

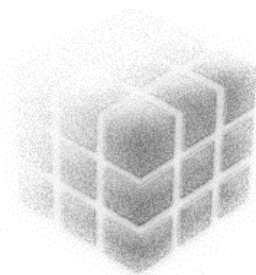


Universidad de las Ciencias Informáticas
Facultad 5

Centro de Consultoría y Desarrollo de Arquitecturas Empresariales

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

Título: Almacén de Datos para la Suite de Gestión de Proyectos Xedro-
GESPRO.



Autora:

Claudia Celeste Rivero Hechavarría.

Tutoras:

Ing. Claribel Rojas Luzardo.

Ing. Miroslava Lázara Aldana Cuza.

Ciudad de La Habana, Junio 2015

“Año 56 de la Revolución Cubana”



DECLARACIÓN DE AUTORÍA

Declaro ser autora de la presente tesis y recomiendo a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo. Para que así conste firmo la presente a los ____ días del mes de _____ del año 2015.

Firma de la Autora

Claudia Celeste Rivero Hechavarría

Firma de la Tutora

Ing. Clarisbel Rojas

Firma de la Tutora

Ing. Miroslava L. Aldana



Síntesis de la tutora.

Nombre y Apellidos: Ing. Clarisbel Rojas Luzardo

Especialidad: Ingeniero en Ciencias Informáticas.

Correo electrónico: clarisbel@uci.cu

Síntesis de la tutora.

Nombre y Apellidos: Ing. Miroslava Lázara Aldana Cuza.

Especialidad: Ingeniero en Ciencias Informáticas.

Correo electrónico: mlaldana@uci.cu



“El futuro de nuestra patria tiene que ser necesariamente un futuro de hombres de ciencia, tiene que ser un futuro de hombres de pensamiento, porque precisamente es lo que más estamos sembrando; lo que más estamos sembrando son oportunidades a la inteligencia (...)”

Fidel Castro Ruz



Dedico este trabajo a mi madre Gudelia, a mi padre Francisco y especialmente a mi hermana Yariannis (Lasiri).



Agradezco a:

Mi familia por todo el apoyo que me brindaron durante estos 5 años, especialmente a mi papá, mi mamá, mi hermana Lasiri, mi tío Pedro y mi padrino Cándido.

A mis abuelos por su cariño y a mi prima Ycannis (La Pipa) por estar siempre al pendiente de mí.

A mi novio Randy (Tuty), por el apoyo y la felicidad de estos años juntos.

A Lourdelina por sus consejos, cariño y ayuda incondicional en cada momento.

A mis tutoras Clarisbel y Miroslava por la ayuda brindada en el desarrollo del presente trabajo de diploma.

A mis compañeros de la universidad y mis profesores.



RESUMEN

En la Universidad de las Ciencias Informáticas (UCI) es empleada la suite Xedro-GESPRO como herramienta para el control y seguimiento de los proyectos informáticos de dicha institución. Xedro-GESPRO contaba con el Almacén Operacional de Datos (ODS por sus siglas del inglés *Operational Data Store*) Proyectos Terminados el cual almacenaba información histórica y resumida para ser consultada posteriormente. Sin embargo, el proceso de almacenamiento de los datos en el ODS era ineficiente e incitaba la obstaculización y debilitamiento del envío y recepción de la información almacenada.

Por tanto se desarrolló un almacén de datos a partir de la BD de Xedro-GESPRO, que constituye un repositorio para la información gestionada en Xedro-GESPRO. En el desarrollo de la solución se emplearon las estrategias de indexado y particionado de tablas. Estas estrategias garantizan un rendimiento aceptable de las peticiones que involucran el procesamiento de un elevado número de tuplas o filas de una tabla y mejor organización de la información.

El almacén de datos se desplegó en los centros de desarrollo de software de la UCI obteniéndose resultados satisfactorios en la carga de los datos desde la BD de Xedro-GESPRO, y en el tiempo de respuesta de las consultas. En el presente trabajo de diploma se abordaron los aspectos relacionados anteriormente.

PALABRAS CLAVE: almacén de datos, gestión de proyectos, particionado de tablas.

**ÍNDICE**

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	5
1.1 Gestión de proyectos.....	5
1.2 Estado actual de los almacenes de datos.....	6
1.3 Almacenes de datos.....	6
1.3.1 Características de los almacenes de datos.....	7
1.3.2 Arquitectura general de un almacén de datos.....	8
1.3.2.1 Componentes de la arquitectura de un almacén de datos.....	8
1.3.3 Estructura de los almacenes de datos.....	10
1.3.4 Enfoques de diseño de los almacén de datos.....	11
1.3.4.1 El particionamiento como enfoque de diseño.....	11
1.3.5 Ventajas de los almacenes de datos.....	11
1.3.6 Desventajas de los almacenes de datos.....	12
1.4 Modelo de datos.....	12
1.4.1 Modelo Entidad-Relación y Modelo Entidad-Relación Extendido.....	12
1.4.2 Modelo Dimensional.....	13
1.4.3 Modelo seleccionado.....	14
1.5 Modos de almacenamiento.....	14
1.5.1 Sistema de Procesamiento Analítico en Línea Relacional.....	14
1.5.2 Sistema de Procesamiento Analítico en Línea Multidimensional.....	15
1.5.3 Sistema de Procesamiento Analítico en Línea Híbrido.....	15
1.5.4 Modo de almacenamiento seleccionado.....	15
1.6 Metodología de desarrollo.....	16
1.6.1 Justificación de la metodología seleccionada.....	18
1.7 Herramientas para el desarrollo del almacén de datos de Xedro-GESPRO.....	19
1.7.1 Herramienta para el modelado.....	19
1.7.2 Sistema Gestor de Base de Datos.....	19
1.7.2.1 PostgreSQL Server 9.2.....	20
1.7.3 Administrador de base de datos.....	20
1.7.3.1 PgAdmin III.....	20
1.7.4 Lenguaje que emplea el sistema gestor de base de datos.....	21
1.8 Conclusiones del capítulo.....	21



CAPÍTULO 2: ANÁLISIS Y DISEÑO DEL ALMACÉN DE DATOS DE XEDRO-GESPRO.	23
2.1 Definición del negocio.....	23
2.2 Temas de análisis.....	25
2.3 Reglas del negocio.	26
2.4 Requisitos funcionales.....	26
2.5 Requisitos no funcionales.....	28
2.6 Diagrama de caso de uso.	28
2.6.1 Descripción de caso de uso.	30
2.7 Arquitectura del almacén de datos de Xedro-GESPRO.....	33
2.8 Diseño del subsistema de almacenamiento.....	33
2.8.1 Dimensiones del modelo físico de datos.	34
2.8.2 Hecho del modelo físico de datos.	34
2.8.3 Matriz bus o matriz dimensional.	34
2.8.4 Modelo físico de datos.....	35
2.8.5 Modelo lógico de datos.....	39
2.9 Diseño del subsistema de integración.	41
2.10 Conclusiones del capítulo.	42
CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBA DEL AD DE XEDRO-GESPRO.....	43
3.1 Implementación del subsistema de almacenamiento.....	43
3.2 Implementación del subsistema de integración.	43
3.2.1 Proceso de extracción de los datos.....	45
3.2.2 Proceso de transformación de los datos.	45
3.2.3 Proceso de carga de los datos.	48
3.3 Pruebas aplicadas al almacén de datos de Xedro-GESPRO.	52
3.3.1 Resultados de las pruebas.	53
3.4 Impacto social.	58
3.5 Conclusiones del capítulo.	59
CONCLUSIONES GENERALES	60
RECOMENDACIONES.....	61
REFERENCIAS BIBLIOGRÁFICAS	62
ANEXOS.....	65



ÍNDICE DE FIGURAS Y TABLAS

Figura 1. Arquitectura general de un almacén de datos (Darío, 2010).	8
Figura 2. Ciclo de vida de “Metodología de desarrollo de almacenes de datos en DATEC” (Hernández, 2012).	17
Figura 3. Diagrama de caso de uso.	29
Figura 4. Arquitectura del almacén de datos de Xedro-GESPRO.	33
Figura 5. Modelo físico de datos del almacén de datos de Xedro-GESPRO.	36
Figura 6. Modelo lógico de datos del almacén de datos de Xedro-GESPRO.	40
Figura 7. Vista del almacén de datos de Xedro-GESPRO en la herramienta pgAdmin III.	43
Figura 8. Proceso ETL.	44
Figura 9. Vista de las particiones de la tabla gespro_projects_risks del almacén de datos de Xedro-GESPRO.	50
Figura 10. Rendimiento del almacén de datos de Xedro-GESPRO según condiciones de carga.	53
Figura 11. Rendimiento del almacén de datos de Xedro-GESPRO vs Proyectos Terminados.	54
Tabla 1. Descripción CU 1: Administrar los datos históricos de los riesgos de los proyectos.	33
Tabla 2: Dimensiones del modelo dimensional.	34
Tabla 3: Matriz dimensional.	35
Tabla 4. Funciones del subsistema de integración.	42
Tabla 5. Conversión de datos para la tabla gespro_projects_risks.	45
Tabla 6. Renombramiento de atributos para la tabla gespro_projects_risks.	46
Tabla 7. Promedio del tiempo de respuesta del almacén de datos de Xedro-GESPRO. ..	56
Tabla 8. Lista de chequeo. Evaluación de la funcionalidad del almacén de datos de Xedro-GESPRO (Ocampo Acosta, y otros, 2011).	57
Tabla 9: Lista de chequeo. Evaluación del RNF de soporte del almacén de datos de Xedro-GESPRO (Ocampo Acosta, y otros, 2011).	58



INTRODUCCIÓN

“Un proyecto es el esfuerzo temporal llevado a cabo con el fin de crear un producto, servicio, o resultado único (PMI, 2013).” La creación de estos demanda un conjunto de actividades a las cuales se aplican conocimientos, habilidades, herramientas y técnicas para cumplir con los requisitos del mismo (PMI, 2013). Este proceso se conoce como Gestión de Proyectos (GP).

El proceso de la GP se logra mediante la aplicación e integración adecuada de cuarenta y siete procesos, los cuales se agrupan en diez Áreas de Conocimiento (AC). Un área de conocimiento representa un conjunto completo de conceptos, términos y actividades que conforman un ámbito profesional, un ámbito de la GP o un área de especialización (PMI, 2013).

La GP permite la obtención de resultados con calidad y costes apropiados debido al control continuo del proceso de obtención de los mismos. Esto promueve la necesidad en las organizaciones, empresas o instituciones de concientizarse en una forma de trabajo por proyecto (Pampliega, 2014).

Empresas internacionales como Oracle, Microsoft y Google sustentan su desarrollo económico en la ejecución de proyectos informáticos. De la misma forma Cuba se proyecta a un futuro innovador, donde instituciones como Desoft, Segurmática y la Universidad de las Ciencias Informáticas (UCI) se encuentran inmersas en la creación de este tipo de proyectos.

Herramientas tecnológicas como *GanttProject*, *OpenProj*, *Basecamp* y *Redmine* se desarrollaron para apoyar el proceso de la GP en empresas orientadas al desarrollo de proyectos. Estas herramientas manejan una considerable cantidad de datos, aun cuando no permiten el control total del proceso de la GP, pues no ofrecen soporte a todas las AC. 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 Consultoría y Desarrollo de Arquitecturas Empresariales (CDAE), que posee el Departamento de GP, donde se desarrolla el ecosistema Xedro-GESPRO.

El ecosistema Xedro-GESPRO es una suite para la GP usada actualmente en la UCI y en varias organizaciones externas a la universidad, que trabajan con el concepto de proyecto. Dicha suite está conformada por un sistema informático y un paquete de servicios para la formación de competencias profesionales en GP (Piñero Pérez, 2014).



La suite Xedro-GESPRO cuenta con una base de datos (BD) transaccional por cada centro de desarrollo de la UCI, donde se almacena la información gestionada en la misma. Para el almacenamiento histórico y resumido de los datos contenidos en dichas bases de datos, las versiones anteriores a la versión 15.05 de Xedro-GESPRO emplean el Almacén Operacional de Datos (ODS por sus siglas del inglés *Operational Data Store*) Proyectos Terminados.

Cabe destacar que el ODS presenta crecimiento exponencial del volumen de datos. En este se almacena la información correspondiente a las competencias por roles, las competencias por recursos humanos, los usuarios y errores de la suite y la evaluación de los usuarios según la tarea realizada por este. Otros datos cargados en el ODS son los referentes a los proyectos, tales como: los cambios, los recursos materiales, los costos, los recursos humanos, las desviaciones, los requerimientos, los riesgos, los roles, las tareas y los indicadores.

En el ODS se definen un conjunto de funciones, donde cada una se encarga de realizar los procesos de extracción, transformación y carga de los datos contenidos en la BD de Xedro-GESPRO. Algunas funciones fracasan porque al ejecutarse el proceso de extracción de los datos no existe correspondencia entre los nombres de las tablas y atributos de la BD de Xedro-GESPRO y el ODS. Dentro del proceso de transformación de los datos no se realizan las conversiones de tipos de datos provenientes de la BD de Xedro-GESPRO, ocasionando que no se correspondan con los tipos de datos de los atributos en las tablas del ODS.

Los problemas en los procesos de extracción y transformación conllevan a que no se ejecute el proceso de carga de los datos en el ODS, lo cual provoca la existencia de tablas vacías. De igual forma incitan la pérdida de la información a almacenar en el ODS. Otra dificultad que se presenta es la ambigüedad de la información almacenada en las tablas que contienen datos, debido a que no se establece un rango de fecha para el almacenamiento, de manera que se eviten dichas ambigüedades. Esto genera como inconveniente el crecimiento exponencial de dichas tablas, lo que afecta el proceso de consultas sobre el ODS, provocando largos tiempos de respuesta.

Por todo lo antes planteado se identifica como **problema de investigación**: ¿Cómo organizar la información de la suite de GP Xedro-GESPRO, de manera que se fortalezcan los procesos de lectura y recepción de la misma?



Se define como **objeto de estudio** de la presente investigación los Almacenes de Datos.

El **campo de acción** comprende los Almacenes de Datos en la GP en Xedro-GESPRO.

Para dar solución al anterior problema de investigación se propone el siguiente **objetivo general**:

Desarrollar un almacén de datos que facilite los procesos de lectura y recepción de la información contenida en la base de datos de la suite Xedro-GESPRO.

Para dar cumplimiento al objetivo propuesto se plantean las siguientes tareas de investigación:

1. Elaborar el marco teórico de la investigación a partir del estudio y selección de las herramientas y tecnologías necesarias para la construcción del almacén de datos.
2. Realizar el análisis y el diseño del almacén de datos a partir de la BD de Xedro-GESPRO.
3. Implementar las funciones en plsql para la carga de datos desde la BD de Xedro-GESPRO al almacén de datos.
4. Crear el almacén de datos utilizando la herramienta PostgreSQL 9.2.
5. Probar el rendimiento del almacén de datos.
6. Validación de la implantación del almacén de datos en la herramienta Xedro-GESPRO.

A partir de los elementos planteados se arriba a la siguiente **idea a defender**:

Desarrollando un almacén de datos para organizar la información contenida en la BD de la suite Xedro-GESPRO, se fortalecen los procesos de lectura y recepción de la misma.

Como parte del trabajo investigativo a realizarse, se utilizan los siguientes métodos científicos:

- ❖ **Histórico – lógico**: para la determinación de los antecedentes en su devenir histórico, tendencias y regularidades del objeto de estudio (Almacenes de Datos) y el campo de acción (Almacenes de Datos en la GP en Xedro-GESPRO). Se empleó con el objetivo de analizar la fuente de origen de los datos y el ODS.
- ❖ **Inductivo – deductivo**: se empleó con la finalidad de interpretar el ODS y los almacenes de datos, de manera que fuese posible llegar a una generalización de conceptos que sustenten la solución al problema planteado.
- ❖ **Modelación**: para el diseño del almacén de datos.

La presente investigación consta de tres capítulos, los cuales se sintetizan a continuación.



Capítulo 1: “Fundamentación Teórica”.

En este capítulo se fundamentan teóricamente los aspectos importantes de un almacén de datos como son: su concepto, características, modos de almacenamiento, arquitectura general, enfoques de diseño. Se define la metodología de desarrollo para la solución y se selecciona el entorno de desarrollo para la construcción del almacén de datos.

Capítulo 2: “Análisis y Diseño del Almacén de Datos”.

En este capítulo se realiza un estudio del negocio con el objetivo de determinar las necesidades de información. A partir de dichas necesidades se identifican los requisitos funcionales, los cuales se agrupan por casos de uso. Cada caso de uso es descrito con el objetivo de establecer las acciones principales del proceso a desarrollar. Se define la arquitectura de la propuesta de solución y se diseñan los subsistemas de almacenamiento e integración.

Capítulo 3: “Implementación y prueba del Almacén de Datos”.

Este capítulo se refiere a la implementación de la solución, abordando específicamente cómo se realiza la implementación de los subsistemas de almacenamiento e integración, teniendo en cuenta los requisitos y necesidades del negocio. De igual forma se especifican las pruebas realizadas y los resultados arrojados por estas. Además se exponen los elementos de despliegue de la solución.



CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.

En este capítulo se introduce el tema de la GP, así como también los conceptos y aspectos importantes de los almacenes de datos como sistema de apoyo a la toma de decisiones. Se determina el modo de almacenamiento y se realiza un análisis de la arquitectura general de un almacén de datos. Se seleccionan las herramientas y tecnologías necesarias conjuntamente con la metodología para el desarrollo de la presente investigación.

1.1 Gestión de proyectos.

Dirigir un proyecto por lo general implica identificar requisitos, abordar las diversas necesidades, inquietudes y expectativas de los interesados en la planificación y la ejecución del proyecto. Involucra acciones como establecer, mantener y realizar comunicaciones activas, eficaces y de naturaleza colaborativa entre los interesados. Otra actividad que abarca es la gestión de los interesados para cumplir los requisitos del proyecto y generar los entregables del mismo. Al mismo tiempo debe equilibrar las restricciones contrapuestas del proyecto que incluye el alcance, la calidad, el cronograma, el presupuesto, los recursos y los riesgos (PMI, 2013).

En la GP se emplean herramientas tecnológicas que han sido desarrolladas para proporcionar eficiencia en el trabajo de equipo. Tales herramientas ofrecen una estructura, flexibilidad y control necesario a los miembros del equipo, para alcanzar resultados extraordinarios dentro del tiempo y presupuesto establecido.

Xedro-GESPRO es una de estas herramientas e incluye entre sus funcionalidades: la gestión de portafolios de proyectos, la gestión de alcance, la gestión de tiempo, la gestión de riesgos, la gestión de comunicaciones, la gestión de la calidad, la gestión logística, el control y seguimiento de proyectos, el control de versiones y la gestión documental (Piñero Pérez, 2014).

Esta facilita la generación dinámica de reportes y la toma de decisiones a partir de un tablero de control que combina indicadores, gráficos y sistemas de información geográfica. Permite la gestión de indicadores de proyectos y la evaluación de los mismos a partir de sistemas de inferencia borrosos. Establece un sistema de toma de decisiones en forma de cascada con el objetivo de disminuir el tiempo de control y seguimiento, aumentando su eficiencia (Piñero Pérez, 2014).



Xedro-GESPRO ofrece muchas facilidades, pero no brinda información histórica y resumida y que al mismo tiempo posibilite la obtención de los datos requeridos en un mínimo tiempo. Los almacenes de datos son sistemas de almacenamiento que facilitan el almacenamiento de la información histórica y resumida. A estos se les puede aplicar técnicas que aumenten el rendimiento de la ejecución de consultas sobre ellos.

1.2 Estado actual de los almacenes de datos.

Estos surgen como una tecnología de soporte a la toma de decisiones para las organizaciones, instituciones y empresas como Facebook, HP y Google. Google contaba con el almacén maestro/esclavo para el almacenamiento de la información generada por la plataforma *Google App Engine* (GAE), la cual permite el alojamiento y desarrollo de sus aplicaciones (Google, 2015).

Pero decidió prescindir de este, debido a que presentó problemas de escalabilidad. El sustituto para este fue el Almacén de Datos de Alta Replicación (HRD por sus siglas del inglés *High Replication Datastore*), el cual desde su lanzamiento no ha demostrado problemas de escalabilidad (Soloelectronicos, 2015).

