



Facultad 3

Carga inicial de los datos y migración total del SIPAC 2.4 a SIPAC 3.0

Informe Técnico en opción al título de Ingeniero en Ciencias Informáticas

Autor: Cristhian Yurien Cartaya Garrido

Tutor: Ing. Samuel Ojeda Pereira

La Habana, 2020

Año 62 de la Revolución

AGRADECIMIENTOS

Primeramente, quisiera agradecerle a mi familia en general, especialmente a mi madre, a mi abuelo y a mi abuela, por apoyarme en cada decisión y por estar siempre ahí cuando los necesité.

A mis compañeros de clases que me dieron su apoyo necesario durante los 5 años de la carrera.

A mis profesores por su paciencia y entrega para que en el día de hoy sea un buen profesional.

A mi tutor por su tiempo dedicado, sus consejos.

Al profe Reinier por su paciencia y confianza depositada en mí.

A toda la familia que me llevo de la UCI: Adrián (Culito), David, Naisel, Nailee, Loandry y en especial a Chris muchas gracias por la ayuda a lo largo de estos 5 años.

A todos los que de una forma u otra han contribuido a la realización del presente trabajo.

A todos, muchas gracias.

DEDICATORIA

A mi madre, a mi abuelo y a mi abuela, por apoyarme en cada decisión y por estar siempre presente
cuando los necesité.

A toda mi familia por ser todo para mí.

RESUMEN

El Centro de Gestión de Entidades, perteneciente a la Universidad de Ciencias Informáticas, se encuentra inmerso en el proceso de actualización del Sistema de Planificación de Actividades. Formando parte así del paquete de soluciones integrales de gestión, Xedro, para las entidades presupuestadas y empresariales. Actualmente se encuentra desplegado en su versión 2.4 en varias entidades del país. Teniendo en cuenta este proceso de actualización, se desarrolla la versión 3.0. Con esta nueva versión existe una relación conceptualmente, pero una inconsistencia entre los datos estructuralmente en las bases de datos de ambos sistemas. Además, los datos difieren en la estructura entre versiones generando problemas como: ruido en los datos, pérdida y nuevos valores. Estos datos discrepan en composición de campos, campos que fueron simplificados, agregados y eliminados en la nueva versión, provocando problemas de inconsistencia e incompatibilidad, generando problemas en la integridad de los datos. Por esta razón la presente investigación tiene como objetivo desarrollar una estrategia para la carga inicial y migración total de los datos de la versión 2.4 a la 3.0 de dicho sistema. Para dicha estrategia se tuvo en cuenta la metodología establecida por el Centro de Tecnologías de Gestión de Datos de la propia universidad. Se utilizó la herramienta Kettle perteneciente a la suite de Pentaho, como herramienta de Extracción, Transformación y Carga. Como resultado se obtuvo la estrategia para la carga inicial y migración total de los datos, disminuyendo así la discrepancia entre los datos y el riesgo de pérdida de información entre ambas versiones.

Palabras claves: carga inicial, estrategia, extracción, transformación, migración de datos, migración total.

ÍNDICE GENERAL

INTRODUCCIÓN	1
DESARROLLO.....	5
Introducción.....	5
Conceptos asociados a la investigación	5
Tendencias actuales de la migración de datos	7
Proceso de extracción, transformación y carga	8
Migración de los datos en sistemas informáticos.....	9
Metodología.....	10
Metodología Kimball	11
Metodología Hefesto.....	12
Metodología de desarrollo de proyectos de almacenes de datos.....	13
Herramientas y tecnologías analizadas para la solución.....	16
Herramientas de modelado.....	16
Lenguaje de modelado	17
Sistema gestor de bases de datos.....	17
Herramienta de gestión de bases de datos.....	18
Herramientas para la Extracción, Transformación y Carga.....	18
Estrategia propuesta al problema planteado.....	21
Estudio preliminar y planeación:	21
Requisitos.....	22
Diseño para el subsistema de integración de datos	29
Implementación del subsistema de integración de datos	34
Validación y pruebas	40
CONCLUSIONES.....	44
RECOMENDACIONES	45
REFERENCIAS BIBLIOGRÁFICAS	46
ANEXOS	49

ÍNDICE DE FIGURAS

Figura 1: Etapas de la metodología Kimball	11
Figura 2: Etapas de la metodología Hefesto.....	12
Figura 3: Ciclo de vida de la metodología de almacenes de datos	14
Figura 4: Modelo físico de datos de la solución propuesta	32
Figura 5: Estrategia para la migración de SIPAC.....	34
Figura 6: Carga de las estructuras del SIPAC 2.4 al SIPAC 3.0	34
Figura 7: Carga de los usuarios del SIPAC 2.4 al SIPAC 3.0	35
Figura 8: Entrada de tabla	35
Figura 9: Filtrar Filas	36
Figura 10: Modificando denominación	36
Figura 11: Añadir constantes.....	37
Figura 12: Añadir secuencia	37
Figura 13: Insertar concepto.....	38
Figura 14: Block this step	39
Figura 15: Insertar/Actualizar estructura	39
Figura 16: Resultados obtenidos luego de la migración con la herramienta	41
Figura 17: Estado de los datos en SIPAC 2.4.....	42
Figura 18: Estado de los datos en SIPAC 3.0.....	43

ÍNDICE DE TABLAS

Tabla 1: Resumen de las ventajas y riesgos de las estrategias de migración.....	7
Tabla 2: Tabla comparativa entre las metodologías seleccionadas	15
Tabla 3: Tabla comparativa entre Talend y Pentaho	20
Tabla 4: Fragmento de los RF de SIPAC 2.4.....	22
Tabla 5: Mapa lógico de datos para la solución propuesta	30

INTRODUCCIÓN

En la actualidad, la mayoría de las instituciones, empresas y gobiernos cuentan con sistemas de cómputo que permiten manejar las diferentes áreas organizacionales según la misión en particular en cada caso, por ejemplo: personal, ventas, planificación, mercadotecnia, publicidad y contabilidad. Es por ello que, a través del tiempo, la informática se ha convertido en una herramienta indispensable para el ser humano en todas sus actividades cotidianas. A esto se suma la contribución de las Tecnologías de la Información y las Comunicaciones (TIC), las cuales han evolucionado y están cambiando el modo tradicional de realizar las actividades, con el objetivo de contribuir al tratamiento de la información en las instituciones y empresas.

Para estos sistemas informáticos, la información es un activo esencial para cualquier organización y el potencial de su valor reside en los datos que, en ocasiones, deben migrar para mejorar el desempeño de una base de datos, actualizar versiones, reducir costes o aplicar políticas de seguridad (Data, 2016). Teniendo en cuenta que este activo es tan necesario para cualquier organización, es de suma importancia conservar los volúmenes de información.

Es necesario para cualquier empresa tener el control de los datos y el acceso total a estos cuando se realizan actualizaciones de versiones, cambios de tecnologías o cualquier proceso que provoque la pérdida de credibilidad de la conservación de los datos. Para evitar esta pérdida se debe realizar el correcto y necesario ciclo de vida de la migración de datos, que no es más que el proceso de transferencia de datos entre sistemas (Data, 2016). Este proceso se desarrolla generalmente mediante la implementación de una serie de fases, que parte desde la etapa de descubrimiento, perfilado, limpieza, extracción y conversión de los datos, para finalizar con la carga de los datos transformados en el nuevo destino. A esta última etapa igual se le conoce como carga inicial de los datos. Tras una fusión o migración a un nuevo software, a menudo las aplicaciones y sistemas se abandonan, pero los datos que contienen deben preservarse. Precisamente, transferir los datos existentes al nuevo entorno es el fin que persigue la migración de datos (Data, 2016).

En la Universidad de Ciencias Informáticas (UCI), se desarrolla el proceso de actualización del Sistema de Planificación de Actividades (SIPAC). Dicho sistema se basa en la Instrucción no.1 del año 2011 de los Consejos de Estado y de Ministros en el momento de salir dicha instrucción, y forma parte del paquete de soluciones integrales de gestión Xedro para las entidades presupuestadas y empresariales. Gobernado por los principios de la independencia tecnológica, con funcionalidades

generales de los procesos y las particularidades de la economía cubana. Sus características funcionales fueron diseñadas de acuerdo con las políticas emitidas por los organismos rectores del Estado Cubano.

Todas las versiones desarrolladas hasta el día de hoy de dicho sistema, tienen en común diferentes aspectos, entre los que se puede mencionar los módulos encargados de generar las configuraciones necesarias que otorgan al sistema y al cliente una simulación de los procesos de organización del personal. Además, cuentan con niveles de subordinación necesarios e indispensables para efectuar una planificación de actividades basadas en reglas de la compartimentación de la información. Por su parte permiten que la información planificada sea accedida por la persona autorizada, en el momento indicado.

El SIPAC en su versión 2.4 se encuentra desplegado en varias entidades del país. Con la construcción de una nueva versión 3.0 de SIPAC surge la necesidad de una estrategia para la carga inicial y para la migración total de los datos para la obtención de estos sin una pérdida total de la información.

Entre la versión 2.4 y 3.0 del SIPAC, existe una relación conceptualmente, pero una inconsistencia entre los datos estructuralmente en las bases de datos de ambos sistemas. Los datos difieren en la estructura entre versiones generando problemas como ruido en los datos, datos atípicos, pérdida y nuevos valores. Estos datos discrepan en composición de campos, campos que fueron simplificados, agregados y eliminados en la nueva versión provocando dichos problemas de inconsistencia e incompatibilidad que generan problemas en la integridad de los datos. Haciendo que las bases de datos de ambos sistemas no logren compartir la información a pesar de que tienen conceptualmente el mismo funcionamiento.

De no existir estas dificultades los datos llegarían a ser utilizados en más de un sistema o en la construcción de otros, permitiendo así la reutilización, evitando que discrepen en su composición y el riesgo de perder información de los mismos.

Ante la situación previamente planteada, se define como **problema a resolver**: ¿Cómo realizar la carga inicial y migración total de los datos de la planificación de la versión 2.4 a la versión 3.0 de SIPAC, minimizando la discrepancia entre los datos y el riesgo de pérdida de información?

La investigación tiene como **objeto de estudio** el proceso de la migración de los datos y queda enmarcado como **campo de acción** la carga inicial y la recuperación total de los datos.

Para darle solución al problema se define como **objetivo general** desarrollar una estrategia para la carga inicial y migración total de los datos del SIPAC 2.4 al SIPAC 3.0, de forma tal que se disminuya la discrepancia entre los datos y el riesgo de pérdida de información de los mismos en dichos procesos.

Las **tareas de la investigación** planteadas son:

1. Establecimiento de los referentes teóricos en los que se sustenta la propuesta de solución sobre la carga inicial y la migración de los datos en sistemas informáticos.
2. Seleccionar los datos a tener en cuenta para migrar desde la versión 2.4 de SIPAC a SIPAC 3.0.
3. Estudio y selección de la metodología de desarrollo a utilizar.
4. Selección de las tecnologías, herramientas y estándares que se necesitan para implementar la propuesta de solución.
5. Elaboración de la descripción del proceso de Extracción, Transformación y Carga de los datos a tener en cuenta para migrar del sistema origen SIPAC 2.4 a la versión 3.0.

Por lo tanto, se plantea la siguiente **idea a defender**: si se desarrolla una estrategia para la carga inicial y migración total de los datos del SIPAC 2.4 al SIPAC 3.0, se podrá minimizar la discrepancia entre los datos y el riesgo de pérdida de información de los mismos en dichos procesos.

Como **posibles resultados** se espera alcanzar una estrategia que contemple la carga inicial y migración total de los datos de SIPAC 2.4 a SIPAC 3.0.

Para validar metodológicamente la investigación se utilizan los métodos de investigación, divididos en métodos teóricos y empíricos, los cuales se exponen a continuación:

Métodos teóricos:

- **Analítico-sintético:** este método se utilizó para facilitar el estudio acerca de la migración de los datos, en el cual se analizaron las diferentes bibliografías consultadas y se sintetizaron las partes previamente analizadas para descubrir sus características generales y las relaciones esenciales entre ellas. De igual forma se utilizó para sintetizar los elementos necesarios para la selección de la metodología y tecnologías adecuadas para el desarrollo de la migración de datos.

