesos de gestión académica. Posteriormente, se sintetizó esta información para formar una comprensión integral del funcionamiento del sistema y su impacto en el proceso académico.

También serán utilizados a nivel empírico los métodos:

**Observación:**

El método de observación se utilizó para estudiar la gestión de datos en el proceso académico y las interacciones entre los usuarios y el sistema XAUCE AKADEMOS. Esta observación permitió recopilar información precisa y detallada sobre el rendimiento y la eficacia del sistema, proporcionando una base empírica sólida para las recomendaciones y mejoras propuestas.

El presente trabajo está estructurado por tres capítulos, distribuidos de la siguiente manera:

**Capítulo 1: Fundamentos teóricos relacionados con los Mercados de Datos.**

Se estudian los conceptos, tecnologías y metodologías que son utilizadas para el desarrollo de los Mercado de Datos, además, se realiza un análisis de soluciones similares para determinar arquitectura y herramientas que permitan su correcto desarrollo.

**Capítulo 2: Análisis, diseño e implementación de un Mercado de Datos para el Sistema de Gestión Académica XAUCE AKADEMOS para el MINED**.

Se identifican los requerimientos de los usuarios a través de preguntas específicas; luego, se analizan estas preguntas para identificar indicadores y perspectivas que orientan la construcción del Mercado de Datos; finalmente, se confecciona un modelo conceptual para definir la estructura del Mercado de Datos y se procede a poblarlo con datos, utilizando técnicas de limpieza y calidad de datos.

**Capítulo 3: Prueba del Mercado de Datos para el Sistema de Gestión Académica XAUCE AKADEMOS.**

Se enfoca en las pruebas necesarias para verificar que el Mercado de Datos cumpla con los estándares de calidad y seguridad establecidos. La ejecución de estas pruebas se realizará en un entorno controlado para identificar posibles problemas.

# CAPITULO 1: Fundamentos teóricos relacionados con los Mercados de datos.

En el presente capítulo se exponen los conceptos asociados a los Almacenes de Datos y se estudian las herramientas, técnicas y tecnologías para su desarrollo; además, se realiza un estudio de la metodología que guiará el análisis, diseño y las herramientas a utilizar durante el desarrollo del Mercado de Datos.

## 1.1 Mercado de Datos

Ralph Kimball define los Almacenes de Datos (AD) como una copia de los datos transaccionales estructurados específicamente para consultas y análisis. Los define también como "la unión de todos los mercados de datos de una entidad", donde los Mercados de Datos (en lo adelante, MD) son un repositorio de información, similar a un almacén de datos, pero orientado a un área o departamento específico de la organización (Kimball, 1998).

Por su parte, Inmon, lo define como una colección de datos orientada a un determinado ámbito (empresa, organización), integrado, no volátil y variable en el tiempo, que ayuda a la toma de decisiones en la entidad en la que se utiliza (Inmon, 2002).

A modo de resumen, un MD es una infraestructura que integra datos de diversas fuentes de una organización y los organiza de manera estructurada para facilitar su análisis. Estos datos históricos abarcan un amplio período de tiempo y se utilizan para identificar tendencias, patrones y relaciones que pueden ayudar en la toma de decisiones estratégicas. El MD permite a empresas y organizaciones acceder a información crítica de manera rápida y eficiente, lo que permite mejorar la capacidad de respuesta.

Basándose en la definición dada por Inmon, se determinan las características de los MD

### 1.1.1 Características de los Mercados de Datos

Un Mercado de Datos es una colección de datos orientada al negocio, integrada, variante en el tiempo y no volátil para el soporte del proceso de toma de decisiones de la entidad (Bernabeu Ricardo, 2010).

**Orientada al negocio**: incluyen todos aquellos datos que son necesarios para satisfacer de manera inmediata los requerimientos funcionales en los procesos orientados a las aplicaciones.

**Integrada:** contiene todos los datos producidos por distintos departamentos, secciones y aplicaciones, y deben por lo tanto ser analizados para asegurar su calidad y limpieza.

**Variante en el tiempo**: los datos se encuentran almacenados junto a sus respectivos históricos, los cambios producidos en los datos a lo largo del tiempo quedan registrados.

**No volátil:** los datos una vez que entran en el MD no cambian debido a que solo existen dos tipos de operaciones: la carga de datos y el acceso a los mismos (escritura y lectura). Por esta razón es que en el MD no se requieren mecanismos de control de concurrencia y recuperación.

### 1.1.2 Arquitectura del mercado de datos

Las soluciones de MD exhiben una arquitectura de sistemas genérica que abarca todos los procesos necesarios para la construcción de estos.

Considerando las características de los MD, se procede a definir y describir los componentes que forman parte de su arquitectura, detallando su función e interacción dentro del sistema (Bernabeu Ricardo, 2010). En la ilustración 1 se muestra la arquitectura general de un MD.

**Figura *1****Arquitectura general de un almacén de datos*



*Nota.* (Bernabeu Ricardo, 2010).

**Componentes de la arquitectura de un almacén de datos**

* **Orígenes de datos** *(OLTP OnLine Transactional Processing):* representa toda aquella información transaccional que genera la empresa en su accionar diario, además, de las fuentes externas con las que puede llegar a disponer.
* **Administrador de carga** o más conocido como Proceso de Extraer, Transformar y Cargar (**ETL** por sus siglas del inglés *Extract, Transform and Load*): es el proceso en el que se extraen los datos que se encuentran almacenados en los sistemas OLTP.
* **Administrador de Almacén de Datos**: almacena los datos de forma multidimensional, a través de tablas de hechos y tablas de dimensiones, gestiona y mantiene metadatos.
* **Administrador de Consulta**: realiza las operaciones necesarias para soportar los procesos de gestión y ejecución de consultas relacionales.
* **Herramienta de Consulta y Análisis**: son sistemas que permiten a los usuarios realizar la exploración de datos del Almacén de Datos
* **Usuarios**: son aquellos que se encargan de tomar las decisiones y planificar las actividades del negocio, es por ello que se hace tanto énfasis en la integración y limpieza de datos, para poder conseguir que la información posea toda la calidad posible.

