

*Universidad de las Ciencias Informáticas*

*Centro de Informatización Universitaria*

*Facultad 1*



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

**Título:** Migración de los datos manejados por la Dirección de Cooperación Internacional hacia el Sistema de Gestión de Cooperación Internacional.

**Autores:**

Yunisleidys Gámez Gutiérrez

Reynel Almanza Vázquez

**Tutores:**

Ing. Javier González La Nuez

Ing. Reinier Morejón Novales

**La Habana, 2012**

**“Año del 54 Aniversario de la Revolución”**



*¡No sueñes tu vida... vive tus sueños!*

*Lo que puedes hacer, o has soñado que podrías hacer, debes comenzar. La osadía lleva en sí, genio, poder y magia.*

*Goethe Johann Wolfgang Von*

### **Declaración de autoría**

Declaramos que somos los únicos autores del trabajo “Migración de los datos manejados por la Dirección de Cooperación Internacional hacia el Sistema de Gestión de Cooperación Internacional” y autorizamos al Centro de Informatización Universitaria de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmamos los presentes a los \_\_\_\_ días del mes de \_\_\_\_\_ del año 2012.

\_\_\_\_\_  
Yunisleidys Gámez Gutiérrez  
Autor

\_\_\_\_\_  
Reynel Almanza Vázquez  
Autor

\_\_\_\_\_  
Javier González La Nuez  
Tutor

\_\_\_\_\_  
Reinier Morejón Novales  
Tutor

## *Dedicatoria*

*Yunisteidy G3mez Guti3rrez*

A mi familia, por apoyarme en todo momento y porque a ellos les debo lo que soy.

*Reynel Almanza V3zquez*

A toda mi familia por el apoyo incondicional que me ha brindado siempre y en especial a mi mamá, mi mayor motivaci3n en toda mi carrera.

### *Agradecimientos*

*Yunisteidy G3mez Guti3rrez*

A mima por su infinito apoyo a lo largo de mi carrera. Nunca le podr3 estar lo suficientemente agradecida por todo lo que ha hecho por m3.

A mi abuelita por su infinito esp3ritu de sacrificio, y cuidarme todos los a3os de mi vida.

A mi familia en general, a pipo, a mi abuelo, a mis primos, a mis t3as, a Ania.

No quiero dejar de agradecerles a mis amigos Alejandro y Daniel por sus constantes consejos y rega3os cuando hicieron falta.

Agradecer tambi3n a mis amigas Ani, Lillian, Mayara, y a las que no est3n aqu3 pero su amistad siempre fue un apoyo para m3.

Gracias a mis compa3eros de aula, a todos los que han compartido conmigo desde primer a3o hasta quinto, los quiero mucho, especialmente a aquellos que me apoyaron en los momentos dif3ciles.

A las ni3as de mi apto, Zule, Yari, Dai, Aylena, Ani, por soportarme tanto tiempo.

Un especial agradecimiento a Riki por todo el apoyo incondicional que me brind3.

A mis tutores Javier y Reinier, por su conocimiento y ayuda.

A mi compa3ero de tesis por soportarme y poner todo su empe3o en la realizaci3n de este trabajo.

A una persona que ya no est3 en mi vida pero que es muy importante para m3.

A todos los que de una manera u otra han dejado su huella en m3 durante estos 5 a3os.

¡Muchas Gracias!

*Reynel Almanza Vázquez*

Agradezco de todo corazón y de manera muy especial a mi mamá, la persona sin la cual no hubiese sido posible finalizar mis estudios. A ella que tanto esfuerzo ha realizado durante toda mi carrera para que pudiera llegar hasta este momento en que se ve el fruto de tanto sacrificio y dedicación. Doy gracias a Dios por permitirme ser hijo tuyo, por darme la satisfacción de llamarte madre, por permitir que fueras tú la persona que me diera la educación que hoy tengo. Gracias a ti soy lo que soy y a ti te debo todo en mi vida. Espero te sientas orgullosa de mí, que es lo que más me interesa en este mundo. Creo que toda una vida no me es suficiente para retribuirte todo lo que has hecho por mí, solo me queda decirte que aquí me tienes para todo y en todo momento. Te quiero mucho.

A mis hermanos Reinier, Leandra y Javier por mantenernos unidos a pesar de la distancia y por lo que representan para mí.

A mi padrastro Alejandro (Leonardo) que ha sido como un padre y aunque no habla mucho me ha demostrado de una manera u otra que me aprecia como un hijo.

A mi papá por todo su apoyo y cariño, quien a pesar de estar lejos de mí siempre ha estado al tanto de mi vida y mis estudios.

A mi tía Sonia, a quien quiero como una madre, le agradezco por soportarme tanto tiempo, por su cariño y apoyo.

A mi abuela Paula, que siempre me ha dado buenos consejos para que las cosas me salgan bien y por su apoyo durante toda mi carrera.

A mis primos Javier, Leandro, Jorgito, David, Roniel y a mi tío Manolo que siempre se ha interesado por mis estudios.

A todos mis amigos que hoy no están aquí, gracias por darme la oportunidad de contar con su amistad y por el apoyo que me brindaron en todos los años que compartimos.

Agradezco de manera especial a Niuver y Dianelys que siempre han estado ahí para apoyarme y a quienes considero como parte de mi familia.

A todos mis compañeros de grupo, a Salgado, Zoe, Jose, Yeni, Maryanis y a mi compañera de tesis por su empeño en la realización de este trabajo.

A mis tutores Javier y Reinier por la ayuda y el conocimiento aportado.

¡A todos muchas gracias!

### **Resumen**

Las tecnologías informáticas avanzan de manera acelerada y con ella la información se ha convertido en un factor muy importante para toda la sociedad. En la Universidad de las Ciencias Informáticas existe actualmente la Dirección de Cooperación Internacional, encargada de la gestión de los servicios de trámites de misión, la gestión de visitas y de colaboración internacional. Esta dirección cuenta con un sistema que almacena solamente una parte de los datos que maneja, el resto se encuentra almacenado en documentos con diferentes formatos. Esta descentralización de la información provoca un manejo complejo de la misma, por lo que se creó el Sistema de Gestión de Cooperación Internacional, el cual no cuenta aún con el historial de los datos manejados por la Dirección de Cooperación Internacional. Es por esto que se hace imprescindible diseñar una estrategia de migración que permita almacenar en dicho sistema todos los datos manejados por esta dirección. Para esto fue necesario realizar un estudio de las herramientas y procedimientos de migración existentes, así como también la validación de la solución propuesta.

**Palabras clave:** base de datos, estrategia, migración, pruebas.

## **Abstract**

Computer technologies are progressing rapidly, and with it the information has become very important for society. At the University of Information Sciences there is now a Department of International Cooperation, responsible for the management of mission processing services, management of visits and international collaboration. This address has a system that stores only a part of the data handled, the rest is stored in documents of different formats. This decentralization of information creates complex management of it, so it created the Management System for International Cooperation, which does not yet have the history of the data handled by the Directorate for International Cooperation. That is why it is essential to design a migration strategy that allows data stored in the system all data handled by this address. This required a study of existing migration tools and migration procedures performed previously, as well as validation of the proposed solution.

**Keywords:** database, strategy, migration, tests.



**Índice**

Introducción .....	1
Capítulo 1. Fundamentación teórica.....	6
1.1 Introducción.....	6
1.2 Conceptos asociados al dominio del problema.....	6
1.2.1 Dirección de Cooperación Internacional.....	6
1.2.2 Base de Datos .....	7
1.2.3 Sistemas Gestores de Base de Datos.....	8
1.2.4 SQL Server .....	9
1.2.5 Migración de datos.....	10
1.2.6 Concepto de estrategia .....	10
1.2.7 Tendencias actuales de la migración de datos.....	11
1.2.8 Pruebas de migración .....	11
1.3 Estudio de procedimientos de migración de datos.....	12
1.3.1 Procedimiento aplicado en Innovative System, Inc. ....	12
1.3.2 Migración integrada de datos para el Sistema Nacional de Salud. ....	12
1.4 Aportes obtenidos de los procedimientos estudiados: .....	14
1.5 Herramientas informáticas utilizadas en los procesos de migración de datos.....	14
1.5.1 Kettle .....	14
1.5.2 Talend Open Studio .....	14
1.5.3 AdventNet SwisSQL.....	15

1.5.4 Softek Transparent Data Migration Facility UNIX (IP).....	16
1.5.5 Ora2Pg .....	16
1.6 Resultados del estudio de las herramientas de migración de datos.....	17
1.7 Herramientas y tecnologías a utilizar .....	17
1.7.1 Gestor de base de datos PostgreSQL 8.4.....	17
1.7.2 PgAdmin III .....	18
1.7.3 Entorno de desarrollo integrado NetBeans 6.9.1.....	19
1.7.4 Marco de trabajo CodeIgniter.....	19
1.7.6 Lenguaje de programación.....	21
1.7.6.1 PHP 5.3 .....	21
1.7.7 Lenguaje de consulta SQL.....	21
1.7.8 Visual Paradigm.....	22
1.7.9 Servidor Web .....	23
1.7.10 Sistema operativo Ubuntu 11.10 .....	23
1.8 Conclusiones parciales.....	24
Capítulo 2. Descripción y análisis de la solución propuesta .....	25
2.1 Introducción.....	25
2.2 Características de los sistemas involucrados .....	25
2.2.1 Sistema de Gestión de Cooperación Internacional.....	25
2.2.2 Dirección de Relaciones Internacionales (DRI) .....	26
2.3 Migración de datos .....	26
2.3.1 Etapas de la migración de datos .....	26

2.3.2. Técnicas de migración de datos.....	27
2.4 Propuesta de arquitectura para la migración de los datos .....	28
2.5 Proceso de extracción, transformación y carga (ETL).....	29
2.5.1 Extracción .....	29
2.5.2 Transformación y limpieza .....	29
2.5.3 Carga de los datos.....	29
2.6 Calidad de datos.....	29
2.7 Estrategia de migración propuesta .....	30
2.8 Descripción de la solución propuesta .....	31
2.8.1 Desarrollo de la migración de los viajes hacia el módulo de Trámites.....	31
2.8.2 Desarrollo de la migración de la tabla TDPPermisoSalida del DRI hacia el SGCI ..	36
2.9 Conclusiones parciales.....	40
Capítulo 3. Validación de la solución propuesta.....	41
3.1 Introducción.....	41
3.2 Problemas encontrados en el proceso de migración de datos.....	41
3.3 Estrategia de validación y prueba.....	41
3.3.1 Perfilado de datos .....	42
3.3.2 Pruebas de aceptación .....	55
3.4 Conclusiones parciales.....	55
Conclusiones .....	56
Recomendaciones .....	57
Referencias bibliográficas .....	58

Bibliografía consultada..... 61

Anexos..... 62

## Introducción

En la actualidad las Tecnologías de la Información y las Comunicaciones (TIC) representan el eslabón vital del desarrollo y progreso de la sociedad, cuya característica principal es el conocimiento y el mejoramiento de la calidad de vida, permitiendo la interactividad, velocidad, interconexión y digitalización de la información.

El acelerado avance de las TIC ha influido considerablemente en el campo de la informática, la cual surgió por la necesidad social creciente de desarrollar nuevos métodos y medios eficaces para recopilar, conservar, buscar y divulgar la información, donde esta última juega un papel fundamental para el análisis en la toma de decisiones.

La información en estos tiempos ha alcanzado un valor significativo para la mayoría de las organizaciones. Con el objetivo de preservarla, se han desarrollado herramientas informáticas como son las bases de datos, las cuales posibilitan almacenar y recuperar datos de manera eficiente y rápida. Las bases de datos son utilizadas en muchas aplicaciones y ocupan un lugar importante en cualquier área del quehacer humano, comercial, y tecnológico.

Para su definición, desarrollo y manipulación se utilizan los Sistemas Gestores de Base de Datos (SGBD), cuyo principal objetivo es manejar de manera clara, sencilla y ordenada un conjunto de datos y servir de interfaz entre bases de datos, el usuario y las aplicaciones. Los SGBD además tienen el fin de mejorar las respuestas a los clientes y obtener de manera organizada y rápida la información, logrando la independencia e integridad de los datos.

Cada día, con los grandes adelantos en la informática, se han ido perfeccionando los sistemas de almacenamiento de datos, por lo que muchos de los que existían anteriormente han ido quedando obsoletos, impidiendo una buena manipulación de la información que guardan. Surge entonces la necesidad de transferir esos datos a sistemas de almacenamiento de datos actuales que cuenten con mayores funcionalidades y capacidades.

Para este tipo de transferencia es recomendable realizar primero todo un proceso de transformación y limpieza de los datos, garantizando que no exista redundancia y problemas de actualización en los mismos. Este proceso asegura la integridad en el sistema final al cual se desea salvaguardar la información.

Estas transferencias de datos son realizadas en la Universidad de las Ciencias Informáticas (UCI), que a pesar de tener pocos años de creada se ha convertido en una pequeña empresa de software que tiene entre sus necesidades traspasar o actualizar datos en nuevos sistemas.

La UCI fue creada con el objetivo de formar profesionales integrales, comprometidos con los ideales del socialismo y la revolución cubana, y tiene la misión de ser una universidad innovadora de

excelencia científica, informatizar el país y desarrollar la industria del software para contribuir al desarrollo económico de la nación cubana.

En su estructura organizativa se encuentra la Dirección de Cooperación Internacional (DCI) que es la unidad encargada de gestionar las relaciones internacionales de la universidad, proceso que se traduce en la coordinación, apoyo y promoción de las actividades relacionadas con la cooperación internacional. La DCI procura además formar alianzas con entidades académicas, las cuales se formalizan a través de convenios de cooperación, además de llevar un control de todas las visitas recibidas en la UCI, con el fin de fomentar relaciones de prestigio, pertinentes y sostenibles.

En el Centro de Informatización Universitaria (CENIA) de la UCI se viene gestando un programa para la informatización de los procesos sustantivos de la propia universidad, cuyo producto principal es el Sistema de Gestión Universitaria (SGU), dentro del cual y para mejorar los procesos que tienen lugar en la DCI, se concibió el Sistema de Gestión de Cooperación Internacional (SGCI), el cual permite al cliente trabajar de forma sencilla y eficaz, reduciendo costos, agilizando acciones y ofreciendo respuestas rápidas.

Actualmente en la universidad se fortalecen los lazos de solidaridad con países de todo el mundo, por lo que la colaboración se ha hecho evidente desde sus comienzos y se hace necesario manejar los procesos que se llevan a cabo para satisfacer esta actividad. Debido a esto la DCI es la encargada de coordinar la cooperación internacional y por tanto tramitar todo lo referente al tema. La necesidad de centralizar los datos se hace indispensable en dicha dirección, ya que una parte de la información se encuentra almacenada en diferentes formatos, provocando: lentitud en la búsqueda, demora en las gestiones a realizar, trabajo complejo y engorroso e impide llevar a cabo estadísticas reales sobre las relaciones que sostiene la UCI con otros países como parte de la colaboración, lo cual es imprescindible conocer para la toma de decisiones.

Otra parte de los datos se encuentra almacenada en el sistema "Dirección de Relaciones Internacionales (DRI)", que contiene solamente los datos del proceso de gestión de trámites y está desarrollado en SQL Server, un gestor de base de datos propietario muy costoso, que no cumple con las políticas de utilización de software libre asumidas por el país y por la universidad.

El SGCI, además de gestionar los procesos actuales de la DCI, debe realizar una gestión completa de los datos históricos recogidos en la propia dirección desde su creación en el año 2002, sin embargo, no se ha realizado aún una estrategia que permita transferir esta información hacia dicho sistema, lo cual es precisamente el objetivo principal de la investigación.

Por los motivos anteriormente expuestos se plantea el siguiente **problema científico**: ¿Cómo lograr que el SGCI cuente con los datos históricos manejados por la DCI en la UCI?

Teniendo como **objeto de estudio** la migración de datos.

Por cuanto se especifica el siguiente **campo de acción**: Proceso de migración de los datos manejados por la DCI hacia el SGCI en la UCI.

El **objetivo general de la investigación** es desarrollar la migración de los datos manejados por la DCI hacia el SGCI aplicando una estrategia que garantice que dicho sistema cuente con el historial de los datos referentes a los procesos de cooperación internacional en la UCI. Con vistas a dar cumplimiento al mismo se plantean los siguientes **objetivos específicos**:

- Elaborar el marco teórico conceptual a partir del estudio de procedimientos y herramientas existentes relacionadas con la migración de datos.
- Elegir la herramienta a utilizar para llevar a cabo la migración de los datos.
- Desarrollar una estrategia para la migración de los datos manejados por la DCI hacia el SGCI.
- Realizar un plan de pruebas para comprobar la correcta migración de los datos.
- Realizar las pruebas de integridad de la información migrada.

Para dar cumplimiento al objetivo general se proponen las siguientes **tareas de la investigación**:

- Estudio del estado del arte de procedimientos que permitan apoyar el proceso de migración de datos al Sistema Gestor de Bases de Datos PostgreSQL.
- Estudio del lenguaje plpgsql que permita utilizar funciones y realizar consultas necesarias acordes a la arquitectura de base de datos.
- Estudio del lenguaje de programación PHP.
- Estudio de alternativas para la migración de datos.
- Realización de un estudio bibliográfico que permita fundamentar las herramientas informáticas utilizadas en los procesos de migración de datos.
- Levantamiento de la información a migrar hacia el SGCI.
- Análisis de la estructura de los datos manejados por la DCI.
- Interpretación y evaluación de datos.
- Transformación y limpieza de los datos a migrar.
- Diseño de la estrategia de migración de los datos manejados por la DCI hacia el SGCI.
- Migración de los datos manejados por la Dirección de Cooperación Internacional hacia el Sistema de Gestión de Cooperación Internacional.
- Estudio de pruebas existentes para la comprobación de la integridad y calidad de los datos.
- Realización de pruebas iterativas al proceso de migración de datos.

- Diseño del plan de pruebas para la comprobación de la integridad y calidad de los datos migrados.
- Ejecución del plan de pruebas para comprobar la integridad y calidad de los datos migrados.
- Evaluación de los resultados de la migración de la información.
- Realización de reportes estadísticos que permitan validar el proceso de migración de datos.

Como **idea a defender** se plantea que la realización de la migración de los datos manejados por la DCI hacia el SGCI permitirá que dicho sistema cuente con los datos históricos referentes a los procesos de cooperación internacional que realiza la UCI desde su surgimiento.

Para el desarrollo de la investigación se utilizaron diferentes **métodos científicos**, tanto teóricos como empíricos, que se ajustan al objeto de estudio y al cumplimiento de los objetivos trazados.

### **Métodos teóricos empleados:**

- **Analítico-sintético:** Permitirá la recopilación de información sobre las migraciones de datos para ser estudiadas y llegar a una conclusión para elaborar una estrategia a seguir en la migración de los datos.
- **Histórico-lógico:** Se utilizará este método para conocer la trayectoria histórica del tema de la migración de datos, posibilitando de esta manera crear una estrategia para dar solución al problema planteado.