- **Modelación:** se utilizó para poder abstraerse del problema social y establecer la estrategia para la migración de los datos, haciendo una representación abstracta de la solución, facilitando así el desarrollo de la misma.

Métodos empíricos:

Análisis documental: este método se utilizó para analizar todos los documentos de referencias, de acuerdo a la carga inicial y migración de los datos, para así tener mayor esclarecimiento sobre los principales conceptos que sustentan la investigación y así seleccionar los más adecuados para la misma.

DESARROLLO

Introducción

La migración de base de datos es un proceso complejo que cuenta con sus determinadas fases y sobre todo requiere tiempo. Este tiempo se define dependiendo del volumen de la base de datos del sistema que se desea migrar. Cuando se realiza un cambio, una modificación o actualización de una base de datos, los datos necesitan ser preservados en el nuevo sistema, por tal motivo deben ser transformado al formato correspondiente para el nuevo sistema y este a su vez debe ser capaz de preservarlo (Tarazona, 2017). Teniendo en cuenta lo anteriormente planteado, la investigación parte del estudio de varios conceptos asociados al objeto de estudio, así como un conjunto de aspectos teóricos-técnicos para lograr que la carga inicial y la migración total del SIPAC en su versión 2.4 a la versión 3.0 se realice de forma satisfactoria.

Conceptos asociados a la investigación

Con el objetivo de lograr un entendimiento del objeto de estudio de la presente investigación, se requiere el conocimiento previo de los conceptos enunciados a continuación:

Migración de los datos:

Según (Angeles Cruz Manjarrez Antaño, 2014), la migración de datos consiste en la transferencia de materiales digitales de un origen de datos a otro. Se trata de una consideración clave para cualquier implementación, actualización o consolidación de un sistema informático.

Por otra parte, (Data, 2016) plantea que es el proceso para transferir los datos de un sistema a otro, que puede ser por cambios en el sistema de almacenamiento donde se encuentran los datos mientras se practican las modificaciones necesarias en la base de datos o la aplicación que los gestiona.

Por su parte (Tarazona, 2017) define la migración de bases de datos como el cambio de datos de un sistema a otro sistema, proceso por el cual grandes volúmenes de bases de datos son trasladados desde un sistema existente hacia un sistema nuevo.

Una vez analizadas las definiciones anteriormente descritas, el autor asume la migración de los datos como el traslado entre sistemas de la información, con el objetivo de conservarlos durante actualizaciones, reparaciones o mantenimiento de aplicaciones o sistemas, que dependen o utilizan estos datos.

Calidad de los datos:

Para el caso de este término, el autor asume lo establecido por la norma ISO/IEC 25023, donde se plantea la calidad de los datos como el "Grado en que las características de los datos satisfacen necesidades implícitas y establecidas cuando son usados en condiciones específicas". Lo anterior, da un papel relevante a la participación del usuario a la hora de definir si un conjunto de datos es de calidad. La norma define como característica la recuperabilidad que se tendrá en cuenta a lo largo del proceso de migración de datos (ISO/IEC, 2016).

La calidad de los datos es más que una percepción. No existen estandarizaciones y si bien no es un requisito de la migración de datos, es aconsejable realizar un proceso de calidad de datos antes de migrar los datos (Data, 2016).

Integración de los datos:

Sánchez en el 2017 plantea que la integración de datos es el proceso de combinar datos de diferentes fuentes en una sola aplicación, unificándolos y centralizándolos para facilitar la recuperación, inspección y análisis de estos datos (SANCHEZ, 2017).

Por su parte, (ESPINOZA, 2018) asocia la integración de datos al proceso que permite combinar datos heterogéneos de muchas fuentes diferentes en la forma y estructura de una única aplicación, facilitando que diferentes tipos de datos sean fusionados por usuarios, organizaciones y aplicaciones.

Teniendo en cuenta las definiciones descritas, el autor de la presente investigación asume la definición por (SANCHEZ, 2017), que asocia la integración de los datos al proceso de combinar datos de diferentes fuentes en una sola aplicación, unificándolos y centralizándolos para facilitar la recuperación, inspección y análisis de estos. Agregando que la misma se realiza con el objetivo de eliminar inconsistencias, errores y duplicaciones que puedan existir centralizando los datos para el análisis en busca de información relevante y que permita la recuperación, en este caso, total de los datos.

Planificación:

Según (Franklin, 1987) es el proceso de seleccionar información y hacer suposiciones respecto al futuro para formular las actividades necesarias para realizar objetivos organizacionales.

Por su parte (Stoner, 1996), define este término como una forma concreta de la toma de decisiones, estableciendo metas y eligiendo medios para alcanzar estas metas. Es un proceso continuo que refleja los cambios de la organización, adaptándose a estos.

Al referirse a este término (Borjas, 2018), asocia la planificación empresarial a la conjunción de distintos tipos de planes, reforzados en la idea de su necesaria coherencia e integración, que coordina e integra los aspectos productivos con el máximo de eficiencia y la activa participación de los trabajadores.

Una vez establecidas estas definiciones, el autor de la presente investigación asume la planificación como un sistema de trabajo en el cual se conciben las actividades necesarias para cumplir con los objetivos organizacionales, lo que facilita la organización del mismo, así como el funcionamiento y control de la productividad en las empresas e instituciones.

Tendencias actuales de la migración de datos

Entre las estrategias de datos que existen se analizarán tres de las más conocidas según (Tarazona, 2017):

- ✓ **Ejecución en paralelo:** esta estrategia se basa en validar por un tiempo estipulado los resultados que tiene el nuevo sistema comparado con el anterior, corriendo en forma paralela los dos sistemas (Tarazona, 2017).
- ✓ **Migración Incremental:** en esta estrategia el nuevo sistema se activa en forma incremental de acuerdo a como se va llevando a cabo las migraciones. No se debe esperar que todo el nuevo sistema esté listo para implementar su funcionamiento (Tarazona, 2017).
- ✓ **Big Bang:** esta estrategia se basa en seguir utilizando el sistema actual mientras se va implementado la migración al nuevo sistema durante todo el proyecto planteado, de modo tal que, hasta que todos los datos no estén montados en el nuevo sistema, el sistema de base de datos anterior no se desactiva (Tarazona, 2017).

A continuación, se realiza un resumen de las ventajas y riesgos de estas estrategias:

Tabla 1: Resumen de las ventajas y riesgos de las estrategias de migración

Estrategias de migración	Ventajas	Riesgos
Ejecución en paralelo	Garantiza el funcionamiento del	El costo de tener dos sistemas en

	nuevo sistema en un entorno real. Se comprueban los datos en el nuevo sistema mientras se ejecuta la migración garantizando el control sobre los datos.	funcionamiento. El surgimiento de nuevos requisitos durante la migración complicaría el proceso, provocando replantear la estrategia.
Incremental	Garantiza que no exista tiempo de inactividad del sistema. Define fases para la organización de la migración.	El no tener en cuenta que los sistemas fuente y de destino operan en paralelo afectaría la calidad de los datos y el proceso de migración.
Big Bang	Control sobre el proceso de migración. Alcance bien definido. Es una estrategia de rapidez, para pocos períodos de tiempo.	Debido a la rapidez de la ejecución, está en riesgo la calidad de los datos. Es necesario planear una secuencia de ejecución previamente.

Debido a las dependencias funcionales que existe entre los datos y el volumen de estos, se decidió por una estrategia de tipo incremental, para evitar el colapso de los datos en el proceso y migrar por partes todos los datos de SIPAC 2.4. Por otra parte, se propone tener en cuenta una ejecución en paralelo para la etapa de pruebas, comprobándose el estado de los datos según las respuestas de SIPAC 2.4.

Proceso de extracción, transformación y carga

✓ Extracción de los datos

El proceso de extracción consiste en obtener los datos del sistema origen (Data, 2016). El objetivo de esta fase es extraer solamente los datos que serán útiles para responder preguntas que se plantean inicialmente reduciendo los datos y facilitando su entendimiento (Jhoan Esteban, 2018). Esta fase requiere una verificación de los formatos de origen en los que están almacenados los datos, ya que estos pueden ser distintos entre sí (Borjas, 2018). Los datos pueden ser extraídos en su totalidad mediante el barrido de las tablas por completo, procesando por lotes únicamente lo que fue modificado o agregado, o extraer los datos a medida que se produce una actualización (Borjas,

2018). En la migración de datos el proceso de extracción de los datos varía su complejidad en dependencia de los orígenes de la información que se quiere obtener teniendo en cuenta las dependencias y relaciones entre estos.

✓ **Transformación de los datos**

En esta fase se aplica una serie de reglas de negocio o funciones sobre los datos extraídos para convertirlos en datos que serán cargados (Angelino Feliciano, René Cuevas, José Martínez, 2016). Los datos necesitan ser procesados, limpiados y adaptados para lograr compatibilidad con el nuevo sistema teniendo en cuenta las dependencias del sistema. Se incluyen operaciones básicas sobre los datos, como el filtrado para reducir ruidos y decidir qué hacer con los datos faltantes (Reyes Saldaña, García Flores 2005). Además, se le aplican una serie de reglas y transformaciones a los datos para que estos se acoplen al nuevo sistema. Johan Esteban plantea como procedimientos de limpieza la división de datos en campos más sencillos y unitarios, la corrección de errores sintácticos y buscar la relación entre los datos para simplificarlo (Jhoan Esteban, 2018).

Este paso de filtrado debe determinar que coincidencias existen entre los campos origen-destino, y en cada caso aplicar la transformación necesaria para cargar los datos en el destino. Es en esta etapa donde se dará solución a la incompatibilidad e inconsistencia de los datos garantizando que estos sean recuperados en su totalidad en la nueva versión.

✓ **Carga de los datos**

La carga de datos consiste en la incorporación de estos en el destino con el formato adecuado (Johan Esteban, 2018). Luego de terminar el procesamiento y la transformación de los datos estarán listos y adaptados para el destino planteado, monitoreando el correcto estado y calidad de los datos.

Una vez resuelto los problemas de inconsistencia e incompatibilidad de los datos y campos de ambas bases de datos, se realiza un chequeo de datos comprobando que la información es consistente, fiable e íntegra. Se mantiene un control estricto sobre los datos resultantes para garantizar una total recuperación con el objetivo de confirmar que toda la información está integrada y cargada con éxito en el destino.

Migración de los datos en sistemas informáticos

El proceso de migración requiere de etapas para llevarse a cabo como, por ejemplo: la observación de la fuente de datos y el destino de los mismos, así como los formatos en los que se presenta la información para identificar las conversiones para adecuarse a las nuevas necesidades (Angeles Cruz Manjarrez Antaño, 2014). Es necesario saber si se realizará una migración total o parcial del sistema para adaptar la migración al grupo de datos deseados.

Las migraciones parciales tienen como características enfocarse en datos específicos, pueden ser manejadas con sentencias SQL ya que está destinada a algunos datos. En el caso de la migración total se busca el traslado de todos los datos, con el fin de conservarlos en su totalidad. El uso de herramientas destinadas para este tipo de procesos, resulta más efectivo por la gran carga de datos que presenta.

Tomando como punto de partida lo definido anteriormente al proponer una migración híbrida entre el enfoque incremental y en paralelo, el autor define que se realizará una migración total de los datos, realizando una extracción con enfoque incremental y una carga en paralelo como referente, garantizando la migración y conservación total de los datos.

Una vez esclarecidos los referentes teóricos descritos anteriormente, es necesario estudiar un conjunto de metodologías para llevar a cabo la migración de bases de datos. A continuación, se describirán un conjunto de metodologías de almacenes de datos. Este estudio se basa partiendo de que uno de los procesos para realizar un almacén de datos, es la migración de datos. Una vez seleccionada la metodología a utilizar, se delimitará hasta qué etapa o fase de la misma se tendrá en cuenta en la propuesta de solución, partiendo de las especificaciones que se persigue con la presente investigación.

Metodología

Las metodologías de desarrollo de software son *“un conjunto de procedimientos, herramientas y un soporte documental que ayuda a los desarrolladores a realizar un nuevo software”* (Pacienza, 2015).