### 1.1.3 Modelo multidimensional de los Mercados de Datos

A diferencia de los clásicos sistemas de Base de Datos que presentan sus estructuras diseñadas mediante el modelo Entidad-Relación (E-R) los MD se diseñan mediante un modelo multidimensional compuestos por dimensiones y hechos. Poseen la misma información que el de E-R, pero la organiza de forma diferente para garantizar la velocidad y eficiencia en la recuperación de la misma. A continuación, se describen las características de este modelo (Bernabeu Ricardo, 2010):

**Dimensiones**: Las tablas de dimensiones definen como están los datos organizados lógicamente y proveen el medio para analizar el contexto del negocio. Contienen datos cualitativos. Representan los aspectos de interés, mediante los cuales los usuarios podrán filtrar y manipular la información.

**Hechos**: los hechos son datos instantáneos en el tiempo, que son ﬁltrados, agrupados y explorados a través de condiciones deﬁnidas en las dimensiones. Los datos presentes en los hechos constituyen el volumen del MD, y pueden estar compuestos por millones de registros dependiendo de su granularidad y de los intervalos de tiempo de los mismos. Los más importantes son los de tipo numérico.

## 1.2 Estado del arte

Con el objetivo de establecer el estado del arte en la presente investigación, se realiza un estudio para detectar y estudiar aspectos, metodología y herramientas empleadas en el desarrollo de soluciones homólogas, con el fin de evaluar su factibilidad en el entorno donde se desenvuelve la presente investigación.

Sistemas estudiados:

1. Implementación de un almacén de datos en el Centro de Enseñanza del Laboratorio Nacional de Informática Avanzada (LANIA) aplicado al estudio de perfiles de deserción.
2. Almacén de Datos para la gestión de estudios de Peligro, Vulnerabilidad y Riesgo en Cuba.
3. Aplicación de un almacén de datos para la toma de decisiones en la Dirección Técnica Normativa Docente del Ministerio de Educación.
4. Mercado de datos: Análisis de Tendencias para la Actividad de Desarrollo-Producción UCI.

**Tabla 1**  
*Análisis de sistemas homólogos*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sistema** | **Metodología** | **Fases** | **Clasificación** | **Herramientas** |
| 1 | CRISP-DM (Cross Industry Standard Process for Data Mining). | 6 | Híbrida | Kettle, PostgreSQL, Visual Paradigm |
| 2 | HEFESTO. | 4 | Ágil | Pentaho Data Integration, Mondrian Schema Workbench |
| 3 | Ralph Kimball  "enfoque dimensional" o "modelo dimensional. | - | Tradicional | SQL Server |
| 4 | “Metodología de Desarrollo para Proyectos de Almacenes de Datos”. | 9 | Ágil | PostgreSQL, PgAdmin, Pentaho Data Integration (PDI) |

En la Tabla 1 se plasma una muestra de los sistemas estudiados y algunos de los principales aspectos que se tuvieron en cuenta a la hora de realizar el análisis de los mismos.

Todos los MD persiguen un objetivo común: consolidar datos dispersos y variados de diversas fuentes para facilitar un acceso rápido y proporcionar una fuente para análisis y generación de informes. Sin embargo, no se pueden aplicar soluciones existentes de otras empresas, ya que cada una requiere una solución única que se adapte a sus necesidades específicas, debido a que tienen sus propias particularidades y requerimientos únicos.

### 1.2.1 Metodologías para el diseño e implementación de un MD

Una metodología es una colección de procedimientos, técnicas, herramientas y documentos auxiliares que ayudan a los desarrolladores de software en sus esfuerzos por implementar nuevos sistemas de información. Una metodología está formada por fases, cada una de las cuales se puede dividir en sub-fases, que guiarán a los desarrolladores de sistemas a elegir las técnicas más apropiadas en cada momento del proyecto y también a planificarlo, gestionarlo, controlarlo y evaluarlo (Avison & Fitzgerald, 1995).

A continuación, se describen las metodologías más utilizadas para el diseño e implementación de un almacén de datos.

**Metodología de Kimball**

Esta metodología adopta este nombre en honor a su creador Ralph Kimball, considerado el inventor del Modelo Dimensional y pionero en AD e Inteligencia de Negocios. Define un almacén de datos como "la unión de todos los Mercados de Datos de una entidad" (Kimball, 2010).

La metodología se basa en lo que Kimball denomina Ciclo de Vida Dimensional del Negocio. Este ciclo de vida del proyecto de AD, está basado en cuatro principios básicos (Peñafiel et al., 2019):

* Centrarse en el negocio.
* Construir una infraestructura de información adecuada.
* Realizar entregas en incrementos significativos.
* Ofrecer la solución completa.

**Metodología de Inmon**

Desde el punto de vista arquitectónico, la mayor diferencia entre Inmon y Kimball es el sentido de la construcción del AD, Inmon plantea una metodología descendente (top-down) a la hora de diseñar un AD, ya que de esta forma se considerarán mejor todos los datos corporativos. En esta metodología los MD se crearán después de haber terminado el AD completo de la organización (Peñafiel et al., 2019).

De forma contraria a la de Kimball, esta metodología puede tener una implementación tardada, y es recomendada cuando se hace demasiado difícil representar el modelo a través de dimensiones y la complejidad de la solución se hace demasiado grande, pero no es muy recomendable para proyectos sencillos pues va de lo general, el AD, a lo más específico, el MD.

**Metodología Hefesto**

Es una de las metodologías más difundidas y utilizadas por su fácil implementación y aporte práctico. La metodología Hefesto propone cuatro fases fundamentales para su aplicación, cada una con una serie de puntos que se deben realizar y que se resumen en el siguiente gráfico (Peñafiel et al., 2019):

**Figura 2***Fases Metodología Hefesto*

****

*Nota*. (Bernabeu Ricardo, 2010)

**Metodología de Desarrollo para Proyectos de Almacenes de Datos**

La Metodología de Desarrollo de Proyectos de Almacenes de Datos toma como base la metodología de Kimball para definir los aspectos específicos del desarrollo de almacenes de datos (DW). Para incorporar los principios básicos que permiten una adecuada gestión del proyecto, utiliza la Guía para los Fundamentos de la Dirección de Proyectos (Guía del PMBOK). Los temas asociados a CMMI se incorporan a partir del Programa de Mejora por lo tanto hereda algunos de sus enfoques, artefactos y actividades(González Hernández, 2013) . La Metodología de Desarrollo para Proyectos de Almacenes de Datos tiene como objetivos:

* Proveer una guía referencial de la forma en que debe organizarse el proceso de desarrollo y las actividades que deben ser realizadas para garantizar la correcta realización de un proyecto de DW.
* Facilitar la adaptación y formación de los especialistas y estudiantes que se enfrentan por primera vez al desarrollo o liderazgo de un proyecto de DW.
* Establecer un marco para la gestión del proyecto como parte del proceso de desarrollo de las soluciones de DW.
* Alinear la Metodología de Desarrollo para Proyectos de Almacenes de Datos a CMMI para poder optar por la certificación de este modelo en su nivel 2.