### **Métodos empíricos empleados:**

- **Entrevista:** La entrevista es una conversación planificada entre el investigador y el entrevistado para obtener información. Se utilizará para obtener en qué situación se encuentran los datos y entender sus especificidades para poder llegar a una solución.

Con la investigación se obtendrán los siguientes beneficios:

- Tener la información organizada y centralizada en un mismo sistema.
- Mantener la seguridad e integridad de los datos migrados.
- Garantizar que el nuevo sistema cuente con el historial de los procesos de trámites que se realizan en la universidad.

El contenido del trabajo está estructurado en tres capítulos fundamentales:

**Capítulo 1. Fundamentación teórica:** Aborda toda la fundamentación teórica referente al tema de investigación, así como un estudio de procedimientos homólogos y los lenguajes y tecnologías utilizadas para desarrollar el diseño propuesto.



**Capítulo 2. Descripción y análisis de la solución propuesta:** Este capítulo aborda la descripción de los sistemas involucrados en la migración de datos, así como las herramientas utilizadas y la estrategia diseñada para la solución propuesta.

**Capítulo 3. Validación de la solución propuesta:** En este capítulo se incluye la validación de los resultados de la migración y un resumen total de la información que será trasladada hacia el SGCI.

## Capítulo 1. Fundamentación teórica

### 1.1 Introducción

Cuba está realizando transformaciones importantes para el desarrollo de la informática. Una de ellas es el proceso de migración a software libre que es una de las tareas fundamentales de la informatización de la sociedad, la cual se lleva a cabo también en el CENIA empleando tecnologías de código abierto para el trabajo en proyectos productivos.

En el presente capítulo se hará una breve descripción de los elementos teóricos de la investigación con el objetivo de lograr una mejor comprensión del objeto de estudio, donde se abordarán diferentes conceptos estrechamente relacionados con el tema de la investigación. Además, se expondrán las principales herramientas y tecnologías a emplear para dar cumplimiento al objetivo del presente trabajo.

### 1.2 Conceptos asociados al dominio del problema

#### 1.2.1 Dirección de Cooperación Internacional

La Dirección de Cooperación Internacional es la unidad responsable de las relaciones internacionales de la UCI. Su misión es gestionar, coordinar y asesorar la cooperación internacional de la universidad con organismos e instituciones internacionales. Además procura formar alianzas con entidades extranjeras y llevar a cabo convenios de cooperación.

La DCI está compuesta por tres grupos fundamentales, el grupo de cooperación, el de trámites y el administrativo (que incluye la gestión económica y de visitas), los cuales poseen características funcionales independientes, con el objetivo de lograr una mayor especialización de cada uno de los procesos con vistas a enriquecer el producto final.

Las funciones actuales de la DCI son las siguientes:

- Dirigir metodológica y funcionalmente el Sistema de Gestión de Cooperación Internacional de la UCI.
- Promover, apoyar y dar seguimiento a los vínculos de la UCI con entidades extranjeras.
- Asesorar y representar a la UCI en organizaciones internacionales.
- Coordinar y apoyar la atención a las visitas de extranjeros a la UCI.
- Coordinar y gestionar la información para las misiones de la UCI en el exterior.
- Tramitar la información y documentación necesaria para la realización de las misiones de los trabajadores y estudiantes, así como de los extranjeros que visitan la UCI.
- Gestionar todo lo relacionado con el aseguramiento económico de las actividades de trámites y visitas. (Dirección de Cooperación Internacional, 2010)

### 1.2.2 Base de Datos

El uso de bases de datos se ha hecho indispensable en las sociedades industriales desde que fue escuchado por primera vez en un simposio celebrado en California en 1963. Existen varias definiciones de una base de datos como son:

Una base de datos es un conjunto de datos persistentes que es utilizado por los sistemas de aplicación de alguna empresa dada. (Date, 2001)

Colección o depósito de datos, donde los datos están lógicamente relacionados entre sí, tienen una definición y descripción común y están estructurados de una forma particular. Una base de datos es también un modelo del mundo real y, como tal, debe poder servir para toda una gama de usos y aplicaciones. (Arias, 2007)

De manera general, para la presente investigación, se puede definir una BD como un conjunto de datos organizados, que se encuentran agrupados, estructurados e interrelacionados entre sí, almacenados con carácter más o menos permanente en la computadora.

Una BD se compone de una o más tablas que guarda un conjunto de datos. Cada tabla tiene una o más columnas y filas. Las columnas guardan una parte de la información sobre cada elemento que se quiera guardar en la tabla, cada fila de la tabla conforma un registro.

#### **Arquitectura de las bases de datos:**

La arquitectura propuesta por el grupo del Instituto Nacional Estadounidense de Estándares-Normas de Planificación y Requisitos (ANSI/SPARC)<sup>1</sup> se divide en tres niveles: interno, lógico-global y externo o de vistas.

El nivel interno es el más cercano al almacenamiento físico, o sea, es el relacionado con la forma en que los datos están realmente almacenados.

El nivel externo es el más cercano a los usuarios, o sea, es el relacionado con la forma en que los datos son vistos por cada usuario individualmente.

El nivel lógico-global es un nivel intermedio que describe la estructura de toda la BD para una comunidad de usuarios mediante un esquema conceptual. Este esquema oculta los detalles de las estructuras de almacenamiento y describe entidades, atributos, relaciones, operaciones de los usuarios y restricciones. (Mato, 1999)

Existen varias clasificaciones de BD y entre ellas podemos mencionar la relacional, donde este modelo es el que ha marcado la línea de investigación por muchos años y tiene sus orígenes en el año 1970

---

<sup>1</sup>ANSI/SPARC es un grupo de normalización creado en 1969 para estudiar el impacto de los SGBD en los sistemas de información.

cuando E.F. Codd de IBM<sup>2</sup> introdujo la idea en un documento titulado “*A Relational Model of data for Large Shared Banks*” (Un Modelo de Datos Relacional para Grandes Bancos Compartidos); antes de eso la mayoría de las BD estaban basadas en un modelo de red o una simple estructura de archivo plano.

Dentro de las principales características de una base de datos relacional se pueden mencionar las siguientes: (Mato, 1999)

- Independencia lógica y física de los datos: Los datos no dependen del programa y por tanto cualquier aplicación puede hacer uso de ellos.
- Minimización de la redundancia: Al reducir ésta al máximo se logra un mayor aprovechamiento del espacio y además se evita que existan inconsistencias entre los datos.
- Integración y sincronización: La integración consiste en garantizar una respuesta a los requerimientos de diferentes aspectos de los mismos datos por diferentes usuarios. Y la sincronización consiste en la necesidad de garantizar el acceso múltiple y simultáneo a la BD.
- Integridad de los datos: Como se evita la redundancia de datos e inconsistencia de los mismos y se manejan transacciones, la integridad está garantizada. El concepto básico de integridad es que la información obtenida de la BD es correcta en todo momento.
- Seguridad: Garantiza el acceso autorizado a los datos. Se crea una jerarquía de acceso, que permitirá o prohibirá a los usuarios hacer una u otra acción sobre la BD.
- Facilidad de manipulación de la información: Permite la capacidad de una búsqueda rápida y aísla al usuario de las complejidades del tratamiento de los datos.
- Compartición de datos: Se puede tener acceso a los datos con la última actualización prácticamente en tiempo real.

### 1.2.3 Sistemas Gestores de Base de Datos

El software que permite la utilización y/o la actualización de los datos almacenados en una (o varias) base(s) 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. (Mato, 1999)

Es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y

---

<sup>2</sup> International Business Machines es una empresa multinacional estadounidense de tecnología y consultoría. Fabrica y comercializa hardware y software para computadoras, y ofrece servicios de infraestructura, alojamiento de Internet, y consultoría en una amplia gama de áreas relacionadas con la informática, desde computadoras centrales hasta nanotecnología.

las aplicaciones. El objetivo fundamental de un SGBD consiste en suministrar al usuario las herramientas que le permitan manipular, en términos abstractos, los datos; o sea, de forma que no le sea necesario conocer el modo de almacenamiento de los datos en la computadora, ni el método de acceso empleado. (Mato, 1999)

Es una aplicación que permite a los usuarios definir, crear y mantener la BD y proporciona un acceso controlado a la misma, asegurando su integridad, confidencialidad y seguridad.

Por tanto debe permitir:

- Definir una base de datos: especificar tipos, estructuras y restricciones de datos.
- Construir la base de datos: guardar los datos en algún medio controlado por el mismo SGBD.
- Manipular la base de datos: realizar consultas, actualizarla, generar informes.

## **Clasificación de los SGBD.**

Los SGBD se clasifican en libres o propietarios.

### **SGBD libres:**

- PostgreSQL
- FireBird
- SQLite

### **SGBD propietarios:**

- Oracle (Oracle Corp.)
- MySQL (SUN Microsystems)
- DB2 (IBM)
- InterBase (CodeGear)
- MaxDB (SAP)
- JavaDB (SUN Microsystems)
- SQL Server (Microsoft Corp.)
- Infomix (IBM)

## **1.2.4 SQL Server**

SQL Server es un sistema para la gestión de BD creado por Microsoft, el mismo se basa en el modelo relacional. SQL Server utiliza como lenguajes de consulta T-SQL y ANSI SQL. Entre sus principales características se encuentran:

- Facilidad de instalación, distribución y utilización: incluye un conjunto de herramientas administrativas y de desarrollo que mejora el proceso de instalación, distribución, administración y uso de SQL Server en varios sitios. (Alma Carrillo, 2010)
- Características de bases de datos corporativas: protege la integridad de los datos a la vez que minimiza la carga de trabajo que supone la administración de miles de usuarios modificando la base de datos simultáneamente. (Alma Carrillo, 2010)
- Soporta procedimientos almacenados.
- Incluye también un entorno gráfico de administración, que permite el uso de comandos DDL<sup>3</sup> y DML gráficamente.
- Permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y los terminales o clientes de la red sólo acceden a la información.

### 1.2.5 Migración de datos

La migración de datos constituye el proceso por el cual grandes volúmenes de datos son trasladados desde sistemas existentes hacia nuevos sistemas. (Consuelo Inés González Lau, 2007)

Existen diversos motivos para hacer una migración, tales como: mejorar el desempeño de la BD, cumplir con nuevos requerimientos de usuario, de la aplicación o políticas de seguridad; así como la compatibilidad con otras aplicaciones, la actualización de versiones, la estandarización de la tecnología de información, la reducción de costos que se puede tener al cambiar por software libre y nuevos procesos de negocio.

Durante la migración se realizan procesos de extracción, transformación y carga, los cuales permiten obtener los datos desde su origen, modificarlos para cumplir con la integridad y la consistencia e insertarlos finalmente en la BD destino.

### 1.2.6 Concepto de estrategia

Existen diversas definiciones referentes al concepto de estrategia, entre ellas podemos mencionar:

Henry Mintzberg define el término estrategia como “*el patrón de una serie de acciones que ocurren en el tiempo*”. (Henry Mintzberg, 1997)

La estrategia es un plan, una especie de curso de acción conscientemente determinado, una guía para abordar una situación específica. (Cruz, 2006)

Teniendo en cuenta los conceptos anteriormente expuestos se puede decir que una estrategia es un

---

<sup>3</sup> DDL (Data Definition Language): Lenguaje de Definición de Datos.

conjunto de acciones planificadas sistemáticamente en el tiempo que se llevan a cabo para lograr un determinado fin. En otras palabras, una estrategia es el proceso seleccionado a través del cual se prevé alcanzar un cierto estado futuro.

## 1.2.7 Tendencias actuales de la migración de datos

La migración de datos se realiza generalmente cuando una empresa u organización decide cambiar el sistema que ha venido empleando, por razones económicas, o cuando desea sustituir el SGBD que utiliza en ese momento. El proceso de migración de datos es una operación muy compleja. No cabe duda de que el proceso no es directo y es preciso planificarlo con cuidado para que progrese sin problemas ni contratiempos. A nivel técnico, pocas aplicaciones cuentan con herramientas potentes para migraciones y en la mayoría de los casos, la tarea suele realizarse construyendo los procesos directamente contra la BD. Además se requiere de un equipo de trabajo con avanzados conocimientos sobre el tema, y muchas veces dejan de estar disponibles las aplicaciones debido a que se encuentran en estos procesos migratorios.

A fin de garantizar que la migración de datos sea efectiva, es esencial diseñar una estrategia que permita mover datos de forma segura y con la calidad requerida.

## 1.2.8 Pruebas de migración

Este tipo de pruebas se realizan para brindar apoyo al proceso de migración de datos, verificando que los mismos cumplen con las especificaciones definidas en el lugar de destino. Normalmente incluye la maduración de datos, lo cual implica verificar el comportamiento de los datos migrados al nuevo sistema, prueba que se realiza después de haber certificado la estabilidad funcional del sistema.

Las pruebas funcionales de la migración de datos, determinan que la información sea traspasada con éxito al nuevo sistema.

Entre las pruebas recomendadas se encuentran:

- Verificación de las reglas de dominio de las columnas: Ciertas columnas sólo deberían permitir ciertos valores.
- Verificación de las reglas de valores por defecto de las columnas: Si no se suministra un valor se coloca uno por defecto.
- Verificación de las reglas de existencia de valores: Las columnas no pueden tener valores nulos.
- Verificación de las reglas de los valores de una fila: El valor de un dato en una fila puede depender de otro valor en la misma fila. (Javier Lamas Collejo, 2010)

### 1.3 Estudio de procedimientos de migración de datos

Realizar un estudio de procedimientos que apoyan la migración de datos tiene como objetivo principal tomar experiencias en cuanto al funcionamiento de los procesos que llevan a cabo los sistemas actuales en el mundo y al mismo tiempo analizar en qué medida los procedimientos encontrados ayudan en la solución de la problemática planteada.

#### 1.3.1 Procedimiento aplicado en Innovative System, Inc.

La migración de datos de aplicaciones y bases de datos antiguas a nuevas es un subproducto necesario del crecimiento organizacional y de la adaptación a las cambiantes necesidades de negocios. Pero el mezclar datos con diferentes formatos, estructuras y parámetros, continúa siendo una de las prácticas más riesgosas de las áreas de Tecnología de Información, resultando frecuentemente en grandes retrasos, costos elevados más de lo pronosticado e inclusive en fallas completas del proyecto. Innovative utiliza el sistema de perfilamiento (“profiling”) automático de datos y las soluciones de principio-a-fin para la integración y manejo de calidad de datos. En este proceso se pueden llevar a cabo pasos como:

- Realizar una inspección a los datos para identificar formatos inconsistentes, estructuras de datos dañadas, información faltante y otras anomalías que corrompen la calidad de los mismos.
- Usar los resultados del perfilamiento de datos, escribir especificaciones de mapeo para transformar los datos que ayudarán a asegurar una conversión exitosa la primera vez.
- Conducir una prueba de conversión usando una muestra de datos estadísticamente significativa para confirmar que las especificaciones de mapeo están correctas antes de aplicarlas al universo de los datos.
- Limpiar, estandarizar, identificar clientes duplicados y enlazar clientes relacionados durante el proceso de transformación para maximizar la exactitud y eliminar duplicidades.
- Implementar procedimientos para monitorear y mejorar la calidad de datos en una forma continua. (Innovative Systems, Inc, 2005)

#### 1.3.2 Migración integrada de datos para el Sistema Nacional de Salud.

El Centro Nacional de Genética Médica (CNGM) tiene dentro de sus funciones las investigaciones de la genética médica.

En la actualidad la Genética Médica enfrenta un grupo de dificultades asociado al crecimiento del volumen de información que generan los estudios genéticos y el descubrimiento vertiginoso del genoma humano, y en este sentido Cuba a través del CNGM ha desarrollado diversos estudios en consonancia con esta tendencia mundial. En este centro el volumen de datos histórico es valioso y no



debe perderse por lo que se llevó a cabo una migración de los datos al Sistema Informático de Genética Médica (alásMEDIGEN), que resuelve entre otros problemas el intercambio de información entre los genetistas.

Para esto se puso en práctica el desarrollo de una propuesta del proceso de migración de los datos que permitió que el sistema nuevo funcione reutilizando la información del sistema origen.

A continuación se describen algunos de los procesos empleados:

- Se revisó el diseño de la estructura de las tablas.
  - Se revisó si el diseño de la estructura de las tablas origen se podían migrar al diseño de la estructura de las tablas nuevas, elaborándose un documento donde se detalló por cada tabla origen y campo si era posible migrarlo, y se detalló la tabla destino con los campos que contiene.
  - Se revisó la calidad de la información de las tablas viejas, para desechar la información que no fuese valiosa, y no era necesario migrar.
  - Se realizó un script para el traspaso de la información de una BD a otra.
  - Se diseñaron nuevas tablas para poder realizar el traspaso de los datos de unas tablas a otras.
- (Lucía E. Domínguez Abreu, 2011)

### **1.3.3 Migración de la información del Sistema de Gestión Académica, Akademos, al Sistema de Gestión Académica de Pregrado.**

Akademos contiene información de gran importancia que constituye el grosor histórico de la universidad, pues almacena todos los datos académicos de los estudiantes que han cursado en la misma.

En el CENIA de la UCI se realizó una migración de toda la información contenida en la BD de Akademos para el nuevo modelo de datos del Sistema de Gestión Académica de Pregrado, debido a que con el surgimiento del mismo, era necesario que éste contara con todos los datos referente a los estudiantes que se encontraban almacenados en la BD del anterior Sistema de Gestión Académica, garantizando así la continuidad de la gestión de los procesos involucrados en la formación docente en la universidad. (Javier Lamas Collejo, 2010)

Para llevar a cabo esta migración de Akademos al Sistema de Gestión Académica de Pregrado se pusieron en marcha una serie de fases de la migración de datos, para luego poder aplicar los procedimientos o estrategias establecidas para llegar al resultado final.

En la estrategia para la migración de los datos se realizó un análisis de los sistemas, la descripción de los campos con los que contaba cada sistema, más tarde se realizó la migración haciendo uso de una

herramienta de migración de datos y luego una descripción de la cantidad de datos trasladada de un sistema a otro. Además se concluyó la migración con reportes de los datos migrados y no se tuvo en cuenta la fase de pruebas y cargas reales debido a que el sistema Pregrado estaba aún en fase de desarrollo.

### **1.4 Aportes obtenidos de los procedimientos estudiados:**

Después de realizado un estudio de estos procedimientos, se arribó a la conclusión de que existe una tendencia común en la forma de trabajo para llevar a cabo una migración de datos, en la que se realizan una serie de pasos que guían dicho proceso. Cada uno de ellos fue exitoso en su objetivo, sin embargo, la problemática a la que dan solución no es la misma que la existente en la presente investigación. Es por esto que sólo se tendrán en cuenta los aspectos positivos que ayuden a diseñar una estrategia de migración que permita resolver el problema en cuestión.