En el desarrollo de software la necesidad de organizar o estructurar de forma correcta y disciplinada, es uno de los factores más importantes para un mayor aprovechamiento del tiempo y los recursos. Por tal razón, es preciso definir una estrategia que ordene las posibles tareas a desarrollar, así como una guía de cómo efectuar las actividades, procedimientos y pasos a tener en cuenta en el desarrollo de un software.

Para el caso de la presente investigación, se tendrá en cuenta las metodologías que guían el desarrollo de almacenes de datos. Esta selección se basa debido a que una de sus principales actividades es realizar una migración de datos, siendo esto el objetivo de la presente investigación. Por tal motivo no es necesario realizar todas las fases del ciclo de vida de estas metodologías, por lo que abarcar las fases hasta la integración de datos, es suficiente para darle cumplimiento al objetivo trazado en la presente investigación.

En este sentido, muchos profesionales se guían por metodologías basadas en las corrientes de Kimball y/o Inmon, así como muchos otros diseñan sus propias metodologías basadas en alguna de las corrientes mencionadas o que sean afines a la tecnología seleccionada. (Gravitar, 2010). A continuación, se describen las principales características de las metodologías estudiadas.

✓ Metodología Kimball

Esta metodología apunta a una solución completa que se puede implementar en poco tiempo. Esto cumple con lo que se necesita en una metodología de inteligencia de negocios, dar resultados rápidos para demostrar el valor de la solución al negocio y no perder la confianza de los usuarios. Se enfoca principalmente en el diseño de la base de datos que almacenará la información para la toma de decisiones. El ciclo de vida de Kimball está basado en cuatro principios básicos, los cuales se definen a continuación (Juhasz, 2014):

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

En la siguiente figura se muestran las diferentes fases establecidas por Ralph Kimball, que han sido diseñadas para que puedan ser desarrolladas en paralelo o en forma secuencial durante el proceso de creación de almacenes de datos (Geovanny Euclides Silva Peñafiel, 2019).

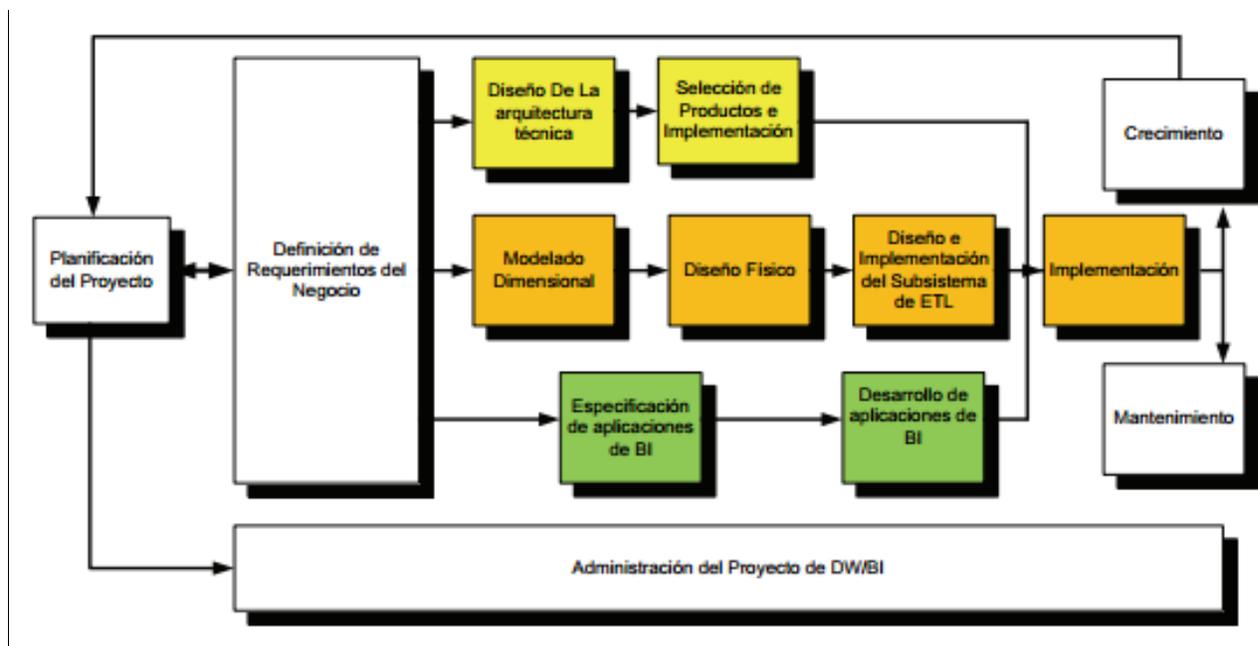


Figura 1: Etapas de la metodología Kimball

Fuente: (Hernández, 2013)

✓ Metodología Hefesto

Su idea está fundamentada en una amplia investigación, comparación de metodologías existentes y experiencias propias en procesos de confección de almacenes de datos. Para trabajar con la metodología Hefesto se comienza recolectando las necesidades de información de los usuarios y se obtienen las preguntas claves del negocio. Luego, se deben identificar los indicadores resultantes de los interrogativos y sus respectivas perspectivas de análisis, mediante las cuales se construirá el modelo conceptual de datos del almacén de datos. Después, se analizarán los Procesamientos de Transacciones en Línea (OLTP por sus siglas en inglés de *On Line Transaction Processing*) para señalar las correspondencias con los datos fuentes y seleccionar los campos de estudio de cada perspectiva. Una vez hecho esto, se pasará a la construcción del modelo lógico del depósito, precisándose las jerarquías que intervendrán. Por último, se definirán los procesos de carga, transformación, extracción y limpieza de los datos fuente (Dario, 2010). A continuación, se muestran las etapas de esta metodología.



Figura 2: Etapas de la metodología Hefesto

Fuente: (Ismael, y otros, 2013)

✓ Metodología de desarrollo de proyectos de almacenes de datos

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

- ✓ Un ciclo de vida para el desarrollo de proyectos de almacenes de datos, claramente definido a partir de la descripción de las fases y flujos de trabajo que presenta.
- ✓ Un conjunto de roles, artefactos y herramientas adaptados a las características de los proyectos de almacenes de datos y las necesidades de gestión del proyecto que posibilitan la ejecución adecuada del proceso de desarrollo.
- ✓ Una descripción de los procesos y actividades que se realizan a lo largo de todo el ciclo de vida de la metodología relacionándolos con los roles, artefactos y herramientas necesarios para su ejecución según las características de este tipo de soluciones.

El ciclo de vida de la metodología se divide en siete fases y un flujo de trabajo (Hernández, 2013).

Flujo de Trabajo:

1. Gestión de Proyecto:

- Se ejecuta a lo largo de todo el ciclo de vida del proyecto.
- Lo componen procesos que mantienen la adecuada gestión del proyecto.
- Solo se incluye en los procesos que se ejecutan a lo largo de todo el desarrollo y los que están asociados únicamente a la gestión del proyecto.

Fases:

1. Estudio Preliminar y Planeación

- Planear la planeación.
- Planificar estudio preliminar.
- Realizar un diagnóstico integral de la organización.
- Realizar análisis de factibilidad.

2. Requisitos

- Realizar el levantamiento.
- Definición y especificación.

3. Arquitectura

- Definir la arquitectura de la solución de DW.
- 4. Diseño e implementación
 - Diseño e Implementación del Subsistema de almacenamiento.
 - Diseño e Implementación del Subsistema de integración de datos.
 - Diseño e Implementación del Subsistema de visualización de la información.
- 5. Prueba
 - Realizar las pruebas al Sistema
- 6. Despliegue
 - Desplegar el sistema.
 - Capacitación y transferencia de tecnológica.
- 7. Soporte y Mantenimiento
 - Operaciones de mantenimiento y soporte.

A continuación, se muestra el ciclo de vida de dicha metodología:

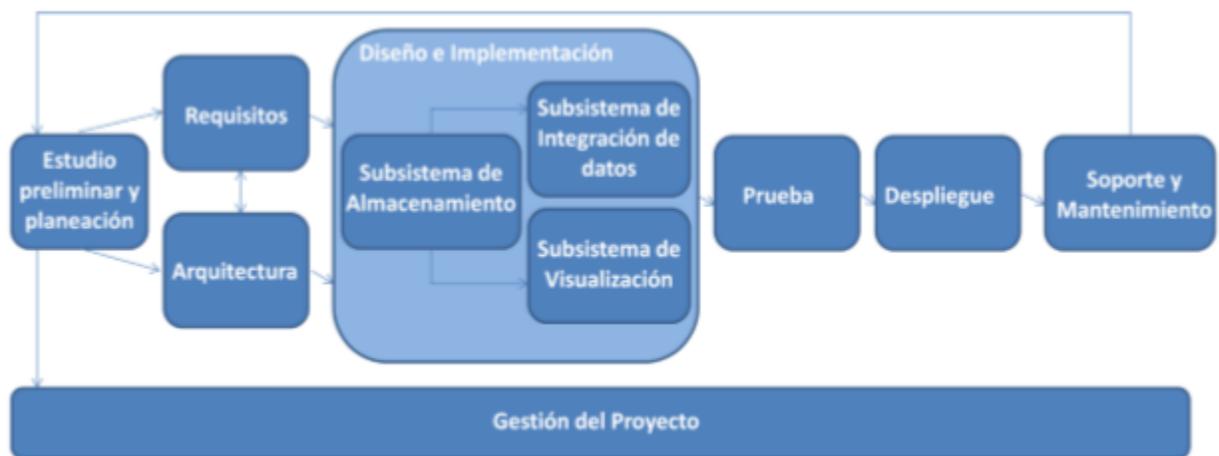


Figura 3: Ciclo de vida de la metodología de almacenes de datos

Fuente: (Hernández, 2013)

Una vez realizado el estudio sobre las diferentes metodologías a tener en cuenta en la presente investigación, se decide realizar una comparación entre las mismas teniendo en cuenta los siguientes criterios de selección:

- ✓ Ciclo de vida
- ✓ Nivel de entendimiento para principiantes
- ✓ Documentación

En la tabla que se muestra a continuación se resume como influye cada uno de estos criterios en cada una de las metodologías estudiadas.

Tabla 2: Tabla comparativa entre las metodologías seleccionadas

Criterios	Kimball	Hefestos	Metodología de desarrollo de proyectos de almacenes de datos
Ciclo de vida	Deja claro qué se debe hacer, pero no cómo hacerlo. Es rápido y plantea la división de los procesos, permitiendo su realización en paralelo o secuencial, por lo que es muy flexible a la hora de definir la ruta a seguir.	Menciona puntualmente los pasos a seguir en cada fase, pero no cómo realizarlos. Su diseño plantea el proceso ETL como último paso a realizar, lo que dificulta la migración de datos.	Está claramente definido a partir de la descripción de las fases y flujos de trabajo que presenta.
Nivel de entendimiento para principiantes	Es complejo por la falta de detalle para seguir su ciclo de vida.	Sencillo, ya que describe paso a paso lo que hay que hacer en cada fase de su ciclo de vida.	Sencillo, ya que describe todo los pasos de su ciclo de vida y establece roles, artefactos y herramientas y los relaciona con las actividades y los procesos.
Documentación	Abundante y precisa.	Abundante y precisa.	Abundante y precisa.

Como resultado de la comparación realizada, el autor de la presente investigación selecciona como metodología a utilizar en la estrategia para la carga inicial y migración total de los datos del SIPAC en su versión 2.4 a la versión 3.0, la Metodología de Desarrollo para Proyectos de Almacenes de Datos. Dicha decisión se tomó con el objetivo de aprovechar el ciclo de vida claramente definido que esta presenta, la claridad de los artefactos que genera, así como la facilidad de adaptarse a cualquier proyecto de almacenes de datos. Además, existe experiencia y dominio de esta metodología en la universidad, específicamente en el centro al que pertenece la presente investigación. Esto trae como principal ventaja la celeridad en el proceso, ya que no es necesario capacitar a los desarrolladores sobre el proceder con dicha metodología.