Facebook lanzó el almacén de datos escalable RocksDB de código abierto, desarrollado para servir contenido a sus 1,2 mil millones de usuarios. Este no tiene la capacidad de indexar los datos de sus tablas ni soporta las consultas SQL (Jackson, 2013).

HP desarrolló el almacén de datos *Vertica OnDemand*. Está orientado al almacenamiento de información para proveedores de comunicaciones y servicios tales como correo electrónico, video, Servicio de Mensajes Cortos (SMS por sus siglas del inglés *Short Message Service*) y Servicio de Mensajería Multimedia (MMS por sus siglas del inglés *Multimedia Messaging Service*). Presentando un procesamiento masivo en paralelo para una escalabilidad extrema (HP Website, 2015).

Los elementos anteriormente expuestos constituyen un punto de partida para la creación del almacén de datos de Xedro-GESPRO, pues demuestran ser altamente escalables aunque los almacenes de datos HRD y *Vertica OnDemand* están orientados a satisfacer necesidades de información que no se corresponden con la GP. En el caso del almacén de datos RocksDB no tiene capacidades de indexación ni soporta consultas SQL.

1.3 Almacenes de datos.



Los almacenes de datos surgieron en la década de los 90 del siglo pasado. Bill Inmon y Ralph Kimball son dos grandes autores que han dedicado sus estudios a este tema. Bill Inmon plantea que: *“un almacén de datos es una colección de datos orientados al dominio, integrados, no volátiles y variables en el tiempo, organizados para dar apoyo al proceso de toma de decisiones”* (Inmon, 2005).

Ralph Kimball lo define como: *“un sistema que extrae, limpia, ajusta e integra los datos del origen en un almacenamiento de datos dimensionales y luego apoya e implementa la consulta y el análisis con el fin de la toma de decisiones”* (Kimball, 2004). También lo define como *“la unión de todos los mercados de datos de una entidad”*, donde los Mercados de Datos (DM por sus siglas del inglés *Data Marts*) son un repositorio de información, similar a un almacén de datos, pero orientado a un área o departamento específico de la organización (Hernández, 2012).

Analizados ambos conceptos se entiende como almacén de datos (AD) el sistema que extrae, limpia, ajusta e integra los datos del origen en un almacenamiento de datos dimensionales, donde dicha colección de datos estará orientada a temas y permanecerán no volátiles y variables en el tiempo, con el fin de apoyar la toma de decisiones.

1.3.1 Características de los almacenes de datos.

Bill Inmon define un conjunto de características para identificar a los almacenes de datos las cuales son brevemente argumentadas a continuación.

Los almacenes de datos son:

Orientado a temas: constituye una forma de organizar los datos de acuerdo al interés del usuario y en correspondencia a las facilidades de consulta de la información por parte de este (Inmon, 2005).

Integrado: el AD en su proceso de integración, limpia y transforma los datos cargados desde diferentes sistemas operacionales y los almacena consistentemente (Inmon, 2005).

No volátil: los almacenes de datos solo pueden ser consultados, lo que significa que no pueden realizarse modificaciones sobre ellos. Una vez que se desea registrar nuevos datos, estos son insertados a continuación del registro anterior, lo cual permite mantener un registro histórico de estos (Inmon, 2005).

Variables en el tiempo: en un AD los datos deben incrementarse periódicamente por lo cual el tiempo constituye un aspecto importante a la hora de realizarse un registro, y por



tanto resulta necesario establecer un instante o intervalo de tiempo a los datos (Inmon, 2005).

1.3.2 Arquitectura general de un almacén de datos.

La arquitectura de un almacén de datos comprende un conjunto de componentes relacionados entre sí, los cuales desempeñan determinada función dentro del propio sistema. En la figura 1 se muestra la arquitectura general de un AD según Bernabeu R. Darío.

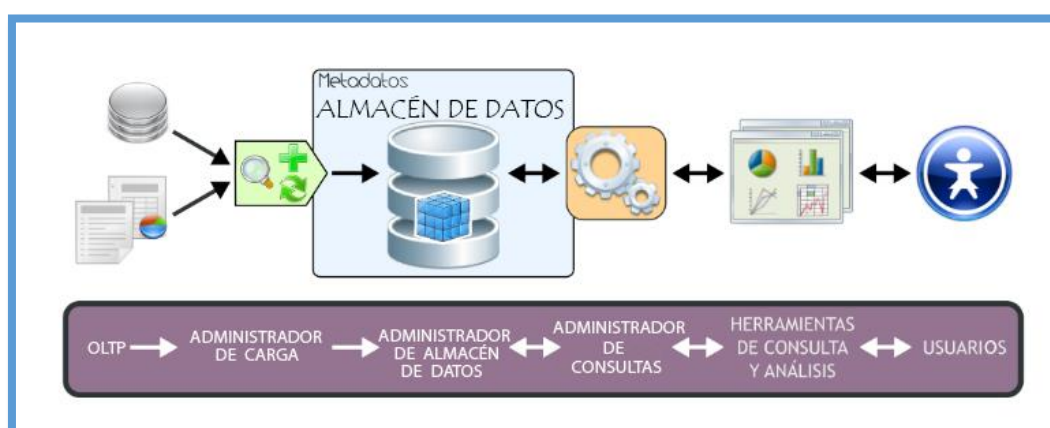


Figura 1. Arquitectura general de un almacén de datos (Darío, 2010).

Bernabeu R. Darío en su publicación “DATA WAREHOUSIN: Investigación y Sistematización de Conceptos”, plantea que la forma de operar del esquema anterior es la siguiente:

- ❖ Los datos son extraídos desde diferentes fuentes de datos. Esta información generalmente reside en diferentes tipos de sistemas, orígenes y arquitecturas y tienen formatos muy variados (Darío, 2010).
- ❖ Los datos son integrados, transformados y limpiados, para luego ser cargados en el AD (Darío, 2010).
- ❖ Principalmente, la información del AD, se estructura en cubos multidimensionales, ya que estos preparan esta información para responder a consultas dinámicas con un buen rendimiento (Darío, 2010).
- ❖ Los usuarios acceden a los cubos multidimensionales del AD utilizando diversas herramientas de consulta, exploración, análisis y reportes (Darío, 2010).

1.3.2.1 Componentes de la arquitectura de un almacén de datos.



- ❖ Procesamiento Transaccional en Línea (OLTP por sus siglas del inglés *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. (Darío, 2010).

Estas fuentes de información, son de características muy disímiles entre sí, en formato, procedencia y función. Entre los OLTP más habituales que pueden existir en cualquier organización se encuentran:

- Hojas de cálculo.
 - BD transaccionales.
 - Informes semanales, mensuales o anuales.
 - Hipertextos.
 - Archivos de texto.
- ❖ 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, y que luego son integrados y transformados con el objetivo de eliminar las inconsistencias en estos para ser posteriormente cargados en el AD (Darío, 2010).
 - ❖ Administrador de AD: almacena los datos de forma multidimensional, es decir, a través de tablas de hechos y tablas de dimensiones, gestiona y mantiene metadatos. Es el encargado de: transformar e integrar los datos fuentes y del almacenamiento intermedio en un modelo adecuado para la toma de decisiones, realizar todas las funciones de definición y manipulación del depósito de datos, para poder soportar todos los procesos de gestión del mismo (Darío, 2010).
 - ❖ Administrador de Consulta: este componente realiza las operaciones necesarias para soportar los procesos de gestión y ejecución de consultas relacionales. Este recibe las consultas de los usuarios, las aplica a la estructura de datos correspondiente (cubo multidimensional, Business Models) y devuelve los resultados obtenidos. (Darío, 2010).
 - ❖ Herramientas de Consulta y Análisis: son sistemas que permiten a los usuarios realizar la exploración de datos del AD. Básicamente constituyen el nexo entre el depósito de datos y los usuarios. Utilizan la metadata de las estructuras de datos que han sido creadas previamente para trasladar a través de consultas SQL los



requerimientos de los usuarios, para luego devolver el resultado obtenido. (Darío, 2010).

- ❖ Usuarios: son los que interactúan con las herramientas de consulta y análisis para obtener información almacenada en el AD (Darío, 2010).

Para el AD a desarrollar en el marco de la presente investigación se define una arquitectura que incluirá los componentes OLTP, Administrador de Carga y Administrador de AD. Los componentes Administrador de Consultas, Herramientas de Consulta y Análisis y Usuarios son precedentes a esta investigación. Más adelante se abordarán las características específicas de la arquitectura de la solución propuesta.

1.3.3 Estructura de los almacenes de datos.

De acuerdo con el nivel de detalle que posean los datos, se organizan en diferentes niveles en el almacén. Estos niveles de datos se argumentan a continuación:

- ❖ Detalle de datos actuales: son aquellos que reflejan las ocurrencias más recientes. Generalmente se almacenan en disco, aunque su administración sea costosa y compleja, con el fin de conseguir que el acceso a la información sea sencillo y veloz, ya que son bastante voluminosos. Su gran tamaño se debe a que los datos residentes poseen el más bajo nivel de granularidad, o sea, se almacenan a nivel de detalles (Darío, 2010).
- ❖ Detalle de datos históricos: representan aquellos datos antiguos, que no son frecuentemente consultados. También se almacenan a nivel de detalle, normalmente sobre alguna forma de almacenamiento externa, ya que son muy pesados y en adición a esto, no son requeridos con mucha periodicidad. Este tipo de datos son consistentes con los detalles de datos actuales (Darío, 2010).
- ❖ Datos ligeramente resumidos: son los que provienen desde un bajo nivel de detalle y agrupan los datos bajo algún criterio o condición de análisis. Habitualmente son almacenados en disco (Darío, 2010).
- ❖ Datos altamente resumidos: son aquellos que compactan aún más a los datos ligeramente resumidos. Se guardan en disco y son muy fáciles de acceder (Darío, 2010).
- ❖ Metadatos: representan la información acerca de los datos. De muchas maneras se sitúa en una dimensión diferente al de otros datos del AD, pues su contenido no es tomado directamente desde el ambiente operacional (Darío, 2010).



1.3.4 Enfoques de diseño de los almacén de datos.

Existen diferentes enfoques de diseños para los almacenes de datos. En la cuarta edición del libro *Building the Data Warehouse*, Inmon se refiere a la granularidad de datos y al particionamiento de datos como las dos más importantes decisiones de diseño.

Con respecto a la granularidad plantea que este es el aspecto más importante en lo que respecta al diseño de almacenes de datos, pues representa el nivel de detalle con que se puede almacenar información referente a un negocio. Para obtener mayores posibilidades analíticas es preciso determinar un mayor nivel de granularidad de los datos (Inmon, 2005).

Sobre el particionamiento plantea: *“la partición de datos se refiere a la ruptura de los datos en unidades físicas separadas que pueden ser manejados de manera independiente”* (Inmon, 2005).

1.3.4.1 El particionamiento como enfoque de diseño.

El particionamiento es empleado en un AD para dividir una tabla en varias tablas más pequeñas según un criterio definido. Dos razones por las cuales se emplea el particionamiento en un AD son:

- ❖ Posibilitar un fácil y optimizado mantenimiento del AD y de su proceso ETL (Darío, 2010).
- ❖ Aumentar el rendimiento de las consultas (Darío, 2010).

Con el particionado de una tabla se reducen el número de registros en la misma y por tanto se limita su crecimiento. Posibilita mejor organización de la información y la ejecución de consultas eficientemente. A partir de las ventajas expresadas sobre el particionado de tablas, se define el particionamiento como enfoque de diseño para la solución propuesta.

1.3.5 Ventajas de los almacenes de datos.

Algunas de las ventajas más significativas de los almacenes de datos son:

- ❖ Transforma datos orientados a las aplicaciones en información orientada a la toma de decisiones (Darío, 2010).
- ❖ Integra y consolida diferentes fuentes de datos (internas y/o externas) y departamentos empresariales, que anteriormente formaban islas, en una única plataforma sólida y centralizada (Darío, 2010).



- ❖ Aumento de la eficiencia de los encargados de tomar decisiones (Darío, 2010).
- ❖ Mejora la entrega de información, es decir, información completa, correcta, consistente y accesible. Información que los usuarios necesitan, en el momento adecuado y en el formato apropiado (Darío, 2010).

1.3.6 Desventajas de los almacenes de datos.

El empleo de un AD también presenta inconvenientes, los cuales son:

- ❖ Requiere una gran inversión, debido a que su correcta construcción no es tarea sencilla y consume muchos recursos, además, su implementación implica desde la adquisición de herramientas de consulta y análisis, hasta la capacitación de los usuarios (Darío, 2010).
- ❖ Infravaloración de los recursos necesarios para la captura, carga y almacenamiento de los datos (Darío, 2010).
- ❖ Infravaloración del esfuerzo necesario para su diseño y creación (Darío, 2010).
- ❖ Incremento continuo de los requerimientos de los usuarios (Darío, 2010).

A pesar de las desventajas expuestas, cabe destacar que el desarrollo de la propuesta de solución permitirá el almacenamiento de los datos gestionados en Xedro-GESPRO de forma resumida y que permanecerán no volátiles. Dicha solución tendrá un diseño orientado a la facilidad y rapidez de acceso de la información.

1.4 Modelo de datos.

Un modelo de datos es una definición lógica, independiente y abstracta de los objetos, operadores, que en conjunto constituyen la máquina abstracta con la que interactúan los usuarios. Los objetos nos permiten modelar la estructura de los datos. Los operadores nos permiten modelar su comportamiento (Date, 2001).

1.4.1 Modelo Entidad-Relación y Modelo Entidad-Relación Extendido.

Actualmente existen diversos modelos de datos. Uno de ellos es el Modelo Entidad-Relación (MER), el cual ha ganado gran aceptación como instrumento para modelar un negocio en el proceso de diseño de una BD. Este patrón representa un esquema de BD mediante entidades y asociaciones. Además describe a la BD de una forma sencilla y global, debido a que se crea de acuerdo a los requisitos de los datos propuestos para conformar el sistema operacional.



El Modelo Entidad–Relación Extendido (MER/E) incluye todos los conceptos del MER. En adición, incluye los conceptos de subclase y superclase además de los conceptos de especialización y generalización. Otro concepto incluido en el MER/E es la categorización. Asociado con estos conceptos se encuentra el mecanismo de herencia en el atributo (Date, 2001). Cuando se emplea un MER/E las tablas se tratan independientemente, lo que trae consigo el uso de varios *joins* y esto conlleva a que el proceso de consulta sobre el sistema sea lento y costoso.

1.4.2 Modelo Dimensional.

El Modelo Dimensional (MD), ha sido defendido por Ralph Kimball y ha ganado gran auge en la construcción de almacenes de datos. El MD es una técnica de diseño lógico que permite presentar los datos de manera que el acceso a estos sea de alto rendimiento. Cada MD se compone de una tabla con una clave de varias partes, llamada tabla de hechos, y un conjunto de tablas más pequeñas llamadas tablas de dimensiones. Cada tabla de dimensiones tiene una clave principal que corresponde exactamente a uno de los componentes de la clave de varias partes de la tabla de hechos.

Los hechos contienen atributos de hecho o medida a analizar (Trujillo, y otros, 2011). Es decir, representan una actividad del negocio constituida por un conjunto de indicadores para su análisis.

Una dimensión se compone de una serie de atributos organizados jerárquicamente. Estos atributos permiten analizar las medidas de los hechos a diferente nivel de detalle según se agreguen o desagreguen los datos (Trujillo, y otros, 2011).

1.4.2.1 Tipos de modelamientos de un almacén de datos.

Estrella: 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 ante las consultas de los usuarios. Su diseño es fácilmente modificable (Darío, 2010).

Esquema Copo de Nieve: Representa una extensión del modelo en estrella cuando las tablas de dimensiones se organizan en jerarquías de dimensiones. Presenta una tabla de hechos central que está relacionada con una o más tablas de dimensiones, quienes a su vez pueden estar relacionadas o no con una o más tablas de dimensiones. Posee mayor



complejidad en su estructura y al existir muchas uniones y relaciones entre tablas, el desempeño puede verse reducido (Darío, 2010).

Esquema Constelación: Este modelo está compuesto por una serie de esquemas en estrella. En este tipo de esquema no es necesario que las diferentes tablas de hechos compartan las mismas tablas de dimensiones porque las tablas de hechos auxiliares pueden vincularse con solo algunas de las tablas de dimensiones asignadas a la tabla de hechos principal, y también pueden hacerlo con nuevas tablas de dimensiones. Su diseño y cualidades son muy similares a las del esquema en estrella, pero a diferencia de estos el esquema constelación permite tener más de una tabla de hechos. Esta particularidad posibilita analizar más aspectos claves del negocio con un mínimo esfuerzo adicional de diseño. Contribuye a la reutilización de las tablas de dimensiones, ya que una misma tabla de dimensión puede utilizarse para varias tablas de hechos (Darío, 2010).

1.4.3 Modelo seleccionado.

Para la construcción de un AD no es conveniente el uso de los modelos MER o MER/E debido a que ambos están enfocados a modelar sistemas transaccionales, no representando así las propiedades multidimensionales de un AD como son el hecho, las dimensiones y jerarquías de atributos de dimensiones. Se determina para la propuesta de solución utilizar el MD, modelándolo mediante el esquema en estrella, que optimiza los tiempos de respuesta ante las consultas de los usuarios y es fácilmente modificable.

1.5 Modos de almacenamiento.

El Procesamiento Analítico en Línea (OLAP por sus siglas del inglés *OnLine Analytical Processing*) es el método ágil y flexible para organizar datos, especialmente metadatos, sobre un objeto o jerarquías de objetos como en un sistema u organización multidimensional. Este tiene como objetivo recuperar y manipular datos y combinaciones de los mismos a través de consultas o incluso informes (Díaz, 2010).

1.5.1 Sistema de Procesamiento Analítico en Línea Relacional.

Procesamiento Analítico en Línea Relacional (ROLAP por sus siglas del inglés *Relational OnLine Analytical Processing*), es un tipo de OLAP implementado sobre tecnología relacional. Según Bernabeu R. Darío este tipo de organización física cuenta con todos los beneficios de un sistema gestor de BD relacional a los cuales se les provee extensiones y herramientas para poder utilizarlo como un Sistema Gestor de AD (Darío, 2010).



En este sistema los cubos son calculados y creados dinámicamente, y finalmente mostrados al usuario en un formato multidimensional, luego de enviada una petición de análisis multidimensional al motor ROLAP y posterior selección de los elementos que compondrán el cubo (Darío, 2010).

La posibilidad de generar dinámicamente los cubos provee de mucha flexibilidad. No obstante es preciso resaltar que esta oportunidad conlleva a que ROLAP no sea tan eficiente al dar respuesta a las consultas realizadas por los usuarios (Darío, 2010).

1.5.2 Sistema de Procesamiento Analítico en Línea Multidimensional.

En los sistemas de Procesamiento Analítico en Línea Multidimensional (MOLAP por sus siglas del inglés *Multidimensional OnLine Analytical Processing*), a diferencia de los sistemas ROLAP, se crean y se calculan los cubos previamente. Debido a esto el tiempo de respuesta de las consultas realizadas por los usuarios es corto (Darío, 2010).

Con respecto al almacenamiento y gestión de los datos en el AD, los sistemas gestores de BD son utilizados para ambas tareas, sin embargo para la creación y administración de los cubos se precisa de un software para realizar el trabajo multidimensionalmente (Darío, 2010).

La disminución de flexibilidad constituye una desventaja para este procesamiento y es debido al pre-cálculo de los cubos. MOLAP requiere de unidades de almacenamiento grandes. En determinado momento esto puede llegar a generar gastos económicos (Darío, 2010).

1.5.3 Sistema de Procesamiento Analítico en Línea Híbrido.

Procesamiento Analítico en Línea Híbrido (HOLAP por sus siglas del inglés *Hybrid OnLine Analytical Process*), utiliza un motor ROLAP para almacenar unos datos y una BD multidimensional para guardar otros. Con esta combinación de implementaciones se dividen las tareas, ROLAP para navegar y explorar datos y MOLAP para la creación de tableros (Darío, 2010). Este tipo de sistema intenta tomar las mejores características de MOLAP y ROLAP, sin embargo sacrifica el tiempo de respuesta de los datos detallados para conseguir una menor latencia y mejorar los tiempos de carga.