Para dar cumplimiento a los objetivos propuesto se define el ciclo de vida de la metodología.

**Figura 3***Ciclo de vida de la Metodología de Desarrolllo para Proyectos de Almacenes de Datos*



Nota. (González Hernández, 2013).

### Selección de la metodología.

Después de avaluar cada una de las metodologías mencionadas, se ha decidido guiar el proceso de diseño e implementación del MD por la metodología Hefesto, debido a su adaptabilidad, capacidad para facilitar el complejo proceso de construcción de un MD desde cero y su enfoque en una entrega incremental. Esta posee varias características que la hacen una metodología ideal para personas que entrar por primera vez al mundo de los MD, entre las que destaca que se basa en los requerimientos de los usuarios, por lo que su estructura es capaz de adaptarse con facilidad a los cambios en el negocio, involucrando así a los usuarios finales en cada etapa de la construcción el MD, se utilizan modelos conceptuales y lógicos sencillos de interpretar y analizar además de ser independiente del ciclo de vida del software y de las herramientas que se utilicen para su implementación.

## 1.3 Herramientas y tecnologías utilizadas

Las herramientas, son programas, aplicaciones o simplemente instrucciones usadas para efectuar otras tareas de modo más sencillo. En un sentido amplio del término, se puede decir que una herramienta es cualquier programa o instrucción que facilita una tarea. Dentro del desarrollo del AD, como toda elaboración de un software también existen un grupo de herramientas que facilitarán su construcción.

Las tecnologías se refiere al conjunto de conocimientos, técnicas y métodos utilizados para desarrollar productos o servicios, así como los dispositivos y sistemas derivados de estos conocimientos. En el contexto del desarrollo de almacenes de datos, las tecnologías abarcan desde las bases de datos y lenguajes de programación hasta las plataformas de procesamiento de datos, proporcionando el marco esencial para la construcción y operación eficiente de sistemas de información. A continuación, se describen las herramientas y tecnologías utilizadas en el desarrollo de la solución al problema que da paso a la siguiente investigación.

### 1.3.1 Herramienta para el modelado

Como herramienta de modelado se emplea *Visual Paradigm* (VP), en su versión 8.0. VP es una herramienta de Ingeniería de Software Asistida por Computación (CASE, por sus siglas en inglés) para el modelado mediante el Lenguaje de Modelado Unificado (UML). Esta herramienta proporciona un conjunto completo de funciones de modelado y diagramación para diversos fines, como desarrollo de software, análisis de sistemas y modelado de procesos comerciales. Permite a los usuarios crear representaciones visuales de conceptos, procesos y relaciones utilizando diferentes tipos de diagramas, incluidos diagramas UML (Lenguaje de Modelado Unificado), diagramas BPMN (Modelo y Notación de Procesos de Negocio), diagramas ER (Entidad-Relación), y otros. VP admite el trabajo colaborativo, el control de versiones y la integración con otras herramientas de desarrollo, lo que lo convierte en una opción popular entre los desarrolladores de software, analistas de sistemas y gerentes de proyectos (*Software de diagramas y solución de gráficos en línea*, s. f.).

### 1.3.2 Lenguaje de programación

Un lenguaje de programación se define como un sistema de símbolos y reglas sintácticas y semánticas que establecen la estructura y el significado de sus elementos y expresiones. Su propósito fundamental radica en el control de la operación física y lógica de un sistema informático.

Como lenguaje de programación se utiliza Python en su versión 3.11.

*Python 3.11* es un lenguaje de programación de propósito general, independiente de la plataforma y orientado a objetos. Se debe tomar en cuenta que Python no fue diseñado específicamente para el análisis de datos o la computación científica, y que a últimas fechas ha surgido como una herramienta de primera clase para tareas de computación científica, incluido el análisis y visualización de grandes conjuntos de datos (VanderPlas, 2016).

### 1.3.3 Pentaho Data Integration (PDI)

*Pentaho Data Integration* (PDI) *9.3*, también conocido como *Kettle*, es una herramienta de integración de datos diseñada para ETL (*Extract, Transform, Load*). PDI permite a los usuarios extraer datos de diversas fuentes, transformarlos según las necesidades del negocio y cargarlos en un sistema de destino, como bases de datos, aplicaciones de datos grandes, o archivos de texto. Es conocida por su interfaz gráfica intuitiva y capacidades robustas, facilitando la creación de flujos de datos complejos sin necesidad de programación extensa.

### 1.3.4 Visual Studio Code

*Visual Studio Code* *1.86.1* (VS Code) es un editor, gratuito, de código fuente desarrollado por Microsoft, popular por su flexibilidad y personalización. VS Code admite diversas extensiones y tiene características avanzadas como depuración, control de versiones con Git, utilizado para programar en cualquier lenguaje de programación, sin cambiar de editor, es compatible con muchos lenguajes y combina la sencillez de un editor de código fuente con potentes herramientas lo que lo hace ideal para desarrolladores (*Get Started with Visual Studio Code*, s. f.).

### 1.3.5 Sistema Gestor de Base de Datos

Los Sistema de Gestión de Base de Datos (SGBD*)* son un tipo de software muy específico, dedicados a servir de interfaz entre la base de datos, los usuarios y las aplicaciones que lo utilizan. Se compone de lenguajes de definición, manipulación, consulta y seguridad de datos. El propósito general de los SGBD es el de manejar de manera clara, sencilla y ordenada un conjunto de datos (Bernabeu Ricardo, 2010).

Como SGBD se emplea PostgreSQL.