Teniendo en cuenta lo definido por el autor de la presente investigación al inicio de este epígrafe, se establece llevar a cabo exclusivamente las siguientes fases de dicha metodología:

- Estudio preliminar y planeación.
- Requisitos.
- Diseño e implementación (subsistema de integración de datos).
- Pruebas

Una vez seleccionada la metodología que guiará el proceso de desarrollo en la presente investigación, a continuación, se describen las herramientas y tecnologías a utilizar para la solución.

Herramientas y tecnologías analizadas para la solución

Contar con las herramientas y tecnologías apropiadas y que estas se adapten a las necesidades, resulta imprescindible para las empresas productoras de software, ya sea para consumo interno, o para la venta y distribución de los mismos. Es por ello que a continuación se describen las herramientas y tecnologías a utilizar de forma tal que se adapten a las necesidades de la solución propuesta.

Herramientas de modelado

Las herramientas de modelado o de Ingeniería de Software Asistida por Computadora (CASE por sus siglas del inglés *Computer Aided Software Engineering*) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo de las mismas en términos de tiempo y dinero. Estas herramientas, pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software, en tareas como el proceso de realizar un diseño del proyecto (Flores, 2013).

Teniendo en cuenta la herramienta CASE que utiliza el centro al que pertenece la presente investigación, el autor del presente trabajo decide utilizar Visual Paradigm en su versión 8.0. A continuación se describen sus principales características:

✓ **Visual Paradigm**

Esta es una herramienta de Lenguaje Unificado de Modelado (UML por sus siglas en inglés *Unified Modeling Language*) que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Ayuda a una rápida construcción de aplicaciones con calidad y a un menor costo. Permite crear todos los tipos de diagramas de clases, ingeniería inversa, generar código desde diagramas y generar documentación (Paradigm, 2017).

Entre sus principales características se encuentran (Gaines, y otros, 2017):

- Permite la alineación automática para dibujos limpios y nítidos.
- Ofrece amplias características de modelado de casos de uso incluyendo la función completa de UML, diagrama de casos de uso y editor de flujo de eventos.
- Permite diseñar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Proporciona abundantes tutoriales del Lenguaje de Modelado, demostraciones interactivas de UML y proyectos UML.
- Se integra con herramientas Java, como son: Eclipse/IBM, NetBeansIDE, entre otras.

Lenguaje de modelado

El UML fue creado para forjar un lenguaje de modelado visual común, semántica y sintácticamente rico para la arquitectura, el diseño y la implementación de sistemas de software complejos, tanto en estructura como en comportamiento. Consiste en diferentes tipos de diagramas que describen los límites, la estructura y el comportamiento del sistema y los objetos que contiene. El UML se perfecciona continuamente de forma tal que la facilidad de uso, la implementación y la adaptación se pueden realizar de manera sencilla. Algunas de las actualizaciones de los diagramas UML son: mayor integración entre modelos estructurales y de comportamiento, capacidad de definir jerarquía y desglosar un sistema de software en componentes y subcomponentes y UML 2.0 eleva el número de diagramas de 9 a 13 (Lucid, 2018).

El análisis de este lenguaje se realiza debido a que UML es la notación utilizada por la herramienta CASE que se emplea para la propuesta de solución.

Sistema gestor de bases de datos

Un Sistema Gestor de Base de Datos (SGBD) es un sistema que permite la creación, gestión y administración de bases de datos, así como la elección y manejo de las estructuras necesarias para el almacenamiento y búsqueda de información del modo más eficiente posible (Marin, 2019). Para la gestión de bases de datos en la propuesta de solución se decide utilizar PostgreSQL en su versión 10.5, que a continuación se describe:

✓ **PostgreSQL**

Es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia de Distribución de software de Berkeley (BSD por su nombre en inglés *Berkeley Software Distribution*) y con su código fuente disponible libremente. PostgreSQL utiliza un modelo cliente/servidor y usa

multiprocesos en vez de multihilos¹ para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando. Su desarrollo comenzó hace más de 16 años, y durante este tiempo, estabilidad, potencia, robustez, facilidad de administración e implementación de estándares han sido las características que más se han tenido en cuenta durante su desarrollo. PostgreSQL funciona muy bien con grandes cantidades de datos y una alta concurrencia de usuarios accediendo a la vez al sistema (Martínez, 2010). Entre sus principales características se encuentran (Ibarra, 2008):

- Alta concurrencia: mediante un sistema denominado MVCC, (acceso concurrente multiversión, por sus siglas en inglés), PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos.
- Arrays: los usuarios pueden crear sus propios tipos de datos los cuales pueden ser completamente indexables gracias a la infraestructura de PostgreSQL.

Herramienta de gestión de bases de datos

Las herramientas de gestión de bases de datos proporcionan una interfaz que permite administrar las bases de datos. Estas herramientas también permiten ejecutar consultas SQL² desde este interfaz de usuario (Martínez, 2010). Para la propuesta de solución se decide utilizar la herramienta establecida por el centro, PgAdmin en su versión v4.23, que a continuación se describe:

✓ **PgAdmin**

Es una aplicación de diseño y manejo de bases de datos para su uso con PostgreSQL. Se puede utilizar para manejar PostgreSQL 7.3 y superiores, funciona sobre casi todas las plataformas. Este software fue diseñado para responder a diversas necesidades, desde la escritura de simples consultas SQL (*Structured Query Language*) a la elaboración de bases de datos complejas. La interfaz gráfica es compatible con todas las características de PostgreSQL (Page, y otros, 2018).

Herramientas para la Extracción, Transformación y Carga

Existen herramientas que facilitan el proceso de migración de datos y en la actualidad se encuentran cada vez más involucradas en este tipo de procesos. Las herramientas de integración permiten a las empresas conseguir una mayor eficiencia, productividad y agilidad ya que conecta diferentes aplicaciones de software y servicios. Además, relaciona interfaces de programación de aplicaciones,

¹En programación, nos estamos refiriendo a los lenguajes de programación que permiten la ejecución de varias tareas en forma simultánea.

²Lenguaje Estructurado de Consultas (SQL siglas en inglés de *Structured Query Language*)

datos y dispositivos para automatizar los procesos empresariales y ofrecer al usuario información exacta y a tiempo (Navarro, 2018). Algunas de estas herramientas están respaldadas por organizaciones que tienen un claro modelo de negocio como Pentaho, Talend, Oracle, que a continuación se describen.

Pentaho Data Integration (PDI): cuyo nombre en clave es *Kettle*, es una de las herramientas o componentes de *Pentaho Suite* que permite que se utilicen técnicas ETL, es decir, poder implementar procesos de extracción, transformación y carga de datos. *Kettle* ofrece, además, datos analíticos muy precisos, eliminando las complejidades involucradas en la codificación al proporcionar bibliotecas en profundidad para el mismo (Martín, 2019).

Características principales:

- ✓ Tiene una interfaz visual con indicadores de las transformaciones.
- ✓ Es una aplicación escrita en Java con algunas características avanzadas escritas en Java Script.
- ✓ Basado en metadatos.
- ✓ Como soporte se encuentran los foros de Pentaho y la comunidad Pentaho.
- ✓ Con respecto a escalabilidad, soporta la arquitectura de procesamiento en paralelo para distribuir las tareas de ETL a través de múltiples servidores (González, 2011).

Talend Data Integration (TDI): Es una herramienta de código abierto hecha en Java, especializada en la integración de grandes volúmenes de datos (Big Data), que proporciona la extracción, transformación y carga de datos en la nube, gestión de datos maestros, calidad de datos, preparación de datos y la integración de aplicaciones empresariales de software y servicios (Borja, 2018).

Características principales:

- ✓ Tiene una interfaz visual implementada en Eclipse.
- ✓ Genera código en Java o Scripts en Pearl.
- ✓ Basado en Jobs o Trabajos.
- ✓ Como soporte se encuentran los foros de Talend (Bustillos, 2014).

Con respecto a escalabilidad, soporta la arquitectura de procesamiento en paralelo para distribuir las tareas de ETL a través de múltiples servidores.

Oracle Data Integration (ODI): Es una aplicación de software basada en ETL, que se utiliza para la transformación y fusión de datos o la integración de datos de alto volumen, alto rendimiento, hasta

procesos basados en eventos y servicios de datos habilitados para SOA (*Service Oriented Architecture*) mediante el agregado de paralelismo (Borja, 2018).

Características principales:

- ✓ Solamente compatible con bases de datos Oracle.
- ✓ Fácil cuando se trata de información almacenada en bases de datos Oracle, debido a las herramientas Data Pump y transportable tablespaces, pero no ofrece mucha compatibilidad a otras bases de datos (Bustillos, 2014).

Con respecto a escalabilidad, soporta la arquitectura de procesamiento en paralelo para distribuir las tareas de ETL a través de múltiples servidores.

Entre las herramientas descritas anteriormente se destacan Talend y Pentaho por ser herramientas de código abierto y tener varios años de aplicación en diferentes instituciones nacionales o internacionales. Además, son compatibles con la base de datos utilizada en el desarrollo del SIPAC. Por lo antes descrito se realizó un estudio de dichas herramientas para el desarrollo de la propuesta de solución.

Tabla 3: Tabla comparativa entre Talend y Pentaho

Aspectos	Talend	Pentaho
Facilidad de uso	Requiere previo conocimiento	Intuitiva y fácil de usar
Visualización	Se visualizan los datos mediante una consola con una funcionalidad.	Permite visualizar mediante una forma sencilla.
Velocidad	Requiere configuración específica y manual con conocimiento previo de la data a utilizar.	Requerir de Java Database Connector disminuye la velocidad de transacciones pero es más rápido que Talend.
Información	Limitada	Cuenta con blogs, guías, tutoriales, ejemplos.
Conectividad	Varias bases de datos, archivos planos, xml, Excel, servicios web, la nube, big data.	
Implementación	Lenguaje Java y compatible con cualquier plataforma java	

Se seleccionó la herramienta Kettle perteneciente a la suite de Pentaho, la cual es fácil de utilizar, presenta basta información en la red para la preparación y uso de la misma. Además, dicha herramienta funciona en distintos sistemas operativos como Windows, Unix y Linux. Ofrece una

licencia pública GPL y soporta SQL Server y PostgreSQL. Por último, posibilitará realizar una solución efectiva ya que existen experiencias previas positivas del trabajo con la herramienta en el centro al que pertenece la presente investigación.

Estrategia propuesta al problema planteado

Teniendo como referencia la metodología que guiará la propuesta de solución al problema planteado, para la carga inicial y migración total de los datos del SIPAC en su versión 2.4 a la versión 3.0, a continuación, se describe el proceder en cada una de las fases establecidas anteriormente.

Estudio preliminar y planeación:

Esta fase se compone por dos procesos: el estudio preliminar y planeación inicial del proyecto. En su parte el estudio preliminar está dividido en 3 áreas, de las que se identifica desarrollar el área de diagnóstico de los datos para el proceso de migración. Así como el análisis de los datos teniendo en cuenta aspectos como la fuente de datos, el formato en el que se encuentran almacenados, sistemas informáticos que los manejan, utilización e importancia de los mismos, entre otros aspectos (Hernández, 2013). El objetivo de esta etapa es comprender la fuente de datos mediante el estudio preliminar y realizar la planeación de cómo intervenir en esta, identificando áreas, campos, tablas o datos que son críticos para el usuario.

Se propone que en esta fase se evalúen y se definan los datos existentes que serán cargados en la nueva base de datos, con el objetivo de ir conociendo su estructura, formato y nivel de calidad. Esto implica adquirir conocimiento del área de estudio de la base de dato origen, principalmente por correlaciones campo a campo con la base de dato de destino, y la meta que se quiere obtener. Además, es necesario comprender el funcionamiento del negocio y la estructura de los datos para poder establecer las dependencias funcionales existentes entre estos en el sistema origen. Se debe tener en cuenta que en esta investigación solo existe una fuente de datos y un sistema que maneja estos datos, esto se refiere a SIPAC 2.4. Como resultado de este proceso se identificaron como objetivos:

- ✓ mod_estructuracom.nom_dominio.
- ✓ mod_seguridad.seg_usuarios.
- ✓ mod_seguridad.seg_usuarios_nom_dominio.
- ✓ mod_objetivos.dat_plan_pdo.
- ✓ mod_objetivos.dat_fip.
- ✓ mod_objetivos.dat_objetivo.