### **1.5 Herramientas informáticas utilizadas en los procesos de migración de datos.**

#### **1.5.1 Kettle**

Es una de las herramientas libres más antiguas, tiene una gran cantidad de usuarios y una nueva dirección por parte del soporte técnico de Pentaho<sup>4</sup>. Permite implementar los procesos de extracción, transformación y carga de datos (ETL). El uso de Kettle posibilita evitar grandes cargas de trabajo manual frecuentemente difícil de mantener y de desplegar.

Con Kettle se pueden realizar diversas tareas, entre ellas cabe resaltar:

- Exportar de bases de datos a ficheros u otras bases de datos.
- Importar en bases de datos ficheros en formato Excel o texto.
- Migración de datos entre diferentes bases de datos.
- Explotación de los datos existentes en bases de datos (tablas, vistas).
- Limpieza de datos aplicando transformaciones de datos con condiciones complejas. (Kettle, 2007)

#### **1.5.2 Talend Open Studio**

Es una herramienta de código abierto de integración de datos. Ofrece soluciones de calidad de datos, completamente complementarias con sus soluciones de integración. (Cutro, 2009)

---

<sup>4</sup>Es una plataforma de código abierto para la inteligencia de negocios.

Talend Open Studio es una herramienta con licencia GPL<sup>5</sup> que permite de forma visual modelar transformaciones de datos. Se compone de tres aplicaciones principales: Modelado de negocios, Diseño de trabajo y Administración de Metadatos dentro de un entorno de desarrollo gráfico individual. (Talend Open Studio, 2010)

Es multiplataforma pues se puede ejecutar en sistemas operativos como Windows y Linux.

Los usos de Talend Open Studio incluyen:

- Migración.
- Transformación de datos.
- Integración de datos.
- Calidad de datos.
- Procesamiento de los datos en el servidor origen.
- Está basada en los procesos ETL.
- Tiene soporte a través de la Wiki de Talend y foro de Talend.
- Gran disponibilidad de componentes para conexión a múltiples sistemas y orígenes de datos como por ejemplo archivos CSV y Excel, y bases de datos de tipo MS SQL Server y PostgreSQL.
- Generación continua de versiones.

Con su probado rendimiento, facilidad de uso, extensibilidad y robustez, las soluciones de integración de Talend son muy utilizadas y desplegadas en todo el mundo.

### 1.5.3 AdventNet SwisSQL

AdventNet SwisSQL ofrece una herramienta que ayuda a la migración de las estructuras y de los datos de base de datos como Oracle, PostgreSQL y SQL Server.

#### Características:

- Migración de datos de forma rápida asegurando la integridad de los mismos.
- Realiza la migración de datos desde documentos Excel y CSV hacia una BD.
- Migra tablas, índices y restricciones junto con los datos.

---

<sup>5</sup>Licencia Pública General creada por la Federación de Software Libre en 1989 (la primera versión), y está orientada principalmente a proteger la libre distribución, modificación y uso de software.

- Soporta conectividad JDBC<sup>6</sup>.
- Suspende o reanuda la migración en cualquier momento.
- Fácil de usar e intuitiva interfaz gráfica de usuario.
- Ofrece un excelente soporte a sus clientes desde la evaluación del producto hasta la finalización con éxito de su migración. (SwisSQL, 2009)

### 1.5.4 Softek Transparent Data Migration Facility UNIX (IP)

Softek Transparent Data Migration Facility UNIX (IP) es un producto de migración de datos diseñado para trasladar datos sin interrupciones, en remoto con entornos UNIX.

#### Características:

- Realiza migraciones de datos en segundo plano mientras mantiene la disponibilidad de las aplicaciones.
- Excelente para migraciones de datos en remoto a través de la red.
- Ayuda a garantizar la integridad de los datos y permite realizar copias puntuales de datos para que estén disponibles en cualquier momento tras la sincronización inicial.
- Ofrece monitorización centralizada, control y gestión de migraciones a través de una consola común que gestiona entornos Softek TDMF Windows (IP), TDMF UNIX (IP), Softek Replicator UNIX, y Softek Replicator Windows.
- Ofrece soporte para migraciones en múltiples sistemas operativos como Red Hat Enterprise Linux, AIX, HP-UX y Solaris. (IBM, 2008)

### 1.5.5 Ora2Pg

Ora2Pg es un módulo de Perl para exportar una BD Oracle a un esquema de PostgreSQL. Esta aplicación se conecta a la BD Oracle, extrae su estructura, y genera un script SQL que puede cargar en la base de datos PostgreSQL.

La misma hace un respaldo de todo el esquema (tablas, vistas, secuencias, índices, privilegios) con todas las llaves ya sean primarias, foráneas y únicas exportadas a código PostgreSQL sin necesidad de editar el SQL generado. Posee una extensa guía de migración muy intuitiva en su sitio oficial. (Marcos Ortíz Valmaseda, 2009)

---

<sup>6</sup> Java Database Connectivity: permite la ejecución de operaciones sobre bases de datos desde el lenguaje de programación Java, independientemente del sistema operativo donde se ejecute o de la base de datos a la cual se accede, utilizando el dialecto SQL del modelo de base de datos que se utilice.

El Centro de Tecnologías de Gestión de Datos (DATEC) de la UCI adaptó a sus necesidades esta herramienta y creó la nueva herramienta (Ora2PgMigration), la cual está escrita en Python 2.5 utilizando el marco de trabajo de desarrollo multiplataforma Qt4 liberado bajo la licencia GPL, y como base la aplicación Ora2Pg por las excelentes ideas que promueve y por la experiencia de sus desarrolladores.

### **1.6 Resultados del estudio de las herramientas de migración de datos**

Luego del estudio de algunas de las herramientas que realizan procesos de migración de datos, se propone utilizar Talend Open Studio debido a las facilidades de uso que brinda. Además, cumple con las políticas establecidas por el CENIA al ser una herramienta libre. Cuenta con un foro donde se le da solución a los problemas que se pueden presentar en el momento de usar esta herramienta, lo que hace posible obtener soporte técnico de manera gratuita. Permite migrar datos desde documentos Excel y desde una BD de tipo SQL Server hacia otra en PostgreSQL. En el CENIA, lugar donde se desarrolla la presente investigación, se probó su funcionamiento en un proceso de migración de datos de manera exitosa, por lo que existen especialistas con amplio dominio y experiencia a los cuales acudir para obtener asistencia técnica. Otro aspecto importante es que cuenta con una herramienta para el análisis de la calidad de los datos, la cual lleva por nombre Talend Open Profiler. Esta herramienta permite examinar diferentes fuentes de datos (por ejemplo archivos CSV y bases de datos de tipo PostgreSQL y SQL Server) y recoger las estadísticas y la información sobre los datos que contienen para luego mostrar los resultados mediante informes o gráficos, lo cual será de utilidad en la comprobación de la calidad de los datos migrados.

### **1.7 Herramientas y tecnologías a utilizar**

#### **1.7.1 Gestor de base de datos PostgreSQL 8.4 o superior**

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD<sup>7</sup> y con su código fuente disponible libremente. (Martínez, 2010)

PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en lugar de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando. (Martínez, 2010)

---

<sup>7</sup> Berkeley Software Distribution o BSD (en español, "distribución de software berkeley"). Es una licencia de software libre permisiva flexible respecto a la distribución, de modo que el software pueda ser redistribuido como software libre o software propietario, siendo libre la licencia original del autor.

Diseñado para ambientes de alto volumen, PostgreSQL incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, triggers (disparadores), reglas, procedimientos almacenados, secuencias, relaciones y vistas.

Posee una gran escalabilidad, haciéndolo idóneo para su uso en sitios Web que atienden un gran número de solicitudes.

Soporta varios lenguajes (PHP, C, C++, Perl y Python), protocolo de comunicación encriptado por SSL<sup>8</sup>, y también integridad referencial, la cual es utilizada para garantizar la validez de los datos de la BD.

Ejemplos de su avanzada funcionalidad son consultas SQL declarativas, control de concurrencia multi-versión, soporte multi-usuario, transacciones, optimización de consultas y arreglos. PostgreSQL se caracteriza por ser un sistema estable, de alto rendimiento, gran flexibilidad, gran compatibilidad, además tiene características que permiten extender fácilmente el sistema. Además cuenta con una documentación muy bien organizada, pública y libre. (González, 2012)

### 1.7.2 PgAdmin III

PgAdmin III es una aplicación gráfica con licencia de código abierto para administrar bases de datos PostgreSQL. Está escrita en C++ usando la librería gráfica multiplataforma wxWidgets, lo que permite que se pueda usar en Linux, FreeBSD, Solaris, Mac OS X y Windows. Es capaz de gestionar versiones a partir del PostgreSQL 7.3 ejecutándose en varias plataformas, así como versiones comerciales de PostgreSQL como Pervasive Postgres, EnterpriseDB, Mammoth Replicator y SRA PowerGres. (PgAdmin, 2010)

Es una herramienta diseñada para responder a las necesidades de todos los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas. La interfaz gráfica soporta todas las características de PostgreSQL y facilita la administración. La aplicación también incluye un editor SQL con resaltado de sintaxis, un editor de código de la parte del servidor y un agente para lanzar scripts programados. La conexión al servidor puede hacerse mediante conexión TCP/IP o Unix Domain Sockets (en plataformas \*nix), y puede encriptarse mediante SSL para mayor seguridad. (PgAdmin, 2010)

Es posible combinar el uso de esta herramienta con el Visual Paradigm sirviendo una de apoyo a la otra, además permite la creación y visualización de diagramas entidad-relación. PgAdmin está

---

<sup>8</sup> Secure Sockets Layer (SSL; en español «capa de conexión segura»), protocolo criptográfico que proporciona comunicaciones seguras por una red.

desarrollado por una comunidad de expertos en PostgreSQL y está disponible en varios idiomas. (PgAdmin, 2010)

### 1.7.3 Entorno de desarrollo integrado NetBeans 6.9.1 o superior

Netbeans IDE es un entorno de desarrollo integrado visual, hecho principalmente para aplicaciones programadas mediante Java. Además, ofrece soporte para desarrollar en otros lenguajes de programación tales como: C, C++, Fortran, Perl, Python, PHP.

Es una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Existe además un número importante de módulos para extender el Netbeans IDE. Netbeans IDE es un producto libre y gratuito sin restricciones de uso. (Netbeans, 2012)

Consta de una gran comunidad de usuarios en constante crecimiento, lo que le ha permitido, al igual que a otros sistemas libres, el progreso paulatino de sus prestaciones.

Netbeans permite que las aplicaciones se desarrollen a partir de un conjunto de módulos o componentes de software. Brinda una barra de navegación para el acceso rápido a funciones en una clase muy extensa, detecta errores de sintaxis en tiempo real, además de un completamiento de código fuente eficiente y seguro.

Con Netbeans IDE no solo es posible elaborar potentes aplicaciones para el escritorio, también para la web y para dispositivos portátiles.

### 1.7.4 Marco de trabajo CodeIgniter

CodeIgniter es un *framework* (marco de trabajo) de código abierto desarrollado en PHP, libre de uso para el desarrollo de aplicaciones Web, ofreciendo una serie de librerías que facilitan el trabajo. Marca una manera específica de codificar las páginas Web y clasificar sus diferentes scripts, que sirve para que el código esté organizado y sea más fácil de crear y mantener. CodeIgniter implementa el proceso de desarrollo Modelo Vista Controlador (MVC), que es un estándar de programación de aplicaciones. (Álvarez, 2009)

#### Características generales de CodeIgniter 1.7.3

Los puntos más interesantes en este marco de trabajo, sobre todo en comparación con otros productos similares, son los siguientes:

- Versatilidad: CodeIgniter es capaz de trabajar en la mayoría de los entornos o servidores, incluso en sistemas de alojamiento compartido.
- Compatibilidad: CodeIgniter es compatible con la versión PHP 4 y PHP 5.
- Facilidad de instalación: No es necesario más que una cuenta de FTP para subir CodeIgniter al servidor y su configuración se realiza con apenas la edición de un archivo.

- **Flexibilidad:** Codelgniter es bastante menos rígido que otros marcos de trabajo. Define una manera de trabajar específica. Algunos módulos como el uso de plantillas son totalmente opcionales. Esto ayuda muchas veces a que la curva de aprendizaje sea más sencilla al principio.
- **Ligereza:** El núcleo de Codelgniter es bastante ligero, lo que permite que el servidor no se sobrecargue interpretando o ejecutando grandes porciones de código. La mayoría de los módulos o clases que ofrece se pueden cargar de manera opcional.
- **Documentación:** La documentación de Codelgniter es fácil de seguir y de asimilar, porque está escrita en modo de tutorial.
- **Accesibilidad:** Es lo más destacable de Codelgniter ya que se puede utilizar en varios entornos. (Lozano, 2008)

A partir de la utilización de Codelgniter, el equipo de arquitectura del CENIA propuso el nuevo marco de trabajo GUUD<sup>9</sup>. El mismo integra a su vez Codelgniter 1.7.3 y jQuery en una sola infraestructura, razón por la cual engloba las características de ambos. En esta integración se incluyen además un conjunto de novedades o mejoras y algunas modificaciones hechas específicamente al Codelgniter.

### 1.7.5 Talend Open Profiler

Talend Open Profiler es una aplicación muy útil, diseñada para ofrecer a los usuarios una herramienta de generación de perfiles de datos sofisticada, pero fácil de usar. El programa puede definir contenido, estructura y calidad de estructuras de datos muy complejas. Analiza los datos de forma continua y examina los cambios en los datos de origen con el tiempo para ayudar a mejorar la calidad de los mismos. La herramienta nos permite, a partir de los gráficos de resultados realizados, acceder a los registros específicos asociados, lo que ayuda a corregir los errores que puedan existir en la información contenida en los campos de la BD en análisis. Ofrece la posibilidad de navegar por nuestros esquemas de tablas en la BD y además realizar una serie de análisis como:

- **Overview Analysis** (Información general sobre el análisis): analiza los esquemas de base de datos, devolviendo información sobre tablas, filas, número de registros, índices, etc.
- **Table Analysis** (Análisis de la tabla): análisis sobre la definición de una tabla, verificaciones de dependencias, etc.

---

<sup>9</sup> Nombre del marco de trabajo conformado con las iniciales de los departamentos del CENIA: Gestión Universitaria, Universidad Digital y Gestión Documental.



- **Column Analysis** (Análisis por columna): análisis específico sobre el contenido de un campo. Para cada campo, se pueden seleccionar los indicadores de análisis (tales como valores estadísticos, número de registros, valores nulos, valores duplicados, valores únicos, etc.).

### 1.7.6 Lenguaje de programación

Un lenguaje de programación es un idioma artificial que puede utilizarse para definir una secuencia de instrucciones para su procesamiento por un ordenador o computadora. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana. (EcuRed, 2011)

Está formado de un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones.

#### 1.7.6.1 PHP 5.3

PHP<sup>10</sup>, es un lenguaje de código abierto muy popular especialmente adecuado para desarrollo Web y que puede ser incrustado en HTML. (PHP, 2012)

Es un lenguaje multiplataforma. Tiene capacidad de conexión con varios de los manejadores de BD que se utilizan en la actualidad como MySQL, PostgreSQL, mSQL, Oracle, dbm, filepro, Hyperwave, Informix, InterBase y Sybase.

Posee una biblioteca nativa de funciones amplia e incluida y no requiere definición de tipos de variables, además tiene manejo de excepciones. Es independiente del navegador, sin embargo, para que sus páginas PHP funcionen el servidor donde están alojadas debe soportar PHP.

Generalmente es utilizado como módulo de Apache. Permite a la mayoría de los programadores crear aplicaciones complejas con un tiempo de aprendizaje muy corto. Es libre y de código abierto. Utiliza programación por procedimientos, programación orientada a objetos (POO), o una mezcla de ambas. (PHP, 2012).

### 1.7.7 Lenguaje de consulta SQL

SQL (*Structured Query Language*) es un lenguaje de consulta estructurado de alto nivel, no procedural, normalizado, que permite la consulta y actualización de los datos de BD relacionales. (López, 2003)

Una de sus características es el manejo del álgebra y el cálculo relacional permitiendo efectuar consultas con el fin de recuperar, de manera sencilla, información de interés de una BD, así como también hacer cambios sobre ella.

---

<sup>10</sup> Hypertext Pre-processor: Procesador hipertexto.

Es utilizado por los programas de aplicaciones para realizar consultas a las bases de datos y ha sido adoptado por los principales proveedores y programadores de aplicaciones de clientes. (López, 2003)

Es un lenguaje que permite una alta productividad en codificación y la orientación a objetos. Está compuesto por comandos, cláusulas, operadores y funciones de agregado. Estos elementos se combinan en las instrucciones para crear, actualizar y manipular las bases de datos.

Los mandatos de SQL se dividen en tres grandes grupos diferenciados:

- **DDL:** Es el encargado de la definición de BD, tablas, vistas e índices entre otros.  
Son comandos propios de este lenguaje:  
CREATE: Utilizado para crear nuevas tablas, columnas e índices.  
DROP: Empleado para eliminar tablas e índices.  
ALTER: Utilizado para modificar las tablas agregando columnas o cambiando la definición de las mismas.
- **DML:** Su misión es la manipulación de datos. A través de él se puede seleccionar, insertar, eliminar y actualizar datos. Es la parte que más frecuentemente se utiliza y con ella se construyen las consultas.  
Son comandos propios de este lenguaje:  
SELECT: Utilizado para consultar registros de la BD que satisfagan un criterio determinado.  
UPDATE: Utilizado para modificar los valores de los campos y registros especificados.  
INSERT: Utilizado para cargar lotes de datos en la BD en una única operación.  
DELETE: Utilizado para eliminar registros de una tabla de la BD.
- **DCL:** Encargado de la seguridad de la BD, en todo lo referente al control de accesos y privilegios entre los usuarios.  
Son comandos propios de este lenguaje:  
GRANT: Permite dar permisos a uno o varios usuarios o roles para realizar tareas determinadas.  
REVOKE: Permite eliminar permisos que previamente se han concedido con GRANT.

### 1.7.8 Visual Paradigm 8.0

Visual Paradigm para UML<sup>11</sup> (*Unified Modeling Language*) es una herramienta profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Esta herramienta de modelado ayuda a una rápida construcción

---

<sup>11</sup> Lenguaje Unificado de Modelado: Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

de aplicaciones de calidad mejores y a un menor costo. Permite construir los diferentes tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. (Visual Paradigm, 2009)

### **1.7.9 Servidor Web**

Es un programa que gestiona cualquier aplicación en el lado del servidor realizando conexiones bidireccionales y/o unidireccionales y síncronas o asíncronas con el cliente generando una respuesta en cualquier lenguaje o aplicación en el lado del cliente. El código recibido por el cliente suele ser compilado y ejecutado por un Navegador Web. Para la transmisión de todos estos datos se utiliza algún protocolo, generalmente HTTP<sup>12</sup>. (EcuRed, 2011)

#### **1.7.9.1 Servidor Web Apache 2.2**

Apache 2.2 es una plataforma de servidores Web de código fuente abierto. Tiene amplia aceptación en la red y es considerado el servidor Web por excelencia. Además brinda algunas ventajas que son fundamentales para el desarrollo Web como son: fiabilidad, software libre y extensibilidad. Es modular, por lo que puede ser adaptado a diferentes necesidades y entornos.