1.5.4 Modo de almacenamiento seleccionado.

Según los elementos abordados ROLAP no posibilita la ejecución de consultas óptima debido al cálculo de los cubos dinámicamente. En el caso de HOLAP, este se enfoca a los



tiempos de carga y no al tiempo de respuesta de consultas, de manera que el rendimiento de las consultas es bajo. Sin embargo MOLAP permite el pre-cálculo de los cubos de información, lo que posibilita que el tiempo de respuesta de las consultas realizadas por los usuarios sea corto. A partir de esta ventaja de MOLAP se decide tomar este como modo de almacenamiento.

1.6 Metodología de desarrollo.

En el proceso de desarrollo de un sistema es preciso disponer de una guía que facilite dicho proceso. Para la construcción de almacenes de datos hay definidas varias metodologías, cada una con un enfoque diferente. Las principales son las de Bill H. Inmon, con un enfoque descendente (del inglés *top-down*) y Ralph Kimball con un enfoque ascendente (del inglés *bottom-up*).

El enfoque descendente por Bill Inmon se refiere a que los mercados de datos se nutrirán del AD. Propone la creación de un repositorio de datos corporativo como fuente de información consolidada, consistente e histórica. Al ser construido descendentemente los mercados de datos se nutren del AD corporativo, convirtiéndose en un complejo empresarial de bases de datos relacionales (Hernández, 2012).

El enfoque ascendente de Ralph Kimball resulta diferente. Este plantea que el AD se compone por el conglomerado de todos los mercados de datos generados en una empresa y la información siempre se almacena en un modelo dimensional. La idea es construir almacenes de datos independientes para evaluar las ventajas del nuevo sistema a medida que se avanza en su construcción. En él, las partes individuales se diseñan con detalle y luego se enlazan para formar componentes más grandes, que a su vez se enlazan hasta que se forma el sistema completo (Hernández, 2012).

El Centro de Desarrollo de Tecnologías de Datos (DATEC) de la UCI, propone la metodología de desarrollo de almacenes de datos: “Metodología de desarrollo de almacenes de datos en DATEC”. Entre otros objetivos, esta metodología se enfoca en 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 AD. Se basa en la metodología de Kimball, reconocida internacionalmente, considerando el empleo del MD (Hernández, 2012).

La metodología propuesta por DATEC satisface las necesidades de gestión que presentan los proyectos de AD, pues ofrece mejor descripción y fundamentación de las



actividades y artefactos vinculados en el flujo del proceso de gestión de los mismos. Propone utilizar los casos de uso para representar los requisitos del AD con el objetivo de estandarizar la definición de los mismos y, su entendimiento por parte del cliente y del equipo de desarrollo. Entre las herramientas para la realización de las actividades asociadas al proceso de desarrollo del AD esta Visual Paradigm for UML (Hernández, 2012).

A continuación se muestra el ciclo de vida para esta metodología.

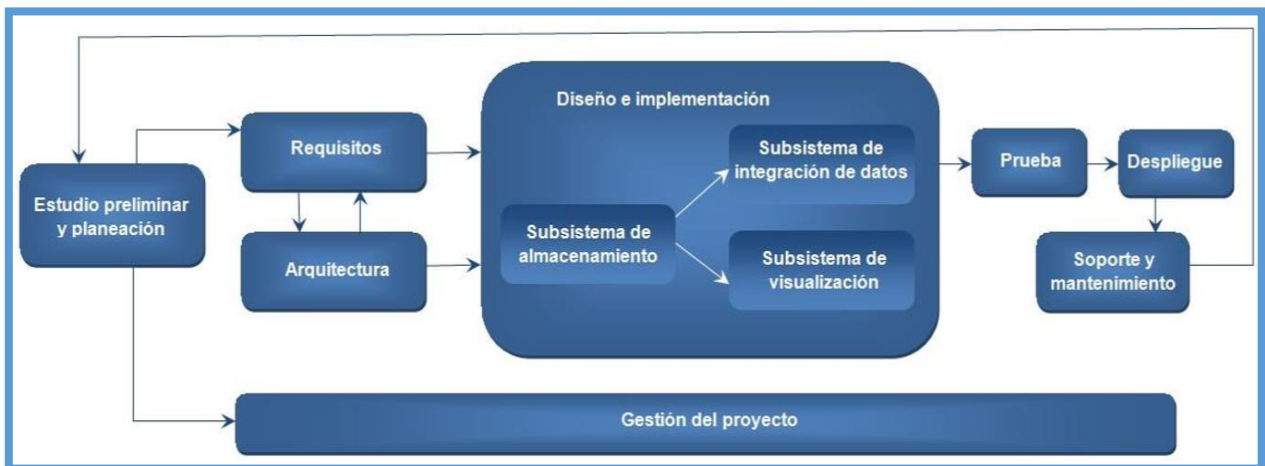


Figura 2. Ciclo de vida de “Metodología de desarrollo de almacenes de datos en DATEC” (Hernández, 2012).

Estudio preliminar y planeación: en un primer momento de esta fase se realiza el estudio de la entidad cliente, para ello se considera un diagnóstico integral que provee la información general de la organización desde el negocio de la misma hasta el estado de los datos que manejan y la infraestructura tecnológica. Estos resultados son de vital importancia para las fases de Requisitos y Arquitectura, ya que establecen los aspectos iniciales que se deben tener en cuenta. En un segundo y último momento de esta fase se realizan las tareas de planeación del proyecto (Hernández, 2012).

Requisitos: el cliente es entrevistado con el objetivo de obtener los requisitos de información, y a su vez se realiza un levantamiento de cada una de las fuentes de datos disponibles que permitan el acceso a la información. Se precisan los requisitos funcionales y no funcionales con los cuales debe cumplir la solución (Hernández, 2012).

Arquitectura: se definen las vistas de arquitectura de sistema, las vistas de arquitectura de tecnología y las vistas de arquitectura de infraestructura (Hernández, 2012).



Diseño e implementación: se diseñan e implementan los tres subsistemas que conforman el AD, teniendo en cuenta que los subsistemas de integración de datos y visualización de información pueden desarrollarse paralelamente si el subsistema de almacenamiento ha sido implementado anteriormente (Hernández, 2012).

Prueba: una vez que finaliza la implementación de la solución se le realizan las pruebas de unidad, las pruebas de integración y de sistema. Realizar tales pruebas permite determinar la calidad del producto obtenido. Cabe destacar que durante todo el proceso de desarrollo pueden efectuarse aseguramientos de calidad (Hernández, 2012).

Despliegue: es llevado a cabo un despliegue piloto, donde se configuran los servidores necesarios y se instalan las herramientas según la arquitectura definida, se cargan una muestra de los datos en un ambiente controlado, con el fin de mostrarle al cliente final el sistema en funcionamiento. Una vez aceptada la solución por el cliente, se realiza la carga histórica de los datos, puede ser en el mismo entorno que el despliegue piloto u otro, todo depende de las condiciones que establezca el cliente. Durante esta etapa se realiza la capacitación y transferencia tecnológica de la solución a los clientes y se da un acompañamiento técnico al cliente para adiestrarlo ante cualquier problema que pueda ocurrir. El resultado fundamental es la solución desplegada en el entorno real y en correcto funcionamiento (Hernández, 2012).

Soporte y mantenimiento: comienza cuando la solución está implantada y en explotación, y se ejecuta según el contrato firmado y las condiciones de soporte establecidas. Puede realizarse a través de variados servicios, que pueden ser soporte en línea, vía telefónica, web, correo u otros y el acompañamiento al cliente. Además se realizan las tareas de manteniendo de la aplicación tan necesarias para este tipo de desarrollo y que garantiza el adecuado funcionamiento y crecimiento del AD (Hernández, 2012).

Gestión del proyecto: esta fase se ejecuta a lo largo de todo el ciclo de vida del proyecto. Es aquí donde se controla, gestiona y chequea el desarrollo, los gastos, las utilidades, los recursos, las adquisiciones, los planes y cronogramas entre otras actividades relacionadas con la GP. Esta fase es el pilar base del proyecto y si no se ejecuta de forma continua y correcta el proyecto puede fracasar (Hernández, 2012).

1.6.1 Justificación de la metodología seleccionada.



Se decidió tomar como guía de desarrollo del AD de Xedro-GESPRO la metodología propuesta por DATEC, donde solo serán abordadas las fases: Estudio preliminar y planeación, Requisitos, Arquitectura, Diseño e implementación (Subsistema de almacenamiento y Subsistema de Integración), Prueba y Despliegue. La selección de la metodología se debe a que tiene bien definidas las fases de desarrollo y las tareas que presenta en cada fase, así como también los roles a desempeñar y herramientas a emplear. Es además, especializada en la creación de almacenes de datos permitiendo el correcto desarrollo de cada subsistema que integran los mismos e incluye los casos de uso como técnica para representar los requisitos del AD. Propone el empleo de Visual Paradigm for UML como herramienta para la realización de la documentación, planificación, estimación y administración del proyecto como actividades del proceso de desarrollo del AD.

1.7 Herramientas para el desarrollo del almacén de datos de Xedro-GESPRO.

1.7.1 Herramienta para el modelado.

Actualmente existen herramientas que le facilitan el trabajo a un analista de software. Las llamadas herramientas de Ingeniería de Software Asistida por Computación (CASE por sus siglas del inglés *Computer Aided Software Engineering*) son aplicables a diversos software para automatizar el proceso de diseño del mismo.

Para el modelado del AD se empleará Visual Paradigm for UML 8.0, pues se cuenta con la licencia de dicha herramienta. A su vez es propuesta por la metodología de desarrollo que se empleará para realizar las actividades del proceso desarrollo de la propuesta de solución.

Algunas de las características de esta herramienta son:

- ❖ Presenta variedad de lenguaje posibilitándose con ello la comunicación del equipo de desarrollo (Visual Paradigm Website., 2015).
- ❖ Multiplataforma (Visual Paradigm Website., 2015).
- ❖ Las ediciones son compatibles unas con otras (Visual Paradigm Website., 2015).
- ❖ Exportación de imágenes en formato jpg, png y svg (Visual Paradigm Website., 2015).

1.7.2 Sistema Gestor de Base de Datos.



El software que permite la utilización y/o la actualización de los datos almacenados en una o varias bases de datos por uno o varios usuarios desde diferentes puntos de vista y a la vez, se denomina Sistema de Gestión de Bases de Datos (SGBD) (Mato García, 2005).

1.7.2.1 PostgreSQL Server 9.2.

PostgreSQL Server es un servidor de BD relacional libre que presenta una arquitectura cliente - servidor, donde Postgres es el nombre dado al servidor. (PostgreSQL Website, 2012). A continuación se mencionan las características de PostgreSQL Server 9.2.

- ❖ Simula la herencia entre tablas mediante el particionado de estas (PostgreSQL Website, 2012).
- ❖ Soporta distintos tipos de datos como: los tipos primitivos, de tipo fecha, monetarios, elementos gráficos. También permite la creación de tipos propios (PostgreSQL Website, 2012).
- ❖ Presenta mejoras en cuanto a escalabilidad y rendimiento que le permiten manejar mayor cargas de trabajo de volumen (PostgreSQL Website, 2012).

PostgreSQL Server se utilizó en el desarrollo de la BD de Xedro-GESPRO, por tanto con el objetivo de mantener la simetría entre ambos sistemas se decide emplear en la solución propuesta este SGBD. Específicamente se utilizará PostgreSQL Server en su versión 9.2, pues constituye un requisito del cliente. Debido a que el equipo de desarrollo posee más de 5 años de experiencia con dicha herramienta.

1.7.3 Administrador de base de datos.

1.7.3.1 PgAdmin III.

Es una aplicación de diseño y manejo de bases de datos para su uso con PostgreSQL Server. La aplicación se puede utilizar para manejar PostgreSQL 7.3 y superiores. Este software fue diseñado para responder a las necesidades de todos los usuarios, desde la escritura de simples consultas SQL hasta la elaboración de bases de datos complejas. La interfaz gráfica es compatible con todas las características de PostgreSQL Server 9.2 y facilita la administración. La aplicación también incluye, entre otros elementos, un editor de la sintaxis SQL y un editor de código del lado del servidor. Permite realizar la conexión del servidor mediante el protocolo TCP/IP. No se requieren controladores adicionales para comunicarse con la BD del servidor (ArPUG, 2015).



Las características de pgAdmin III son las siguientes:

- ❖ Posibilita ver y trabajar con casi todos los objetos de la BD, examinar sus propiedades y realizar tareas administrativas.
- ❖ Una característica interesante de pgAdmin III es que, cada vez que se realiza alguna modificación en un objeto, escribe la sentencia o las sentencias SQL correspondientes, lo que hace que, además de una herramienta muy útil, sea a la vez didáctica.
- ❖ Incorpora funcionalidades para realizar consultas, examinar su ejecución (como el comando *Explain Analyze*) y trabajar con los datos.

1.7.4 Lenguaje que emplea el sistema gestor de base de datos.

El Lenguaje Procedimental / PostgreSQL Lenguaje Estructurado de Consultas (PL/pgSQL por sus siglas del inglés *Procedural Language / PostgreSQL Structured Query Language*) permite crear funciones y procedimientos disparados por eventos, añadir estructuras de control al lenguaje SQL y realizar cálculos complejos. Además hereda todos los tipos definidos por el usuario, las funciones y los operadores.

PL/pgSQL es un lenguaje orientado a bloques, lo que significa que puede existir cualquier número de sub-bloques en la sección de sentencia de un bloque. Los sub-bloques pueden usarse para ocultar variables a otros bloques de sentencias. Las variables declaradas en la sección de declaraciones se inicializan a su valor por defecto cada vez que se inicia el bloque, no cada vez que se realiza la llamada a la función. PL/pgSQL no es sensible a las letras mayúsculas (The PostgreSQL Global Development Group, 2015).

1.8 Conclusiones del capítulo.

- ❖ Para el almacenamiento histórico y resumido de los datos contenidos en la BD de Xedro-GESPRO se determinó, basado en las características referenciadas en el presente capítulo, el empleo de los AD.
- ❖ La utilización del Almacén de Datos de Alta Replicación, el AD Vertica OnDemand y el AD RocksDB constituyó un punto de partida para el desarrollo de la investigación, pero no pueden ser empleados como solución porque no están orientados al almacenamiento de la información asociada a la GP. En el caso del AD RocksDB no soporta consultas SQL.
- ❖ De acuerdo con el estudio de los modelos conceptuales y análisis de los modos de almacenamiento, se identificó que MOLAP es el modo de almacenamiento que se Almacén de Datos para la Suite de Gestión de Proyectos Xedro-GESPRO.



adapta para el de desarrollo de la solución. Para lo cual se determina el diseño de un modelo dimensional mediante el esquema estrella.

- ❖ Para el desarrollo del almacén de datos de Xedro-GESPRO se selecciona como guía de apoyo la Metodología de desarrollo de almacenes de datos en DATEC. Como SGBD se determina emplear PostgreSQL Server 9.2. Para la administración de los datos se decide utilizar pgAdmin III. Para diseñar el modelo físico y lógico, además de los casos de uso se determina usar *Visual Paradigm for UML 8.0*.



CAPÍTULO 2: ANÁLISIS Y DISEÑO DEL ALMACÉN DE DATOS DE XEDRO-GESPRO.

En este capítulo se realiza el análisis de la suite Xedro-GESPRO y el sistema de almacenamiento para la información gestionada en dicha suite. Se identifican los temas de información según las necesidades del cliente. Se define la arquitectura de la solución y se diseñan los subsistemas de integración y almacenamiento del AD partiendo del levantamiento de requisitos y el agrupamiento de estos en casos de uso.

2.1 Definición del negocio.

Xedro-GESPRO como ha sido referenciada en el capítulo 1, es una suite para la GP que puede ser empleada por cualquier organización, empresa o institución que trabaje con el concepto de proyecto. El sistema se rige por la Guía de los Fundamentos para la Dirección de Proyectos del Instituto de Gestión de Proyectos (PMI por sus siglas del inglés *Project Management Institute*) y las buenas prácticas del programa de formación del Máster en Gestión de Proyectos de la UCI (Piñero Pérez, 2014). Tiene una estructura por módulos, los cuales se argumentan a continuación.

- ❖ Módulo de planificación de proyectos: posibilita la dirección integrada de portafolios de proyectos. Permite la gestión de alcance y tiempo posibilitando la identificación de requisitos y control de la calidad así como la construcción semiautomática de cronogramas. Posibilita además la identificación de la línea base de los proyectos y la asignación de recursos materiales y humanos (Piñero Pérez, 2014).
- ❖ Módulo de control y seguimiento: se destaca por un cuadro de mando integral que permite analizar tanto un portafolio de proyectos como proyectos independientes. Con facilidades para la identificación del camino crítico, la cadena crítica e indicadores como el Índice de Rendimiento de la Planificación (IRP) e Índice de Rendimiento de Costos (IRC) que garantizan un vistazo completo del estado del proyecto respecto a: la calidad, costos, tiempo, recursos humanos y los contratos (Piñero Pérez, 2014).
- ❖ Módulo de gestión de riesgos: permite la identificación, análisis y respuesta de los riesgos, compartir riesgos entre diversos proyectos y la generación semiautomática de planes de mitigación y contingencia (Piñero Pérez, 2014).
- ❖ Módulo de gestión costos y facturaciones: permite la gestión de los presupuestos y los costos de los proyectos con facilidades para la integración con Software de



Gestión Integrada. Incluye facilidades para el seguimiento de las facturaciones y la gestión del flujo de caja (Piñero Pérez, 2014).

- ❖ Módulo de gestión de la logística y contratos con proveedores: gestiona recursos materiales asociados a la organización y la asignación de los mismos a los proyectos. Incluye funcionalidades para la gestión de los contratos con los proveedores y la evaluación (Piñero Pérez, 2014).
- ❖ Módulo gestión de contrataciones e interesados: permite el seguimiento de los contratos y sus modificaciones como parte del expediente de proyecto (Piñero Pérez, 2014).
- ❖ Módulo de gestión documental: incluye tres variantes para la gestión de los documentos del proyecto en forma de expediente. Las variantes se diferencian por el equilibrio entre la cantidad de funcionalidades y los requerimientos tecnológicos para su implantación (Piñero Pérez, 2014).
 - Variante 1: más sencilla está basada en facilidades del *Redmine* permite asociar documentos al proyecto solo depende para su implantación de espacio en disco (Piñero Pérez, 2014).
 - Variante 2: recomendada, combina las potencialidades de la plataforma GESPRO y de herramientas para el control de versiones, garantiza un alto nivel de integración, consumo moderado de recursos computacionales, facilidades para el control de versiones y el manejo de expedientes de proyectos (Piñero Pérez, 2014).
 - Variante 3: es la más costosa pero posibilita la integración con el gestor documental Alfresco y mayores funcionalidades para la gestión documental (Piñero Pérez, 2014).
- ❖ Módulo de integración con Sistemas de Información Geográfica (SIG): permite la ayuda a la toma de decisiones en la identificación de rutas de distribución, fortalezas y debilidades de la organización de proyectos y sus recursos partir del análisis del comportamiento por regiones geográficas (Piñero Pérez, 2014).
- ❖ Módulo de integración con otras plataformas: este módulo garantiza la integración con la plataforma desde tres perspectivas posibles. La primera basada en la interoperabilidad a nivel de ficheros valores separados por comas (CSV por sus siglas del inglés *Comma-Separated Values*), la segunda la integración a nivel de bases de datos y el tercer mecanismo basado en servicios a partir del protocolo de



Transferencia de Estado Representacional (REST por sus siglas del inglés *Representational State Transfer*) (Piñero Pérez, 2014).

- ❖ Módulo de configuración y seguridad: incluye facilidades para la configuración y adaptación del sistema para diferentes escenarios (Piñero Pérez, 2014).

Xedro-GESPRO se presenta como una plataforma web con facilidades para la personalización visual (Piñero Pérez, 2014), que emplea una BD para el almacenamiento de los datos gestionados en el mismo. Dicha BD presenta 723 funciones, 26 vistas, 246 tipos y 198 secuencias. Contiene además 237 tablas en la que se almacenan los datos arrojados por las operaciones realizadas en los módulos expuestos anteriormente como son: la información de los usuarios vinculados a un proyecto, las tareas que les son asignadas a estos, los recursos asignados a los locales donde trabaja el personal de un proyecto, además de riesgos y desviaciones, cambios, requerimientos e indicadores de los proyectos terminados o que se encuentran en ejecución, los indicadores de los recursos humanos de un proyecto, los indicadores de los centros de desarrollo de la UCI, los clientes y las organizaciones que emplean Xedro-GESPRO, los errores producidos en Xedro-GESPRO, las competencias por roles y por usuario, entre otros aspectos relacionados con la GP.