- ✓ mod_objetivos.dat_arc.
- ✓ mod_objetivos.dat_actividades.
- ✓ mod_objetivos.dat_elementos.

Requisitos

Esta fase se divide en dos procesos importantes: el levantamiento de los requisitos y el análisis de los requisitos definidos. En el levantamiento de los requisitos se identifican todos los requerimientos de la migración y luego se identifica las estructuras bases del modelo lógico dimensional. Esta fase propone con el objetivo de no realizar un estudio de toda la extensión de la fuente de datos que solo se analicen los datos identificados en el paso anterior (Hernández, 2013). Estos requisitos se agrupan en dos grupos: requisitos funcionales y no funcionales.

- ✓ Los requisitos funcionales (RF) de un sistema, son aquellos que describen cualquier actividad que este deba realizar, en otras palabras, el comportamiento o función particular de un sistema o software cuando se cumplen ciertas condiciones. Por lo general, estos deben incluir funciones desempeñadas por pantallas específicas, descripciones de los flujos de trabajo a ser desempeñados por el sistema y otros requerimientos de negocio, cumplimiento, seguridad u otra índole (Pressman, 2010).
- ✓ Los requisitos no funcionales (RnF) representan características generales y restricciones de la aplicación o sistema que se esté desarrollando. Suelen presentar dificultades en su definición dado que su conformidad o no conformidad podría ser sujeto de libre interpretación, por lo cual es recomendable acompañar su definición con criterios de aceptación que se puedan medir (Pressman, 2010).

Teniendo como base que el objetivo de esta investigación es realizar una estrategia para la carga inicial y migración total de los datos del SIPAC en su versión 2.4 a la versión 3.0, y que este en su nueva versión se ejecute satisfactoriamente, el autor de la presente investigación propone tener en cuenta todo los RF definidos por el equipo de análisis y diseño del SIPAC en su versión 2.4. En la tabla que se muestra a continuación, se describe un fragmento de estos RF:

Tabla 4: Fragmento de los RF de SIPAC 2.4

No	Nombre	Descripción
RF1.	Adicionar plan	El sistema permitirá adicionar un plan al sistema, con los atributos: <ul style="list-style-type: none"> • Denominación • Estado de cumplimiento

		<ul style="list-style-type: none"> • Categoría del plan • Porcentaje de cumplimiento • Fecha de inicio • Fecha de fin • Descripción
RF2.	Modificar plan	<p>El sistema permitirá modificar un plan previamente adicionado, con los atributos:</p> <ul style="list-style-type: none"> • Denominación • Estado de cumplimiento • Categoría del plan • Porcentaje de cumplimiento • Fecha de inicio • Fecha de fin • Descripción
RF3.	Adicionar actividad en el calendario	<p>El sistema permitirá adicionar una actividad en el calendario con los siguientes atributos:</p> <ul style="list-style-type: none"> • Denominación • Fecha de inicio • Fecha de fin • Porcentaje de cumplimiento • Hora de inicio • Hora de fin • Lugar • Categoría de la actividad • Tipo de actividad • Descripción • Involucrados • Participantes • Dirigentes • Principal • Estado de cumplimiento
RF4.	Modificar actividad en el calendario	<p>El sistema permitirá modificar una actividad en el calendario previamente adicionada, con los siguientes atributos:</p> <ul style="list-style-type: none"> • Denominación • Fecha de inicio • Fecha de fin • Porcentaje de cumplimiento • Hora de inicio • Hora de fin • Lugar

		<ul style="list-style-type: none"> • Categoría de la actividad • Tipo de actividad • Descripción • Involucrados • Participantes • Dirigentes • Principal • Estado de cumplimiento
RF5.	Adicionar objetivo	<p>El sistema permitirá adicionar un objetivo con los siguientes datos:</p> <ul style="list-style-type: none"> • Denominación • Fecha de inicio • Fecha de fin • Descripción • Estado de cumplimiento • Porcentaje de cumplimiento • Involucrados • Participantes • Dirigentes
RF6.	Modificar objetivo	<p>El sistema permitirá modificar un objetivo previamente adicionado, con los siguientes datos:</p> <ul style="list-style-type: none"> • Denominación • Fecha de inicio • Fecha de fin • Descripción • Estado de cumplimiento • Porcentaje de cumplimiento • Involucrados • Participantes • Dirigentes
RF7.	Adicionar actividad	<p>El sistema permitirá adicionar una actividad, con los siguientes atributos:</p> <ul style="list-style-type: none"> • Denominación • Fecha de inicio • Fecha de fin • Categoría de la actividad • Hora de inicio • Hora de fin • Lugar • Estado de cumplimiento • Porcentaje de cumplimiento

		<ul style="list-style-type: none"> • Tipo de actividad • Descripción • Involucrados • Participantes • Dirigentes • Principal • Extraplan
RF8.	Modificar actividad	<p>El sistema permitirá modificar una actividad previamente adicionada, con los siguientes atributos:</p> <ul style="list-style-type: none"> • Denominación • Fecha de inicio • Fecha de fin • Categoría de la actividad • Hora de inicio • Hora de fin • Lugar • Estado de cumplimiento • Porcentaje de cumplimiento • Tipo de actividad • Descripción • Involucrados • Participantes • Dirigentes • Principal • Extraplan
RF9.	Adicionar área de resultado clave	<p>El sistema permitirá adicionar una nueva área de resultados clave con los siguientes atributos:</p> <ul style="list-style-type: none"> • Denominación • Fecha de inicio • Fecha de fin • Descripción • Involucrados • Capítulos
RF10.	Modificar área de resultado clave	<p>El sistema permitirá modificar un área de resultados clave previamente adicionada, con los siguientes atributos:</p> <ul style="list-style-type: none"> • Denominación • Fecha de inicio • Fecha de fin • Descripción • Involucrados

		<ul style="list-style-type: none">• Capítulos
RF11.	Adicionar criterio de medida	El sistema permitirá adicionar un criterio de medida, introduciendo los siguientes datos: <ul style="list-style-type: none">• Denominación• Fecha de inicio• Fecha de fin• Unidad de medida• Descripción• Porcentaje de cumplimiento• Estado de cumplimiento• Valor meta• Valor real• Involucrados• Participantes
RF12.	Modificar criterio de medida	El sistema permitirá modificar un criterio de medida previamente adicionado, con los siguientes datos: <ul style="list-style-type: none">• Denominación• Fecha de inicio• Fecha de fin• Unidad de medida• Descripción• Porcentaje de cumplimiento• Estado de cumplimiento• Valor meta• Valor real• Involucrados• Participantes
RF13.	Adicionar categoría del plan	El sistema permitirá adicionar una nueva categoría del plan con los siguientes atributos: <ul style="list-style-type: none">• Denominación• Descripción• Rango de publicidad• Fecha de inicio• Fecha de fin
RF14.	Modificar categoría del plan	El sistema permitirá modificar una categoría del plan previamente adicionada, con los siguientes atributos: <ul style="list-style-type: none">• Denominación

		<ul style="list-style-type: none"> • Descripción • Rango de publicidad • Fecha de inicio • Fecha de fin
RF15.	Adicionar actividad hija	<p>El sistema permitirá adicionar una actividad hija, con los siguientes atributos:</p> <ul style="list-style-type: none"> • Denominación • Fecha de inicio • Fecha de fin • Categoría de la actividad hija • Hora de inicio • Hora de fin • Lugar • Estado de cumplimiento • Porcentaje de cumplimiento • Tipo de actividad hija • Descripción • Involucrados • Participantes • Dirigentes • Principal • Extraplan
RF16.	Modificar actividad hija	<p>El sistema permitirá modificar una actividad hija previamente adicionada, con los siguientes atributos:</p> <ul style="list-style-type: none"> • Denominación • Fecha de inicio • Fecha de fin • Categoría de la actividad hija • Hora de inicio • Hora de fin • Lugar • Estado de cumplimiento • Porcentaje de cumplimiento • Tipo de actividad hija • Descripción • Involucrados • Participantes • Dirigentes • Principal

		<ul style="list-style-type: none"> • Extraplan
RF17.	Adicionar objetivo hijo	<p>El sistema permitirá adicionar un objetivo hijo con los siguientes datos:</p> <ul style="list-style-type: none"> • Denominación • Fecha de inicio • Fecha de fin • Descripción • Estado de cumplimiento • Porcentaje de cumplimiento • Involucrados • Participantes • Dirigentes
RF18.	Modificar objetivo hijo	<p>El sistema permitirá modificar un objetivo hijo previamente adicionado, con los siguientes datos:</p> <ul style="list-style-type: none"> • Denominación • Fecha de inicio • Fecha de fin • Descripción • Estado de cumplimiento • Porcentaje de cumplimiento • Involucrados • Participantes • Dirigentes

En cuanto a los RnF a continuación, se detallan los identificados para la solución propuesta:

✓ **Requisito de restricción de diseño**

RnF1. Utilizar la herramienta CASE Visual Paradigm teniendo en cuenta sus ventajas para modelar el ciclo de vida del proceso.

RnF2. Utilizar la herramienta PostgreSQL v10.5 para la gestión de los datos en la propuesta de solución, así como la herramienta Kettle v7.1 perteneciente a la suite de Pentaho para el proceso de ETL, que requiere a su vez de Java 8.0 para establecer conexiones con la fuente de datos.

✓ **Requisito de compatibilidad**

RnF3. Realizar todas configuraciones de las que depende el correcto funcionamiento de PostgreSQL v10.5.

✓ **Requisito de seguridad**

RnF4. Garantizar el uso de servicios de red, puertos y protocolos acordes con las políticas de seguridad trazadas por la entidad que soportará el despliegue de la aplicación.

RnF5. Garantizar los mecanismos para proteger las cadenas de conexión y las contraseñas para conectarse a la base de datos en el servidor de aplicaciones mediante el software y hardware ya definido para el sistema.

RnF6. Garantizar en el sistema, que cada usuario para consultas a la base de datos, cuente solo con los privilegios mínimos e indispensables.

✓ **Requisito de escalabilidad**

RnF7. Garantizar que el servidor de base de datos debe posea una capacidad mínima de almacenamiento de 500 GigaByte.

✓ **Requisito de confiabilidad**

RnF8. Configurar un sistema de respaldo ante fallos del servidor que garantice una copia de la aplicación así como del estado diario de la base de datos.

Por otra parte, con el objetivo de definir el significado de los términos del negocio para que posean una única interpretación por parte de todos los involucrados al proyecto, se establece el Glosario de términos con las siguientes definiciones:

ARC: Área de resultados claves

FIP: Factores que influyen en el plan

SIPAC: Sistema de Planificación de Actividades

Diseño para el subsistema de integración de datos

En esta etapa se obtiene el producto de software, se diseñan e implementa el subsistema de integración de datos, que forma parte de la solución propuesta. A continuación, se detallan cada uno de los pasos a seguir.

✓ **Estrategia para garantizar la calidad de datos**

Esta tarea dentro de la etapa de diseño tiene como objetivo cuantificar los datos que poseen la fuente de datos de origen (Data, 2016). Además de eliminar errores, duplicidades e inconsistencias para lograr homogenizar los conceptos, garantizando la consistencia, fiabilidad e integridad de las bases de datos (Tarazona, 2017). Según (ESPINOZA, 2018) es un proceso para analizar los datos en

detalle, comparándolos con sus metadatos, calculando estadísticas de datos e informando de las medidas de calidad de los datos que se deben aplicar en cada momento. Para esta tarea se propone el siguiente ciclo (Data, 2016):

- Identificar los datos que requieran de una limpieza previa.
- Definir reglas para evaluar los datos identificados previamente y diagnosticar el estado actual.
- Ejecutar las reglas de variables, almacenamiento o transformación establecidas a los datos para realizar la limpieza de estos.
- Evaluar los resultados obtenidos para identificar excepciones y definir nuevas reglas.

En esta etapa se realiza el pre-procesamiento de los datos para determinar la correspondencia entre los campos de origen-destino y diagnosticar el estado de los datos actuales. En caso de que los datos tengan inconsistencias e incompatibilidad, se debe detectar estos datos para realizar una reestructuración, limpieza o transformación de los mismos. Cuando se analiza la información que la nueva base de datos debe conservar y a la que debe permitir total acceso, en ocasiones hay datos redundantes, incompletos, innecesarios o despreciables que no se tienen en cuenta en el destino. La migración tiene como objetivo la recuperación total de los datos por lo que estos deben ser conservados en su totalidad para ser consultados siempre que estos datos no afecten a la información ya almacenada.