PostgreSQL es un SGBD relacional y de código abierto que presenta una arquitectura cliente - servidor, conocido por su robustez, extensibilidad y cumplimiento con los estándares SQL. Soporta distintos tipos de datos, así como la creación de tipos propios. PostgreSQL se destaca por su capacidad de manejar transacciones complejas y su soporte para funciones avanzadas como procedimientos almacenados, triggers y vistas. Es ampliamente utilizado en aplicaciones web, sistemas de análisis de datos y cualquier entorno que requiera una base de datos fiable y escalable (*PostgreSQL: About*, s. f.).

*pgAdmin 4* es una herramienta de administración y desarrollo de código abierto para PostgreSQL, diseñada para facilitar la gestión y el uso de bases de datos. Proporciona una interfaz gráfica de usuario (GUI) intuitiva y rica en funcionalidades, permitiendo a los usuarios realizar tareas como la creación de bases de datos, la ejecución de consultas SQL, la configuración de usuarios y permisos, y la visualización de estadísticas y gráficos de rendimiento. Es ampliamente utilizada por desarrolladores y administradores de bases de datos para interactuar de manera eficiente con PostgreSQL (*pgadmin*, s. f.).

PostgreSQL se utilizó en el desarrollo de la Base de Datos de XAUCE-AKADEMOS, por tanto, con el objetivo de mantener la simetría entre ambos sistemas se decide emplear en la solución propuesta este SGBD.

## Conclusiones del capítulo

El Sistema de Gestión Académica XAUCE AKADEMOS es una herramienta multiplataforma que contribuye al perfeccionamiento de los procesos académicos de una institución. Su uso permite el desarrollo coherente de una estrategia organizacional que articule todos los niveles de decisión presentes en los procesos universitarios. Todos los roles del proceso educativo están involucrados en la solución, permitiendo el acceso a la información de forma segura en todos los niveles y facilitando la toma de decisiones (*XAUCE AKADEMOS 1.0 | Universidad de las Ciencias Informáticas*, s. f.).

Sin embargo, se ha identificado que este sistema no cuenta con una gestión de información estadística integrada. Actualmente, AKADEMOS solo almacena los datos de las estructuras educacionales del país, y el análisis estadístico necesario se realiza de forma manual, lo que limita la eficiencia y la capacidad de respuesta rápida a las necesidades de información y análisis.

El proceso de almacenamiento de información en sistemas informáticos abarca desde la entrada y procesamiento de datos hasta su organización, almacenamiento físico, y gestión continua. Este es fundamental para garantizar que la información esté disponible, sea accesible y pueda ser recuperada cuando sea necesario, asegurando seguridad y eficiencia en su manejo.

Para la correcta estructuración de la información, en esta investigación, se desarrolla un mercado de datos. Con este propósito, se estudiaron varias metodologías de las que escogió Hefesto para guiar el proceso debido a su características y facilidad de implementación, asimismo se mencionan y analizan las herramientas y tecnologías empleadas en el desarrollo de esta investigación, lo que permitió definir el ambiente tecnológico, manteniendo la simetría entre la solución propuesta y el sistema XAUCE-AKADEMOS, sentando las bases para la ejecución de la solución propuesta en la investigación.

# CAPÍTULO 2: Analisis, diseño e implementacion del Almacen de Datos

En este capítulo se muestran las fases a seguir para la realización del análisis y el diseño del AD utilizando la metodología Hefesto y se explicarán los pasos de cada una de ellas, y se definirá un modelo conceptual que luego será ampliado a partir del establecimiento de los indicadores y perspectivas identificados en los requisitos. Además, se establecerán las correspondencias entre el modelo conceptual y los OLTP. Finalmente, se diseñará el modelo lógico del Almacén de Datos.

## 2.1 Análisis de requerimientos

Como punto de partida se deben identificar los requerimientos de los usuarios a través de preguntas que evidencien los objetivos de su organización. Luego se analizarán estas preguntas a fin de identificar cuáles serán los **Indicadores y Perspectivas** que serán tomadas en cuenta para la construcción del MD. Finalmente se confeccionará un **Modelo Conceptual** en donde se podrá visualizar el resultado obtenido en este primero paso.

#### 2.1.1 Preguntas de negocio

En el contexto del proceso de levantamiento de requisitos para la construcción del Almacén de Datos destinado al Sistema de Gestión Académica XAUCE AKADEMOS para el MINED, se realizó una entrevista con el Ing. Sandy Nuñez Padrón, un directivo con profundo conocimiento de los procesos del negocio. El objetivo de la entrevista fue identificar preguntas clave que orientaron el desarrollo del almacén de datos, asegurando una comprensión completa de las necesidades y expectativas del usuario en esta fase inicial del proyecto.

A partir de la entrevista realizada se definieron las siguientes cuestiones que conformarán los requerimientos para el desarrollo del Almacén de Datos, agrupadas por temas en común.

Las preguntas de Negocio obtenidas fueron las siguientes:

**Movimiento de estudiantes**

* ¿Cuántas altas, por motivo, se han registrado en un periodo de tiempo determinado?
* ¿Cuántas bajas, por motivo, se han registrado en un periodo de tiempo determinado?
* ¿Cuántas otras bajas, por motivo, se han registrado en un periodo de tiempo determinado?
* ¿Cuántos graduados o egresado se han registrado en un periodo de tiempo determinado?

#### 2.1.2 Indicadores y Perspectivas

Una vez establecidas las cuestiones claves, se procede a su descomposición para descubrir los indicadores que se utilizarán y las perspectivas de análisis que intervendrán.

Los **indicadores** constituyen valores numéricos, que representan de forma concreta el objeto analizado, por ejemplo: saldos, importes, promedios, cantidades, sumatorias, fórmulas.

En cambio, las **perspectivas** se refieren a los objetos mediante los cuales se quiere examinar los indicadores, con el fin de responder a las cuestiones planteadas; por ejemplo: clientes, proveedores, sucursales, países, productos, rubros (Bernabeu, 2010).

A continuación, se analizan las preguntas obtenidas en el paso anterior y se detallan cuáles son sus respectivos indicadores y perspectivas.