2.2 Temas de análisis.

Al desarrollar un AD es imprescindible identificar los temas de análisis. Estos permiten obtener una estructura de diseño adecuada y factible, por lo cual se realizaron entrevistas a los directivos del negocio para conocer las carencias de información que estos presentan. A partir de las necesidades de información se determinaron los temas a analizar para la creación del AD. Los tópicos de información o análisis identificados se exponen a continuación.

- ❖ Históricos de los proyectos.
- ❖ Históricos de los riesgos de los proyectos.
- ❖ Históricos de los cambios de los proyectos.
- ❖ Históricos de las desviaciones de los proyectos.
- ❖ Históricos de los requerimientos de los proyectos.
- ❖ Históricos de las tareas de los proyectos.
- ❖ Históricos de los recursos de los proyectos.
- ❖ Históricos de los indicadores de los proyectos.



- ❖ Históricos de los indicadores de los recursos humanos (RR.HH) de los proyecto.

2.3 Reglas del negocio.

En la BD de Xedro-GESPRO se almacenan los valores según se gestionan en la herramienta. En el caso de la fecha de cierre de los proyectos, cuando un proyecto no ha cerrado, el valor de ese campo toma valor nulo. Se establece que al realizarse las transformaciones para el almacenamiento de la información se guarde como nulo dicho campo. De igual forma, para los demás campos se decide no tenerse en cuenta los valores nulos.

2.4 Requisitos funcionales.

Los requisitos funcionales (RF) describen las funcionalidades que el equipo de desarrollo debe construir. Estos requisitos incluyen las funcionalidades que deben implementarse en los tres subsistemas que se desarrollan en soluciones de AD (Hernández, 2012).

A partir del análisis del negocio se lograron determinar los requisitos funcionales con los que debe cumplir la propuesta de solución.

RF 1: Extraer los datos de la tabla que contiene la información referente a los riesgos de los proyectos a partir del año 2010 hasta la fecha.

RF 2: Transformar los datos extraídos de la tabla que contiene la información referente a los riesgos de los proyectos a partir del año 2010 hasta la fecha.

RF 3: Cargar los datos extraídos y transformados referente a los riesgos de los proyectos a partir del año 2010 hasta la fecha.

RF 4: Extraer los datos de la tabla que contiene la información referente a los cambios de los proyectos a partir del año 2010 hasta la fecha.

RF 5: Transformar los datos extraídos de la tabla que contiene la información referente a los cambios de los proyectos a partir del año 2010 hasta la fecha.

RF 6: Cargar los datos extraídos y transformados referente a los cambios de los proyectos a partir del año 2010 hasta la fecha.

RF 7: Extraer los datos de la tabla que contiene la información referente a las desviaciones de los proyectos a partir del año 2010 hasta la fecha.

RF 8: Transformar los datos extraídos de la tabla que contiene la información referente a las desviaciones de los proyectos a partir del año 2010 hasta la fecha.

RF 9: Cargar los datos extraídos y transformados referente a las desviaciones de los proyectos a partir del año 2010 hasta la fecha.



RF 10: Extraer los datos de la tabla que contiene la información referente a los requerimientos de los proyectos a partir del año 2010 hasta la fecha.

RF 11: Transformar los datos extraídos de la tabla que contiene la información referente a los requerimientos de los proyectos a partir del año 2010 hasta la fecha.

RF 12: Cargar los datos extraídos y transformados referente a los requerimientos de los proyectos a partir del año 2010 hasta la fecha.

RF 13: Extraer los datos de la tabla que contiene la información referente a un proyecto a partir del año 2010 hasta la fecha.

RF 14: Transformar los datos extraídos de la tabla que contiene la información referente a los proyectos a partir del año 2010 hasta la fecha.

RF 15: Cargar los datos extraídos y transformados referente a los proyectos a partir del año 2010 hasta la fecha.

RF 16: Extraer los datos de la tabla que contiene la información referente a las tareas de los proyectos a partir del año 2010 hasta la fecha.

RF 17: Transformar los datos extraídos de la tabla que contiene la información referente a las tareas de los proyectos a partir del año 2010 hasta la fecha.

RF 18: Cargar los datos extraídos y transformados referente a los las tareas de los proyectos a partir del año 2010 hasta la fecha.

RF 19: Extraer los datos de la tabla que contiene la información referente a los recursos de los proyectos a partir del año 2010 hasta la fecha.

RF 20: Transformar los datos extraídos de la tabla que contiene la información referente a los recursos de los proyectos a partir del año 2010 hasta la fecha.

RF 21: Cargar los datos extraídos y transformados referente a los recursos de los proyectos a partir del año 2010 hasta la fecha.

RF 22: Extraer los datos de la tabla que contiene la información referente a los indicadores de los proyectos a partir del año 2010 hasta la fecha.

RF 23: Transformar los datos extraídos de la tabla que contiene la información referente a los indicadores de los proyectos a partir del año 2010 hasta la fecha.

RF 24: Cargar los datos extraídos y transformados referente a los indicadores de los proyectos a partir del año 2010 hasta la fecha.

RF 25: Extraer los datos de la tabla que contiene la información referente a los indicadores de los recursos humanos de los proyectos a partir del año 2010 hasta la fecha.



RF 26: Transformar los datos de la tabla que contiene la información referente a los indicadores de los recursos humanos de los proyectos a partir del año 2010 hasta la fecha.

RF 27: Cargar los datos de la tabla que contiene la información referente a los indicadores de los recursos humanos de los proyectos a partir del año 2010 hasta la fecha.

2.5 Requisitos no funcionales.

Los Requisitos No Funcionales (RNF) describen las propiedades y cualidades que debe tener la solución. Representan las características del producto (Hernández, 2012). Los RNF definidos para la solución son:

Funcionalidad:

❖ Interoperabilidad:

- El sistema debe ser capaz de interactuar con la base de datos del sistema Xedro-GESPRO.

Eficiencia:

❖ Comprobación del tiempo:

- El sistema debe ser capaz de dar respuestas a las consultas de los usuarios en un tiempo menor a los tiempos de respuesta del ODS Proyectos Terminados y bajo diferentes condiciones de carga.

Portabilidad:

❖ Adaptabilidad:

- Debe ser capaz de adaptarse a los ambientes donde se emplee la suite de GP Xedro-GESPRO.

Soporte:

- Para el desarrollo de la solución se debe emplear PostgreSQL Server 9.2.

Interfaz de hardware: están definidos atendiendo a las posibilidades de tecnológicas que puede proporcionar la UCI para el despliegue en la red de centros de desarrollo de dicha institución, definiéndose como requisitos:

- Servidor de Base de Datos: 10 GB de HDD, 4GB de RAM y 2 CPU virtuales.

2.6 Diagrama de caso de uso.

Un caso de uso es una secuencia de pasos establecidos por las relaciones creadas entre el actor de un sistema y el sistema, reflejándose el comportamiento de este último.



Se determinaron para la propuesta de solución un total de 9 casos de uso, en los cuales se agrupan los 27 RF definidos anteriormente. A continuación se mencionan los caso de uso funcional (CUF) definidos.

CUF 1: Administrar los históricos de los riesgos de los proyectos.

CUF 2: Administrar los históricos de los cambios de los proyectos.

CUF 3: Administrar los históricos de las desviaciones de los proyectos.

CUF 4: Administrar los históricos de los requerimientos de los proyectos.

CUF 5: Administrar los históricos de los proyectos.

CUF 6: Administrar los históricos de las tareas de los proyectos.

CUF 7: Administrar los históricos de los recursos de los proyectos.

CUF 8: Administrar los históricos de los indicadores de los proyectos.

CUF 9: Administrar los históricos de los indicadores de los RR.HH de los proyectos.

En la figura 3 se muestra el diagrama de CUF de la propuesta de solución.

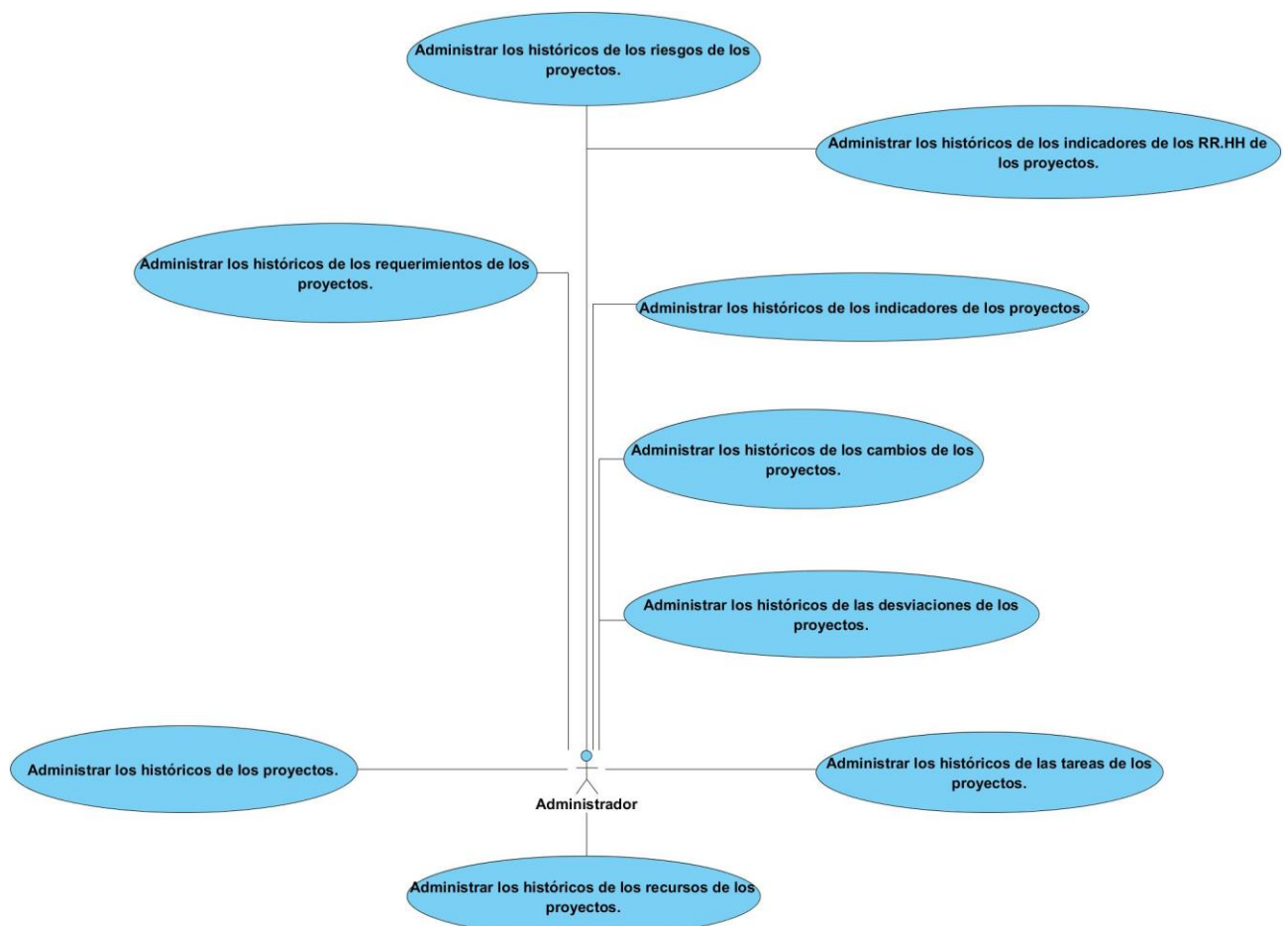


Figura 3. Diagrama de caso de uso.

Descripción del actor.

Almacén de Datos para la Suite de Gestión de Proyectos Xedro-GESPRO.



Administrador: encargado de llevar a cabo los procesos de extracción, transformación y carga de los datos del sistema fuente.

2.6.1 Descripción de caso de uso.

La siguiente tabla muestra la descripción del caso de uso: Administrar los datos históricos de los riesgos de los proyectos. Los restantes se encuentran en los Anexos del 9 al 16.

Objetivo	Permitir el almacenamiento de los datos históricos de los riesgos de los proyectos.	
Actores	Administrador.	
Resumen	Administrador: (inicia) Extrae, transforma y carga los datos históricos de los riesgos de los proyectos.	
Complejidad	Simple.	
Prioridad	Alta.	
Precondiciones	La BD de Xedro-GESPRO y la información contenida en ella deben estar disponibles.	
Postcondiciones	Los datos históricos de los riesgos de los proyectos deben ser almacenados en el AD.	
Flujo de eventos		
Flujo básico <Almacenar Datos>		
	Actor	Sistema
1	1 El administrador configura el fichero <i>bash</i> para la BD de Xedro-GESPRO.	
2	2 El administrador programa la ejecución del fichero <i>bash</i> mediante el comando <i>crontab</i> y este se encarga de la ejecución	2.1 Al ejecutarse el fichero <i>bash</i> el sistema realiza las siguientes acciones: <ul style="list-style-type: none"> - Extraer los datos de la tabla que contiene la información referente a los riesgos de los proyectos a partir del año 2010 hasta la



	<p>de dicho fichero automáticamente.</p>	<p>fecha.</p> <ul style="list-style-type: none"> - Transformar los datos extraídos de la tabla que contiene la información referente a los riesgos de los proyectos a partir del año 2010 hasta la fecha. - Cargar los datos extraídos y transformados referente a los riesgos de los proyectos a partir del año 2010 hasta la fecha.
<p>Flujos alternos</p>		
<p>Sección 1: “Extraer los datos de la tabla que contiene la información referente a los riesgos de los proyectos a partir del año 2010 hasta la fecha.”</p>		
<p>Flujo básico < Extraer los datos de la tabla que contiene la información referente a los riesgos de los proyectos a partir del año 2010 hasta la fecha></p>		
	<p>Actor</p>	<p>Sistema</p>
<p>1</p>		<p>2.1 Establece la conexión a la BD origen.</p>
<p>2</p>		<p>3.1 Extrae los datos especificados, en correspondencia con la fecha de actualización.</p>
<p>Flujos alternos</p>		
<p>Nº Evento <Datos incorrectos></p>		
	<p>Actor</p>	<p>Sistema</p>
<p>1.</p>		<p>2.1 Muestra los siguientes mensajes o uno de ellos en dependencia del dato introducido incorrectamente:</p> <ul style="list-style-type: none"> - “Usuario incorrecto” - “Contraseña incorrecta”



		<ul style="list-style-type: none"> - “Puerto incorrecto” - “Host incorrecto”
Flujos alternos		
Nº Evento <Fecha fuera de rango.>		
	Actor	Sistema
1		3.1 No extrae nueva información.
Sección 2: “Transformar los datos extraídos de la tabla que contiene la información referente a los riesgos de los proyectos a partir del año 2010 hasta la fecha.”		
Flujo básico <Transformar los datos extraídos de la tabla que contiene la información referente a los riesgos de los proyectos a partir del año 2010 hasta la fecha.>		
	Actor	Sistema
1		2.1 Transforma los datos extraídos de acuerdo a las características que deben presentar.
2		3.1 Almacena la información en una tabla temporal.
Sección 3: “Cargar los datos extraídos y transformados referente a los riesgos de los proyectos a partir del año 2010 hasta la fecha.”		
Flujo básico <Cargar los datos extraídos y transformados referente a los riesgos de los proyectos a partir del año 2010 hasta la fecha.>		
	Actor	Sistema
1		2.1 Crea las particiones para la tabla de los históricos referentes a los riesgos de los proyectos.
2		3.1 Almacena los datos en cada partición de



		acuerdo a la fecha.
Flujos alternos		
Nº Evento <Partición creada.>		
	Actor	Sistema
1		2.1 No crea nuevas particiones y almacena la información en la partición existente correspondiente al año de los datos a insertar.

Tabla 1. Descripción CU 1: Administrar los datos históricos de los riesgos de los proyectos.

2.7 Arquitectura del almacén de datos de Xedro-GESPRO.

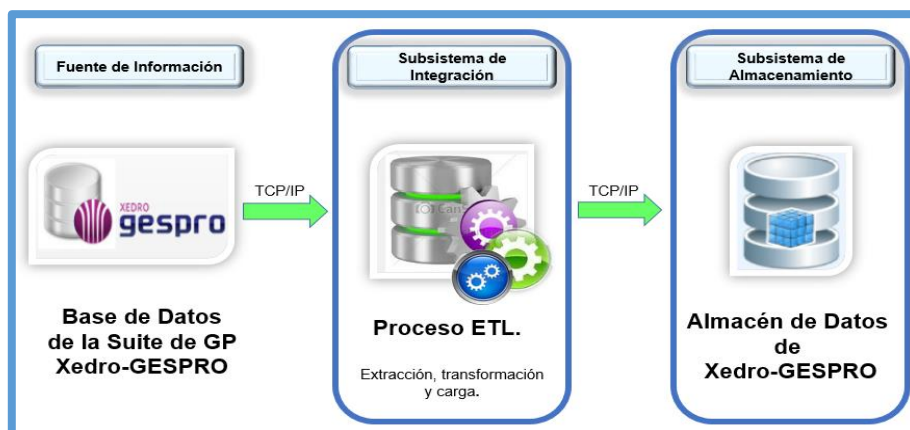


Figura 4. Arquitectura del almacén de datos de Xedro-GESPRO.

Fuente de Información: la fuente de datos de la cual se extrae la información de forma resumida para ser almacenada en el almacén de datos. En este caso se tiene como fuente a la BD de la suite de GP Xedro-GESPRO.

Subsistema de Integración: la información contenida en la BD de la suite de GP Xedro-GESPRO se le realizan procesos de limpieza, transformación e integración con el objetivo de que los datos estén listos para efectuar la carga en el AD de Xedro-GESPRO.

Subsistema de Almacenamiento: el AD es creado en este subsistema, en el cual se almacenan los datos obtenidos una vez efectuado el proceso ETL.

2.8 Diseño del subsistema de almacenamiento.

Para la creación y funcionamiento del almacén de datos se realiza el modelo físico de datos en una estructura multidimensional. Este está conformado por 9 tablas que



contienen información asociada a los proyectos. Se diseña un modelo físico que permite el análisis de los proyectos desde perspectivas tales como: cambios, riesgos, tareas, desviaciones, indicadores, requerimientos, recursos y recursos humanos.

2.8.1 Dimensiones del modelo físico de datos.

En la siguiente tabla se presentan las dimensiones del modelo dimensional con una breve descripción de la información que se almacena en cada una de ellas.

No.	Dimensión	Descripción
D1	<i>gespro_projects_risks</i>	En esta tabla se almacena la información de los riesgos de los proyectos.
D2	<i>gespro_projects_requirements</i>	En esta tabla se almacena la información de los requerimientos de los proyectos.
D3	<i>gespro_projects_deviations</i>	En esta tabla se almacena la información de las desviaciones de los proyectos.
D4	<i>gespro_projects_changes</i>	En esta tabla se almacena la información de los cambios de los proyectos.
D5	<i>gespro_projects_tareas</i>	En esta tabla se almacena la información de las tareas de los proyectos.
D6	<i>gespro_projects_indicators</i>	En esta tabla se almacena la información de los indicadores de los proyectos.
D7	<i>gespro_projects_resources</i>	En esta tabla se almacena la información de los recursos de los proyectos.
D8	<i>gespro_rh_project_indicator_values</i>	En esta tabla se almacena la información de los indicadores de los recursos humanos de los proyectos.

Tabla 2: Dimensiones del modelo dimensional.

2.8.2 Hecho del modelo físico de datos.

H1: *gespro_projects*: En esta tabla se almacena la información de los proyectos.