✓ Mapa lógico de datos

Con el fin de representar el origen y destino físico de cada uno de los atributos que conforman el proceso de migración, en la tabla que se muestra a continuación se describe un fragmento del mapa lógico de datos según la estructura física de las fuentes de datos y el modelo físico de la propuesta de solución.

Tabla 5: Mapa lógico de datos para la solución propuesta

Mapa lógico de datos					
Nombre de BD Fuente: SIPAC 2.4			Nombre de BD Destino: SIPAC 3.0		
Nombre de la tabla	Nombre del campo	Tipo de dato	Nombre de la tabla	Nombre del campo	Tipo de dato
nom_dominio	denominacion	character varying (255)	cal_estructura	denominacion	character varying (255)
			cal_estructura	fecha_inicio	date

			cal_estructura	fecha_fin	date
seg_usuario	nombreusuario	text	auth_user	username	character varying (150)
seg_usuario	contrasenna	character varying (255)	auth_user	password	character varying (128)
dat_fip	fechaemision	date	cal_fip	fechaemision	date
dat_arc	denarc	character varying (255)	cal_arc	denominacion	character varying (2000)
dat_actividades	esprincipal	numeric	schedule_event	es_actividad_principal	boolean
dat_actividades	lugar	character varying (255)	schedule_occurrence	lugar	character varying (500)

✓ **Modelo físico de datos**

Con el objetivo de representar los objetos de datos y sus relaciones, a continuación, se muestra un fragmento del modelo físico de datos. En este caso sólo se modelan las tablas objetivos para un mayor entendimiento del mismo:

✓ Plan detallado ETL

Este proceso permite la transferencia de datos desde múltiples fuentes, este caso en particular se propone desde la base de datos perteneciente al SIPAC 2.4, donde posteriormente los datos fueron reformateados y limpiados para ser cargados en la base de datos del SIPAC 3.0.

Extracción de datos de la base de datos origen, la cual tiene como característica ser una base de datos relacional, esta extracción debe convertir los datos en un formato preparado para iniciar el proceso de transformación. Una parte intrínseca que se debe tener en cuenta, es analizar los datos extraídos. Este chequeo permite verificar si los datos cumplen la pauta o estructura que se esperaba y de no ser así los datos se deben rechazar. Un requerimiento importante a tener en cuenta es que la extracción debe causar un impacto mínimo en el sistema origen, ya que, si los datos a extraer son muchos, el sistema de origen podría ralentizar e incluso colapsar.

Para la extracción de los datos de SIPAC 2.4 se estudió la estructura y la composición de las tablas en la base de datos y las coincidencias con la base de datos destino, en este caso, SIPAC 3.0. Es necesario de igual forma definir la ubicación de los datos y cómo extraerlos en dependencia de las relaciones funcionales entre ellos. También se define la ubicación de los datos fundamentales para proceder a la extracción de estos y poder aplicarles las transformaciones necesarias en el siguiente paso. Se procede a extraer los datos de las tablas identificadas previamente en la fase de estudio preliminar y planeación.

En la fase de **transformación** se deben aplicar una serie de reglas de negocio o funciones sobre los datos extraídos de la base de datos de SIPAC 2.4 para convertirlos en datos cargados. A continuación, se mencionan algunas de las transformaciones que se deben realizar:

- ✓ Fusión de campos.
- ✓ Cambios en la cantidad de caracteres permitidos.
- ✓ Cambio de estructura.
- ✓ Eliminar recursividad.
- ✓ Cambio del tipo de dato en algunos valores (por ejemplo, convertir de int a boolean).
- ✓ Agregar constantes a los nuevos campos para evitar columnas vacías.
- ✓ Establecer herencias.
- ✓ Se generaron campos claves en el destino.

Para la fase de **carga**: los datos de la fase anterior, deben ser cargados en el sistema destino, es decir, en la base de datos del SIPAC en la versión 3.0. Para este proceso el autor de la presente investigación define un conjunto de pasos que se detallan en la siguiente figura:

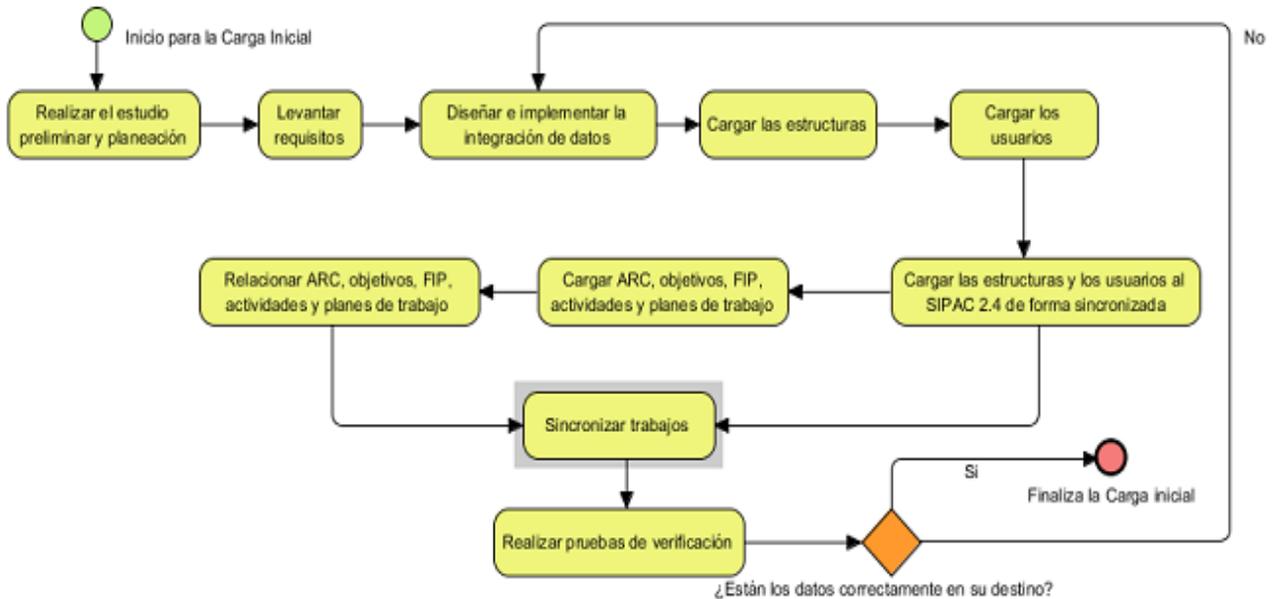


Figura 5: Estrategia para la migración de SIPAC

Implementación del subsistema de integración de datos

Se propone realizar transformaciones para la carga de las estructuras y la carga de los usuarios. A continuación, se muestra un ejemplo de cómo se llevarían a cabo estos procesos.

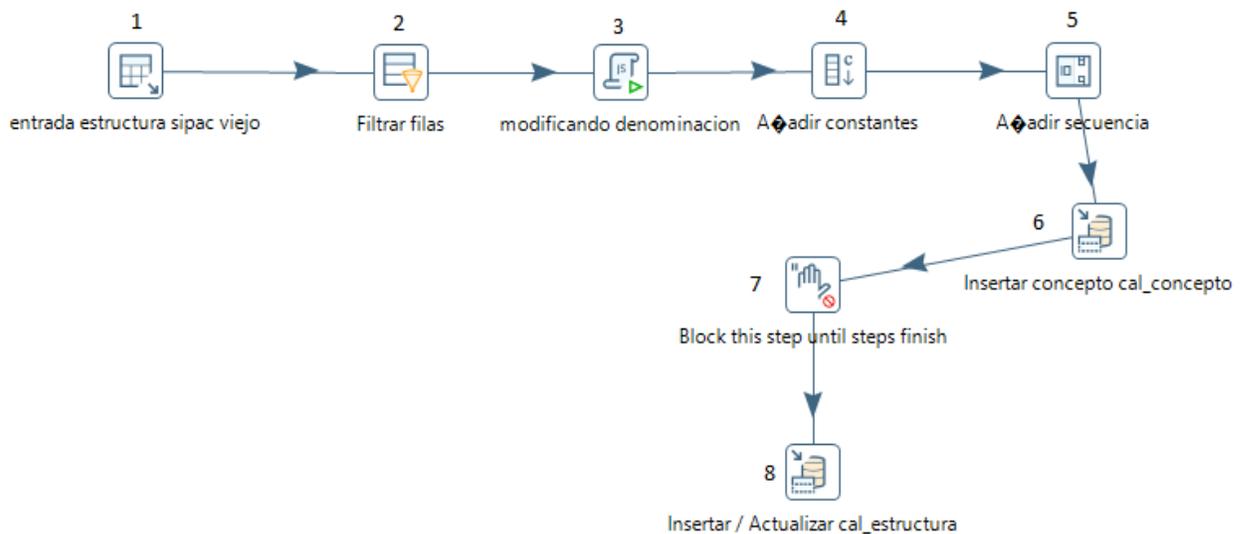


Figura 6: Carga de las estructuras del SIPAC 2.4 al SIPAC 3.0

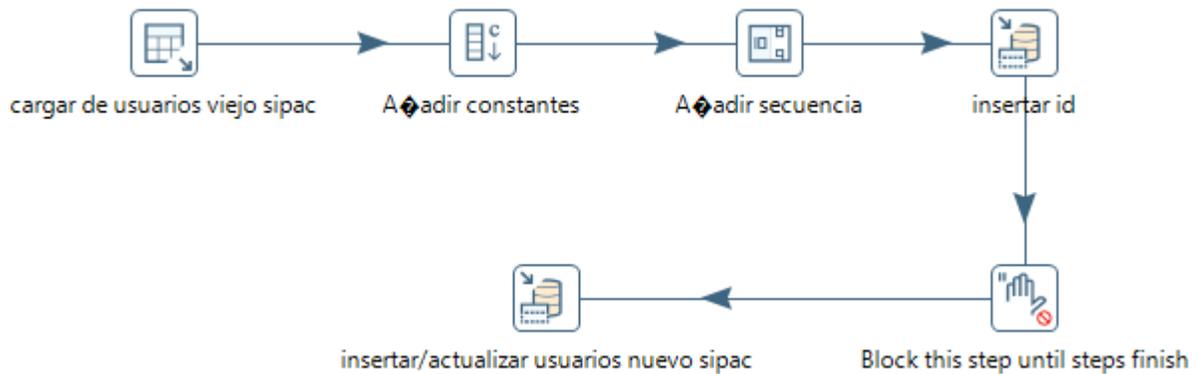


Figura 7: Carga de los usuarios del SIPAC 2.4 al SIPAC 3.0

En la carga de las estructuras el primer paso es la extracción de los datos de SIPAC 2.4 para proceder a las transformaciones necesarias como los pasos 2, 3 y 4 (ver figura 7).

Nombre paso

Conexión

SQL

```

SELECT
iddominio,
denominacion
FROM mod_estructuracom.nom_dominio
WHERE
iddominio <> 1
ORDER BY
iddominio
  
```

Line 1 Column 0

Enable lazy conversion

Reemplazar variables en

Insertar datos del paso

Ejecutar para cada fila?

Limitar tamaño

Figura 8: Entrada de tabla

En el paso 2 se filtran las filas para seleccionar todas las denominaciones que comiencen en Dom (ver figura 8).

The screenshot shows a configuration window for a data transformation step. The title is 'Nombre de paso' with the value 'Filtrar filas'. Below it are two dropdown menus: 'Enviar 'verdadero' a paso:' and 'Enviar 'falso' a paso:'. The main configuration area is titled 'La condición:' and contains a text input field with 'denominacion', a dropdown menu with 'LIKE', another text input field with 'Dom%', and the text '(String)' to its right. A small green arrow icon is visible on the right side of the condition area.