Presenta mensajes de error altamente configurables. Proporciona una gran variedad de Módulos de Multiprocesamiento (MPM) que permiten que se ejecute en un proceso basado en híbridos (procesos y subprocesos), para adaptarse mejor a las exigencias de cada infraestructura particular. Fue diseñado para reducir la latencia y aumentar el rendimiento. Es un software libre y multiplataforma que permite a clientes o instituciones construir sistemas confiables con fines experimentales o para resolver un problema específico de la organización. Es fácil conseguir información y soporte de Apache. Está en constante actualización y puede agregar funcionalidad a medida que esté disponible.

La licencia Apache es una descendiente de la licencia BSD, no es GPL. Esta licencia permite hacer lo que se quiera con el código fuente siempre que se le reconozca su trabajo. (Ciberaula, 2010)

Apache permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Además, es posible configurar Apache para que ejecute un determinado script cuando ocurra un error en concreto.

### **1.7.10 Sistema operativo Ubuntu 11.10**

Ubuntu es una distribución Linux que ofrece un sistema operativo predominantemente enfocado a ordenadores de escritorio aunque también proporciona soporte para servidores.

---

<sup>12</sup> Hypertext Transfer Protocol: Protocolo de Transferencia de Hipertexto.

Basado en Debian GNU/Linux, Ubuntu concentra su objetivo en la facilidad de uso, la libertad en la restricción de uso, los lanzamientos regulares (cada 6 meses) y la facilidad en la instalación. Ubuntu es patrocinado por Canonical Ltd., una empresa privada fundada y financiada por el empresario sudafricano Mark Shuttleworth.

## **Características:**

- Los desarrolladores de Ubuntu se basan en gran medida en el trabajo de las comunidades de Debian y GNOME.
- Las versiones estables se liberan cada 6 meses y se mantienen actualizadas en materia de seguridad hasta 18 meses después de su lanzamiento.
- El entorno de escritorio oficial es Gnome y se sincronizan con sus liberaciones.
- El navegador Web oficial es Mozilla Firefox.
- El sistema incluye funciones avanzadas de seguridad y entre sus políticas se encuentra el no activar, de forma predeterminada, procesos latentes al momento de instalarse. Por eso mismo, no hay un firewall predeterminado, ya que no existen servicios que puedan atentar a la seguridad del sistema.
- Para labores y tareas administrativas en terminal incluye una herramienta llamada sudo (similar al Mac OS X), con la que se evita el uso del usuario root (administrador).
- Mejora la accesibilidad y la internacionalización, de modo que el software está disponible para tanta gente como sea posible. (LinuxZone, 2010)

## **1.8 Conclusiones parciales**

En este capítulo se realizó un análisis de varios procedimientos homólogos y herramientas de migración de datos, obteniéndose como resultado un amplio conocimiento sobre las características fundamentales que debe tener todo proceso de migración de datos y la utilización de Talend Open Studio como herramienta para la realización de este proceso, teniendo en cuenta la problemática existente en la DCI de la UCI. Se utilizará el lenguaje de programación PHP 5.3 para realizar algoritmos que permitan la migración de datos y el lenguaje SQL para consultas en la BD. Como servidor Web Apache 2.2, PostgreSQL 8.4 como gestor de bases de datos, como entorno de desarrollo Netbeans 6.9.1 y como cliente de base de datos PgAdmin III. Todos estos utilizados sobre el sistema operativo Linux en la distribución Ubuntu 11.10.

### Capítulo 2. Descripción y análisis de la solución propuesta

#### 2.1 Introducción

Actualmente muchas organizaciones llevan a cabo la migración de datos para preservar información histórica actualizada en un sistema y utilizarla en la elaboración de estadísticas reales y en la toma de decisiones de un conjunto de personas. Esta no es una tarea simple por lo que requiere un buen diseño de una estrategia que permita que se realice con la calidad requerida y sin pérdida de información.

En el siguiente capítulo se exponen las principales características de los sistemas involucrados, así como también se explican los pasos que se llevarán a cabo en la estrategia de migración que se propone para dar solución a la problemática por la cual surge la presente investigación.

#### 2.2 Características de los sistemas involucrados

##### 2.2.1 Sistema de Gestión de Cooperación Internacional

El Sistema de Gestión de Cooperación Internacional es una aplicación Web que tiene como punto de acceso principal la integración, gestión, y organización de los procesos de colaboración internacional en la UCI. Este sistema permitirá la gestión de procesos que incluyen:

- Intenciones de colaboración.
- Trámites para entrada y salida al exterior.
- Atención a visitas.
- Gestión de gastos por concepto de: colaboración, trámites y visitas.
- Generación de reportes.
- Gestión de notificaciones.
- Gestión de alertas.

Permite además la participación activa de decanos, rector, grupo de cooperación internacional, especialistas de trámites y de economía, y otros directivos, agilizando así los mecanismos propios de la gestión de misiones y visitas. Está compuesto por los siguientes módulos:

- Cooperación: Permite gestionar las intenciones de colaboración de la universidad, la solicitud de misiones de proyecto, cooperación y proyectos especiales, así como también la gestión de membresías y becas.
- Trámite: Permite gestionar todos los trámites migratorios de los misioneros de la universidad.
- Visita: Permite gestionar la atención a visitas en la universidad.
- Economía: Permite gestionar los gastos por concepto de trámites, intenciones de colaboración y visitas.

- Reportes: Permite generar reportes relacionados con todos los procesos de la DCI.
- Notificaciones: Gestiona las notificaciones y alertas del sistema.

El sistema está desarrollado bajo el gestor de base de datos PostgreSQL e implementado sobre el marco de trabajo CodeIgniter, trabajando con el patrón Modelo-Vista-Controlador (MVC) para una mejor organización del trabajo a realizar. La BD está compuesta por: 59 tablas en el módulo de cooperación, 28 tablas en trámites, 23 en visita y 13 tablas en economía.

### 2.2.2 Dirección de Relaciones Internacionales (DRI)

El DRI es un sistema basado en el gestor de BD SQLServer. Está implementado para el módulo de trámites a misioneros pero no abarca todos los procesos y flujos precisos que se requieren en el negocio. Este sistema sólo es utilizado cuando se desea introducir alguna información como por ejemplo: los datos personales de un misionero, y para generar documentos que le sean necesarios al grupo de especialistas que gestionan los trámites.

## 2.3 Migración de datos

### 2.3.1 Etapas de la migración de datos

Existen varias etapas a seguir en todo proceso de migración de datos entre las cuales se destacan las siguientes:

- **Levantamiento de información:** Contempla la definición de la información que requiere el nuevo sistema y se identifica en la fuente de origen.
- **Análisis de Calidad:** Se verifica la calidad de los datos, con el propósito de establecer los tiempos reales de las etapas de transformación, limpieza y mapeo de datos.
- **Transformación/Mapeo de datos:** Contempla la definición de los diccionarios con las reglas de transformación y el mapeo de campos entre las fuentes de origen y destino.
- **Limpieza de datos:** Contempla la depuración de los datos en las fuentes de origen.
- **Empleo del ETL (Extractor, Transformador y Cargador):** Se realiza el empleo de extracción, transformación y carga de los datos.
- **Pruebas:** Se realizan las pruebas funcionales de la migración, determinando que la información sea traspasada con éxito al nuevo sistema.
- **Cargas reales:** Se realiza la migración real de los datos desde el antiguo sistema al nuevo sistema.
- **Revisión de resultados:** Se realiza la revisión final de la información cargada al nuevo sistema. (Herrera, 2007)

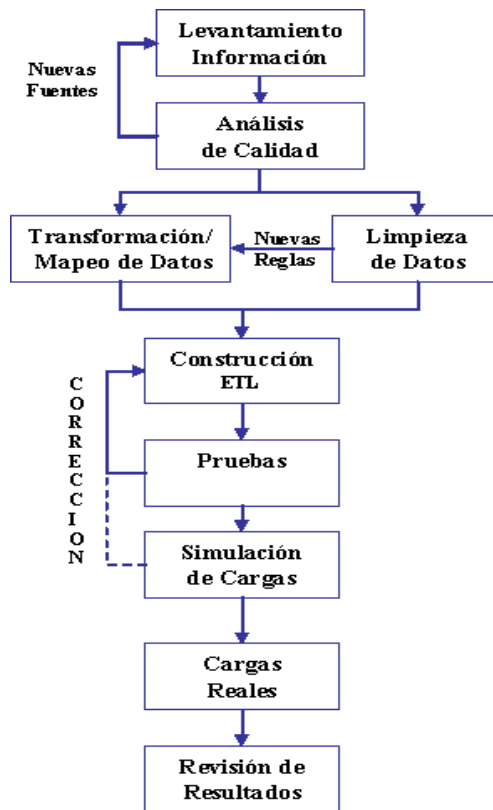


Fig 1. Etapas de la migración de datos (Herrera Jaque, 2007).

### 2.3.2. Técnicas de migración de datos

Para lograr una migración exitosa hay que tener en cuenta determinadas técnicas, entre las cuales se encuentran:

- **Planeación:** Este proceso de planeación y análisis del trabajo puede llevarse algún tiempo adicional, pero esta fase es la de mayor importancia para la migración de datos, repercutiendo en el desarrollo exitoso de la misma.
- **Contador de registros:** Si la migración se realiza de forma manual mediante alguna consulta de inserción es recomendable inicializar un contador para cada registro insertado con éxito y otro para los no insertados, así obviamente la suma de ambos debe ser igual a los registros originales.
- **Mapeado de tipos de datos:** Existen determinadas plataformas que no soportan algunos tipos de datos, así que es necesario planificar el mapeo de los campos en la nueva BD.
- **Restricciones y disparadores:** Antes de iniciar la migración de la BD, es recomendable

deshabilitar los triggers y restricciones que puedan generar error al momento que el SGBD ejecute el proceso de escritura de los datos.

- **Codificación de caracteres:** Cuando el copiado se efectúa de forma automática, es preciso identificar la codificación de caracteres que la BD destino espera, pues así se impide la sustitución automática de caracteres o en su caso, pérdida de los mismos. (Zaragoza, 2010)

### 2.4 Propuesta de arquitectura para la migración de los datos

La arquitectura en el campo de la informática no es más que un conjunto de estructuras que proveen un esqueleto para el diseño de un sistema en general. (Tamargo, 2011)

En el proceso de migración de datos se deben identificar las fuentes, el esquema, el movimiento de los datos y su enfoque de almacenamiento de datos.

#### Fuente de datos:

Son los ficheros que se encuentran almacenados en los sistemas fuentes o carpetas que guardan la información histórica. Los datos en esta área se recogen en formatos: Word, Excel, y en tablas de una BD SQLServer. Estos datos sufrirán un proceso de extracción hacia un área de almacenamiento intermedio o temporal, donde serán preparados para la posterior migración.

#### Área temporal (AT):

Constituye el área de preparación de los datos para luego ser cargados al destino. Esta área temporal estará compuesta por tablas en un esquema público de la BD del sistema para dicho montaje, donde se realizará el proceso de transformación y limpieza de los datos para lograr que estos tengan la calidad requerida.

#### Área de datos (AD):

Esta área está compuesta por las tablas del negocio, que recogerán la información que fue migrada.

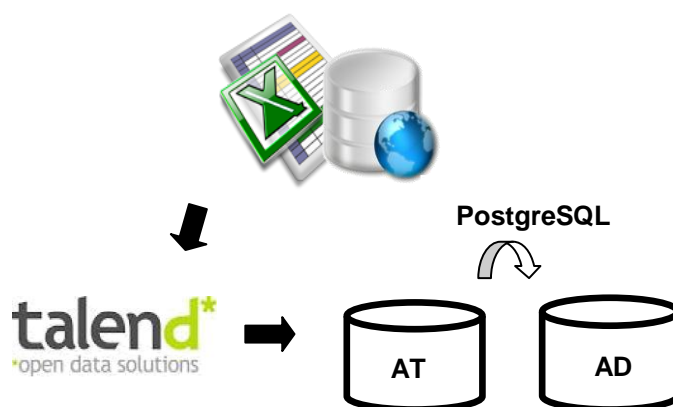


Fig 2. Arquitectura para la migración de datos.



### **2.5 Proceso de extracción, transformación y carga (ETL)**

El proceso ETL permite mover datos desde distintas fuentes, reformatearlos, limpiarlos y cargarlos en bases de datos. Es utilizado para migrar datos de una o más bases de datos a terceros y también para convertir bases de datos de un tipo o formato a otro.

#### **2.5.1 Extracción**

La primera parte del proceso ETL consiste en extraer los datos desde los sistemas de origen. En este subproceso se convierten los datos a un formato preparado para iniciar el proceso de transformación. Aquí se verifican los datos extraídos, donde se comprueba si los datos cumplen con lo que se espera y se adaptan al formato estándar diseñado, de lo contrario son rechazados.

#### **2.5.2 Transformación y limpieza**

La fase de transformación aplica una serie de reglas de negocio<sup>13</sup> o funciones sobre los datos extraídos para convertirlos en datos que serán cargados.

Para esta fase debe hacerse una limpieza de los datos para eliminar errores e inconsistencias en los mismos resolviendo el problema de identidad de los objetos.

Luego se procede a aplicar las reglas de transformación que pueden ser: combinar los datos de distintas fuentes, realizar búsqueda de valores en distintas tablas y darle tratamiento a valores nulos.

#### **2.5.3 Carga de los datos**

La fase de carga es el momento en que los datos son insertados en la BD del sistema destino, una vez terminada la transformación y limpieza de los mismos. El principal objetivo de esta fase es lograr que los datos estén listos para ser consultados.

### **2.6 Calidad de datos**

Según la ISO 8402 la calidad “es la suma de todos aquellos aspectos o características de un producto o servicio que influyen en su capacidad para satisfacer las necesidades, expresadas o implícitas”.

La calidad de los datos es fundamental para toda migración pues determina el nivel de exactitud y estandarización en que se encuentra la información. Para esto es preciso realizar un perfilamiento de datos, donde se estudia su contenido, estructura, calidad y dependencias, y de ahí obtener estadísticas e información de los mismos.

Este proceso es muy importante pues permite conocer el tipo de información que contiene cada fuente

---

<sup>13</sup>Son aquellas condiciones, validaciones y normas que se deben cumplir y controlar dentro de la organización y que son definidas de acuerdo con el comportamiento esperado del negocio y de la organización.

y campo, además de detectar todos aquellos formatos, registros nulos, valores indebidos, datos ausentes o duplicados, y definir como tratar cada una de las casuísticas que van apareciendo.

### **2.7 Estrategia de migración propuesta**

Para dar solución al problema existente se diseñó una estrategia de migración la cual inicia con el levantamiento de la información que se desea salvaguardar en el nuevo sistema. En este paso se recogerán los datos manejados por la DCI para ser migrados hacia el SGCI. Luego se analizan los sistemas involucrados en el proceso de migración de los datos. En este caso se estudiará el modelo de datos del SGCI que es el que contará con los datos migrados, y además la BD del DRI en la que se encuentran los trámites de misioneros. Posteriormente se crea un respaldo a la BD del DRI para evitar la pérdida de información durante la migración. Si las fuentes de origen de la información son documentos digitales, el siguiente paso sería organizarlos en un mismo formato (Excel). A continuación se crean tablas temporales en el esquema público de la BD del sistema destino para migrar de forma provisional los datos que se trasladarán a las tablas del negocio, para lo cual se utiliza la herramienta Talend Open Studio. Una vez cargados los datos en dichas tablas, se lleva a cabo la transformación y limpieza de los mismos para dar cumplimiento a los siguientes objetivos:

- Simplificar los datos quedando solamente aquellos que son necesarios migrar.
- Lograr que no existan datos repetidos y distribuidos en cualquier parte.
- Evitar anomalías en inserciones, modificaciones y borrados.

En el caso de las visas y la habilitación de los pasaportes, la migración hacia las tablas del negocio del SGCI se efectuará utilizando la herramienta Talend. Para el resto de la información se realiza la implementación de los algoritmos necesarios garantizando que la misma pase de forma correcta hacia el sistema final. Luego se lleva a cabo la ejecución de los algoritmos implementados para realizar una primera iteración del proceso de migración de los datos. En caso de que exista información que no se pueda migrar se almacenará en un arreglo, el cual se imprimirá en el navegador Web para conocer los errores encontrados y mitigarlos, preparando la información para una nueva iteración del algoritmo hasta obtener como resultado final la completa migración.

Un factor importante para el éxito de la migración es la realización de pruebas. Es por esto que posteriormente se propone pasar a esta etapa donde se verifica la calidad de los datos, lo cual permite validar que la migración se realice de forma satisfactoria.

### 2.8 Descripción de la solución propuesta

Una vez trazada la estrategia a seguir en la migración de los datos, se procede a la aplicación de la misma para dar solución al problema planteado. A continuación se describen en dos ejemplos ilustrativos, los pasos empleados en dicho proceso.

#### 2.8.1 Desarrollo de la migración de los viajes hacia el módulo de Trámites

Inicialmente se realizó un levantamiento de información donde se recogieron los viajes a migrar, contenidos en documentos en diversos formatos. Posteriormente se analizó el SGCI para conocer el negocio identificando las tablas que guardarán la información, así como también los campos que contienen y el tipo de dato para cada uno de ellos. Luego se organizaron las fuentes de origen de los viajes llevándolas a formato Excel para poder cargar la información con la herramienta Talend. A continuación se crearon tablas temporales en el esquema público del SGCI a las que se importaron los datos contenidos en los documentos Excel. En la siguiente imagen se muestra un ejemplo de como quedan estructuradas dichas tablas.

public.viajes_2011	
 id_viaje	integer
 nombre_persona	varchar(150)
 mision	varchar
 pais	varchar
 fechaSalida	varchar
 fechaEntrada	varchar
 tipo_mision	varchar(50)

Fig 3. Tabla temporal que contiene los viajes cargados desde un documento Excel.

Haciendo uso del Talend se creó un diseño de trabajo para cargar los datos contenidos en cada uno de los documentos hacia las tablas temporales creadas. El mismo consta de un componente de entrada (tFileInputExcel) para la conexión a los archivos Excel, un componente de salida (tPosgresqlOutput) para la conexión al SGCI y otro para el mapeo de los datos (tMap). En este último se enlazan los campos de la fuente de origen con los correspondientes en el destino. Este proceso se muestra en las siguientes imágenes.