2.8.3 Matriz bus o matriz dimensional.

La relación existente entre las dimensiones y los hechos del AD de Xedro-GESPRO se representa mediante la matriz bus. Es un medio para verificar que no exista solapamiento entre los hechos. Las columnas de la matriz representan el hecho identificado para el AD



y las filas las dimensiones utilizadas. Las celdas marcadas con una X indican que la fila de la dimensión está relacionada con la columna del hecho.

hecho/dimensiones	D1	D2	D3	D4	D5	D6	D7	D8
<i>gespro_projects</i>	X	X	X	X	X	X	X	X

Tabla 3: Matriz dimensional.

2.8.4 Modelo físico de datos.

Una vez identificada la tabla hecho y las tablas dimensiones se diseña el modelo físico de datos para el AD de Xedro-GESPRO, tomando como base la topología estrella. Esta topología optimiza los tiempos de respuesta de las consultas, permite un diseño fácilmente modificable y establece las relaciones físicas y lógicas entre el hecho y dimensiones.

Para el diseño del AD se decide no tomar la característica de la topología de estrella de establecer relación física entre la tabla hecho y sus dimensiones. Por lo cual se realiza una combinación de datos cualitativos y cuantitativos en las diferentes tablas. Posibilitando una reducción considerable de los conceptos de tabla hecho y dimensión, y por consiguiente, evitar consultas combinadas realizadas por un usuario, que tienden a aumentar el tiempo de respuesta del sistema.

Igualmente se determina cambiar la forma en que se realizan las relaciones lógicas entre la tabla hecho y dimensiones. Según definiciones especificadas en el capítulo 1, el atributo clave de las tablas dimensiones se añaden a la tabla hecho. Sin embargo para el diseño del AD de Xedro-GESPRO, el atributo clave *name_project* definido en la tabla hecho, pasa a constituir una columna de las tablas dimensiones. En la figura 5 se muestra el diseño multidimensional del AD de Xedro-GESPRO.



CAPÍTULO 2: ANÁLISIS Y DISEÑO DEL AD DE XEDRO-GESPRO

schema_ods_gespro.gespro_projects_indicators	schema_ods_gespro.gespro_projects_changes	schema_ods_gespro.gespro_projects_requirements	schema_ods_gespro.gespro_projects_resources
project_name varchar(255)	id int4	project_name text	issues_subject text
created_on date	name text	name text	issues_start_date date
updated_on date	description text	description text	issues_end_date date
cant_issues int4	priority int4	priority int4	issues_closed_on date
cant_users int4	baseline_modified text	prototype text	issues_created_on date
cant_requirements int4	configuration_element text	functional bool	issues_updated_on date
cant_risks int4	project_change_status_id int4	status text	project_name text
cant_resources int4	justified text	created_on date	ir_planned_amount int4
cant_changes int4	environment text	updated_on date	ir_plan_date date
cant_deviations int4	created_on date	gjp int4	ir_amount_hour int4
dbname text	updated_on date	gc int4	ir_real_amount int4
dbhost text	impact int4	gid int4	ir_real_date date
IE float8	active text	gis int4	resources_name text
IRE float8	author_detection text	gif int4	resources_description text
IRP float8	approved_by text	solicitud_date date	resources_code text
ICD float8	charge text	parent_id int4	resources_available int4
IRC float8	date_detection date	type text	resources_existence int4
IRL float8	date_closed date	dbname text	resources_um text
IRRH float8	project_name text	dbhost text	resources_limit int4
IREF float8	author text		resources_rate float8
ICDP float8	dbname text		resources_status int4
ICDE float8	dbhost text		resource_category_id int4
ICDV float8			resources_type text
ICDPG float8			resources_area float8
PV float8			resource_location_id int4
EV float8			workstations int4
AC float8			resources_rate_cuc float8
BAC float8			resources_rate_usd float8
IRHT float8			dbname text
IRHE float8			dbhost text
IRHA float8			
IRHF float8			
Real-IRE float8			
Plan-IRE float8			
Real-Contabilidad float8			
Plan-Hitos float8			
Real-IRP float8			
Plan-IRP float8			
TPTP float8			
TPTR float8			
TRTR float8			
Fondo de Tiempo float8			
Tiempo Planificado float8			
Tiempo Trabajado float8			
Financiamiento float8			
Cant. Riesgos Exp. Alto float8			
Cant. Riesgos Exp. Medio float8			
Cant. Riesgos Exp. Bajo float8			
Impacto Total Hitos Ejec float8			
Costo Total Hitos Ejec float8			
Cant. Devs. Calidad float8			
Cant. Devs. Costo float8			
Cant. Devs. Tiempo float8			
Cant. Devs. Logística float8			
Cant. Devs. Desempeño float8			
Cant. Devs. Impacto Alto float8			
Cant. Devs. Impacto Medio float8			
Cant. Devs. Impacto Bajo float8			
Cant. Requisitos Propuestos float8			
Cant. Requisitos Aprobados float8			
Cant. Requisitos Pospuestos float8			
Cant. Requisitos Terminados float8			
Cant. NC Formato float8			
Cant. NC Error técnico float8			
Cant. NC Correspondencia con otro artefacto float8			
Cant. NC Ortografía float8			
Cant. NC Redacción float8			
Cant. NC Validación float8			
Cant. NC Opciones que no funcionan float8			
Cant. NC Error interfaz float8			
Cant. NC Error idioma float8			
Cant. NC Funcionalidad float8			
Cant. NC Excepciones float8			
Cant. NC Seguridad float8			
Cant. NC Diseño CP float8			
Cant. NC Recomendación float8			
Cant. NC Otros float8			
Cant. RRHH Eval B float8			
Cant. RRHH Eval R float8			
Cant. RRHH Eval M float8			
Cant. Estudiante 1ro float8			
Cant. Estudiante 2do float8			
Cant. Estudiante 3ro float8			
Cant. Estudiante 4to float8			
Cant. Estudiante 5to float8			
Cant. RGA-Reclen Graduado en Adiestramiento float8			
Cant. IRG-Instructor Recien Graduado float8			
Cant. Especialista float8			
Cant. Profesor de Centro float8			
Cant. Profesor de Facultad trabaja en Centro float8			
Cant. Profesor de otras áreas trabaja en Centro float8			
Cant. Personal externo a la universidad float8			
Cant. Técnico float8			
Cant. Asistente de control float8			
Cant. Otros trabajadores float8			

schema_ods_gespro.gespro_projects
id_project int4
name text
description text
is_public bool
parent_id int4
created_on date
updated_on date
identifier text
status int4
cf_clasificacion_naturaleza text
cf_clasificacion_tipo_cliente text
cf_clasificacion_programa text
cf_clasificacion_estado text
cf_no_registro text
cf_entidad_desarrolladora text
cf_fecha_fin date
cf_fecha_inicio date
cf_prioridad int4
cf_tiempo_estimado_hh text
cf_modalidad_proyecto text
cf_alcance_proyecto text
cf_objetivo_general text
cf_objetivos_especificos text
cf_impacto_proyecto text
cf_responsable_desarrollador text
cf_telefono_desarrollador text
cf_correo_desarrollador text
cf_responsable_cliente text
cf_cargo_desarrollador text
cf_cargo_cliente text
cf_telefono_cliente text
cf_correo_cliente text
cf_nombre_oficial text
cf_cliente text
cf_no_contrato text
cf_fuente_financiamiento text
cf_codigo_centro_costo text
cf_patrocinador_proyecto text
cf_pais_cliente text
cf_oace_cliente_ministerio text
cf_monto_total_contrato_usd varchar(2147483647)
cf_monto_ejecutor_cubano_usd varchar(2147483647)
cf_monto_ejecutor_cubano_cuc varchar(2147483647)
cf_presupuesto_gasto_usd varchar(2147483647)
cf_presupuesto_gasto_cup varchar(2147483647)
cf_presupuesto_gasto_cuc varchar(2147483647)
cf_monto_ejecutor_cubano_cup varchar(2147483647)
cf_monto_total_contrato_cup varchar(2147483647)
cf_monto_total_contrato_cuc varchar(2147483647)
dbname text
dbhost text

schema_ods_gespro.gespro_projects_requirements
project_name text
name text
description text
priority int4
prototype text
functional bool
status text
created_on date
updated_on date
gjp int4
gc int4
gid int4
gis int4
gif int4
solicitud_date date
parent_id int4
type text
dbname text
dbhost text

schema_ods_gespro.gespro_projects_resources
issues_subject text
issues_start_date date
issues_end_date date
issues_closed_on date
issues_created_on date
issues_updated_on date
project_name text
ir_planned_amount int4
ir_plan_date date
ir_amount_hour int4
ir_real_amount int4
ir_real_date date
resources_name text
resources_description text
resources_code text
resources_available int4
resources_existence int4
resources_um text
resources_limit int4
resources_rate float8
resources_status int4
resource_category_id int4
resources_type text
resources_area float8
resource_location_id int4
workstations int4
resources_rate_cuc float8
resources_rate_usd float8
dbname text
dbhost text

schema_ods_gespro.gespro_projects_issues
id_issues int4
project_name text
firstname text
lastname text
issue_parent_id int4
tracker_type_name text
subject text
start_date date
due_date date
description text
done_ratio int4
status text
priority text
fixed_version_id int4
created_on date
updated_on date
estimated_hours float8
cf_hito_desarrollo bool
cf_artefacto text
cf_evaluacion text
cf_tipo_tarea text
cf_complejidad text
cf_referencia text
cf_costo_estimado float8
cf_impacto_ejecucion int4
cf_tipo_reunion text
cf_fecha_tomado date
dbname text
dbhost text

schema_ods_gespro.gespro_projects_risks
name text
impact float8
probability float8
mitigation text
contingency text
project_name text
category text
created_on date
updated_on date
detection_datetime date
closed_datetime date
resolution_datetime date
stages_name text
status text
contingency_responsible text
consequence text
cause text
resolution_complexity float8
comments text
dbname text
dbhost text

schema_ods_gespro.gespro_projects_changes
id int4
name text
description text
priority int4
baseline_modified text
configuration_element text
project_change_status_id int4
justified text
environment text
created_on date
updated_on date
impact int4
active text
author_detection text
approved_by text
charge text
date_detection date
date_closed date
project_name text
author text
dbname text
dbhost text

schema_ods_gespro.gespro_projects_requirements
name text
description text
impact int4
detection_datetime date
created_on date
updated_on date
irht float8
irhe float8
irha float8
irhf float8
evaluation varchar(255)
dbname text
dbhost text

schema_ods_gespro.gespro_projects_deviations
name text
description text
impact int4
detection_datetime date
created_on date
updated_on date
author_name text
project_name text
project_risk_name text
deviations_types_name text
dbname text
dbhost text

schema_ods_gespro.gespro_projects_indicators
firstname_user varchar(2147483647)
lastname_user varchar(2147483647)
project_name varchar(2147483647)
created_on date
updated_on timestamp(29)
irht float8
irhe float8
irha float8
irhf float8
evaluation varchar(255)
dbname text
dbhost text

Figura 5. Modelo físico de datos del almacén de datos de Xedro-GESPRO.



2.8.5 Modelo lógico de datos.

Con el objetivo de presentar mejor organización en los datos del negocio y facilitar la ejecución de consultas sobre el almacén en un tiempo mínimo, se crea una estructura lógica a partir de lo planteado por Inmon en su libro *Building the Data Warehousing*: “La partición de datos se refiere a la ruptura de los datos en unidades físicas separadas que pueden ser manejados de manera independiente (Inmon, 2005).”

Además, se toma como fundamento la característica de PostgreSQL Server 9.2 sobre el particionado de tablas. Dicha particularidad es incluida con el objetivo de simular el concepto de herencia de la Programación Orientada a Objetos (OOP por sus siglas del inglés *Object Oriented Programming*). Con respecto a dicho concepto, PostgreSQL Server 9.2 considera las tablas hijas como particiones que contendrán los mismos campos que la tabla padre, denominada tabla maestra.

En la figura 6 se observa el modelo lógico de datos obtenido para el almacén de datos de Xedro-GESPRO. En este se evidencia el particionado horizontal por el atributo *created_on*. Esto significa la ruptura de las tablas por filas, donde el atributo *created_on* representa una fecha con el formato AAAA/MM/DD, es decir año, mes y día respectivamente. Cada partición será creada conforme se almacenen los datos en la tabla maestra, siempre y cuando dicho atributo corresponda al año para el cual no haya sido creada la partición. Cuando una tabla maestra es particionada, sus tablas particiones toman por nombre el nombre de la tabla maestra seguido por el año al cual pertenece la información a almacenar en la partición creada.



CAPÍTULO 2: ANÁLISIS Y DISEÑO DEL AD DE XEDRO-GESPRO

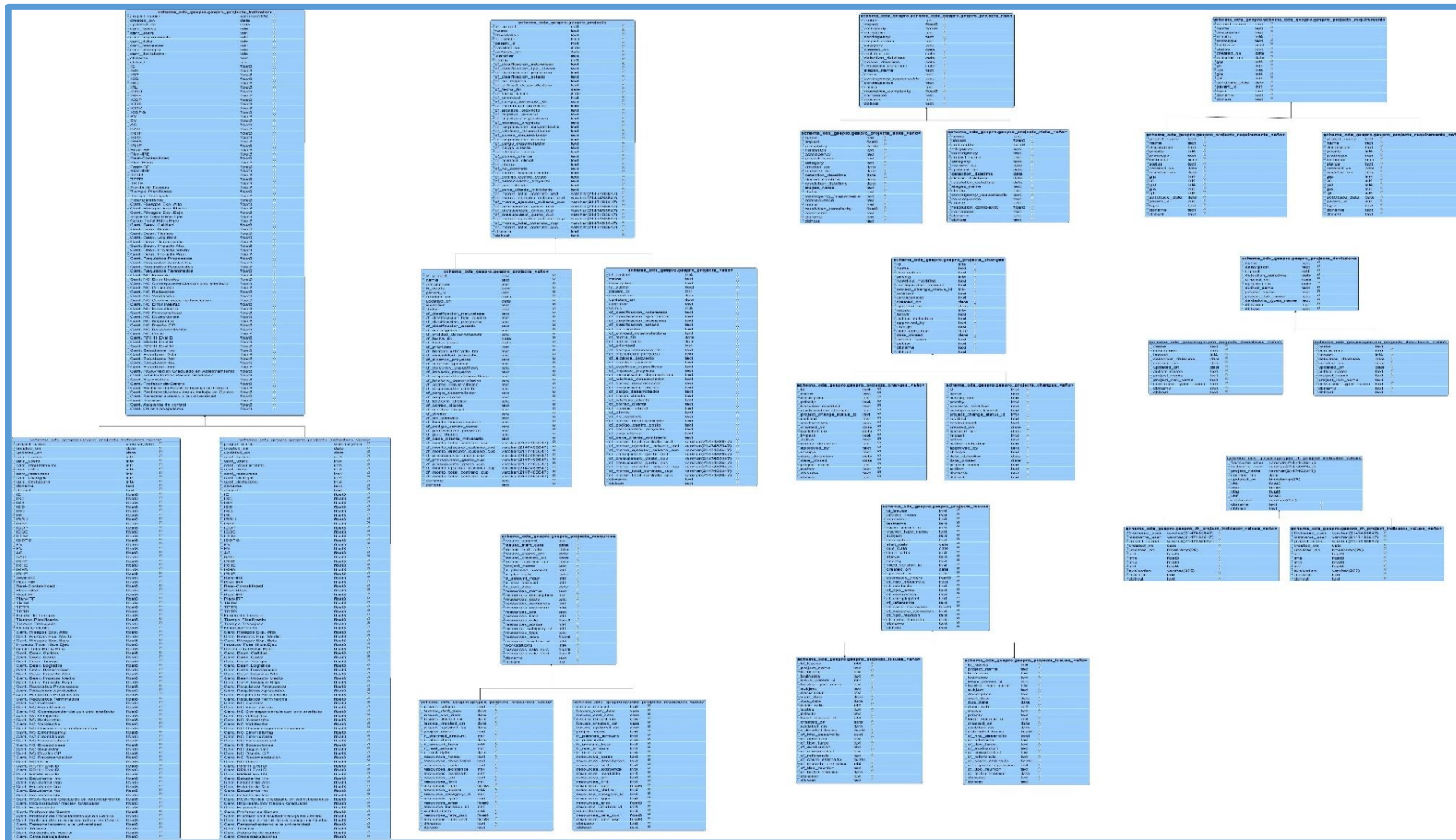


Figura 6. Modelo lógico de datos del almacén de datos de Xedro-GESPRO.



2.9 Diseño del subsistema de integración.

Luego de realizar el diseño del subsistema de almacenamiento se procede a diseñar el subsistema de integración. Con el propósito de lograr una adecuada integración de datos se definen funciones que se encargarán de ejecutar los procesos de extracción, transformación y carga de los datos.

Se definieron 9 funciones ETL que contienen una sub-consulta para ejecutar la función encargada de crear las tablas particiones, por lo que el subsistema de integración cuenta con un total de 18 funciones.

En el capítulo 3 se describe el algoritmo de dichas funcionalidades, las cuales se relacionan a continuación.

Funciones ETL	Función para la creación de las tablas particiones.
<i>_gespro_etl_projects(pdbname: text, pport: int4, phost: text, puser: text, ppassword: text)</i>	<i>_ods_etl_create_dinamyc_table_projects(year_date double precision)</i>
<i>_gespro_etl_projects_changes(pdbname: text, pport: int4, phost: text, puser: text, ppassword: text)</i>	<i>_ods_etl_create_dinamyc_table_changes(year_date double precision)</i>
<i>_gespro_etl_projects_deviations(pdbname: text, pport: int4, phost: text, puser: text, ppassword: text)</i>	<i>_ods_etl_create_dinamyc_table_deviations(year_date double precision)</i>
<i>_gespro_etl_projects_issues(pdbname: text, pport: int4, phost: text, puser: text, ppassword: text)</i>	<i>_ods_etl_create_dinamyc_table_issues(year_date double precision)</i>
<i>_gespro_etl_projects_requirements(pdbname: text, pport: int4, phost: text, puser: text, ppassword: text)</i>	<i>_ods_etl_create_dinamyc_table_requirements(year_date double precision)</i>
<i>_gespro_etl_projects_resources(pdbname: text, pport: int4, phost: text, puser: text, ppassword: text)</i>	<i>_ods_etl_create_dinamyc_table_resources(year_date double precision)</i>
<i>_gespro_etl_projects_risks(pdbname: text, pport: int4, phost: text, puser: text, ppassword: text)</i>	<i>_ods_etl_create_dinamyc_table_risks(year_date double precision)</i>



<i>text, pport: int4, phost: text, puser: text, ppassword: text)</i>	<i>date double precision)</i>
<i>_gespro_etl_projects_indicators(pdbname: text, pport: int4, phost: text, puser: text, ppassword: text)</i>	<i>_ods_etl_create_dinamyc_table_indicators(year_date double precision)</i>
<i>_gespro_etl_rh_project_indicator_values(pdbname text, pport integer, phost text, puser text, ppassword text)</i>	<i>_ods_etl_create_dinamyc_table_rh_project_indicator_values(year_date double precision)</i>

Tabla 4. Funciones del subsistema de integración.

2.10 Conclusiones del capítulo.

- ❖ A partir de la definición del negocio y el análisis de las necesidades de información se identificaron los RF y RNF.
- ❖ Se determinaron los subsistemas de integración y de almacenamiento que conforman el AD y la BD de Xedro-GESPRO como fuente de origen de la cual se obtendrán los datos para ser procesados, mediante el diseño de la arquitectura.
- ❖ Se obtuvo el diseño de los modelos lógico y físico, así como también la definición de las funcionalidades encargadas de procesar la información contenida en la fuente de origen. A partir de lo cual se realizará la implementación de los subsistemas definidos en la arquitectura.



CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBA DEL AD DE XEDRO-GESPRO.

En este capítulo se exponen los elementos de la implementación de la solución. Se explica cómo se realizó la implementación de los subsistemas de integración y almacenamiento del AD de Xedro-GESPRO, a partir de los requisitos y necesidades del negocio. Se hace referencia además, a las pruebas realizadas al producto desarrollado mediante la ejecución de consultas en el ODS y en la solución.

3.1 Implementación del subsistema de almacenamiento.

A partir del modelo físico de datos se realizó la implementación del AD de Xedro-GESPRO en la herramienta pgAdmin III. Los pasos a seguir son los siguientes:

Paso 1: Crear la BD vacía *almacen_de_datos_gespro*.