**Tabla 2**  
*Indicadores y perspectivas*

|  |  |
| --- | --- |
| **Indicadores** | **Perspectivas** |
| Cantidad de altas | Motivo (de otro centro del municipio, otra provincia, etc), tiempo |
| Cantidad de bajas | Motivo (a otro centro de la provincia, a otra provincia, etc), tiempo |
| Cantidad de otras bajas | Motivo (defunción, emigración, etc), tiempo |
| Cantidad de graduados o egresados | Tiempo |

#### 2.1.3 Modelo conceptual

Se construye el modelo conceptual a partir de los indicadores y perspectivas obtenidas en el paso anterior. El modelo conceptual (Figura 4) permite observar con claridad cuál es el alcance del proyecto, ofreciendo una base sobre la cual trabajar. Además, al poseer un alto nivel de definición de datos, permite que pueda ser representado y explicado con facilidad.

**Figura 4***Modelo conceptual.*



*Nota*. En el lado izquierdo pueden observarse las diferentes perspectivas empleadas para obtener como resultado los indicadores requeridos por el cliente, y la relación mediante la cual se unen es el: Movimiento de estudiantes.

## 2.2 Análisis de los OLTP

En esta fase se analizó la base de datos del Sistema de Gestión Académica XAUCE AKADEMOS para el MINED para determinar cómo calcular los indicadores y establecer las correspondencias entre el modelo conceptual creado en el paso anterior y los datos fuente. Se definieron los campos a incluir en cada perspectiva y se amplió el modelo conceptual con la información obtenida.

### 2.2.1 Hechos e indicadores

En este paso se explica cómo se calculan los indicadores, definiendo los siguientes conceptos para cada uno de ellos:

* Hecho o hechos que lo componen, con su respectiva fórmula de cálculo.
* Función de agregación que se utilizará, ejemplo: *SUM, AVG, COUNT, MAX*, entre otras.

Los indicadores se calculan tal como se puede apreciar a continuación:

**Indicador:** Cantidad de altas.

* **Hecho(s):** Cantidad de altas.
* **Función de sumarización:** *COUNT*.

**Aclaración:** el indicador, cantidad de altas, representa la cantidad de altas que se dan en un centro educativo del país.

Debido a que el método de cálculo de los indicadores es consistente y sigue la misma estructura, se optó por presentarlos en una tabla. Esto facilita una comprensión más clara y detallada.

**Tabla 3**  
*Hechos e indicadores*

|  |  |  |  |
| --- | --- | --- | --- |
| **Indicador** | **Hecho(s)** | **Función de sumarización** | **Descripción** |
| Cantidad de altas | Cantidad de altas | *COUNT* | El indicador, cantidad de altas, representa la cantidad de altas que se dan en un centro educativo del país. |
| Cantidad de bajas | Cantidad de bajas | *COUNT* | El indicador, cantidad de bajas, representa la cantidad de bajas que se dan en un centro educativo del país. |
| Cantidad de otras bajas | Cantidad de otras bajas | *COUNT* | El indicador, cantidad de otras bajas, representa la cantidad de otras bajas que se dan en un centro educativo del país. |
| Cantidad de graduados/egresados | Cantidad de graduados/egresados | *COUNT* | El indicador, cantidad de graduados o egresados, representa la cantidad de graduados o egresados en un centro educativo del país. |

### 2.2.2 Mapeo

En este paso se examinaron los OLTP disponibles para asegurar que contienen los datos requeridos y sus características para establecer la correspondencia directa entre los elementos del modelo conceptual y la fuente de datos.

**Las relaciones que se identificaron fueron las siguiente:**

* La perspectiva: **Motivo,** se relaciona con la tabla: **tramite\_tbr\_tramite\_configuracion**.
* La perspectiva: **Tiempo,** se relaciona con el campo: **fecha\_creacion** en la tabla: **tramite\_tbd\_tramite\_estudiante** debido a que es la fecha principal en el proceso.
* El indicador: **Cantidad de altas**, surge del conteo de los registros en la tabla **tramite\_tbd\_registro\_estudiante**.
* El indicador: **Cantidad de bajas**, surge del conteo de los registros en la tabla **tramite\_tbd\_tramite\_baja\_estudiante**.
* El indicador: **Cantidad de otras bajas**, surge del conteo de los registros en la tabla: **tramite\_tbd\_tramite\_traslado\_estudiante**.
* El indicador: Cantidad de graduados/egresados, surge del conteo de los registros en la tabla: **tramite\_tbd\_tramite\_graducacion\_estudiante**.

### 2.2.3 Granularidad

Una vez establecidas las relaciones con los OLTP, se eligieron los campos específicos que cada perspectiva incluye ya que a través de estos se analizaron los indicadores. Basándose en el mapeo establecido en el paso anterior, se muestran a continuación todos los campos disponibles para las perspectivas.

Con respecto a la perspectiva **Motivo**, los datos que se pueden utilizar son:

id, nombre, autor\_id, deshabilitado, fecha\_creacion, fecha\_modificacion, descripcion, cambia\_estado, cambia\_situacion\_escolar, cambia\_estructura, estado\_final, tipo\_tramite, deleted\_at, is\_deleted.

Con respecto a la Perspectiva: **Tiempo**, que es la que determina la granularidad del MD, los datos más comunes son los siguientes: Día, Semana, Mes, Trimestre, Año, entre otros.

Una vez recolectada toda la información pertinente y consultado con el cliente los datos de interés para analizar los indicadores, los resultados obtenidos fueron:

Perspectiva **Motivo**:

* tipo\_tramite\_estudiante de la tabla tramite\_tbd\_tramite\_estudiante. Ya que hace referencia al tipo de trámite que se le realiza al estudiante y por ende el motivo.

Perspectiva **Tiempo**:

* fecha\_creacion de la tabla tramite\_tbd\_tramite\_estudiante. Hace referencia a la fecha en la q se inicia el trámite del estudiante.

### 2.2.4 Modelo conceptual ampliado

En este paso, y con el fin de graficar los resultados obtenidos en los pasos anteriores, se amplía el modelo conceptual, colocando debajo de cada perspectiva los campos seleccionados para el análisis de los indicadores y debajo de estos su respectiva fórmula de cálculo.

**Figura 5**  
*Modelo conceptual ampliado*



*Nota.* En la imagen se muestran los resultados obtenidos en pasos anteriores, que resultó en la ampliación del modelo conceptual. Se muestra de forma gráfica cada una de los campos de las perspectivas, a través de los cuales se analizan los indicadores y estos, a su vez, llevan las fórmulas para el calcular cada uno.