Fig 4. Migración de los viajes desde un documento Excel hacia la tabla temporal en la BD

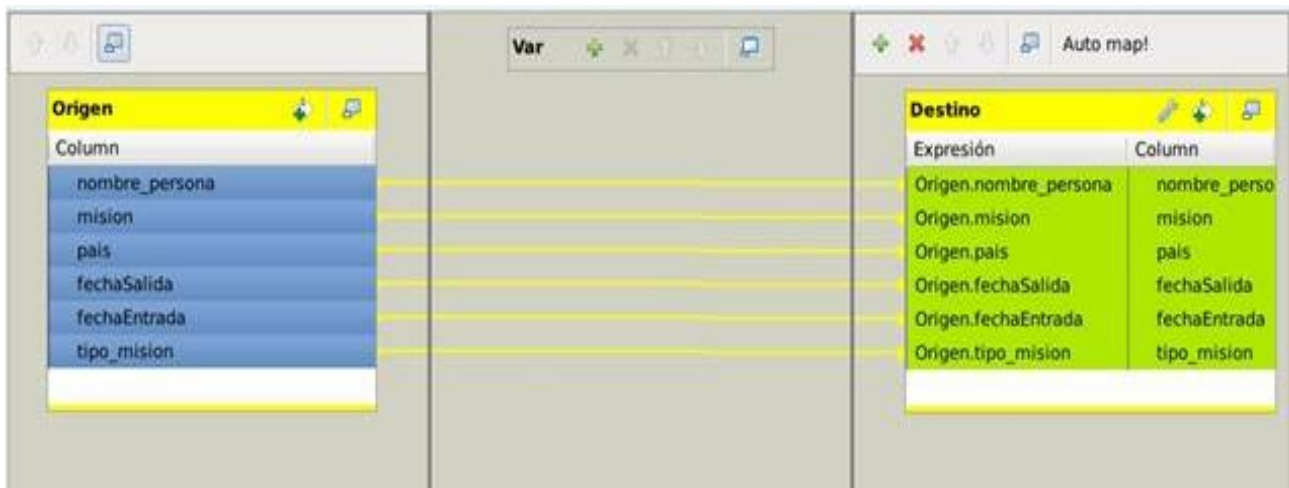


Fig 5. Mapeo de los datos de un documento Excel de viaje con los campos de la tabla temporal creada

Una vez cargados los datos en la tabla temporal, se lleva a cabo la transformación y limpieza de los mismos realizando los siguientes pasos:

- Se verificó que no existan datos repetidos innecesariamente.
- Se le añadió a la tabla temporal los campos `id_tipo_mision` y `codigo_pais` que eran necesarios registrar en el SGCI. Ambos se infieren de los campos `tipo_mision` y `país` respectivamente, por lo que se realizaron consultas en el lenguaje SQL para obtener su valor real. Para la ejecución de las mismas se utilizó la herramienta PgAdmin III.
- Se eliminó la información que no era necesario registrar en el SGCI.

La tabla temporal quedaría de la siguiente forma luego de realizar el paso anterior.



public.viajes_2011			
	<b>id_viaje</b>	<b>integer</b>	
	nombre_persona	varchar(150)	
	fechaSalida	varchar	
	fechaEntrada	varchar	
	id_tipo_mision	integer(10)	
	codigo_pais	char(3)	

**Fig 6.** Tabla temporal luego de la transformación y limpieza de los datos.

Después de cumplir con los pasos anteriores y conociendo el negocio del SGCI, se implementa un método o algoritmo para trasladar los datos desde la tabla temporal hacia las tablas del sistema final. La imagen a continuación muestra de manera general el funcionamiento del algoritmo implementado.

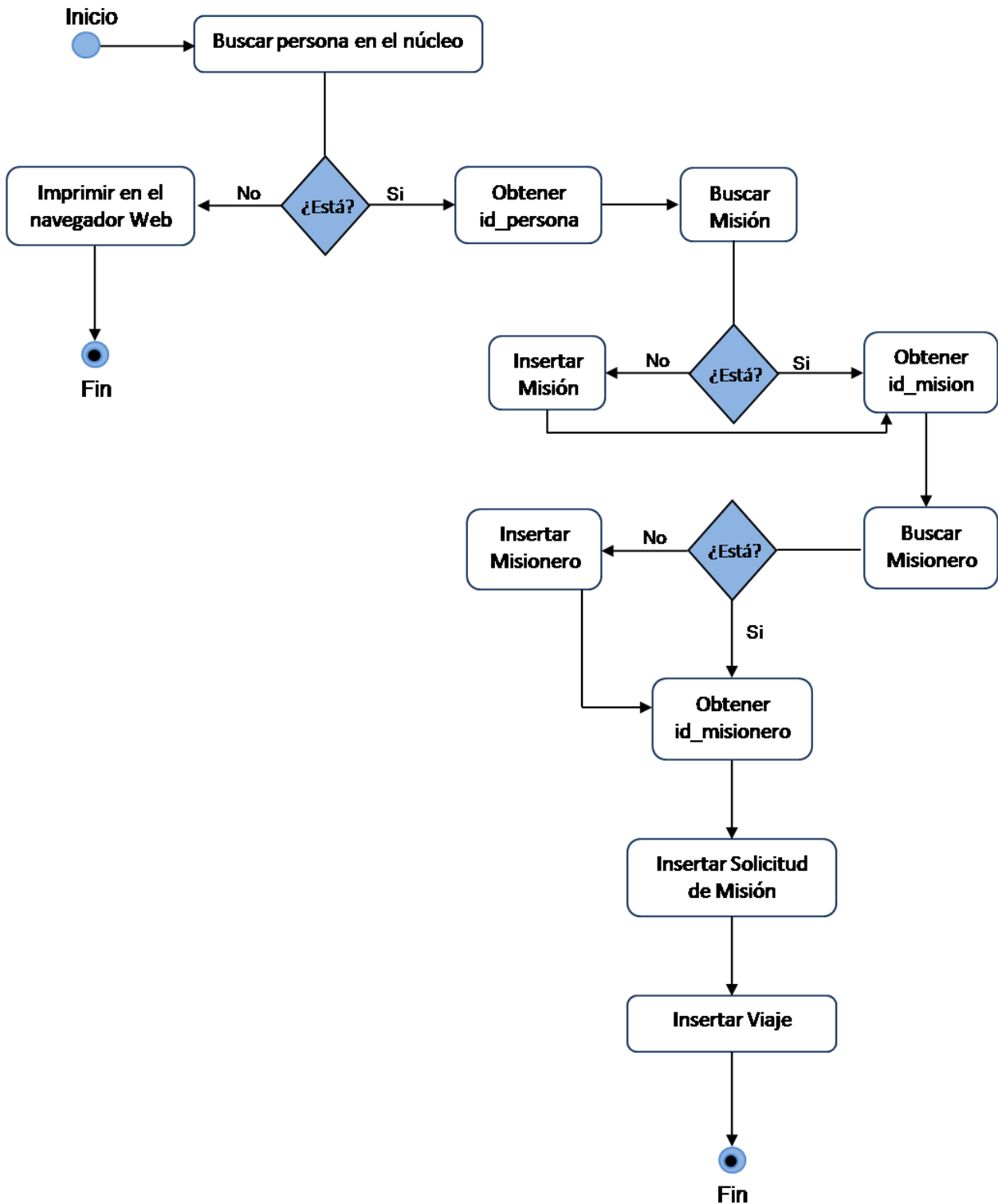


Fig 7. Diagrama que ilustra el funcionamiento del algoritmo implementado.

### Descripción del procedimiento a seguir ilustrado anteriormente

La idea de implementar un algoritmo para la migración de los viajes se debe a que para poder insertar los datos en el sistema final, se debe realizar primero una búsqueda en la BD del núcleo<sup>14</sup> de las personas que viajan como se ilustra en la imagen anterior. A continuación se describe el algoritmo implementado para migrar los viajes:

- Primeramente se realiza una búsqueda en el núcleo de las personas asociadas a los viajes que se migrarán para obtener su id\_persona. Para ello se utilizó una función creada por el grupo de trabajo del CENIA.
- Una vez encontrada esa persona se guarda su id\_persona y se inserta una nueva misión en la tabla tb\_dmision del módulo de cooperación teniendo en cuenta que, en caso de que esa misión ya esté registrada, no se insertará una nueva sino que se tomará el id\_mision de la que ya existe.
- Posteriormente se procede a insertar un misionero en la tabla tb\_dmisionero del mismo módulo mencionado anteriormente obteniéndose el identificador con que fue registrado. Si ya existía algún misionero con el mismo id\_persona entonces se guarda el id\_misionero.
- Luego se inserta una solicitud de misión en la tabla tb\_rmisionero\_mision.
- Finalmente se inserta el viaje en la tabla tb\_dviaje.
- Por último se imprime en el navegador Web la información de las personas que no se migraron por alguna razón, para mitigar los errores y preparar la información para una nueva iteración del método.

Culminada la implementación del algoritmo se procede entonces a ejecutarlo. En esta primera iteración se obtiene como resultado que de los 6329 viajes almacenados en los diferentes documentos Excel se migraron 6209. De los 120 viajes restantes existían errores como:

- Cambio de una letra por otra en los nombres de las personas a buscar.
- Algunos no tenían el segundo nombre o solo aparecía la letra inicial mayúscula del mismo.
- Apellidos almacenados en abreviatura.
- Valor nulo en el campo FechaSalida, el cual es necesario registrar en el sistema final.

Se analizaron los errores encontrados y se corrigieron de la siguiente manera:

---

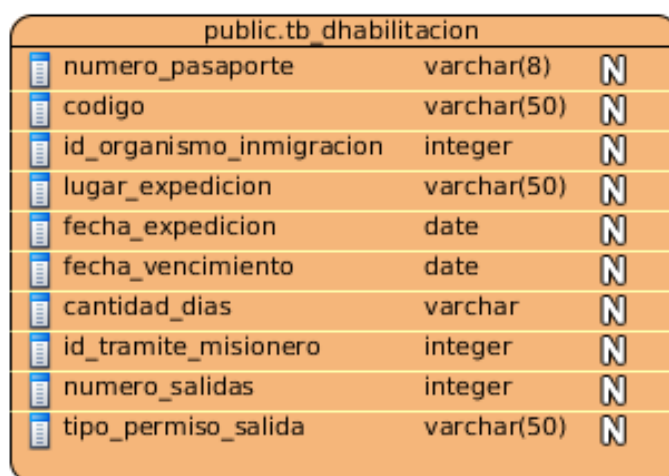
<sup>14</sup> BD que contiene los módulos que son horizontales del SGCI.

- Los nombres que presentaban cambios de letra, falta del segundo nombre, apellidos en abreviatura; fueron modificados por los que aparecían en la tabla persona del núcleo luego de realizar varias búsquedas utilizando los filtros creados en el CENIA.
- Los viajes que tenían valor nulo en el campo FechaSalida fueron verificados con el cliente.

Una vez corregidos estos errores se prosigue a una segunda iteración del algoritmo teniendo como resultado que de los 120 viajes que habían quedado en la primera iteración sólo no se migraron 5 pues no estaban registrados en el núcleo, por lo que se decidió por parte del Departamento de Gestión Universitaria, insertar las personas asociadas a cada uno de estos viajes en el SGCI. De esta manera culmina el proceso de migración de los viajes hacia el SGCI.

### 2.8.2 Desarrollo de la migración de la tabla TDPPermisoSalida del DRI hacia el SGCI

En el levantamiento de la información a migrar se recogieron los datos de la habilitación de los pasaportes, almacenados en la tabla TDPPermisoSalida del DRI. Se creó luego una tabla temporal en el esquema público de la BD del SGCI con el nombre tb\_dhabilitacion. Dicha tabla contiene los mismos campos que la del DRI. A continuación se muestra como queda estructurada la misma.



public.tb_dhabilitacion		
numero_pasaporte	varchar(8)	N
codigo	varchar(50)	N
id_organismo_inmigracion	integer	N
lugar_expedicion	varchar(50)	N
fecha_expedicion	date	N
fecha_vencimiento	date	N
cantidad_dias	varchar	N
id_tramite_misionero	integer	N
numero_salidas	integer	N
tipo_permiso_salida	varchar(50)	N

Fig 8. Tabla temporal de la habilitación de los pasaportes creada en el SGCI.

Utilizando la herramienta de migración Talend se creó un diseño de trabajo que permitió migrar los datos desde la BD del DRI hacia la tabla temporal creada en la BD del SGCI. Primeramente se cargaron los datos de la tabla TDPPermSalida en un componente de entrada que permite la conexión a una BD en SQL Server. Luego se realiza la conexión a la tabla temporal creada en el esquema público del SGCI mediante el componente de salida que brinda Talend para una BD de tipo PostgreSQL, y después se procede al mapeo de los datos enlazando cada campo de la tabla origen con el campo de la tabla destino correspondiente. Todo este proceso se muestra en las figuras 9 y 10.





Fig 9. Migración de datos desde la tabla del DRI hacia la tabla temporal creada en la BD del SGCI.

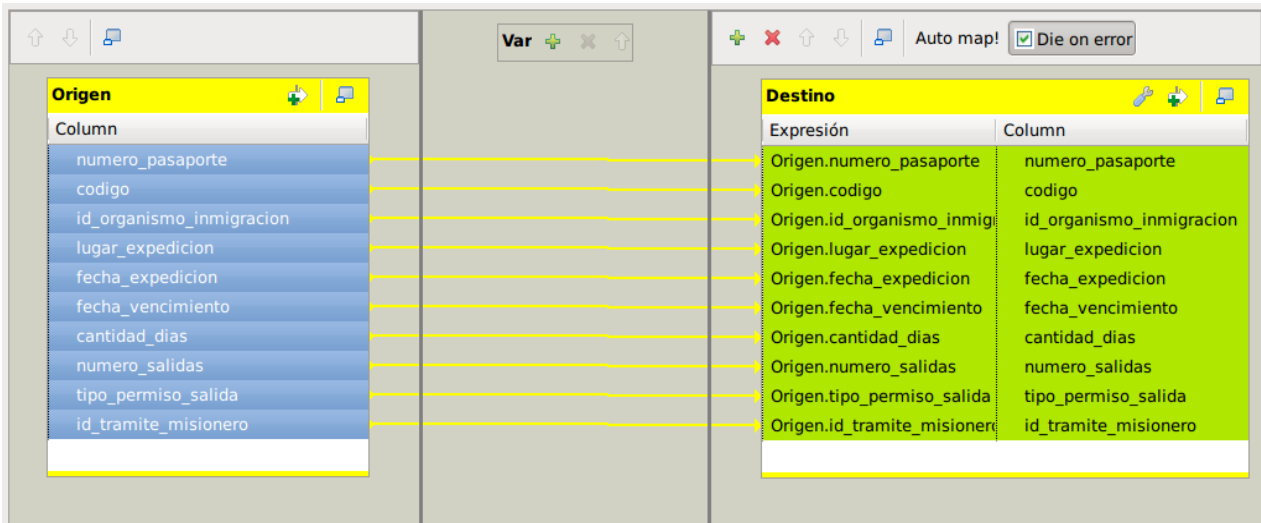


Fig 10. Mapeo de los datos de las habilitaciones hacia la tabla temporal.

La tabla destino de la que se habla anteriormente no es más que la tabla del SGCI a donde se migrarán los datos, la cual se muestra a continuación.

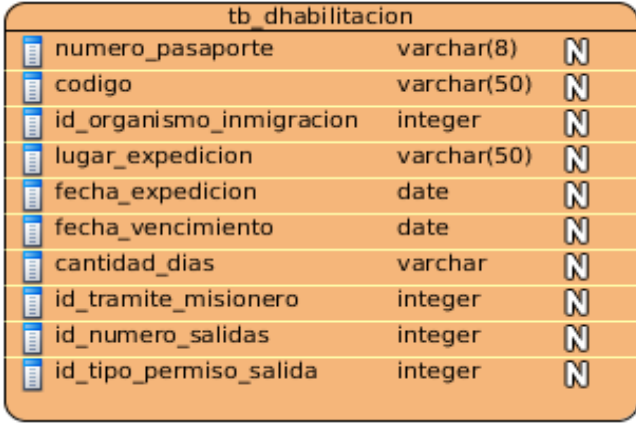
sq_tramite.tb_dhabilitacion		
	<b>id_habilitacion</b>	<b>int4</b>
	numero_pasaporte	varchar(8)
	codigo	varchar(15)
	id_organismo_inmigracion	int4 <b>N</b>
	lugar_expedicion	varchar(50) <b>N</b>
	fecha_expedicion	date
	fecha_vencimiento	date
	cantidad_dias	varchar
	id_numero_salidas	int4 <b>N</b>
	id_tipo_permiso_salida	int4 <b>N</b>
	id_tramite_misionero	int4 <b>N</b>

Fig 11. Tabla del negocio a la que se desea migrar la habilitación de los pasaportes.

Una vez cargada la información en la tabla temporal se procedió a la transformación y limpieza de la misma para la posterior migración hacia la tabla `tb_dhabilitacion` del módulo de trámites de la BD del SGCI. Con este fin se realizaron los siguientes pasos:

- Se le añadió a la tabla temporal los campos `id_numero_salidas` e `id_tipo_permiso_salida` que son necesarios registrar en la tabla del negocio.
- Se llenaron los campos agregados teniendo en cuenta los valores registrados en los nomencladores `tb_ntipo_permiso_salida` y `tb_nnumero_salidas` del módulo de trámites, empleando consultas SQL.
- Se eliminaron los datos duplicados innecesariamente.
- Se eliminó la información que no era necesario migrar al sistema destino.

Así queda entonces la tabla temporal después de realizar el paso anterior.



tb_dhabilitacion		
numero_pasaporte	varchar(8)	N
codigo	varchar(50)	N
id_organismo_inmigracion	integer	N
lugar_expedicion	varchar(50)	N
fecha_expedicion	date	N
fecha_vencimiento	date	N
cantidad_dias	varchar	N
id_tramite_misionero	integer	N
id_numero_salidas	integer	N
id_tipo_permiso_salida	integer	N

**Fig 12. Tabla temporal de la habilitación luego de las transformaciones realizadas.**

Ya lista la información se utilizó nuevamente la herramienta Talend para la migración de los datos de la tabla temporal creada hacia la tabla `tb_dhabilitacion` del módulo de trámites de la BD del SGCI. Se carga la tabla `tb_dhabilitacion` utilizando el componente de entrada para una BD en PostgreSQL, luego se realiza la conexión a la tabla del negocio y después se procede al mapeo de los datos utilizando el componente `tMap` (Mapeo) haciendo corresponder los campos de la tabla origen con los de la tabla destino, garantizando que la información sea migrada correctamente. Dicho proceso se muestra en la siguiente figura, concluyendo así la migración de las habilitaciones.



Fig 13. Diseño de trabajo para migrar los datos desde la tabla temporal hacia la tabla del negocio.