Paso 2: Añadir a la BD creada el esquema *schema_ods_gespro*.

Paso 3: Eliminar el esquema *public*.

Paso 4: En el esquema creado en el **Paso 2**, crear las tablas definidas en el modelo físico de datos (ver Figura 5).

La figura 7 muestra la vista del AD de Xedro-GESPRO en la herramienta pgAdmin III.

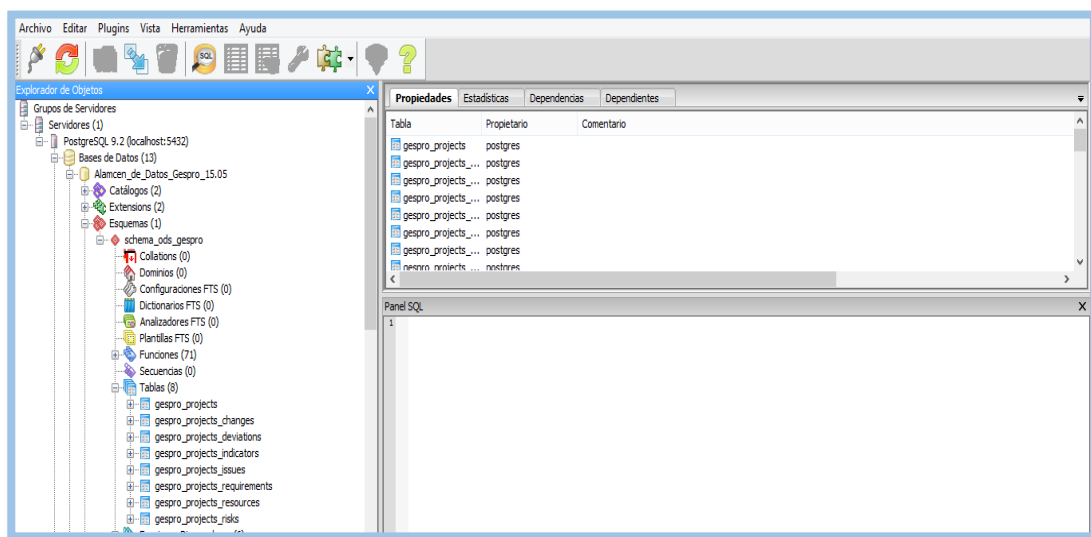


Figura 7. Vista del almacén de datos de Xedro-GESPRO en la herramienta pgAdmin III.

3.2 Implementación del subsistema de integración.

En el subsistema de integración es donde se realizan los procesos de extracción, transformación y carga de los datos. Según Ralph Kimball (Kimball, 2004) la extracción se refiere al proceso de extraer los datos de las distintas fuentes hacia el área de preparación del AD, para su posterior transformación. La transformación consiste en los



procesos de filtrado, limpieza, depurado, homogenización y agrupación de la información luego de ser extraída. Con respecto al proceso de carga plantea que este consiste en cargar los datos desde estas fuentes ya transformadas hacia el AD.

El proceso ETL definido para las funciones definidas en el capítulo 2, se expone a continuación mediante 4 pasos lógicos de manera general.

Paso 1: Extracción de los datos.

- ❖ Los datos son extraídos teniéndose en cuenta un período de tiempo específico, de manera que se eviten las ambigüedades de datos.

Paso 2: Transformar los datos extraídos.

- ❖ Las transformaciones realizadas son:
 - Conversión de datos.
 - Renombramiento de atributos.

Paso 3: Crear las particiones físicas.

- ❖ En caso de existir la partición, se ejecuta directamente el paso 4.

Paso 4: Cargar los datos en el almacén.

En la figura 8 se muestra el diagrama que representa el proceso ETL, para los pasos explicados anteriormente.

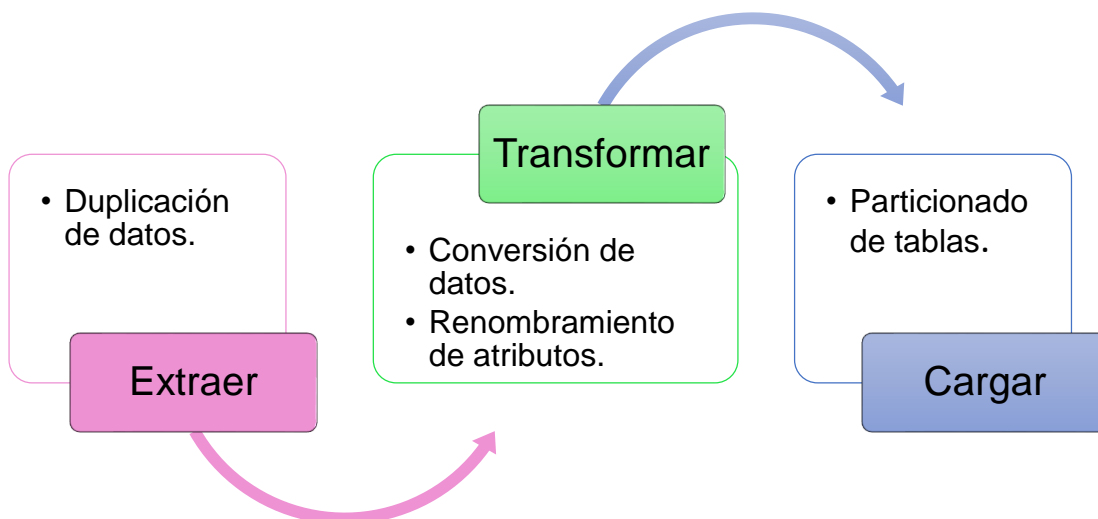


Figura 8. Proceso ETL.

A continuación se realiza la explicación con más detalles del algoritmo definido en las funciones encargadas de la ejecución del proceso ETL. Se ejemplifica mediante la función `_gespro_etl_projects_risks` (`pdname: text`, `pport: int4`, `phost: text`, `puser: text`, `ppassword: text`), encargada de realizar el proceso ETL para los riesgos de los proyectos.



3.2.1 Proceso de extracción de los datos.

Paso 1: Extraer el máximo *created_on* contenido en la tabla *gespro_projects_risks*.

Paso 2: Almacenar el máximo *created_on* obtenido en la variable *last_created_on*.

Paso 3: Crear la tabla temporal *project_risk_temp*.

Paso 4: Establecer la conexión a la BD de Xedro-GESPRO haciendo uso de la función *dblink* del PostgreSQL Server 9.2, a la cual se le pasa por parámetro el nombre de la BD de Xedro-GESPRO, el puerto para la conexión, el host donde se encuentra la BD de Xedro-GESPRO y usuario y contraseña de PostgreSQL Server 9.2.

- a. En el caso de no ser exitosa la conexión se aborta el proceso, de lo contrario se realiza el paso 5.

Paso 5: Extraer la información deseada contenida en la BD de Xedro-GESPRO.

- a. Validar que el valor del atributo *created_on* de la tupla a extraer sea mayor o distinto que el valor almacenado en la variable *last_created_on*, con lo cual se evita la ambigüedad de datos.

3.2.2 Proceso de transformación de los datos.

Las transformaciones de los datos tienen lugar dentro de la propia consulta de extracción de datos. A medida que se extraen los datos se realizan las conversiones de tipos de datos y se renombran atributos, de manera que estos se ajusten al esquema de destino. Los datos extraídos y transformados son almacenados momentáneamente en la tabla temporal *project_risks_temp*.

La siguiente tabla muestra las conversiones realizadas para los riesgos de los proyectos.

Conversiones de datos	
De	A
<i>timestamp without time zone</i>	<i>date</i>
<i>character varying</i>	<i>double precisión</i>
<i>character varying</i>	<i>integer</i>
<i>character varying</i>	<i>date</i>

Tabla 5. Conversión de datos para la tabla *gespro_projects_risks*.

En la tabla 6 se muestran los renombramientos de los atributos para la tabla *gespro_projects_risks*. Algunas columnas tomaban el mismo nombre que tenían en la fuente de origen tal es el caso de *project_stages.name*. Otras tomaban el nombre de la



propia función utilizada para obtener el valor; tal es el caso de *_rpt_get_project_name* para la cual se define el nombre: *project_name*.

Renombramiento de atributos	
De	Como
<i>_rpt_get_project_name</i>	<i>project_name</i>
<i>risk_categories.name</i>	<i>category</i>
<i>coalesce</i>	<i>detection_datetime</i>
<i>coalesce</i>	<i>closed_datetime</i>
<i>coalesce</i>	<i>resolution_datetime</i>
<i>project_stages.name</i>	<i>stages_name</i>
<i>_rpt_get_user_name</i>	<i>contingency_responsible</i>
<i>project_risk_statuses.name</i>	<i>status</i>

Tabla 6. Renombramiento de atributos para la tabla *gespro_projects_risks*.

El siguiente código corresponde al fragmento de la función ETL que realiza los procesos de extracción y transformación de los datos descritos anteriormente.

```
CREATE OR REPLACE FUNCTION
```

```
schema_ods_gespro._gespro_etl_projects_risks(pdbname text, pport integer, phost  
text, puser text, ppassword text)
```

```
RETURNS boolean AS
```

```
$BODY$
```

```
DECLARE
```

```
    rec RECORD;
```

```
    last_created_on DATE;
```

```
    rec_temp RECORD;
```

```
    sql_insert CHARACTER VARYING;
```

```
    name_table CHARACTER VARYING;
```

```
    year_return DOUBLE PRECISION;
```



```
BEGIN
SELECT MAX(created_on) FROM schema_ods_gespro.gespro_projects_risks
INTO last_created_on;
IF last_created_on IS NULL
THEN last_created_on := '2009-01-01';
END IF;
CREATE LOCAL TEMPORARY TABLE project_risk_temp(
    "name" TEXT,
    impact DOUBLE PRECISION,
    probability DOUBLE PRECISION,
    mitigation TEXT,
    contingency TEXT,
    project_name TEXT,
    category TEXT,
    created_on DATE,
    updated_on DATE,
    detection_datetime DATE,
    closed_datetime DATE,
    resolution_datetime DATE,
    stages_name TEXT,
    status TEXT,
    contingency_responsible TEXT,
    consequence TEXT,
    cause TEXT,
    resolution_complexity DOUBLE PRECISION,
    "comments" TEXT,
    dbname TEXT,
    dbhost TEXT
)ON COMMIT DROP;
INSERT INTO project_risk_temp
(SELECT *,pdbname,phost
FROM schema_ods_gespro.dblink('dbname=||pdbname||' '||port=||pport||'
```



```
||'host=||phost||' ||'user=||puser||' ||'password=||ppassword,
'SELECT *
FROM
(SELECT project_risks."name",
    impact,
    probability,
    mitigation,
    contingency,
    _rpt_get_project_name(project_risks.project_id) as project_name,
    risk_categories.name as category,
    cast (project_risks.created_on as date),
    cast (project_risks.updated_on as date),
    COALESCE(detection_datetime, current_date) as detection_datetime
    COALESCE(closed_datetime, current_date) as closed_datetime,
    COALESCE(resolution_datetime, current_date) as resolution_datetime,
    project_stages.name as stages_name,
    project_risk_statuses.name as status,
    rpt_get_user_name(contingency_responsible_id) as contingency_responsible,
    consequence,
    cause,
    resolution_complexity,
    "comments"
FROM project_risks LEFT JOIN risk_categories
ON (project_risks.risk_category_id = risk_categories.id) LEFT JOIN project_stages
ON(project_risks.id = project_stages.id) LEFT JOIN project_risk_statuses
ON(project_risks.project_risk_status_id = project_risk_statuses.id)
WHERE project_risks.created_on > ||quote_literal(last_created_on)||')
as subconsulta
WHERE subconsulta.created_on <> ||quote_literal(last_created_on))
```

3.2.3 Proceso de carga de los datos.



Paso 1: Obtener los diferentes años a los que pertenecen las tuplas almacenadas en *project_risks_temp*.

Paso 2: Almacenar los diferentes años obtenidos en la variable *year_date*.

Paso 3: Para cada año almacenado en *year_date* ejecutar los pasos del 3.1 al 3.4.

Paso 3.1: Ejecutar la función *_ods_etl_create_dinamyc_table_risks* (*year_date*: *double precisión*).

Paso 3.1.1: Verificar si existe al menos un registro del año en *year_date* en la tabla maestra *gespro_projects_risks*. En caso de existir ejecutar paso 3.1.4, de lo contrario se ejecutan los pasos del 3.1.2, al 3.1.4.

Paso 3.1.2: Crear dinámicamente la partición correspondiente al año en *year_date* para la tabla maestra *gespro_projects_risks*. Para cada tabla partición se define el objeto *check*.

Paso 3.1.3: Crear el objeto *índice* a la tabla partición.

Paso 3.1.4: Retornar *year_date*.

Paso 3.2: Obtener de la tabla *project_risks_temp* las tuplas que corresponden al año en *year_date*.

Paso 3.3: Almacenar en la partición correspondiente al año en *year_date* las tuplas obtenidas en el paso 3.2.

Paso 3.4: Retornar *true*.

Los objetos *check* e *índice* se describen a continuación y se define su objetivo.

- ❖ Se estableció la creación de un índice para cada partición con el objetivo de optimizar la búsqueda de la información. Este se define sobre la columna que contiene el atributo *created_on*. La estructura del nombre de este objeto está dada por el mismo nombre de la partición y seguido el atributo *created_on*, quedando de la siguiente manera *gespro_projects_risks_<año>_created_on*.
- ❖ Se decidió definir el objeto *check* con el fin de establecer el rango de fecha para la inserción de las tuplas en cada partición. Este rango de fecha está dado por 1 año, lo que significa que para cada tabla maestra se tendrá una partición por año, teniendo en cuenta el atributo *created_on*.

En la siguiente figura se observa una vista de las particiones de las tablas maestra del almacén en el pgAdmin III, con la estructura especificada anteriormente.

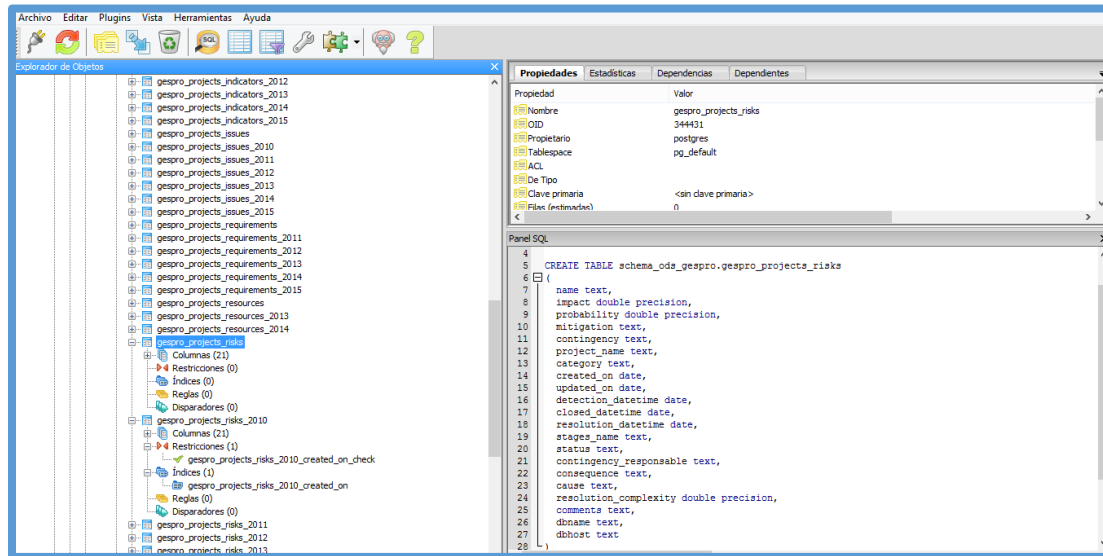


Figura 9. Vista de las particiones de la tabla *gespro_projects_risks* del almacén de datos de Xedro-GESPRO.

A continuación se muestra la parte final de la función ETL, en la que se visualiza el código para el proceso de carga.

```
FOR rec_temp IN
SELECT DISTINCT EXTRACT(YEAR FROM project_risk_temp.created_on)
AS year_date FROM project_risk_temp
LOOP
SELECT *
FROM schema_ods_gespro._ods_etl_create_dinamyc_table_risks (rec_temp.year_date)
INTO year_return;
name_table := schema_ods_gespro.gespro_projects_risks_'||year_return;
sql_insert := 'INSERT INTO '|| name_table ||'
              SELECT * FROM project_risk_temp
              WHERE
              EXTRACT(YEAR FROM project_risk_temp.created_on) = '||year_return||';
EXECUTE sql_insert;
END LOOP;
RETURN TRUE;
END;
$BODY$
LANGUAGE plpgsql VOLATILE
```



```
COST 100;
ALTER FUNCTION schema_ods_gespro._gespro_etl_projects_risks(text, integer, text,
text, text)
OWNER TO postgres;
```

El siguiente código corresponde a la función para la creación de las particiones.

```
CREATE OR REPLACE FUNCTION
schema_ods_gespro._ods_etl_create_dinamyc_table_risks(year_date double precision)
RETURNS double precision AS
$BODY$
DECLARE
    create_table CHARACTER VARYING;
    year_start DATE;
    year_end DATE;
    year_table DOUBLE PRECISION;
    name_table CHARACTER VARYING;
    name_index CHARACTER VARYING;
    create_indice CHARACTER VARYING;
BEGIN
-->Consultando el máximo created_on de la tabla maestra.
year_table := (SELECT EXTRACT (YEAR FROM m.created_on) AS year_select
                FROM schema_ods_gespro.gespro_projects_risks m
                WHERE
                EXTRACT(YEAR FROM m.created_on) = year_date LIMIT 1);
IF year_table IS NULL THEN
    year_start := CAST(year_date||'-01-01' AS DATE);
    year_end := CAST(year_date||'-12-31' AS DATE);
    name_table := 'schema_ods_gespro.gespro_projects_risks_'||year_date;
    name_index := 'gespro_projects_risks_'||year_date||'_created_on';
-->Crear tabla dinámica.
    create_table := 'CREATE TABLE '||name_table||'
```



```
(CHECK (created_on
        BETWEEN '||quote_literal(year_start)||'
        AND '||quote_literal(year_end)||'))
        INHERITS (schema_ods_gespro.gespro_projects_risks);';
create_indice := 'CREATE INDEX ' ||name_index||
        ' ON ' || name_table|| '(created_on)';
EXECUTE create_table;
EXECUTE create_indice;
RETURN year_date;
ELSE
        RETURN year_table;
END IF;
END;
$BODY$
LANGUAGE plpgsql VOLATILE
COST 100;
ALTER FUNCTION schema_ods_gespro._ods_etl_create_dinamyc_table_risks(double
precision)
OWNER TO postgres;
```

3.3 Pruebas aplicadas al almacén de datos de Xedro-GESPRO.

El proceso de prueba del software, se realiza con el objetivo de determinar la calidad del producto. Durante este proceso se detectan los errores del sistema, que no se tuvieron en cuenta durante su desarrollo. Una vez corregidos tales errores el producto puede ser entregado al cliente.

A continuación se detallan las pruebas realizadas al AD de Xedro-GESPRO.

Pruebas de carga: Evalúan el rendimiento del sistema con una carga predefinida. La prueba de carga mide cuánto se tarda un sistema para realizar diversas tareas y funciones del programa bajo condiciones normales o predefinidas. Debido a que el objetivo de las pruebas de carga es determinar si el rendimiento del sistema satisface los requisitos no funcionales de carga, es pertinente determinar, antes de comenzar las pruebas, la configuración mínima y máxima y los niveles de actividad (J Zapata, 2011).



Prueba de volumen: constituyen pruebas típicas de entornos que utilicen bases de datos. Se realizan para analizar el comportamiento de la BD con volúmenes de datos almacenados lo más similar posible a los esperados en la explotación real del sistema.

Prueba de rendimiento: este tipo de prueba se realiza para medir requisitos sensibles al tiempo como son, entre otros, tiempos de respuesta e índices de procesamiento de transacciones. Su objetivo es verificar y validar los requisitos de desempeño que se han especificado.

Pruebas de aceptación: estas pruebas son realizadas por el cliente para verificar que se cumple con los requisitos planteados por el mismo y validar su conformidad con el producto (Hernández, 2012).