Figura 9: Filtrar Filas

Luego en el paso 3, mediante un archivo JavaScript se le eliminan las 3 primeras letras de estos datos, para eliminar el Dom de los datos (ver figura 9), por ejemplo, DOMViceDefensa resultaría en ViceDefensa

The screenshot shows a configuration window for a data transformation step. The title is 'Nombre de paso' with the value 'modificando denominacion'. On the left is a tree view of 'Java script functions' with categories like 'Transform Scripts', 'Transform Constants', 'Transform Functions', 'Input fields', and 'Output fields'. The main area is titled 'Java script:' and contains a code editor with the following code:

```
//Script here
var new_denom = trim(substr(denominacion,3));
```

 Below the code editor, there are fields for 'Número de línea: 0', 'Compatibility mode?', and 'Optimization level' set to '9'.

Figura 10: Modificando denominación

En el paso 4 se definen constantes que se ubicarán en las bases de datos (ver figura 10).

Nombre paso

Campos:

#	Nombre	Tipo	Formato	Longitud	Precisión	Moneda	Decimal	Grupo	Valor	Set empty string?
1	is_group	Boolean							false	N
2	fecha	Date	dd/MM/yyyy						01/07/2019	N
3	id_dpa	Integer							13	N

Figura 11: Añadir constantes

En el paso 5 se añade la secuencia donde añadirá una columna con el nombre id_concepto (ver figura 11) y se inserta en la base de dato en el paso 6 (ver figura 12).

Obtener valor de la secuencia de la base de datos

Nombre de paso

Nombre de valor

Utilizar una base de datos para generar la secuencia

Utilizar base datos para obtener secuencia?

Conexión

Nombre de esquema

Nombre de secuencia

Utilizar un contador de la transformación para generar la secuencia

Utilizar contador para calcular secuencia?

Nombre contador (opcional)

Valor inicial

Incremento

Valor máximo

Figura 12: Añadir secuencia

Nombre de paso:

Conexión:

Esquema destino:

Tabla destino:

Tamaño transacción (commit):

No realizar actualizaciones:

La clave(s) para realizar búsqueda de valor(es):

#	Campo de tabla	Comparador	Campo1	Campo2
1	id_concepto	=	id_concepto	

Campos de actualización:

#	Campo de tabla	Campo de Flujo	Actualizar
1	id_concepto	id_concepto	N
2	tipo	is_group	N

Figura 13: Insertar concepto

El paso 7 es una condicional que bloquea el flujo de datos para que no pueda continuar, este paso no permitirá el flujo hacia el paso 8 hasta que exista algún id_concepto (ver figura 13), en caso de que exista se finaliza insertando o actualizando los datos en la base de datos (ver figura 14) completando la carga de las estructuras.

Step name

Watch the following steps

#	Step name	Copy Nr
1	Insertar concepto cal_concepto	0

Figura 14: Block this step

Nombre de paso

Conexión

Esquema destino

Tabla destino

Tamaño transacción (commit)

No realizar actualizaciones:

La clave(s) para realizar búsqueda de valor(es):

#	Campo de tabla	Comparador	Campo1	Campo2
1	concepto_ptr_id	=	id_concepto	

Campos de actualización:

#	Campo de tabla	Campo de Flujo	Actualizar
1	denominacion	new_denom	N
2	fecha_inicio	fecha	N
3	fecha_fin	fecha	N
4	dpa_id	id_dpa	N
5	is_grupo	is_group	N
6	concepto_ptr_id	id_concepto	N

Figura 15: Insertar/Actualizar estructura

Para la carga de los usuarios se propone emplear el mismo procedimiento, pero con datos diferentes. En este caso no es necesario ejecutar los pasos 2 y 3. Para asociar las estructuras con los usuarios se debe tener en cuenta la tabla `mod_seguridad.seg_usuarios_nom_dominio` que contiene el id de los usuarios y las estructuras, y se procede a asociarlos mediante esta en las transformaciones de Pentaho. Para cargar los planes, FIP, ARC y objetivos se realiza un proceso similar al demostrado por las estructuras aplicando las transformaciones con diferentes datos. En el caso de las actividades intervienen dos tablas como destino: `shedule_event` y `shedule_occurrence`. Estas tablas tienen una relación de padre-hijo respectivamente, por lo que requiere una extracción dividida de los datos y un trabajo por separado cargando en la primera tabla las actividades y en la otra el personal asignado a estas actividades.

Validación y pruebas

En esta fase se realizan las pruebas necesarias para validar la calidad del de la migración y los datos en el sistema de SIPAC 3.0. Las pruebas que se realizan son (Hernández, 2013):

- ✓ Pruebas de integración: permite verificar la correcta integración de los componentes y subsistemas que conforman la solución. Esta prueba se sincronizará con la migración para comprobar el estado de los datos en la base de datos y su correcto traslado.
- ✓ Pruebas de sistema (validación): permiten validar el cumplimiento de los requisitos de información y funcionales definidos por los clientes. Son las pruebas más cercanas a la realidad del cliente, debido a que los probadores utilizan el sistema de la misma manera que será usado por los clientes. Esta prueba es fundamental para comprobar la accesibilidad de los datos.

Para el caso de la estrategia que se plantea como solución al problema que da paso a la presente investigación, se propone tener en cuenta los resultados de las ejecuciones lanzadas por la herramienta (ver figura 15) y las consultas directas en la base de datos. Esto se realiza sincronizando la implementación del subsistema de integración a estas pruebas, comparando una muestra de los datos entre ambos sistemas. Como la migración se realiza de forma incremental estas pruebas se realizan con el objetivo de detectar errores en cada uno de los procesos ETL.

Execution Results

Execution History
 Logging
 Step Metrics
 Performance Graph
 Metrics
 Preview data

\${TransPreview.FirstRows.Label}
 \${TransPreview.LastRows.Label}
 \${TransPreview.Off.Label}

#	iddominio	denominacion	new_denom	is_group	fecha	id_dpa	id_concepto
1	3.0	DomPrimerVicePresidente	PrimerVicePresidente	N	2019/07/01 00:00:00.000	13	802
2	29.0	DomRelacionesInternacionales	RelacionesInternacionales	N	2019/07/01 00:00:00.000	13	803
3	30.0	DomDependenciaInterna	DependenciaInterna	N	2019/07/01 00:00:00.000	13	804
4	31.0	DomINDER	INDER	N	2019/07/01 00:00:00.000	13	805
5	32.0	DomEMPRESCONSUL	EMPRESCONSUL	N	2019/07/01 00:00:00.000	13	806
6	33.0	DomSaladeTV	SaladeTV	N	2019/07/01 00:00:00.000	13	807
7	34.0	DomOficinadelHistoriador	OficinadelHistoriador	N	2019/07/01 00:00:00.000	13	808
8	35.0	DomJusticia	Justicia	N	2019/07/01 00:00:00.000	13	809
9	36.0	DomCombatientes	Combatientes	N	2019/07/01 00:00:00.000	13	810
10	39.0	DomSubdireccion	Subdireccion	N	2019/07/01 00:00:00.000	13	811
11	40.0	DomDepartamentoInformatica	DepartamentoInformatica	N	2019/07/01 00:00:00.000	13	812
12	41.0	DomDepartamentoEconomia	DepartamentoEconomia	N	2019/07/01 00:00:00.000	13	813
13	42.0	DomDptoSegProteccion	DptoSegProteccion	N	2019/07/01 00:00:00.000	13	814
14	43.0	DomDepartamentoRRHH	DepartamentoRRHH	N	2019/07/01 00:00:00.000	13	815
15	44.0	DomDepartamentoServicios	DepartamentoServicios	N	2019/07/01 00:00:00.000	13	816

Figura 16: Resultados obtenidos luego de la migración con la herramienta

En el caso de las pruebas de integración en la base de datos se realizará de forma comparativa, revisando y comprobando que los datos fueron correctamente migrados de SIPAC 2.4 (ver figura 16) a SIPAC 3.0 (ver figura 17)

	concepto_ptr_id [PK] integer	denominacion character varying (255)	codigo character varying (20)	fecha_inicio date	fecha_fin date	abreviatura character varying (20)	dpa_id integer
1	0	Estructuras		2020-07-15	2020-07-15		[null]
2	2	Universidad de las Ciencias In...	UCI	2020-07-15	2020-07-15	UCI	13
3	3	Centro CEIGE	CEIGE	2020-07-15	2020-07-15	CEIGE	13
4	4	Dpto Desarrollo de Componen...	DC	2020-07-15	2020-07-15	DC	13
5	5	Aplicaciones de Gestión Empr...	AGE	2020-07-15	2020-07-15	AGE	13
6	671	CAPCAMGUEY	[null]	2019-07-01	2019-07-01	[null]	13
7	672	PrimerVicePresidente	[null]	2019-07-01	2019-07-01	[null]	13
8	673	SecretariaCAP	[null]	2019-07-01	2019-07-01	[null]	13
9	674	ViceTransporte	[null]	2019-07-01	2019-07-01	[null]	13
10	675	ViceEconomia	[null]	2019-07-01	2019-07-01	[null]	13
11	676	ViceDistribucion	[null]	2019-07-01	2019-07-01	[null]	13
12	677	ViceDefensa	[null]	2019-07-01	2019-07-01	[null]	13
13	678	ViceInversiones	[null]	2019-07-01	2019-07-01	[null]	13
14	679	SecretariaAsamblea	[null]	2019-07-01	2019-07-01	[null]	13
15	680	DOPI	[null]	2019-07-01	2019-07-01	[null]	13

Figura 17: Estado de los datos en SIPAC 2.4

	iddominio [PK] numeric (19)	denominacion character varying (255)	dominiostring character varying (255)	descripcion character varying (250)	seguridad integer
1	1	Dominio de instalación	100000001		1
2	2	DOMCAPCAMGUEY	[null]		0
3	3	DomPrimerVicePresidente	[null]		0
4	4	DOMSecretariaCAP	[null]		0
5	5	DOMViceTransporte	[null]		0
6	6	DOMViceEconomia	[null]		0
7	7	DOMViceDistribucion	[null]		0
8	8	DOMViceDefensa	[null]		0
9	9	DOMViceInversiones	[null]		0
10	10	DOMSecretariaAsamblea	[null]		0
11	11	DOMDOPI	[null]		0
12	12	DOMDireccionCuadro	[null]		0
13	13	DOMDirProvEducacion	[null]		0
14	14	DOMDirProvSalud	[null]		0
15	15	DOMCAMSantaCruz	[null]		0

Figura 18: Estado de los datos en SIPAC 3.0

Además, se propone ejecutar pruebas de sistema para comprobar la estabilidad y calidad de los datos. En esta etapa se comprobará que la migración realizada fue correcta verificando que el SIPAC en su versión 3.0, funcione correctamente una vez realizada la migración total y la carga inicial de los datos. Para esto se propone utilizar los Diseños de Casos de Pruebas (DCP) realizados para el sistema origen, lo que esta vez sobre el sistema destino.

En ingeniería del software, un DCP es un conjunto de condiciones o variables bajo las cuales un analista determinará si una aplicación o una característica de éstos es parcial o completamente satisfactoria (Pressman, 2010).

Para un mayor entendimiento de este proceso, en los Anexos se muestra la descripción de las variables (ver anexo 1) y la definición de los escenarios a probar (ver anexo 2) correspondiente al requisito funcional “Adicionar plan”.

Una vez ejecutados todos los DCP y comprobada la estabilidad y calidad de los datos, se podrá concebir que la migración total y carga inicial de los datos del SIPAC en su versión 2.4 a la versión 3.0 fue satisfactoria, disminuyendo así la discrepancia entre los datos y el riesgo de pérdida de información de los mismos en dichos procesos.

CONCLUSIONES

Una vez terminada la presente investigación se tiene que:

- ✓ El establecimiento de los referentes teóricos en los que se sustenta la propuesta de solución sobre la carga inicial y la migración de los datos en sistemas informáticos, permitió obtener un esclarecimiento del objeto de estudio de la presente investigación.
- ✓ La selección de los datos que se van a migrar desde la versión 2.4 de SIPAC a SIPAC 3.0 favoreció la comprensión de la fuente de datos, identificando áreas, campos, tablas o datos críticos para el cliente.
- ✓ El estudio y selección de la metodología de desarrollo a utilizar permitió organizar las actividades y tareas a ejecutar, durante todo el ciclo de vida de desarrollo de la propuesta de solución.
- ✓ La selección de las tecnologías, herramientas y estándares aseguró el ambiente de desarrollo necesario para la implementación de la propuesta de solución.
- ✓ La elaboración de la descripción del proceso de Extracción, Transformación y Carga de los datos que se van a tener en cuenta para migrar del sistema origen SIPAC 2.4 a la versión 3.0, permitió disminuir la discrepancia entre los datos y el riesgo de pérdida de información de los mismos en dichos procesos.