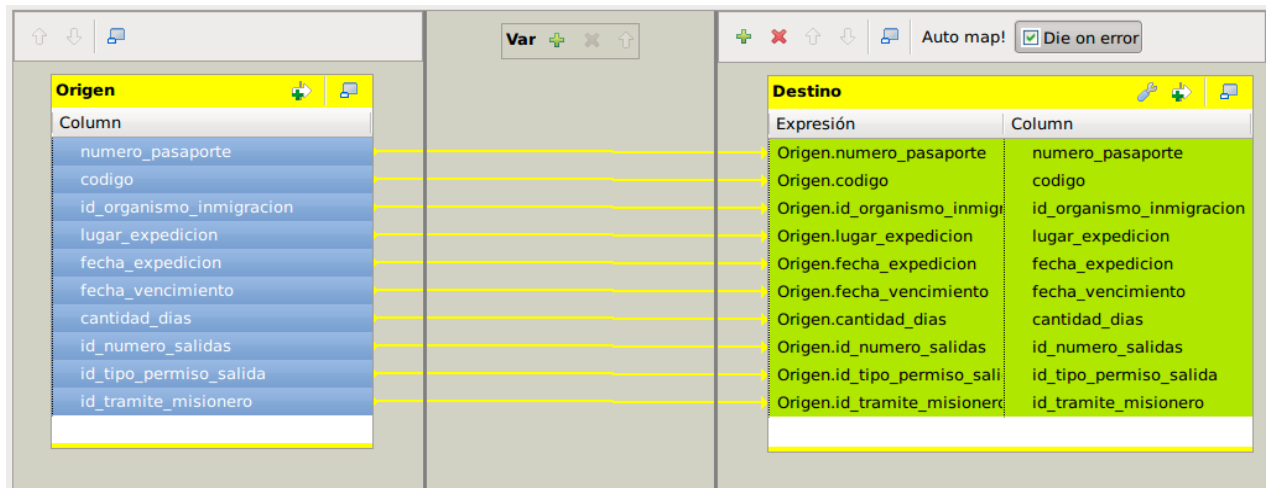
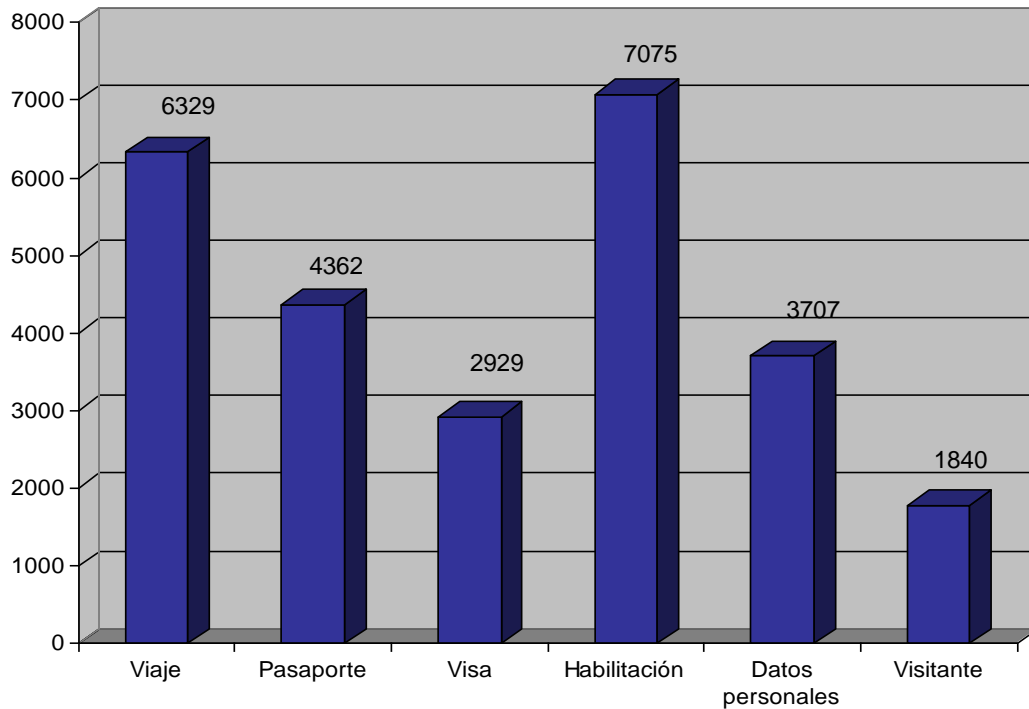


Fig 14. Mapeo de datos de la tabla temporal de habilitaciones hacia el SGCI.

Es necesario destacar que todo el procedimiento de migración de los datos descrito anteriormente se llevó a cabo primero sobre una BD de prueba, la cual es una copia o backup de la BD del SGCI, y luego se desarrolló sobre el SGCI. Esto se realizó con el objetivo de verificar que el proceso se realiza con calidad.

De forma similar fueron tratados los datos de visa, pasaporte y datos personales. Para el caso de los visitantes sólo se realizó el procedimiento hasta comprobar su funcionamiento en la BD de prueba, debido a que el módulo de visita no ha sido implantado aún en el sistema final. Para consultar las restantes descripciones, remitirse al Anexo 1 (Tabla 1-4).

El resultado de la migración fue satisfactorio quedando constancia de la cantidad real de información migrada en la siguiente gráfica.



**Fig 15. Total de las migraciones realizadas.**

### 2.9 Conclusiones parciales

A través de todo el transcurso de este capítulo se llevó a cabo el estudio de los diferentes sistemas que están involucrados en el proceso de migración. Se dirigió la atención a la descripción de la estrategia diseñada, para realizar el proceso de migración de los datos así como la puesta en marcha de la solución que resolverá la problemática existente, a través de la herramienta Talend y de los algoritmos implementados en el lenguaje php para lograr este fin.

### Capítulo 3. Validación de la solución propuesta

#### 3.1 Introducción

Para realizar una buena toma de decisiones, se requiere cierta certeza sobre la confiabilidad de la información que se tiene, la cual está directamente ligada a la calidad de los datos almacenados, cómo son obtenidos y las manipulaciones por las que pasan antes de ser presentados a los usuarios finales. Las pruebas constituyen un paso fundamental en toda migración, ya que permiten operar sobre datos limpios, correctos y útiles; verificando la exactitud y el significado del dato en el sistema.

El objetivo de las pruebas es descubrir el mayor número de errores con la mínima cantidad de esfuerzo y tiempo posible. (Pressman, 2010)

Teniendo en cuenta lo planteado anteriormente, se puede afirmar que validar la solución es un proceso que asegura la correcta migración de los datos a un sistema deseado.

En el presente capítulo se exponen las pruebas que se realizaron en la investigación, dejando así constancia de la exitosa migración de datos provenientes de la DCI hacia el SGCI.

#### 3.2 Problemas encontrados en el proceso de migración de datos

En el proceso de migración de los datos hacia el SGCI se pudieron encontrar algunos problemas con respecto a la calidad de los mismos debido a que provenían de distintas fuentes con diferentes estándares. Dentro de estos problemas se pueden mencionar:

- Faltas de ortografías.
- Valores fuera de rango.
- Ausencia de datos en campos que no pueden ser nulos.
- Errores en el tamaño de los datos.

Para dar tratamiento a estos errores se llevó a cabo el perfilado de los datos a través de la herramienta Talend Open Profiler, la cual permite evaluar el estado en el que se encuentran los datos así como capturar apropiadamente las características de éstos para mejorar su calidad. La solución localiza fácilmente, los datos incorrectos, duplicados o incompletos para identificar problemas potenciales antes de que tenga un impacto serio en el sistema de información. (QData, 2012)

Con esta herramienta de código abierto, se pueden producir informes y gráficos que permiten evaluar el nivel de calidad de los datos.

#### 3.3 Estrategia de validación y prueba

En la Ingeniería de Software, el proceso de pruebas permite verificar la calidad del producto. Es la fase final por donde pasa un sistema o proceso antes de ser entregado para su explotación y validación por parte del cliente.

La estrategia de prueba describe el enfoque y los objetivos generales de las actividades de prueba. Por tanto, después de realizar el proceso de migración de datos es preciso validar que la solución esté correcta, verificando que la cantidad de campos tomados en la fuente origen coincida con la cantidad insertada en la fuente destino. Esto tiene un gran nivel de importancia debido a que da la medida de que la migración se realizó satisfactoriamente.

Para llevar a cabo esta etapa se tomó la decisión de realizar el perfilado de los datos con el uso de la herramienta Talend Open Profiler, para garantizar la integridad, calidad y exactitud de los datos; y se aplicó el nivel de pruebas de aceptación del cliente para comprobar el éxito de la migración y la satisfacción del mismo.

Como parte de la estrategia de prueba se analizaron además los errores que surgieron en cada proceso de migración para posteriormente solucionarlos. Esto trajo consigo que aparecieran no conformidades, las cuales fueron resueltas rápidamente. Además se verificó la migración de los datos en el sistema, antes que el cliente, con el objetivo de comprobar si estaban correctos para ser entregados.

### 3.3.1 Perfilado de datos

Empleando la herramienta Talend Open Profiler, que permite el análisis por columna de una tabla en la BD, se generaron gráficos estadísticos para comprobar el comportamiento de cada campo, tanto en la fuente origen como en la de destino, teniendo en cuenta los siguientes indicadores:

- **Row Count** (contar filas): muestra la cantidad de filas que contiene una columna.
- **Null Count** (contar nulos): muestra la cantidad de valores nulos en una columna.
- **Distinct Count** (contar distintos): muestra la cantidad de valores distintos en una columna.
- **Unique Count** (contar únicos): muestra la cantidad de valores únicos en una columna.
- **Duplicate Count** (contar duplicados): muestra la cantidad de valores duplicados en una columna.
- **Default Value Count** (contar valores por defecto): muestra la cantidad de valores por defecto en una columna.

A continuación se muestran algunos de los gráficos realizados durante el desarrollo de la migración.

En las figuras 16 y 17 se muestra el comportamiento del campo id\_visa tanto en la tabla TDVisa del DRI como en la tabla tb\_dvisa del SGCI. Se puede observar que los indicadores de comparación coinciden totalmente, lo cual demuestra que el proceso de migración se realizó satisfactoriamente.

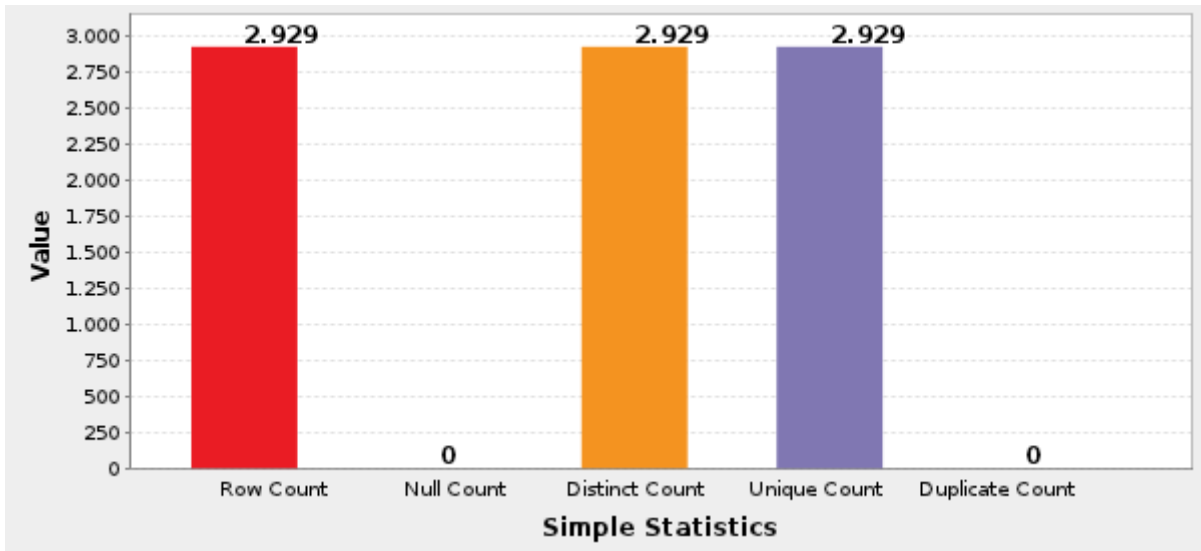


Fig 16. Campo id\_visa de la tabla TDVisa del DRI.

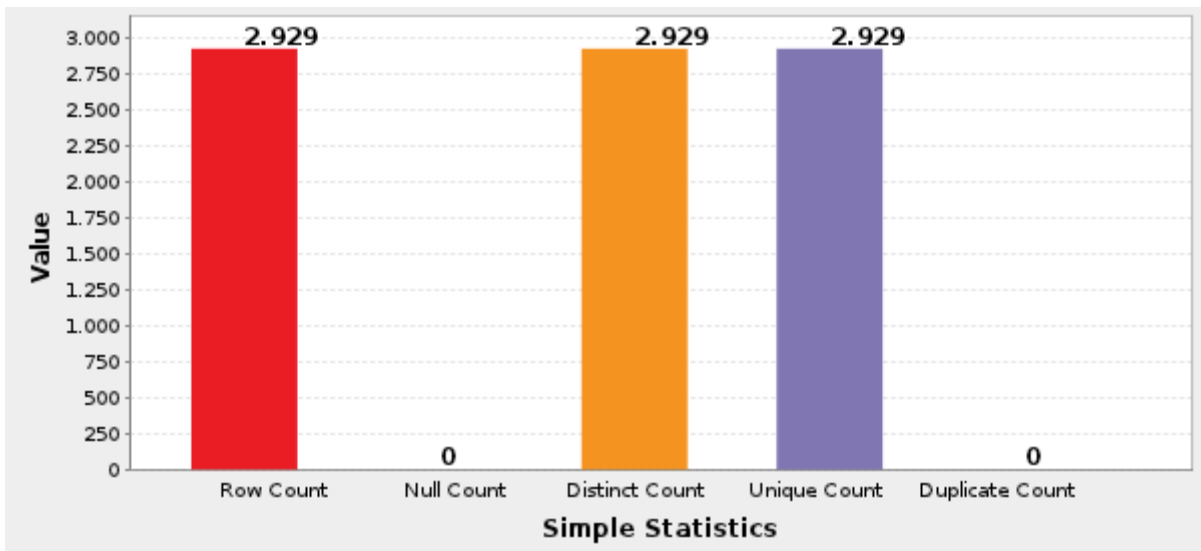


Fig 17. Campo id\_visa de la tabla tb\_dvisa del SGCI.

En las figuras 18 y 19 se muestra el comportamiento del campo numero\_visa tanto en la tabla TDVisa del DRI como en la tabla tb\_dvisa del SGCI, donde se puede observar que los indicadores de comparación coinciden totalmente, lo cual indica que el proceso de migración se realizó satisfactoriamente.

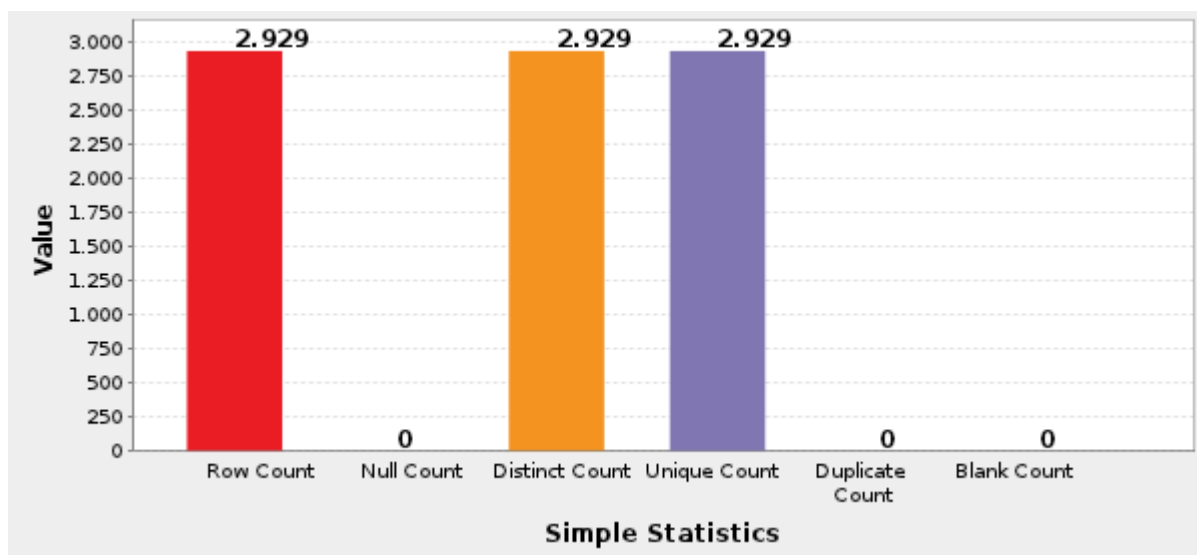


Fig 18. Campo numero\_visa de la tabla TDVisa del DRI.

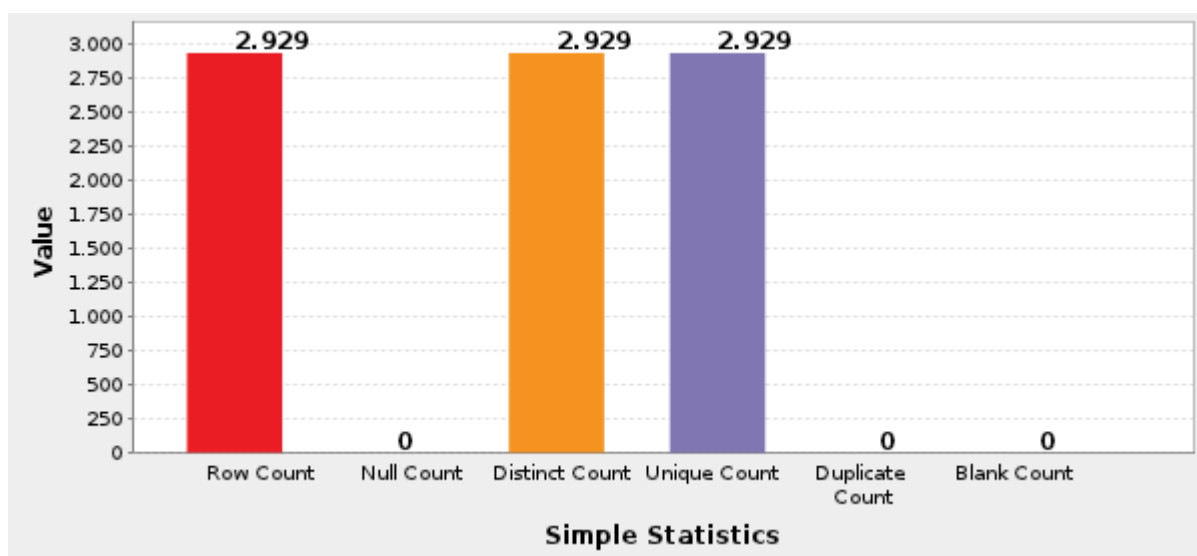


Fig 19. Campo numero\_visa de la tabla tb\_dvisa del SGCI.