Prueba Beta: La prueba beta se lleva a cabo por los usuarios finales del software en los lugares de trabajo de los clientes. El desarrollador no está presente normalmente. Así, la prueba beta es una aplicación en vivo del software en un entorno que no puede ser controlado por el desarrollador. El cliente registra todos los problemas (reales o imaginarios) que encuentra durante la prueba beta e informa a intervalos regulares al desarrollador. Como resultado de los problemas informados durante la prueba beta, el desarrollador del software lleva a cabo las modificaciones (Pressman, 2010).

3.3.1 Resultados de las pruebas.

Evaluación de la prueba de carga.

En el gráfico siguiente se observa el comportamiento de los tiempos de respuesta de una consulta *SELECT* (Anexos 1, 2, 3 y 4) realizada sobre el AD de Xedro-GESPRO en diferentes condiciones de carga.

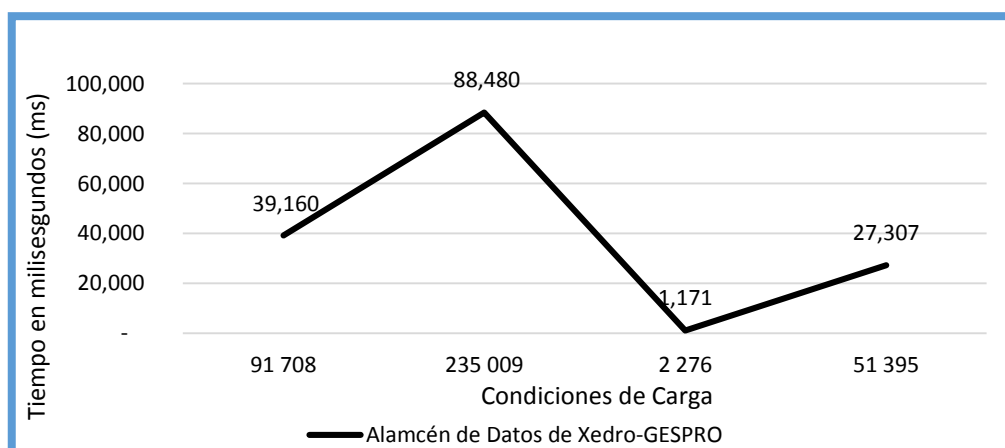


Figura 10. Rendimiento del almacén de datos de Xedro-GESPRO según condiciones de carga.



El gráfico anterior muestra la variabilidad del tiempo de respuesta del almacén de datos en diferentes condiciones de carga. Se aprecia que ante una menor carga se obtuvieron tiempos de respuesta menores, no siendo así con cargas superiores. No obstante de manera general, y desde el punto de vista de una carga por encima de las 200 000 tuplas, el tiempo de respuesta no llegó a 1 minuto. Se muestra que el tiempo máximo que se observa es de 88,480 milisegundos, es decir 0,08848 segundos con una carga a obtener de 235 009 tuplas.

Evaluación de la prueba de volumen.

Se realizó la prueba de volumen al AD de Xedro-GESPRO. Se insertaron grandes volúmenes de datos al almacén, propiciándose un significativo número de tuplas por tablas. Como resultado del proceso, no se presentaron problemas de límite de capacidad ni de volumen de datos. Estos elementos garantizaron que el gestor utilizado y el diseño de las estructuras de la BD implementadas soportan completamente el almacenamiento de los niveles de información requeridos.

Evaluación de la prueba de rendimiento.

La prueba de rendimiento se aplicó tanto al ODS Proyectos Terminados como al AD de Xedro-GESPRO, donde los resultados arrojados por la misma se muestran en la siguiente gráfica.

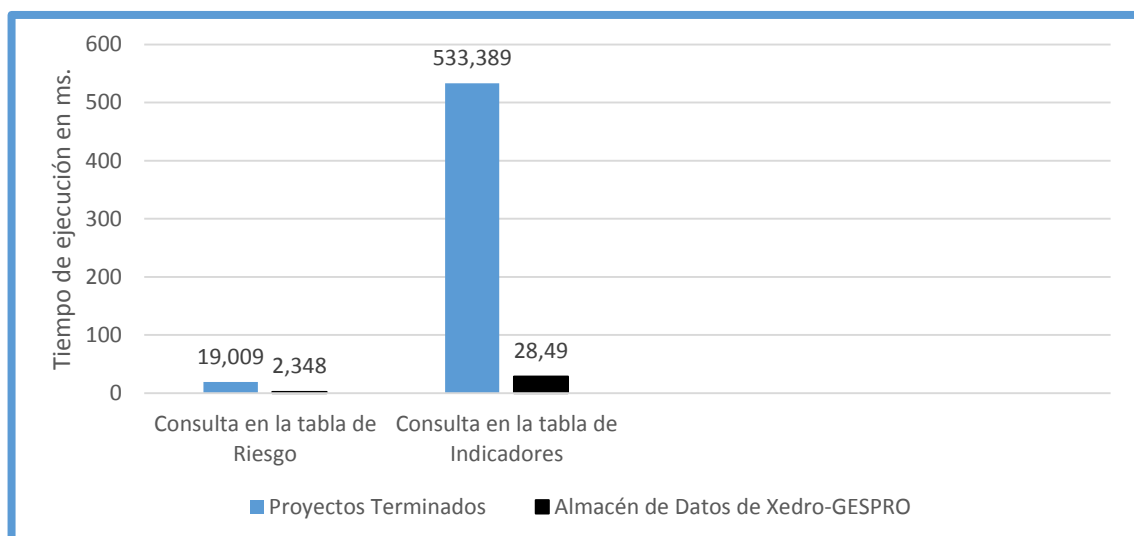


Figura 11. Rendimiento del almacén de datos de Xedro-GESPRO vs Proyectos Terminados.



En el gráfico anterior se evidencia que los tiempos de respuesta del almacén fueron menores con respecto al ODS. El ODS contiene ambigüedad de datos y por tanto las tablas presentan un significativo número de filas que obstaculizan la búsqueda de la información. Además, con respecto a la tabla de los indicadores en el ODS, un proyecto se registra siempre que exista el registro de uno de sus indicadores, lo que conllevó a que existan más filas, no siendo así en el AD de Xedro-GESPRO. Con respecto al AD de Xedro-GESPRO los indicadores son transformados en columnas de manera que un proyecto se registra una sola vez.

En los Anexos del 5 al 8 se puede observar la prueba realizada.

Evaluación de las pruebas de aceptación.

El AD de Xedro-GESPRO fue presentado por el equipo al cliente, el cual comprobó que las funcionalidades implementadas satisfacen sus necesidades. El cliente emitió la carta de aceptación, la cual valida que el sistema está listo para ser desplegado. En el Anexo 17 se muestra la carta.

Evaluación de la prueba beta.

El AD de Xedro-GESPRO se desplegó en los centros de desarrollo de software de la UCI. El proceso fue realizado por el administrador de la institución. Los pasos ejecutados son los siguientes.

Paso 1: Crear una BD vacía.

Paso 2: Restaurar el esquema que contiene el AD de Xedro-GESPRO.

Paso 3: Realizar las configuraciones de los ficheros *bash*, los cuales son ejecutados automáticamente por el comando *crontab*.

- ❖ El comando *crontab* contiene las configuraciones de tiempo de actualización del almacén realizadas por el administrador de la UCI. Se realizó un fichero *bash* para cada instancia de la BD de Xedro-GESPRO. La frecuencia de actualización se determinó que fuese semanalmente.

Al realizarse el despliegue del AD de Xedro-GESPRO en el entorno donde solo es controlado por el cliente, en este caso el administrador del servidor de los centros de desarrollo de software de la UCI, se registraron como no conformidad (NC) las siguientes:

NC 1: No se realiza el proceso ETL para los históricos de los proyectos almacenados en la BD de Xedro-GESPRO.



NC 2: No se realiza el proceso ETL para los históricos de los recursos de proyectos almacenados en la BD de Xedro-GESPRO.

NC 3: No se realiza el proceso ETL para los históricos de las tareas de proyectos almacenados en la BD de Xedro-GESPRO.

Estas NC se registraron porque el cliente no entregó la BD de Xedro-GESPRO correcta y por tanto existían inconsistencias con respecto a tipos de datos de entrada como es el caso de la NC 1 y la NC 3. Con respecto a la NC 2, la dificultad reside en que la función encargada para la creación de las particiones no se incluyó en el conjunto de funciones del almacén. No obstante todas las no conformidades se solucionaron y se entregó la nueva versión al cliente para ser desplegada. Esto demuestra que en un primer momento el producto obtenido no se adaptó al ambiente donde se emplea Xedro-GESPRO, no obstante al solucionarse las NC se logró adaptar al ambiente correspondiente.

Evaluación de la eficiencia del Almacén de Datos de Xedro-GESPRO.

A partir de los valores del gráfico que mide el rendimiento del AD de Xedro-GESPRO con respecto al ODS Proyectos Terminados, se realizó el análisis para comprobar el tiempo de respuesta del AD de Xedro-GESPRO. En la siguiente tabla se visualiza el promedio de tiempo de respuesta de ambas tecnologías de almacenamiento a partir de los valores especificados en el gráfico.

Tecnología de almacenamiento.	Promedio de tiempo de respuesta. (milisegundos)
AD de Xedro-GESPRO	15,419
ODS Proyectos Terminados	276,1965

Tabla 7. Promedio del tiempo de respuesta del almacén de datos de Xedro-GESPRO.

Según los valores de la tabla, estos demostraron que los tiempos de respuesta de una consulta del cliente sobre el AD de Xedro-GESPRO son menores con respecto a los del ODS Proyectos Terminados. Finalmente, se puede afirmar que el AD de Xedro-GESPRO presenta alta eficiencia con respecto al ODS Proyectos Terminados.

Evaluación de la funcionalidad y soporte del AD de Xedro-GESPRO.



Para la evaluación de los RNF correspondientes a los aspectos de funcionalidad y soporte se emplearon las listas de chequeo (LCH). A continuación se especifican las correspondientes a dichos atributos.

Característica de funcionalidad por comprobar (interoperabilidad del producto).	Ponderación de importancia. 1(poco), 5 (fundamental)	Nivel de cumplimiento. (0-100)	Justificación.
1-¿El subsistema de integración establece la conexión a la BD de Xedro-GESPRO, de manera que logra interactuar con la misma?	5	100	Se realizó la ejecución de las funcionalidades para el proceso ETL de los datos contenidos en la BD de Xedro-GESPRO, lográndose realizar dicho proceso y por tanto se estableció la conexión con la fuente de origen. Esto significa que el AD interactuó 100% con la BD de Xedro-GESPRO.

Tabla 8. Lista de chequeo. Evaluación de la funcionalidad del almacén de datos de Xedro-GESPRO (Ocampo Acosta, y otros, 2011).

NC 1: Errores de ortografía en los nombres de funciones.

Característica de soporte a comprobar.	Ponderación de importancia 1(poco), 5 (fundamental)	Nivel de cumplimiento (0-100)	Justificación
1-¿Qué SGBD fue utilizado para el desarrollo del AD?	5	100	Para el desarrollo del AD se empleó PostgreSQL Server 9.2.
2-¿Por qué es usado?	5	100	Es empleado con el objetivo de:



			<ul style="list-style-type: none">▪ Mantener la simetría entre los la BD de Xdero-GESPRO y el AD, ya que dicha BD utiliza como SGBD PostgreSQL.▪ Realizar el particionado de tablas. <p>Además el equipo de desarrollo tiene más de 5 años de experiencia con el SGBD PostgreSQL.</p>
--	--	--	--

Tabla 9: Lista de chequeo. Evaluación del RNF de soporte del almacén de datos de Xedro-GESPRO (Ocampo Acosta, y otros, 2011).

El AD de Xedro-GESPRO permite el almacenamiento la información referente a los riesgos, los cambios, los requerimientos, los indicadores de los proyectos, los recursos, las tareas, las desviaciones, los proyectos y los indicadores de los recursos humanos de un proyecto, mediante la definición de los modelos físico y lógico de datos y un conjunto de funciones. Posibilitó mejor organización de los datos mediante el empleo del particionado de tablas, de manera que se optimiza la búsqueda de la información. El particionado de tablas propició un nivel superior de capacidad del almacén y menor número de filas por cada tabla, lo cual demandó menores tiempos de respuesta de una consulta realizada sobre este, no siendo así en el ODS debido a las dificultades expuestas en la introducción de esta investigación.

3.4 Impacto social.

La creación del AD de Xedro-GESPRO, en cumplimiento con el lineamiento 131 de la Política Económica y Social del Partido y la Revolución del VI Congreso del Partido Comunista de Cuba, el cual se refiere a sostener y desarrollar los resultados alcanzados, entre otros elementos, en la industria del software y el proceso de informatización de la sociedad, constituyó una solución al problema planteado por el Departamento de Gestión de Proyecto del CDAE.



3.5 Conclusiones del capítulo.

Luego de la implementación y prueba del AD de Xedro-GESPRO se arribaron a las siguientes conclusiones:

- ❖ La implementación del subsistema de almacenamiento y el desarrollo de las funciones encargadas de realizar el proceso ETL, permitió la creación de las tablas que conforman el AD de Xedro-GESPRO y el particionado lógico de las mismas respectivamente.
- ❖ La ejecución de las pruebas sobre el AD arrojaron un total de 4 NC, las cuales fueron resueltas en la 2 iteración de pruebas y demostraron el correcto funcionamiento del sistema, así como también la validación por parte del cliente.



CONCLUSIONES GENERALES

Con la realización de la investigación se logró cumplir con los objetivos planteados, desarrollándose el AD de Xedro-GESPRO, el cual soporta el almacenamiento de datos históricos y resumidos. Por lo que se concluye que:

- ❖ El análisis de las características, la arquitectura, la estructura y enfoque de diseño de los almacenes de datos permitió precisar los objetivos reales de estos dentro de la arquitectura de los sistemas de información. A partir de dicho estudio se logró definir el AD de Xedro-GESPRO para el almacenamiento de los datos históricos y resumidos relacionados con la GP contenidos en la BD de Xedro-GESPRO.
- ❖ El empleo de PostgreSQL Server 9.2 permitió la aplicación de la estrategia del particionado de tabla, con lo cual se garantizó mejor organización de la información y tiempos de respuesta al ser consultada la misma por los clientes del AD.
- ❖ Las pruebas efectuadas al sistema permitieron la validación del mismo, obteniendo resultados positivos y la aceptación del cliente.



RECOMENDACIONES

- ❖ Desarrollar el subsistema de visualización para el almacén de datos de Xedro-GESPRO.



REFERENCIAS BIBLIOGRÁFICAS

ArPUG. 2015. Grupo de usuarios PostgreSQL de Argentina Website. [En línea] 2015. Disponible en: <http://www.postgresql.org.ar/trac/wiki/PgAdmin>.

Darío, Bernabeu R. 2010. *DATA WAREHOUSING: Investigación y Sistematización de Conceptos. Hefesto: Metodología para la construcción de un Data Warehouse.* Córdoba, Argentina : s.n., 2010.

Date, C. J. 2001. *Introducción a los Sistemas de bases de datos.* México : PEARSON EDUCATION, 2001. ISBN: 968-444-419-2 .

Díaz, Josep Cuarto. 2010. *Introducción al Business Intelligence.* Barcelona : UOC, 2010. ISBN: 978-84-9788-886-8.

Google. 2015. Ayuda de Administrador de Google Apps. *Acerca de Google App Engine.* [En línea] 2015. Disponible en: <https://support.google.com/a/?hl=es> .

Hernández, Yanisbel González. 2012. *PROPUESTA DE METODOLOGIA PARA EL DASARROLLO DE ALMACENES DE DATOS.* Centro de Tecnologías de Gestión de Datos (DATEC), facultad 6, UCI,La Habana, Cuba : s.n., 2012. s.n.

HP Website. 2015. HP. *Monetice sus datos con análisis de nube de nivel empresarial.* [En línea] Hewlett-Packard Development Company, L.P., 2015. Disponible en: <https://saas.hp.com/es-es/software/vertica>.

Inmon, William H. 2005. *Building the Data Warehouse, Fourth Edition.* Indiana, Indianapolis : Wiley Publishing, Inc., 2005.

J Zapata, Carlos Mario. 2011. Comparación de las características de algunas herramientas de software para herraminetas de software. [En línea] Julio de 2011. Disponible en: <http://www.bdigital.unal.edu.co/28847/1/26734-93661-1-PB.pdf>. 1657-7663.

Jackson, Joab. 2013. IDG News Service. *Facebook entra al código abierto con su almacén de datos embebido.* [En línea] Saya Comunicaciones S.A.C., 26 de Noviembre de 2013. Disponible en : <http://cioperu.pe/articulo/14597/facebook-entra-al-codigo-abierto-con-su-almacen-de-datos-embebido/?p=2>.

Kimball, Ralph. 2004. *The Data Warehouse ETL Toolkit.* Indianapolis : Wiley Publishing, Inc., 2004. SBN: 0-764-57923-1.



Mato García, Rosa María. 2005. *Sistemas de Base de Datos*. Cuba : Pueblo y Educación, 2005. ISBN 959-13-1273-3.

Ocampo Acosta, Alejandro y Correa Tapasco, Luisa Marcela. 2011. *IMPACTO DE LAS PRUEBAS NO FUNCIONALES EN LA MEDICIÓN DE LA CALIDAD DEL PRODUCTO SOFTWARE DESARROLLADO*. Pereira : s.n., 2011.

Pampliega, Carlos J. 2014. *Gestión de Proyectos, ¿moda o necesidad?* *Project Management Institute. Madrid, Spain Chapter*. [En línea] 2014. Disponible en: http://www.pmi-mad.org/index.php?option=com_content&view=article&id=551:gestion-de-proyectos-imoda-o-necesidad&catid=137:articulos&Itemid=88.

Piñero Pérez, Pedro Y y Colectivo de Autores. 2014. *Gespro. Paquete para la Gestión de Proyectos*. [En línea] 21 de Mayo de 2014. Disponible e: http://www.researchgate.net/publication/260418890_GESPRO._Paquete_para_la_gestin_de_proyectos.

Piñero Pérez, Pedro Y y Colectivo de Autores. 2014. *SISTEMA DE INFORMACIÓN PARA LA GESTIÓN DE ORGANIZACIONES ORIENTADAS A PROYECTOS. RIIPRO - Congresos y Jornadas, V Congreso Iberoamericano de Ingeniería de Proyectos*. [En línea] 01 de 10 de 2014. Disponible en: <http://congreso.riipro.org/index.php/CIIP/V-CIIP/paper/view/105>.

PMI. 2013. *Guía de los Fundamentos para la Dirección de Proyectos*. s.l. : Project Management Institute, Inc., 2013. ISBN: 978-1628250091.

PostgreSQL Website. 2012. PostgreSQL. [En línea] The PostgreSQL Global Development Group, 2012. Disponible en: <http://www.postgresql.org.es>.

Pressman, Roger S. 2010. *Software Engineering: A Practitioner's Approach*. [ed.] McGraw-Hil. Seven Edition. New York : The McGraw-Hill Companies Inc., 2010. pág. 469 Chapter 17. ISBN: 978-0-07-337597-7.

Sinnexus Inc. 2012. Sinnexus. *Sistemas de Soporte a la Decision (DSS)*. [En línea] 2012. Disponible en: http://www.sinnexus.com/business_intelligence/sistemas_soporte_decisiones.aspx..



Slideshare Inc. 2011. Slideshare. *Tipos de Sistemas de Información*. [En línea] 3 de Octubre de 2011. Disponible en: <http://es.slideshare.net/GiancarloNebiolonavidad/tipos-sistemas-de-informacion-tpsmisdssess..>

—. **2012.** Slideshare . *Sistemas de apoyo a la toma de decisiones*. [En línea] 2012. Disponible en: <http://www.slideshare.net/heynan/sistemas-de-apoyo-a-la-toma-de-decisiones-12905870..>

Soloelectronicos. 2015. *Google avisa a los usuarios de la plataforma de Google Cloud*. [En línea] 21 de Abril de 2015. Disponible en: <http://soloelectronicos.com/tag/almacen-de-datos-masterslave-de-google/>.

The PostgreSQL Global Development Group. 2015. PostgreSQL. [En línea] 2015. Disponible en: <http://www.postgresql.org/docs/9.2/static/plpgsql.html>.