## 2.3 Modelo lógico del MD

En esta fase se confecciona el modelo lógico de la estructura del MD, teniendo como base el modelo conceptual que ha sido creado. Un modelo lógico es la representación de una estructura de datos, que puede procesarse y almacenarse en algún SGBD (Bernabeu Ricardo, 2010). Para ello se define el tipo de modelo lógico que se utiliza y se diseñan las tablas de dimensiones y de hechos con sus respectivas relaciones.

### 2.3.1 Tipología

Es importante definir el tipo de esquema: Esquema en Estrella, Copo de Nieve o Constelación, que se empleará ya que esta decisión influye directamente en la elaboración del modelo lógico.

Se he seleccionado el Esquema en Estrella ya que cumple con los requerimientos planteados y es el que mejor y de forma más sencilla se adapta a los requerimientos de los usuarios.

El esquema en estrella, consta de una tabla de hechos central y de varias tablas de dimensiones relacionadas a esta, a través de sus respectivas claves. Es el más simple de interpretar y optimiza los tiempos de respuesta. Su diseño es fácilmente modificable (Bernabeu Ricardo, 2010).

En la Figura 6 se puede apreciar el Esquema en Estrella estándar:

**Figura 6**  
*Esquema en estrella.*



*Nota*. (Bernabeu Ricardo, 2010)

### 2.3.2 Tablas de Dimensiones

En este paso se diseñan las tablas de dimensiones que forman parte del MD. Cada perspectiva definida en el modelo conceptual constituye una tabla de dimensiones. Para ello, a partir de cada perspectiva y sus campos se realiza el siguiente proceso:

* Se elige el nombre que identifique la tabla de dimensiones.
* Se añade un campo que represente su clave primaria.
* Se redefinen los nombres de los campos si es que no son lo suficientemente intuitivos.

Perspectiva **Motivo**:

* La nueva tabla de dimensión tendrá el nombre: **dim\_motivo**.
* Se le agrega una clave primaria con el nombre: id\_motivo.
* Se modifica el nombre del campo: tipo\_tramite\_estudiante, por: motivo\_tramite.

**Figura 7**  
*Dimensión: dim\_motivo*



Perspectiva **Tiempo**:

* La nueva tabla de Dimensión tendrá el nombre: **dim\_tiempo**.
* Se le agrega una clave primaria con el nombre: id\_ tiempo.
* Se modifica el nombre del campo: fecha\_creación, por: fecha.

La fecha se guardará con el mismo formato que es almacenada en la base de datos (timestamp with time zone).

**Figura 8**  
*Dimensión: dim\_tiempo*



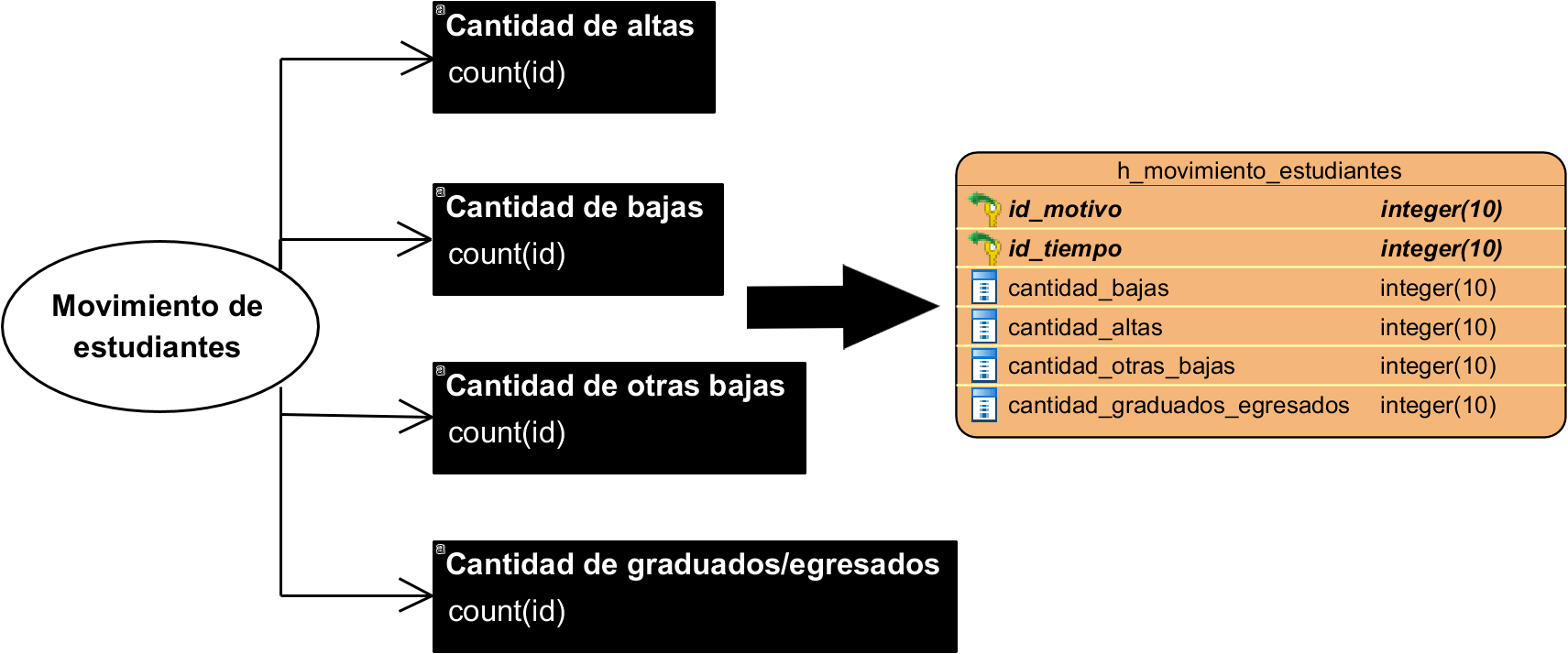
### 2.3.3 Tablas de Hechos

Una vez concebidas las tablas de dimensiones, se procede a definir las tablas de hechos, asignando nombres que representen la información que contienen. Se define además su clave primaria que se compone de la combinación de las llaves primarias de cada tabla de dimensión relacionada. Se crearon tantos campos de hechos como Indicadores se definieron en el modelo conceptual. A continuación, se confecciona la tabla de Hechos del tema: movimiento de estudiantes:

* La tabla de Hechos tendrá el nombre: **h\_movimiento\_estudiantes.**
* Su llave primaria es la combinación de las llaves primarias de las tablas de dimensiones.
* Se crearon tantos campos de hechos como indicadores se definieron en el modelo conceptual.