En las figuras 20 y 21 se muestra el comportamiento del campo `id_persona` tanto en la tabla `TDPersona` del DRI como en la tabla `tb_ddatos_misionero` del SGCI. Como se puede observar los indicadores de comparación coinciden totalmente, lo cual demuestra que el proceso de migración se realizó satisfactoriamente.

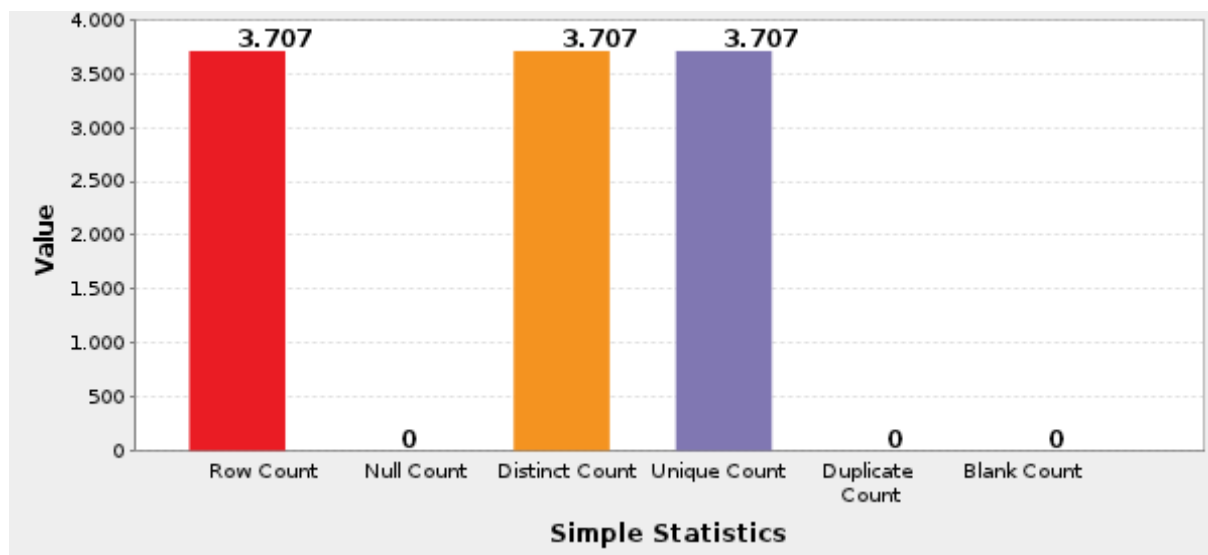


Fig 20. Campo `id_persona` de la tabla `TDPersona` del DRI.

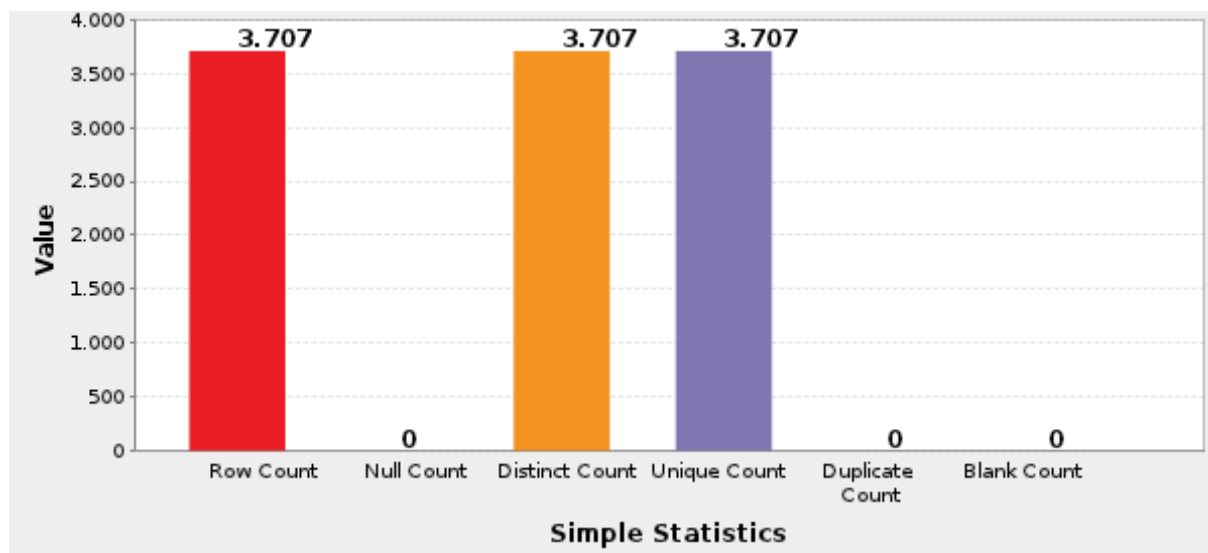


Fig 21. Campo `id_persona` de la tabla `tb_ddatos_misionero` del SGCI.

Las figuras 22 y 23 muestran el comportamiento del campo estatura tanto en la tabla TDPersona del DRI como en la tabla tb\_ddatos\_misionero del SGCI, donde se puede observar que los indicadores de comparación coinciden totalmente, esto demuestra que el proceso de migración se realizó satisfactoriamente.

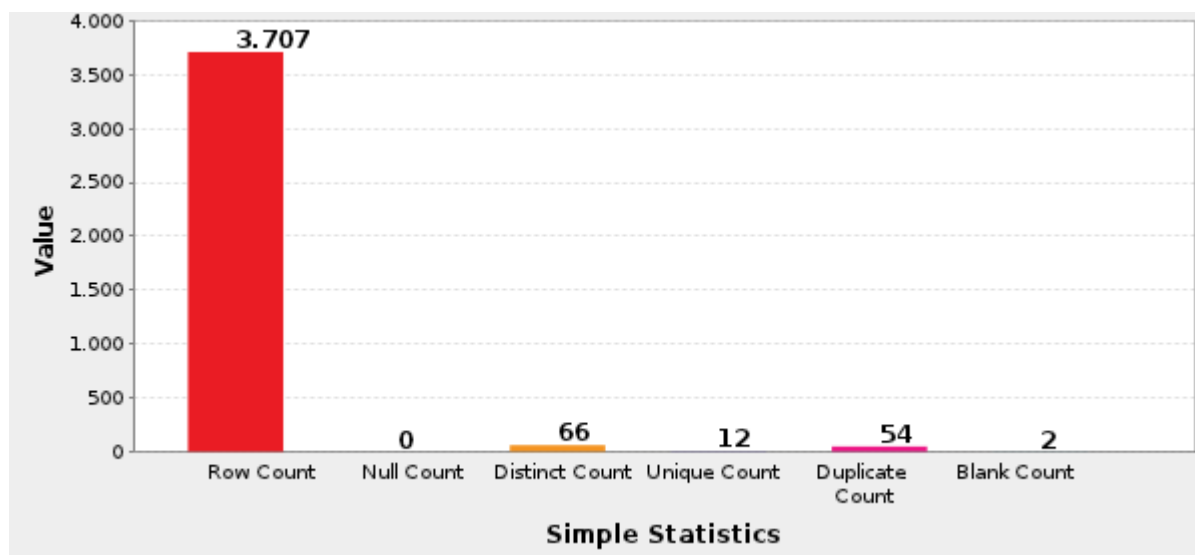


Fig 22. Campo estatura de la tabla TDPersona del DRI.

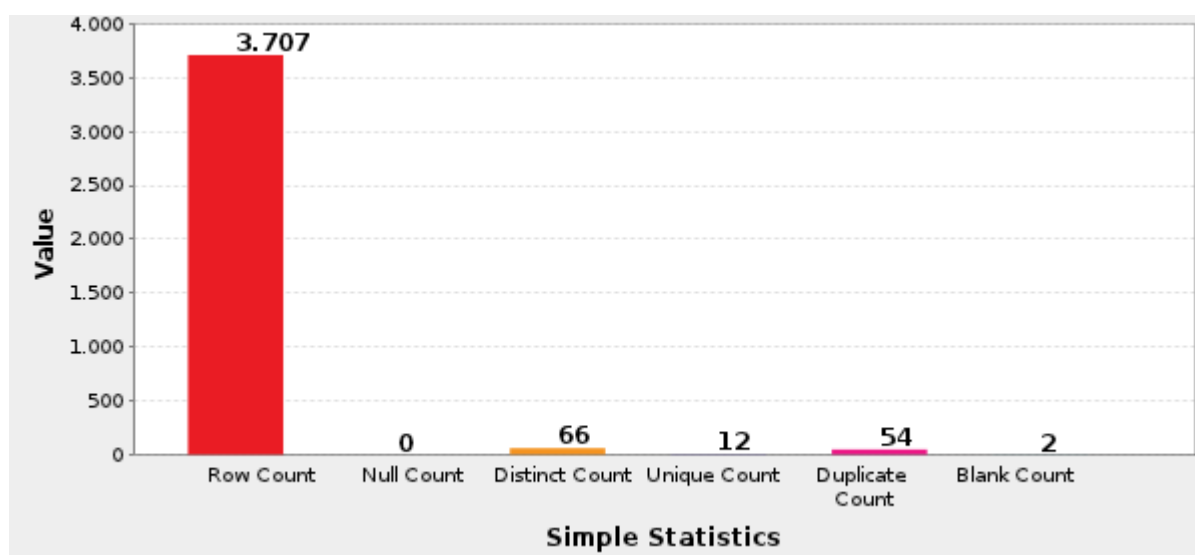


Fig 23. Campo estatura de la tabla tb\_ddatos\_misionero del SGCI.

En las figuras 24 y 25 se muestra el comportamiento del campo `numero_pasaporte` tanto en la tabla `TDPasaporte` del DRI como en la tabla `tb_dpasaporte` del SGCI. Se puede observar que los indicadores de comparación coinciden totalmente, lo cual revela que el proceso de migración se realizó satisfactoriamente.

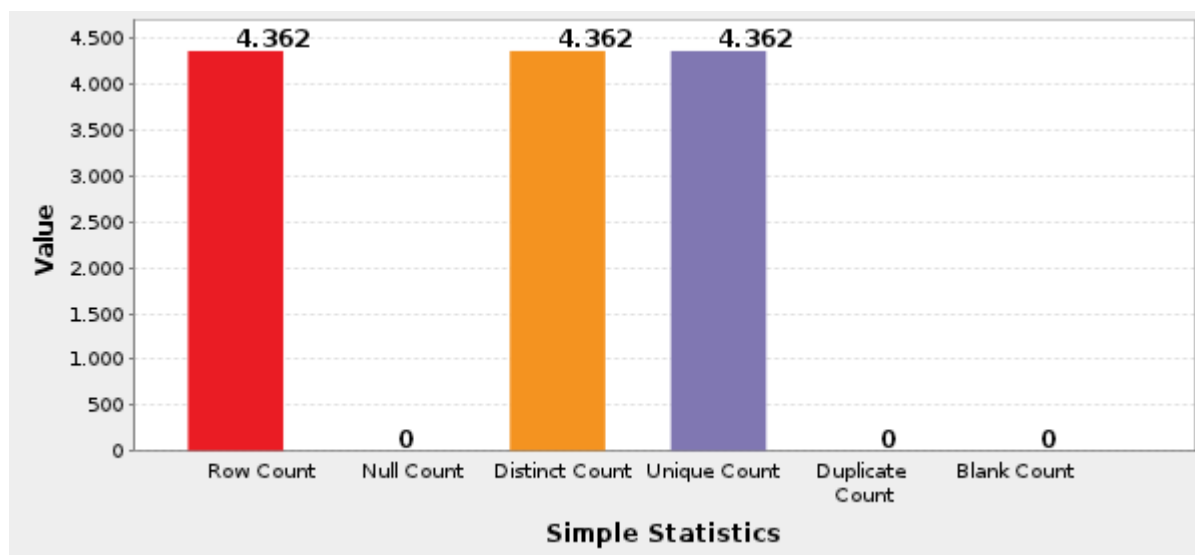


Fig 24. Campo `numero_pasaporte` de la tabla `TDPasaporte` del DRI.

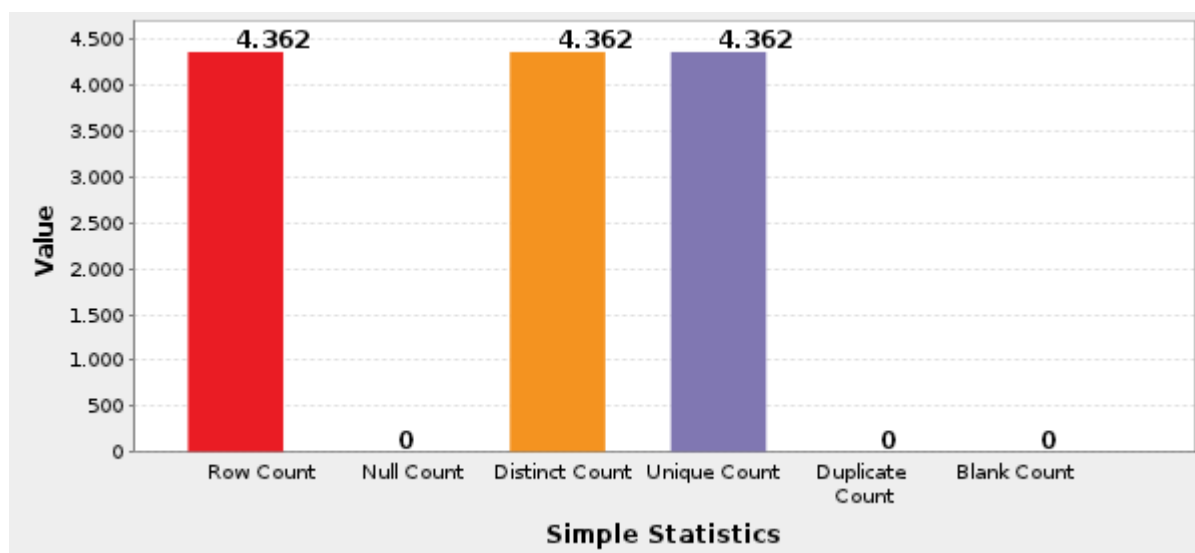


Fig 25. Campo `numero_pasaporte` de la tabla `tb_dpasaporte` del SGCI.

Como se pudo observar, en las figuras 26 y 27 se muestra el comportamiento del campo tipo\_pasaporte tanto en la tabla TDPasaporte del DRI como en la tabla tb\_dpasaporte del SGCI, la coincidencia total de los indicadores de comparación refleja una migración de datos satisfactoria.

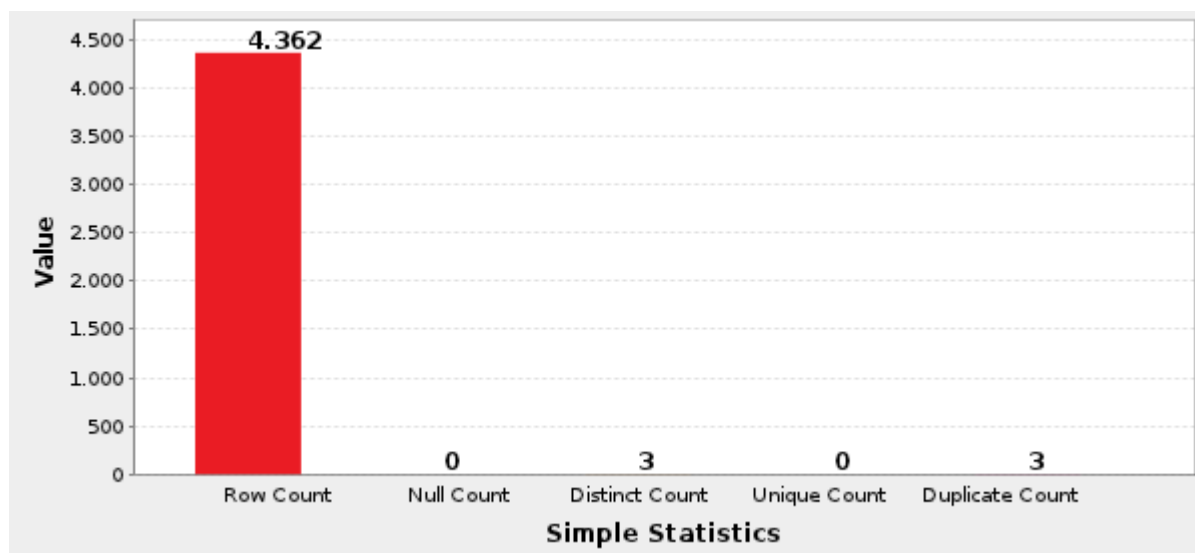


Fig 26. Campo tipo\_pasaporte\_dri de la tabla TDPasaporte del DRI.

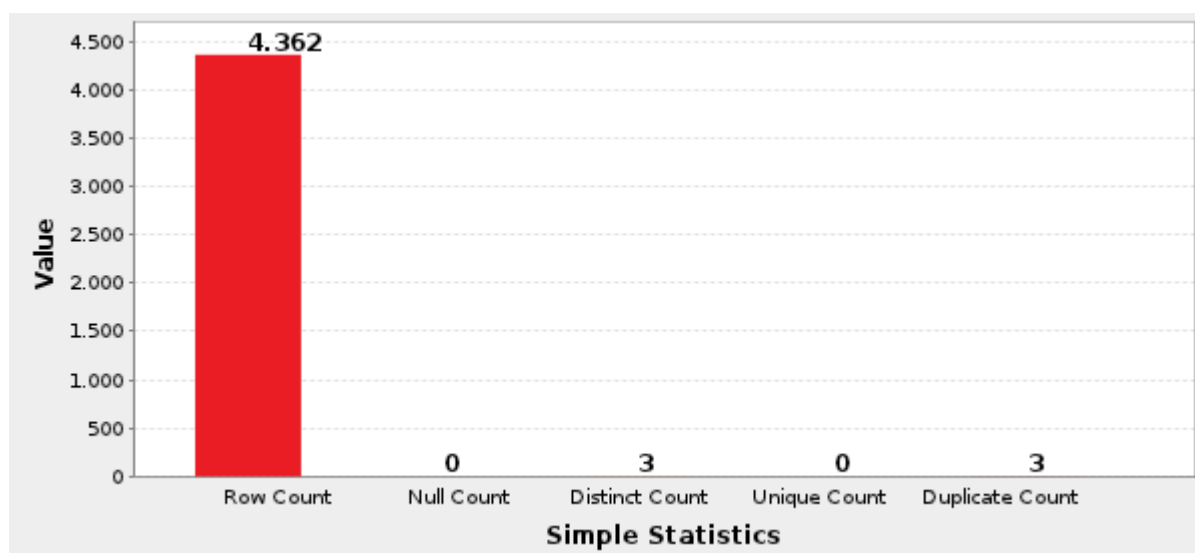


Fig 27. Campo tipo\_pasaporte\_dri de la tabla tb\_dpasaporte del SGCI.