Trujillo, Juan Carlos, Mazón, José Norberto y Pardillo Vela, Jesús. 2011. *DISEÑO Y EXPLOTACIÓN DE ALMACENES DE DATOS. Conceptos Básicos de Modelado Multidimensional*. Alicante : Editorial Club Universitario, 2011. ISBN: 978-84-8454-902-4.

Visual Paradigm Website. 2015. Visual Paradigm Organization. *Visual Paradigm* . [En línea] 2015. Disponible en: <http://www.visual-paradigm.com/>.



ANEXOS

Anexo 1. Consulta *SELECT* en el almacén de datos de Xedro-GESPRO sobre la tabla *gespro_projects_requirements*.

Query - almacen_de_datos_despliegue en postgres@localhost5432*

```
1 EXPLAIN ANALYZE SELECT project_name, name, description, priority, prototype, functional,
2 status, created_on, updated_on, qip, qc, gid, gis, gif, solicitud_date,
3 parent_id, type, dbname, dbhost
4 FROM schema_ods_gespro.gespro_projects_requirements;
```

Panel de Salida

QUERY PLAN
1 Result (cost=0.00..2906.00 rows=91709 width=139) (actual time=0.005..37.637 rows=91708 loops=1)
2 -> Append (cost=0.00..2906.00 rows=91709 width=139) (actual time=0.004..18.759 rows=91708 loops=1)
3 -> Seq Scan on gespro_projects_requirements (cost=0.00..0.00 rows=1 width=297) (actual time=0.000..0.000 rows=0 loops=1)
4 -> Seq Scan on gespro_projects_requirements 2011 gespro_projects_requirements (cost=0.00..104.49 rows=3249 width=142) (actual time=0.000..0.000 rows=0 loops=1)
5 -> Seq Scan on gespro_projects_requirements 2012 gespro_projects_requirements (cost=0.00..2749.69 rows=87469 width=137) (actual time=0.000..37.637 rows=87468 loops=1)
6 -> Seq Scan on gespro_projects_requirements 2013 gespro_projects_requirements (cost=0.00..51.90 rows=990 width=295) (actual time=0.000..0.000 rows=989 loops=1)
7 Total runtime: 39.160 ms

Anexo 2. Consulta *SELECT* en el almacén de datos de Xedro-GESPRO sobre la tabla *gespro_projects_deviations*.

Query - almacen_de_datos_despliegue en postgres@localhost5432*

```
1 EXPLAIN ANALYZE SELECT name, description, impact, detection_datetime, created_on, updated_on,
2 author_name, project_name, project_risk_name, deviations_type_name,
3 dbname, dbhost
4 FROM schema_ods_gespro.gespro_projects_deviations;
```

Panel de Salida

QUERY PLAN
1 Result (cost=0.00..10212.09 rows=235010 width=267) (actual time=0.039..83.112 rows=235009 loops=1)
2 -> Append (cost=0.00..10212.09 rows=235010 width=267) (actual time=0.037..44.116 rows=235009 loops=1)
3 -> Seq Scan on gespro_projects_deviations (cost=0.00..0.00 rows=1 width=272) (actual time=0.000..0.000 rows=0 loops=1)
4 -> Seq Scan on gespro_projects_deviations 2010 gespro_projects_deviations (cost=0.00..3138.28 rows=76328 width=252) (actual time=0.036..83.112 rows=76327 loops=1)
5 -> Seq Scan on gespro_projects_deviations 2011 gespro_projects_deviations (cost=0.00..2889.89 rows=76589 width=232) (actual time=0.030..83.112 rows=76588 loops=1)
6 -> Seq Scan on gespro_projects_deviations 2012 gespro_projects_deviations (cost=0.00..1170.13 rows=22913 width=323) (actual time=0.002..83.112 rows=22912 loops=1)
7 -> Seq Scan on gespro_projects_deviations 2013 gespro_projects_deviations (cost=0.00..2781.60 rows=54860 width=310) (actual time=0.031..83.112 rows=54859 loops=1)
8 -> Seq Scan on gespro_projects_deviations 2014 gespro_projects_deviations (cost=0.00..232.19 rows=4319 width=328) (actual time=0.002..83.112 rows=4318 loops=1)
9 Total runtime: 86.856 ms



Anexo 3. Consulta *SELECT* en el almacén de datos de Xedro-GESPRO sobre la tabla *gespro_projects_risks*.

The screenshot shows a PostgreSQL query editor window titled "Query - almacen_de_datos_despliegue en postgres@localhost:5432 *". The editor contains the following SQL query:

```
1 EXPLAIN ANALYZE SELECT name, impact, probability, mitigation, contingency, project_name,
2 category, created_on, updated_on, detection_datetime, closed_datetime,
3 resolution_datetime, stages_name, status, contingency_responsible,
4 consequence, cause, resolution_complexity, comments, dbname,
5 dbhost
6 FROM schema_ods_gespro.gespro_projects_risks;
```

The "Salida de datos" (Data Output) panel displays the "QUERY PLAN" (Query Plan) for the execution:

Step	Operation	Cost	Rows	Width	Actual Time	Actual Rows	Loops
1	Result	(cost=0.00..154.76 rows=2277 width=414)	2277	414	(actual time=0.005..1.057 rows=2276 loops=1)	2276	1
2	Append	(cost=0.00..154.76 rows=2277 width=414)	2277	414	(actual time=0.004..0.442 rows=2276 loops=1)	2276	1
3	Seq Scan on gespro_projects_risks	(cost=0.00..0.00 rows=1 width=460)	1	460	(actual time=0.000..0.000 rows=0 loops=1)	0	1
4	Seq Scan on gespro_projects_risks 2010 gespro_projects_risks	(cost=0.00..38.67 rows=567 width=430)	567	430	(actual time=0.002..0.092 rows=56)	56	1
5	Seq Scan on gespro_projects_risks 2011 gespro_projects_risks	(cost=0.00..31.66 rows=466 width=409)	466	409	(actual time=0.001..0.076 rows=46)	46	1
6	Seq Scan on gespro_projects_risks 2012 gespro_projects_risks	(cost=0.00..47.54 rows=654 width=450)	654	450	(actual time=0.002..0.129 rows=65)	65	1
7	Seq Scan on gespro_projects_risks 2013 gespro_projects_risks	(cost=0.00..30.94 rows=494 width=374)	494	374	(actual time=0.002..0.077 rows=49)	49	1
8	Seq Scan on gespro_projects_risks 2014 gespro_projects_risks	(cost=0.00..5.95 rows=95 width=307)	95	307	(actual time=0.004..0.017 rows=95)	95	1
9	Total runtime:				1.171 ms		

Anexo 4. Consulta *SELECT* en el almacén de datos de Xedro-GESPRO sobre la tabla *gespro_projects_changes*.

The screenshot shows a PostgreSQL query editor window titled "Query - almacen_de_datos_despliegue en postgres@localhost:5432 *". The editor contains the following SQL query:

```
1 EXPLAIN ANALYZE SELECT id, name, description, priority, baseline_modified, configuration_element,
2 project_change_status_id, justified, environment, created_on,
3 updated_on, impact, active, author, detection, approved_by, charge,
4 date_detection, date_closed, project_name, author, dbname, dbhost
5 FROM schema_ods_gespro.gespro_projects_changes;
```

The "Salida de datos" (Data Output) panel displays the "QUERY PLAN" (Query Plan) for the execution:

Step	Operation	Cost	Rows	Width	Actual Time	Actual Rows	Loops
1	Result	(cost=0.00..3814.95 rows=51396 width=620)	51396	620	(actual time=0.049..26.450 rows=51395 loops=1)	51395	1
2	Append	(cost=0.00..3814.95 rows=51396 width=620)	51396	620	(actual time=0.047..14.263 rows=51395 loops=1)	51395	1
3	Seq Scan on gespro_projects_changes	(cost=0.00..0.00 rows=1 width=480)	1	480	(actual time=0.001..0.001 rows=0 loops=1)	0	1
4	Seq Scan on gespro_projects_changes 2010 gespro_projects_changes	(cost=0.00..2231.07 rows=32607 width=604)	32607	604	(actual time=0.046..8.483)	32607	1
5	Seq Scan on gespro_projects_changes 2011 gespro_projects_changes	(cost=0.00..1435.92 rows=16992 width=656)	16992	656	(actual time=0.039..4.129)	16992	1
6	Seq Scan on gespro_projects_changes 2012 gespro_projects_changes	(cost=0.00..146.93 rows=1793 width=580)	1793	580	(actual time=0.002..0.174)	1793	1
7	Seq Scan on gespro_projects_changes 2013 gespro_projects_changes	(cost=0.00..1.03 rows=3 width=480)	3	480	(actual time=0.001..0.001 rows=3)	3	1
8	Total runtime:				27.307 ms		



Anexo 5. Consulta *SELECT* con condicional *WHERE* en el ODS sobre la tabla *projects_risks*.

The screenshot shows a PostgreSQL query editor window titled "Query - proyectos_terminados en postgres@localhost:5432". The query is as follows:

```
1 EXPLAIN ANALYZE SELECT project_risks_id, name, impact, probability, mitigation, contingency,  
2 project_name, category, created_on, updated_on, detection_datetime,  
3 closed_datetime, resolution_datetime, milestone_name, status,  
4 contingency_responsible, consequence, cause, resolution_complexity, |  
5 comments, dbname, dbhost  
6 FROM project_risks where date_part ('year', project_risks.created_on)=2010;  
7
```

The "Panel de Salida" (Output Panel) shows the "QUERY PLAN" text:

Step	Operation	Cost	Rows	Width	Actual Time
1	Seq Scan on project_risks	cost=0.00..1889.32	rows=122	width=444	(actual time=0.018..18.65
2	Filter: (date_part('year'::text, (created_on)::timestamp without time zone) = 2010::double				
3	Rows Removed by Filter: 14547				
4	Total runtime: 19.009 ms				

Anexo 6. Consulta *SELECT* con condicional *WHERE* en el almacén de datos de Xedro-GESPRO sobre la tabla *gespro_projects_risks*.

The screenshot shows a PostgreSQL query editor window titled "Query - almacen_de_datos_despliegue en postgres@localhost:5432". The query is as follows:

```
1 EXPLAIN ANALYZE SELECT name, impact, probability, mitigation, contingency, project_name,  
2 category, created_on, updated_on, detection_datetime, closed_datetime,  
3 resolution_datetime, stages_name, status, contingency_responsible,  
4 consequence, cause, resolution_complexity, comments, dbname,  
5 dbhost  
6 FROM schema_ods_gespro.gespro_projects_risks where date_part ('year', schema_ods_gespro.gespro_projects_risks.created_on)=  
7 2010;
```

The "Panel de Salida" (Output Panel) shows the "QUERY PLAN" text:

Step	Operation	Cost	Rows	Width	Actual Time
7	-> Seq Scan on gespro_projects_risks_2011 gespro_projects_risks	cost=0.00..35.16	rows=2		
8	Filter: (date_part('year'::text, (created_on)::timestamp without time zone) = 2010::d				
9	Rows Removed by Filter: 466				
10	-> Seq Scan on gespro_projects_risks_2012 gespro_projects_risks	cost=0.00..52.45	rows=3		
11	Filter: (date_part('year'::text, (created_on)::timestamp without time zone) = 2010::d				
12	Rows Removed by Filter: 654				
13	-> Seq Scan on gespro_projects_risks_2013 gespro_projects_risks	cost=0.00..34.65	rows=2		
14	Filter: (date_part('year'::text, (created_on)::timestamp without time zone) = 2010::d				
15	Rows Removed by Filter: 494				
16	-> Seq Scan on gespro_projects_risks_2014 gespro_projects_risks	cost=0.00..6.66	rows=1 w		
17	Filter: (date_part('year'::text, (created_on)::timestamp without time zone) = 2010::d				
18	Rows Removed by Filter: 95				
19	Total runtime: 2.348 ms				



Anexo 7. Consulta **SELECT** con condicional **WHERE** en el ODS sobre la tabla **gespro_projects_indicators**.

```
1 EXPLAIN ANALYZE SELECT gespro_projects_indicators_id, project_name, indicator, indicator_value,
2   created_on, updated_on, cant_issues, cant_users, cant_requirements,
3   cant_risks, cant_resources, cant_changes, cant_deviations, dbname,
4   dbhost
5   FROM gespro_projects_indicators where date_part ('year',gespro_projects_indicators.created_on)=2012;
6
```

QUERY PLAN
text

1	Seq Scan on gespro_projects_indicators (cost=0.00..47335.73 rows=1)
2	Filter: (date_part('year'::text, (created_on)::timestamp without time zone) = 2012::double precision)
3	Rows Removed by Filter: 312210
4	Total runtime: 533.389 ms

Anexo 8. Consulta **SELECT** con condicional **WHERE** en el almacén de datos de Xedro-GESPRO sobre la tabla **gespro_projects_indicators**.

```
1 EXPLAIN ANALYZE SELECT project_name, created_on, updated_on, cant_issues, cant_users,
2   cant_requirements, cant_risks, cant_resources, cant_changes,
3   cant_deviations, dbname, dbhost, "IE", "ISE", "IRI", "ICD", "IRC", "IRL", "IRRH", "IREF", "ICDP", "ICDE", "ICDV", "ICDPO", "EV",
4   "EV", "AC", "BAC", "IRRH", "IRHE", "IRHA", "IRHF", "Real-IRE", "Real-IRL", "Real-Contabilidad", "Plan-Bitos", "Real-IRP", "Plan-IRP",
5   "IRIP", "IRIR", "IRIR", "Fondo de Tiempo", "Tiempo Planificado", "Tiempo Trabajado", "Financiamiento", "Cant. Riesgos Exp. Alto",
6   "Cant. Riesgos Exp. Medio", "Cant. Riesgos Exp. Bajo", "Impacto otal Hitos Ejec", "Costo Total Hitos Ejec", "Cant. Dev. Calidad", "Cant. Dev. Costo",
7   "Cant. Dev. Tiempo", "Cant. Dev. Logísticos", "Cant. Dev. Desempeño", "Cant. Dev. Impacto Alto", "Cant. Dev. Impacto Medio", "Cant. Dev. Impacto Bajo",
8   "Cant. Requisitos Propuestos", "Cant. Requisitos Aprobados", "Cant. Requisitos Propuestos", "Cant. Requisitos Terminados", "Cant. NC Formato", "Cant. NC Error técnico",
9   "Cant. NC Correspondencia con otro artefacto", "Cant. NC Ortografía", "Cant. NC Redacción", "Cant. NC Validación", "Cant. NC Opciones que no funcionan", "Cant. NC Error
10  interfaz", "Cant. NC Error idioma", "Cant. NC Funcionalidad", "Cant. NC Excepciones", "Cant. NC Seguridad", "Cant. NC Diseño CP", "Cant. NC Recomendación", "Cant. NC Otros",
11  "Cant. RRRH Eval B", "Cant. RRRH Eval R", "Cant. RRRH Eval M", "Cant. Estudiante 1ro", "Cant. Estudiante 2do", "Cant. Estudiante 3ro", "Cant. Estudiante 4to", "Cant.
12  Estudiante 5to", "Cant. RGA-Recien Graduado en Adiestramiento", "Cant. IRG-Instructor Recien Graduado", "Cant. Especialista", "Cant. Profesor de Centro", "Cant. Profesor de
13  Facultad trabaja en Centro", "Cant. Profesor de otras áreas trabaja en Centro", "Cant. Personal externo a la universidad", "Cant. Técnico", "Cant. Asistente de control",
14  "Cant. Otros trabajadores"
15  FROM schema_ods_gespro.gespro_projects_indicators where date_part ('year',schema_ods_gespro.gespro_projects_indicators.created_on)=2012;
16
```

QUERY PLAN
text

5	Seq Scan on gespro_projects_indicators_2012 gespro_projects_indicators (cost=0.00..1078.01 rows=45 width=765) (actual time=0.000..0.000 rows=45)
6	Filter: (date_part('year'::text, (created_on)::timestamp without time zone) = 2012::double precision)
7	-> Seq Scan on gespro_projects_indicators_2013 gespro_projects_indicators (cost=0.00..2297.25 rows=97 width=760) (actual time=0.000..0.000 rows=97)
8	Filter: (date_part('year'::text, (created_on)::timestamp without time zone) = 2012::double precision)
9	Rows Removed by Filter: 19443
10	-> Seq Scan on gespro_projects_indicators_2015 gespro_projects_indicators (cost=0.00..14.01 rows=1 width=764) (actual time=0.000..0.000 rows=1)
11	Filter: (date_part('year'::text, (created_on)::timestamp without time zone) = 2012::double precision)
12	Rows Removed by Filter: 9086
13	-> Seq Scan on gespro_projects_indicators_2015 gespro_projects_indicators (cost=0.00..14.01 rows=1 width=764) (actual time=0.000..0.000 rows=1)
14	Filter: (date_part('year'::text, (created_on)::timestamp without time zone) = 2012::double precision)
15	Rows Removed by Filter: 115
16	Total runtime: 28.490 ms

**Anexo 9. Descripción CUF 2: Administrar los datos históricos de los cambios.**

Objetivo	Permitir el almacenamiento de los datos históricos de los cambios.	
Actores	Administrador.	
Resumen	Administrador: (inicia) Extrae, transforma y carga los datos históricos de los cambios.	
Complejidad	Simple.	
Prioridad	Alta.	
Precondiciones	La BD de Xedro-GESPRO y la información contenida en ella deben estar disponibles.	
Postcondiciones	Los datos históricos de los cambios deben ser almacenados en el AD.	
Flujo de eventos		
Flujo básico <Almacenar Datos>		
	Actor	Sistema
1	1 El administrador configura el fichero <i>bash</i> para la BD de Xedro-GESPRO.	
2	2 El administrador programa la ejecución del fichero <i>bash</i> mediante el comando <i>crontab</i> y este se encarga de la ejecución de dicho fichero automáticamente.	2.1 Al ejecutarse el fichero <i>bash</i> el sistema realiza las siguientes acciones: <ul style="list-style-type: none">- Extraer los datos de la tabla que contiene la información referente a los cambios a partir del año 2010 hasta la fecha.- Transformar los datos extraídos de la tabla que contiene la información referente a los cambios a partir del año 2010 hasta la fecha.- Cargar los datos extraídos y transformados



		referente a los cambios a partir del año 2010 hasta la fecha.
Flujos alternos		
Sección 1: “Extraer los datos de la tabla que contiene la información referente a los cambios a partir del año 2010 hasta la fecha.”		
Flujo básico < Extraer los datos de la tabla que contiene la información referente a los cambios a partir del año 2010 hasta la fecha.>		
	Actor	Sistema
1		2.1 Establece la conexión a la BD origen.
2		3.1 Extrae los datos especificados, en correspondencia con la fecha de actualización.
Flujos alternos		
Nº Evento <Datos incorrectos>		
	Actor	Sistema
1		2.1 Muestra los siguientes mensajes o uno de ellos en dependencia del dato introducido incorrectamente: <ul style="list-style-type: none">- “Usuario incorrecto”- “Contraseña incorrecta”- “Puerto incorrecto”- “Host incorrecto”
Flujos alternos		
Nº Evento <Fecha fuera de rango.>		
	Actor	Sistema
1		3.1 No extrae nueva información.
Sección 1: “Transformar los datos extraídos de la tabla que contiene la		



información referente a los cambios a partir del año 2010 hasta la fecha.”		
Flujo básico < Transformar los datos extraídos de la tabla que contiene la información referente a los cambios a partir del año 2010 hasta la fecha.>		
	Actor	Sistema
1		2.1 Transforma los datos extraídos de acuerdo a las características que deben presentar.
2		3.1 Almacena la información en una tabla temporal.
Sección 1: “Cargar los datos extraídos y transformados referente a los cambios a partir del año 2010 hasta la fecha.”		
Flujo básico < Cargar los datos extraídos y transformados referente a los cambios a partir del año 2010 hasta la fecha.>		
	Actor	Sistema
1		2.1 Crea las particiones para la tabla de los históricos referentes a los cambios.
2		3.1 Almacena los datos en cada partición de acuerdo a la fecha.
Flujos alternos		
Nº Evento <Partición creada.>		
	Actor	Sistema
1		2.1 No crea nuevas particiones y almacena la información en la partición existente correspondiente al año de los datos a insertar.