En el gráfico siguiente se aprecia lo antes descrito.

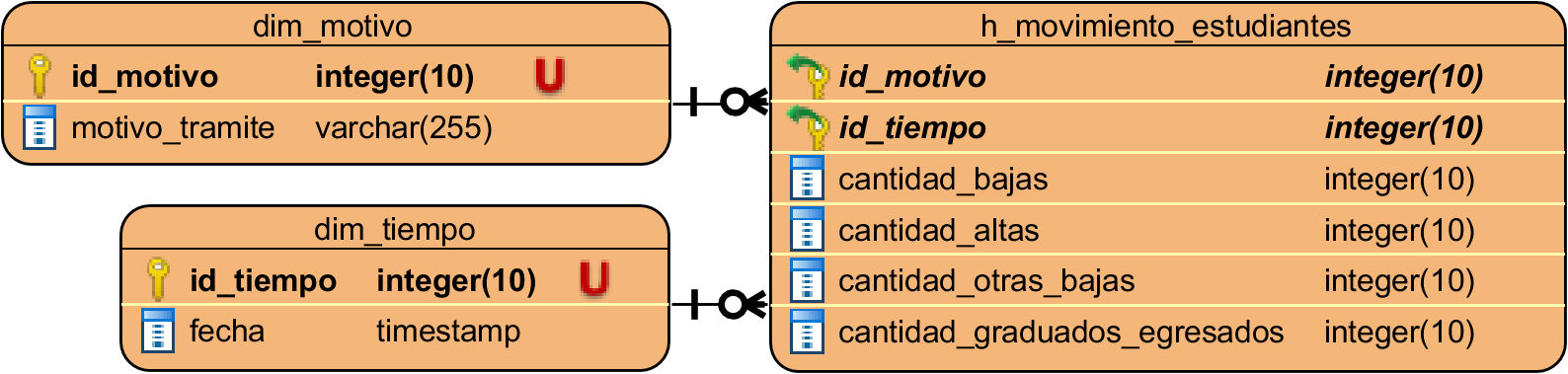
**Figura 9**  
*Diseño de la tabla de hechos*



### 2.3.4 Uniones

En la Figura 10 se aprecian las uniones correspondientes entre las tablas de dimensiones y la tabla de hecho mostrando de forma detallada las relaciones entre ellas. Con esta unión se obtiene el modelo lógico del MD

**Figura 10**  
*Uniones*



## 2.4 Integración de datos

Una vez construido el modelo lógico, se procede a poblarlo con datos, utilizando técnicas de limpieza y calidad de datos.

### 2.4.1 Carga inicial

La carga inicial, se refiere precisamente a la primera carga de datos que se le realiza al MD, poblando el modelo construido en pasos anteriores. Para lo cual se llevaron a cabo una serie de tareas básicas, tales como, asegurar la limpieza y calidad de los datos, por lo general, esta tarea consume un tiempo bastante considerable, ya que se deben insertar registros que han sido generados aproximadamente, y en casos ideales, durante más de cinco años. Se debe evitar que el MD sea cargado con *Missing Values* (valores faltantes), *Outliers* (datos anómalos) o falta de integridad; se debe establecer condiciones y restricciones para asegurar que solo se utilicen los datos de interés.

Primero se cargan los datos de las dimensiones y luego los de las tablas de hechos, esto se debe a la existencia de llaves o claves foráneas y se realiza para que las llaves foráneas en la tabla de hechos puedan hacer referencia a las dimensiones correctamente.

**Proceso ETL Principal**

En la Figura 11 se puede apreciar el proceso ETL principal planteado para la carga inicial del movimiento de estudiantes:

**Figura 11**  
*Proceso ETL principal para la carga inicial del MD*



Las tareas que lleva a cabo este proceso son:

* **Start**: inicia la ejecución de los pasos en el momento en que se le indique.
* **cargar\_motivo**: ejecuta el contenedor de pasos que carga la tabla de dimensión dim\_motivo (más adelante se detallará).
* **cargar\_tiempo**: ejecuta el contenedor de pasos que carga la tabla de dimensión dim\_tiempo (más adelante se detallará).
* **cargar\_movimientos**: ejecuta el contenedor de pasos que carga la tabla de hechos h\_movimiento\_estudiantes (más adelante se detallará).
* **Success**: determina si el flujo de trabajo se completó exitosamente.

A continuación, se detallan las tareas llevadas a cabo por el proceso ETL principal.

**cargar\_motivo**

A continuación, se especifican las tareas llevadas a cabo por cargar\_motivo. Este paso es una secuencia de comandos script, que incluye las siguientes tareas:

* Obtener los datos de la base de datos:

Se obtiene, a través de una consulta SQL, los datos necesarios para cargar la dimensión dim\_motivo. Como fuente de entrada se tomó la tabla tramite\_tbr\_tramite\_configuracion.

Se consultó con el cliente y se determinaron los principales campos o atributos de las estructuras. Esta es la sentencia SQL principal configurada en este paso: SELECT id, nombre AS motivo FROM tramite\_tbr\_tramite\_configuracion.

* Cargar de dimensión:

Este paso almacena en la tabla de dimensión dim\_motivo los datos obtenidos en el paso anterior.

**cargar\_tiempo**

Este paso es una secuencia de comandos script, que incluye las siguientes tareas:

* Obtener los datos de la base de datos:

Se toma como fuente de entrada la tabla tramite\_tbd\_tramite\_estudiante, de esta se selecciona la fecha\_creacion y se realizan una serie de operaciones para determinar los valores del campo id\_tiempo de la tabla:

id\_tiempo = año\*10000 + mes\*100 + día

La fecha se guardará con el mismo formato (timestamp with time zone) que está almacenada en la base de datos. Ejemplo de dato:

20211020; 2021-10-20 02:22:13.155492+02

* Cargar dimensión dim\_tiempo:

almacena en la tabla de dimensión dim\_tiempo los datos obtenidos en el paso anterior.