RECOMENDACIONES

A manera de recomendación se incluyen los siguientes aspectos:

- ✓ Completar la implementación del subsistema de integración para finalizar la carga inicial de los datos de SIPAC en su versión 3.0.
- ✓ Ejecutar la etapa de pruebas en el entorno de producción para verificar el estado de los datos de SIPAC en su versión 3.0.

REFERENCIAS BIBLIOGRÁFICAS

Angeles Cruz Manjarrez Antaño, José Mario Martínez Castro y René E. Cuevas Valencia. 2014. *Migración de Bases de Datos SQL a NoSQL*. México : Tlamati, 2014.

Borjas, Indira Laukión. 2018. *PROCEDIMIENTO PARA LA PLANIFICACIÓN EMPRESARIAL. CASO ESTUDIO EMPRESA CONSTRUCTORA DE OBRAS DE INGENIERÍA No17*. Holguin : s.n., 2018.

Bustillos, Jorge. 2014. *Comparativa Herramientas ETL*. 2014.

Curto, Josep. 2010. *Introducción a la Business Intelligencie*. Barcelona : UOK, 2010.

Dario, Bernabeu Ricardo. 2010. *Hefesto Data Warehousing: Investigación y sistematización de conceptos*. Córdoba : s.n., 2010.

Data, Power. 2016. Migración de datos: definición, desafíos y mejores prácticas para afrontarla. *Power Data*. [En línea] 2016. <https://www.powerdata.es/migracion-de-datos>.

ESPINOZA, OREL OCTAVIO ROMERO. 2018. *GESTIÓN DE PROYECTOS DE TECNOLOGÍAS DE INFORMACIÓN (GEPTI)*. 2018.

Flores, Araceli Soledad Domínguez. 2013. *Unidad 1: UML y el proceso unificado. Desarrollo e implementación de Sistemas de Información*. 2013.

Franklin, Terry G. y. 1987. *Principios de administración*. México : CECSA, 1987.

Gaines, Jeff, Boyd, Geraldine y Copley, Della. 2017. Visual Paradigm Online. *Visual Paradigm Online*. [En línea] 2017. <https://online.visual-paradigm.com/es/features/>.

Geovanny Euclides Silva Peñafiel, Verónica Marcela Zapata Yáñez, Klever Patricio Morales Guamán, Luis Marcelo Toaquiza Padilla. 2019. *Análisis de metodologías para desarrollar Data Warehouse aplicado a la toma de decisiones*. Ecuador : ISSN: 2602-8085, 2019. Vol. 3.

González, Lis. 2011. Catarina.udlap.mx. [En línea] 1 de Febrero de 2011.
http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/gonzalez_m_ac/capítulo3.pdf.

Gravitar, Colectivo de autores de. 2010. Gravitar Información. [En línea] 20 de 10 de 2010.
www.gravitar.biz.

Haider, Aelia. 2019. aстера.com. [En línea] 19 de Febrero de 2019. [Citado el: 12 de Febrero de 2020.]
<https://www.astera.com/es/type/blog/data-integration-tools-for-businesses/>.

Hernández, Yanisbel González. 2013. *METODOLOGÍA DE DESARROLLO PARA PROYECTOS DE*
. UCI : s.n., 2013.

- Ibarra, Antonio Aliaga. 2008.** iessanvicente.com. *iessanvicente.com*. [En línea] 2008. <https://iessanvicente.com/colaboraciones/postgreSQL.pdf>.
- Ismael, Eric Leonard Brizuela y Blanco, Yudi Castro. 2013.** *Metodologías para desarrollar Almacén de Datos*. Matanzas : Revista de Arquitectura e Ingeniería, 2013. Vol. 7.
- ISO/IEC. 2016.** *ISO/IEC 25023 Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) - Measurement of system and software product quality*. Geneva, Switzerland : International Organization for Standardization, 2016.
- Jhoan Esteban, Borja Ruiz. 2018.** *Comparación de herramientas ETL de código abierto*. Universidad Nacional de Colombia. Medellín : s.n., 2018.
- Juhasz, Harun. 2014.** <http://biucv.blogspot.com>. [En línea] 26 de Enero de 2014. [Citado el: 15 de Febrero de 2020.] <http://biucv.blogspot.com/2014/01/ciclo-de-vida-de-kimball.html>.
- Lucid. 2018.** lucidchart.com. *lucidchart.com*. [En línea] 2 de 12 de 2018. [Citado el: 5 de 12 de 2018.] <https://www.lucidchart.com/pages/es/que-es-el-lenguaje-unificado-de-modelado-uml>.
- Marin, Rafael. 2019.** revistadigital.inesem.es. *revistadigital.inesem.es*. [En línea] 16 de abril de 2019. <https://revistadigital.inesem.es/informatica-y-tics/los-gestores-de-bases-de-datos-mas-usados/>.
- Martín, Paula. 2019.** itop.academy. [En línea] 24 de Enero de 2019. [Citado el: 12 de Febrero de 2020.] <https://www.itop.academy/blog/item/que-es-pentaho-data-integration-pdi-y-para-que-sirve.html>.
- Martínez, Rafael. 2010.** Sobre postgresql. *PostgreSQL-es*. [En línea] octubre de 2010. [Citado el: 11 de diciembre de 2019.] http://www.postgresql.org/es/sobre_postgresql.
- Navarro, Manuel. 2018.** revistabyte.es. [En línea] 4 de Diciembre de 2018. [Citado el: 12 de Febrero de 2020.] <https://revistabyte.es/actualidad-byte/herramientas-de-integracion/>.
- Paciencia, Maida EG. 2015.** *Metodología de desarrollo de software*. Argentina : Tesisi de Licenciatura en Sistemas y Computación, 2015.
- Page, Dave, Saito, Hiroshino y Yeatman, Mark. 2018.** PgAdmin. *PgAdmin*. [En línea] 28 de Noviembre de 2018. <http://www.pgadmin.org/>.
- Paradigm, Visual. 2017.** [En línea] 2017. [Citado el: 20 de diciembre de 2019.] <http://www.visualparadigm.com/aboutus/newsreleases/vpuml80.jsp>.
- Pressman, Roger S, Ph.D. 2010.** *Ingeniería de Software. Un enfoque práctico. Séptima Edición*. Mexico, D.F : The McGraw Hill Companies, 2010.
- Pressman, Roger S. 2010.** *Ingeniería de Software. Un enfoque práctico. Séptima*. México DF : McGraw-Hill INTERAMERICA EDITORES, 2010.

Rivadera, Gustavo R. 2010. *La metodología de Kimball para el diseño de almacenes de datos.* Buenos Aires : s.n., 2010.

SANCHEZ, CARLOS ALBERTO AREVALO. 2017. *IDENTIFICACIÓN DE USUARIOS, AUTOMATIZACIÓN DEL PROCESO DE ANÁLISIS Y CLASIFICACIÓN DE INFORMACIÓN EN REPOSITORIOS HISTÓRICOS.* 2017.

Stoner, J. y otros. 1996. *Administración.* México : Prentice - Hall Interamericana, 1996.

Tarazona, Alba Yolima Leguizamon. 2017. *Pautas para un correcta migración de base de datos.* Bogota : s.n., 2017.

ANEXOS**Anexo 1:** Descripción de las variables para el requisito funcional: Adicionar plan

Nombre de campo	Tipo	Válido	Inválido	Inválido
Denominación	Campo de texto	Combinación de letras, números y los caracteres especiales: ""%#@!*()_;;,.?,-.	Vacío	Caracteres especiales: <>/[]{} +=\$^&~`¿¡¨Ççªº
Estado de cumplimiento	Campo de texto	Se actualiza según lo que el usuario introduzca en el campo Porcentaje de cumplimiento.	NA	NA
Categoría del plan	Lista desplegable	Combinación de letras.	Vacío	NA
Porcentaje de cumplimiento	Campo de texto	Combinación de números (Del 0 al 100). Por defecto aparece el cero. Permite el Vacío.	NA	NA
Fecha de inicio	Campo de fecha	Es de tipo fecha con el formato: dd/mm/aaaa. La fecha de inicio debe ser menor o igual que la fecha de fin.	NA	NA.
Fecha de fin	Campo de fecha	Es de tipo fecha con el formato: dd/mm/aaaa. La fecha de fin debe ser mayor o igual que la fecha de inicio.	NA	NA
Descripción	Campo de texto	Combinación de letras, números y los caracteres especiales: ""%#@!*()_;;,.?,-. Permite el Vacío.	NA	NA

Anexo 2: Escenarios a probar para el requisito funcional: Adicionar plan

Id del escenario	Escenario	Denominación	Estado de cumplimiento	Categoría del plan	Porcentaje de cumplimiento	Fecha de inicio	Fecha de fin	Descripción	Respuesta del sistema
EP 1.1	Adicionar un nuevo plan introduciendo datos válidos.	V(Plan de actividades principales)	V(Sin cumplir)	V(Plan de actividades)	V(0)	V(24/02/2020)	V(28/03/2020)	V(Actividades fundamentales del mes de Marzo.)	El sistema muestra una ventana informando que el plan ha sido adicionado correctamente: "El plan ha sido adicionado correctamente."
		V(Plan de visitas para el mes de Abril)	V(Sin cumplir)	V(Plan de visitas)	V(30)	V(04/01/2020)	V(04/03/2020)	V(Vacío)	
EP 1.2	Adicionar un nuevo plan dejando campos requeridos en blanco.	I(Vacío)	V(Sin cumplir)	V(Planes semestrales)	V(0)	V(01/01/2020)	V(31/06/2020)	V(Visitas al área de Deportes en el primer semestre de 2010.)	Se muestra el campo requerido, sin llenar en rojo que indica que no se pueden dejar campos requeridos en blanco, y se muestra el mensaje: "Por favor verifique que hay campo(s) vacío(s)."
		V(Plan mensual de actividades)	V(Sin cumplir)	I(Vacío)	V(100)	V(01/01/2020)	V(31/01/2020)	V(Actividades del mes de Enero en el área de Salud.)	
EP 1.3	Adicionar un nuevo plan	V(Plan de visitas para el mes de Abril)	V(Sin cumplir)	V(Plan de visitas)	V(30)	I(3 de enero)	V(04/03/2020)	V(Vacío)	El sistema no permite que se introduzcan

	introduciendo datos inválidos.	V(Plan de producción de Febrero)	V(Sin cumplir)	V(Planes de producción)	V(100)	V(02/01/2020)	I(1-2-20)	V(Vacío)	caracteres en campos de fecha ni en listas desplegables, así como no permite que se escriban letras o caracteres especiales en campos de textos que solo aceptan valores numéricos.
		V(Plan anual de control)	V(Sin cumplir)	V(Planes de control)	I(1/2)	V(02/01/2020)	V(30/12/2020)	V(Vacío)	
		V(Plan anual de control)	V(Sin cumplir)	V(Planes de control)	I(asd\$%<-+*)	V(02/01/2020)	V(30/12/2020)	V(Vacío)	
		V(Plan de visitas para el mes de Abril)	V(Sin cumplir)	V(Plan de visitas)	V(30)	I(04/02/2020)	I(04/01/2020)	V(Vacío)	El sistema no permite registrar la fecha de inicio posterior a la fecha de fin, el sistema se mantiene en la ventana de edición.
		V(Plan de visitas para el mes de Abril)	V(Sin cumplir)	V(Plan de visitas)	I(101)	V(04/01/2020)	V(04/03/2020)	V(Vacío)	El sistema no permite registrar el porcentaje de cumplimiento mayor que 100, se muestra una ventana
EP 1.4	Cancelar	NA	NA	NA	NA	NA	NA	NA	Se cancela la operación y se cierra la ventana.