**cargar\_movimientos**

Este paso es una secuencia de comandos script, que incluye las siguientes tareas:

* Obtener los datos de la base de datos:

Se obtiene, a través de una consulta SQL, los datos necesarios para cargar la tabla h\_movimiento\_estudiantes. Como fuentes de entrada se tomaron varias tablas, entre ellas: tramite\_tbd\_tramite\_estudiante y tramite\_tbr\_tramite\_configuracion.

Se consultó con el cliente para averiguar cómo determinar los datos a tener en cuenta. A continuación, la sentencia SQL principal configurada en este paso:

**Figura 12**  
*Sentencia SQL principal para cargar la tabla d hechos*



* Cargar tabla h\_movimiento\_estudiantes almacena los datos obtenidos en el paso anterior.

### 2.4.2 Actualización

Para esta etapa del proceso, se definieron las políticas de actualización mediante un acuerdo entre el cliente y el equipo de desarrollo, donde se acordó lo siguiente:

* La información se refrescará todos los días a las 12:00 AM.
* Debido a los pocos cambios y variaciones en los datos, solo se cargarán los registros nuevos.
* Los datos de la dim\_tiempo se cargarán gradualmente.
* Estas acciones se ejecutan durante un período de pruebas, para analizar la manera más eficiente de realizar las actualizaciones, y si es necesario, en alguna medida, efectuar cambios en el proceso de ETL basado en el estudio de los cambios en los OLTP.

## Conclusiones del capítulo

La metodología Hefesto, diseñada específicamente para la construcción de un Mercado de Datos, se estructura de 4 pasos fundamentales. El análisis de requerimientos facilitó la recopilación precisa de los requisitos del cliente, posibilitando que las necesidades sean atendidas desde el inicio, asegurando que todas las funcionalidades críticas fueran identificadas y priorizadas, favoreciendo además la alineación del proyecto con los objetivos estratégicos de la organización, asegurando que el MD aporte valor real y tangible.

La comprensión profunda de los sistemas transaccionales existentes, estructura, contenido y flujo de datos facilitó la identificación de fuentes de datos relevantes para el MD, asegurando que toda la información necesaria esté disponible, detectando también posibles problemas de integridad y consistencia. Se aseguró la planificación eficiente de los procesos de extracción, transformación y carga (ETL).

La creación de un modelo lógico claro y de fácil interpretación facilitó la representación de las relaciones entre los diferentes datos y dimensiones del MD. Esto permite la identificación de posibles problemas en el diseño antes de la implementación física, permitiendo realizar ajustes y optimizaciones tempranas.

La integración de datos facilitó la implementación de procesos ETL robustos y eficientes, permitiendo la transformación de datos en información valiosa y utilizable en la toma de decisiones estratégicas. Proceso que se lleva a cabo de forma manual, el cual es propenso a errores.

# CAPÍTULO 3: Prueba del AD para el Sistema de Gestión Académica XAUCE AKADEMOS.

En este capítulo se explican las reglas y normas que se siguieron al escribir el código, se evidencian las pruebas aplicadas a la solución propuesta.

### 3.1 Estándares de codificación

Un estándar de codificación es una serie de reglas que determinan cómo debe escribirse el código. Los estándares de codificación ayudan a disminuir el esfuerzo necesario para escribir el código. A la vez, el uso de un estándar de codificación hace más fácil la lectura del código escrito por diferentes personas, lo que hace más sencillo el mantenimiento del código a largo plazo (mchojrin, 2022).

Para la implementación del algoritmo en cuestión se hace uso de los siguientes estándares de codificación definidos por los desarrolladores:

* Añadir comentarios descriptivos para explicar determinadas partes del código, en caso de ser necesario que expongan el objetivo del código.
* Elegir nombres descriptivos para las variables.
* Los nombres de variables y funciones deben ser inglés.
* En cuanto a las consultas SQL, escribir con letra mayúscula todas las palabras claves (reservadas) propias del lenguaje.
* Se dejará una línea en blanco inmediatamente antes del inicio de un bloque de código.
* Escribir espaciado, con el fin de mejorar la apariencia del código.
* Evitar el uso de más de una instrucción por línea de código.

**Figura 13**  
*Muestra del estándar de codificación empleado para las consultas SQL*



*Nota*. En la imagen se evidencian las palabras claves del lenguaje SQL escritas en letra mayúscula, para diferenciarlas fácilmente.

### 3.2 Pruebas aplicadas al Mercado de Datos.

Las pruebas de software son un proceso integral y crítico que implica evaluar y verificar que un producto o aplicación de software cumple con los requisitos especificados y realiza las funciones previstas de manera correcta y eficiente. Estas se utilizan para identificar y corregir posibles defectos o fallas en el producto o aplicación de software, asegurando así que el producto final sea de alta calidad y confiabilidad (¿Qué es la prueba de software y cómo funciona?, s. f.).

**Prueba integridad referencial**.

La integridad referencial es una serie de procesos que aseguran que los datos se almacenen y se utilicen uniformemente. Las reglas integradas en la estructura de la base de datos sobre cómo se utilizan claves foráneas para garantizar que tan solo se produzcan cambios, incorporaciones o supresiones de datos adecuados. Las reglas pueden incluir restricciones que eliminen la entrada de datos duplicados, aseguren que los datos son veraces y/o impidan la entrada de datos no pertinentes (Bermúdez León, 2020).

Se basa en la idea de que las relaciones entre claves primarias y foráneas deben ser coherentes y que no debe haber referencias a datos inexistentes, es crucial para mantener la consistencia de las relaciones entre tablas. La integridad referencial garantiza que solo se produzcan las modificaciones, adiciones o eliminaciones requeridas a través de reglas implantadas en la estructura de la base de datos sobre cómo se utilizan las claves externas.



Resultados de las pruebas recogidos en una gráfica o tabla

Integridad de la entidad.

Las pruebas de integridad de entidad se enfocan en la calidad y consistencia de los datos a nivel de cada registro individual dentro de la base de datos, depende de la elaboración de claves primarias. El propósito es garantizar que los datos no se registren varias veces (que cada elemento sea único) y que la tabla no tenga campos nulos (Naeem, 2020).



Resultados de las pruebas recogidos en una gráfica o tabla

### 3.3 Conclusiones del capítulo

En este capítulo, se establecieron estándares de codificación para mejorar la calidad del código. Se llevaron a cabo pruebas de integridad que confirmaron que el algoritmo y la base de datos cumplen con los requisitos

Conclusiones

Recomendaciones

Bibliografía

Anexos