Como se puede observar, en las figuras 28 y 29, el campo nombre\_visitante de la tabla temporal no coincide exactamente con el campo nombre\_visitante de la tabla tb\_dvisitante del SGCI. Esto se debe a que en la tabla del negocio se registra el nombre de un visitante una sola vez y como se puede apreciar en el indicador *Duplicate Count* de la tabla temporal, existen nombres repetidos. El indicador *Distinct Count* de la tabla a migrar, refleja la cantidad real de visitantes que serán insertados en la tabla tb\_dvisitante del negocio.

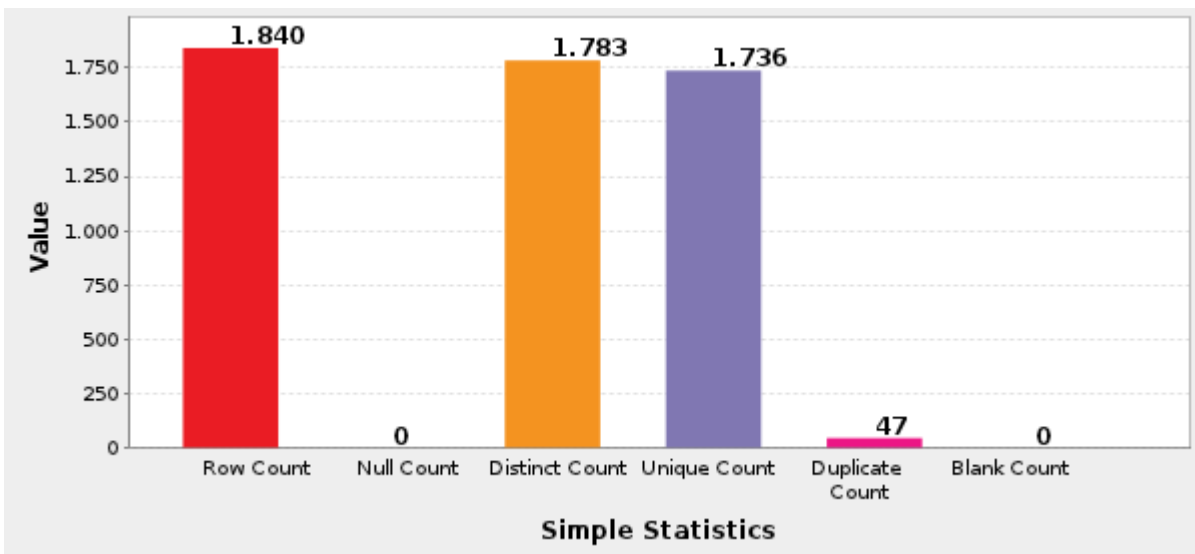


Fig 28. Campo nombre\_visitante\_ de la tabla temporal tb\_dvisitante del SGCI.

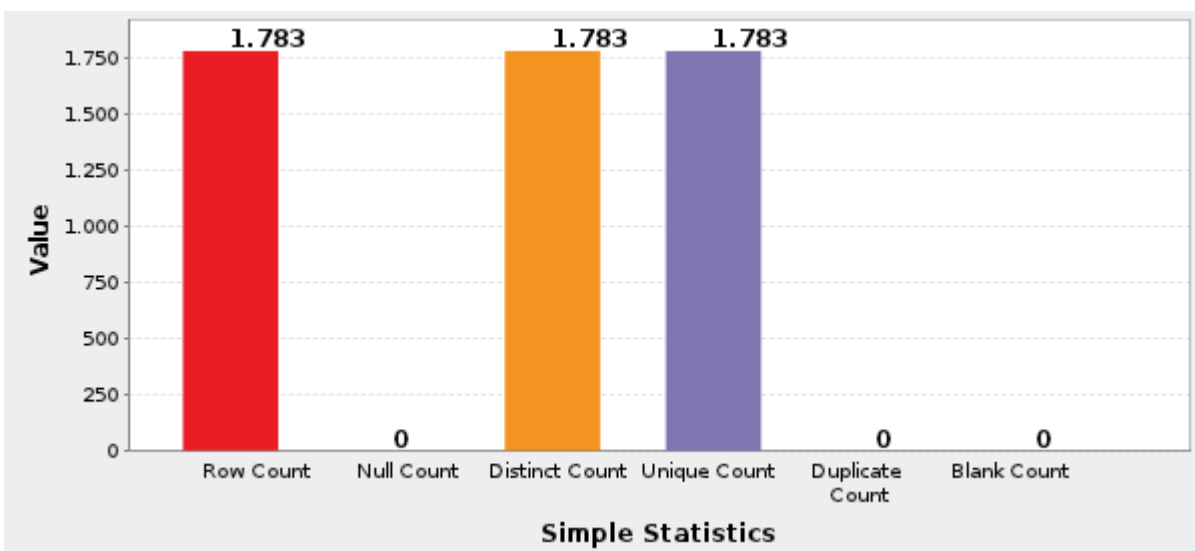


Fig 29. Campo nombre\_visitante\_ de la tabla del negocio tb\_dvisitante del SGCI.

La explicación anterior es válida también para el campo `id_nivel_visitante` que se muestra en las figuras 30 y 31.

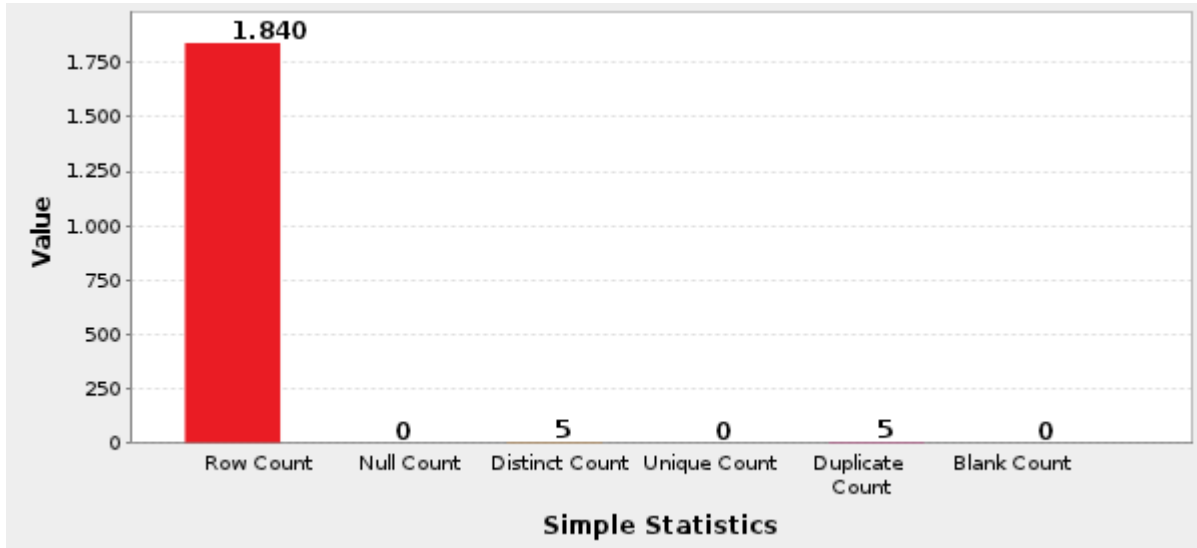


Fig 30. Campo `id_nivel_visitante_` de la tabla temporal `tb_dvisitante` del SGCI.

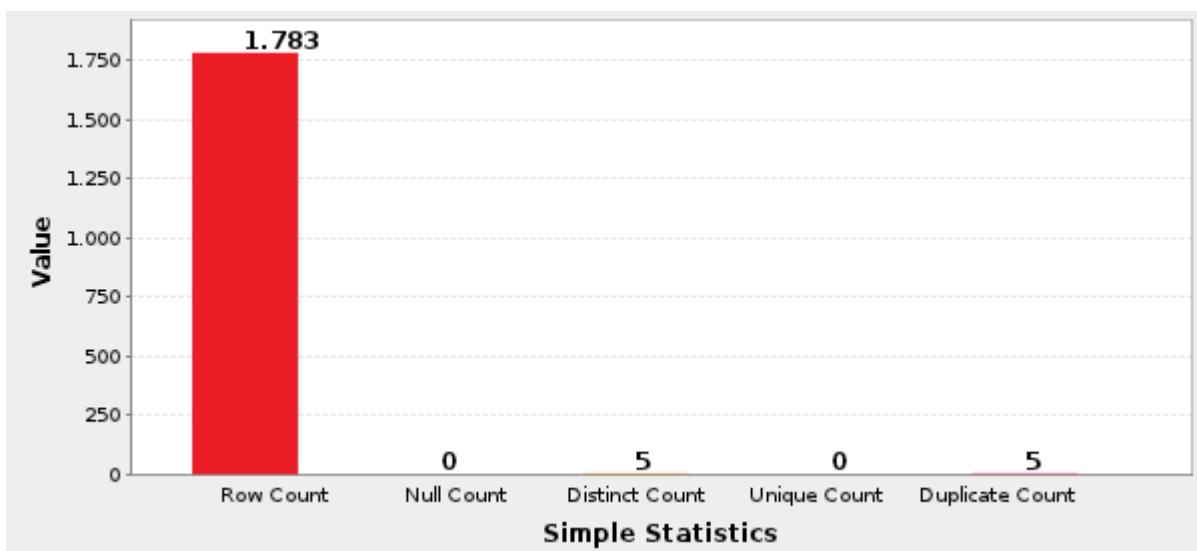


Fig 31. Campo `id_nivel_visitante_` de la tabla del negocio `tb_dvisitante` del SGCI.

Como se puede observar, en las figuras 32 y 33 se muestra el comportamiento del campo codigo tanto en la tabla TDHabilitacion del DRI como en la tabla tb\_dhabilitacion del SGCI. La total coincidencia de los indicadores de comparación demuestra que el proceso de migración se realizó con éxito.

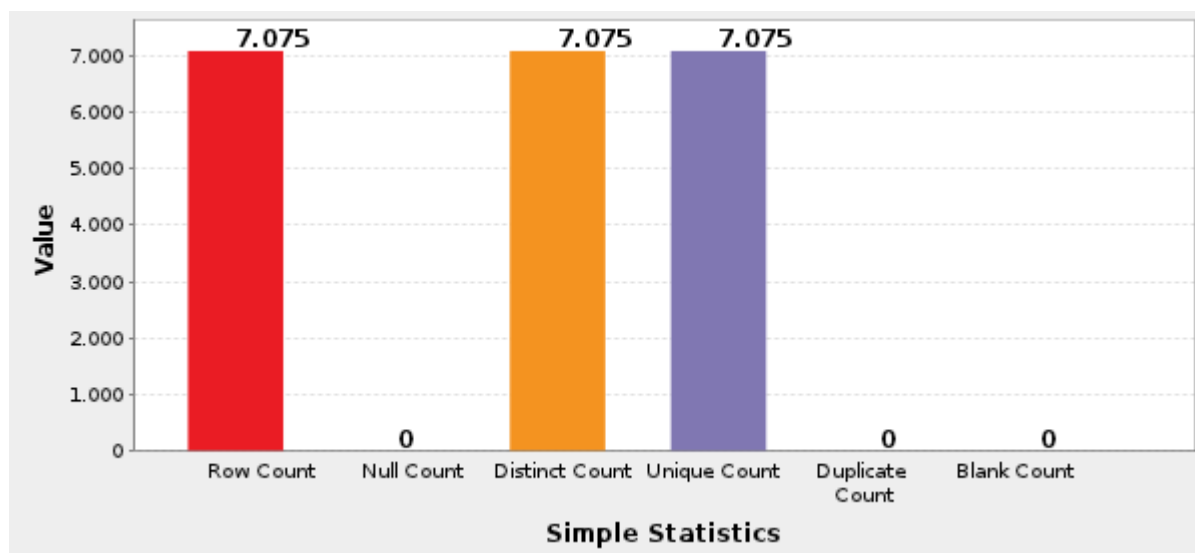


Fig 32. Campo codigo de la tabla TDHabilitacion del DRI.

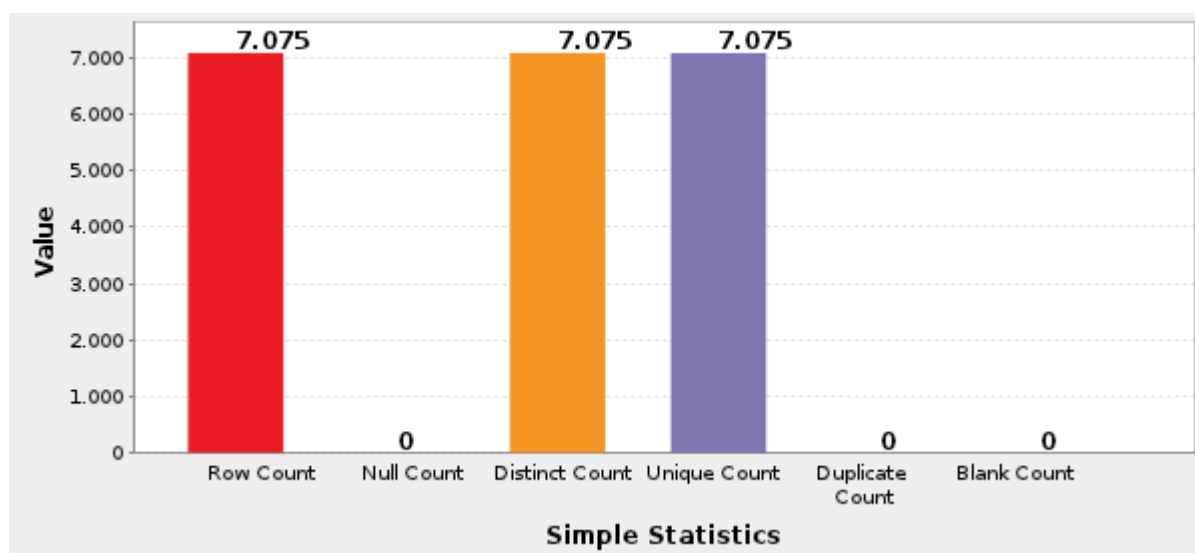


Fig 33. Campo codigo de la tabla tb\_dhabilitacion del SGCI.

Las figuras 34 y 35 muestran el comportamiento del campo fecha\_expedicion tanto en la tabla TDHabilitacion del DRI como en la tabla tb\_dhabilitacion del SGCI, donde se puede observar que los indicadores de comparación coinciden totalmente, lo cual revela que el proceso de migración se realizó satisfactoriamente.

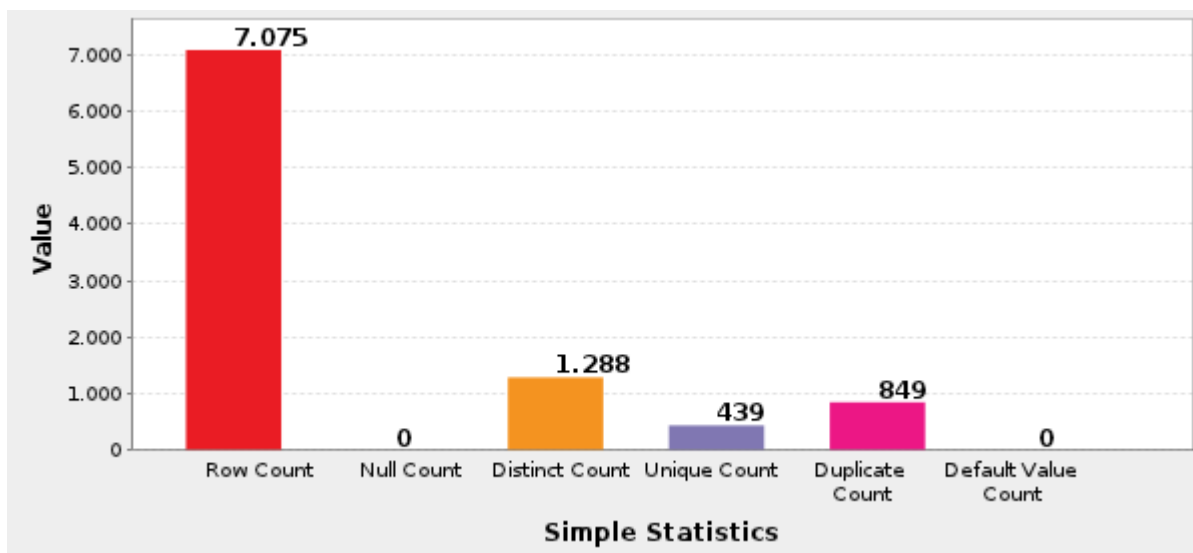


Fig 34. Campo fecha\_expedicion de la tabla TDHabilitacion del DRI.

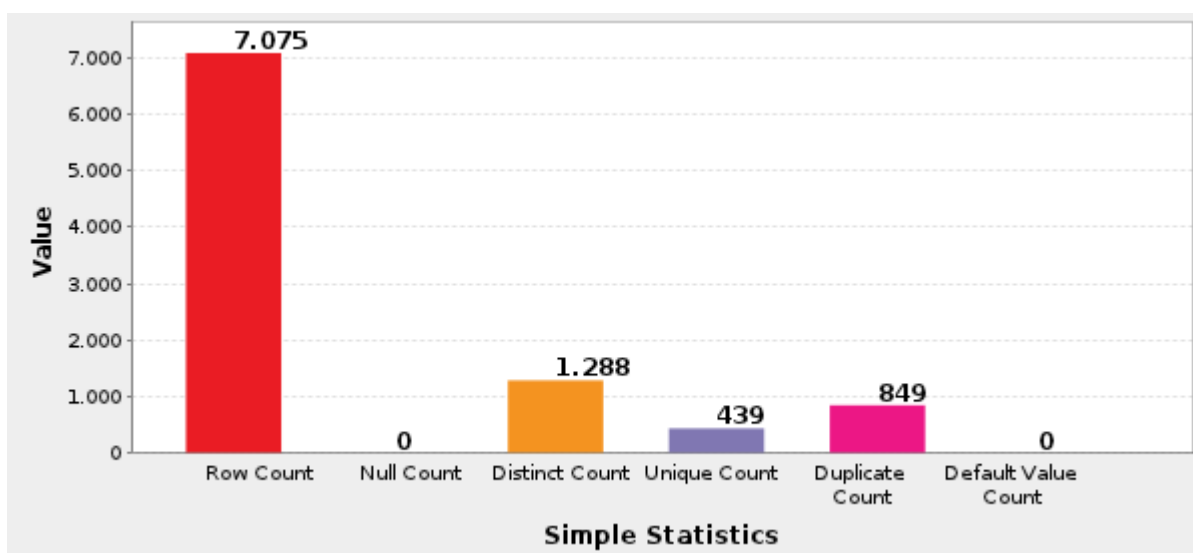


Fig 35. Campo fecha\_expedicion de la tabla tb\_dhabilitacion del SGCI.



A continuación se puede observar el comportamiento del campo `id_mision` tanto en la tabla temporal `tb_dviaje` del SGCI como en la tabla del negocio `tb_dviaje` del mismo sistema, a través de las figuras 36 y 37. La coincidencia de los indicadores de comparación refleja una migración satisfactoria.

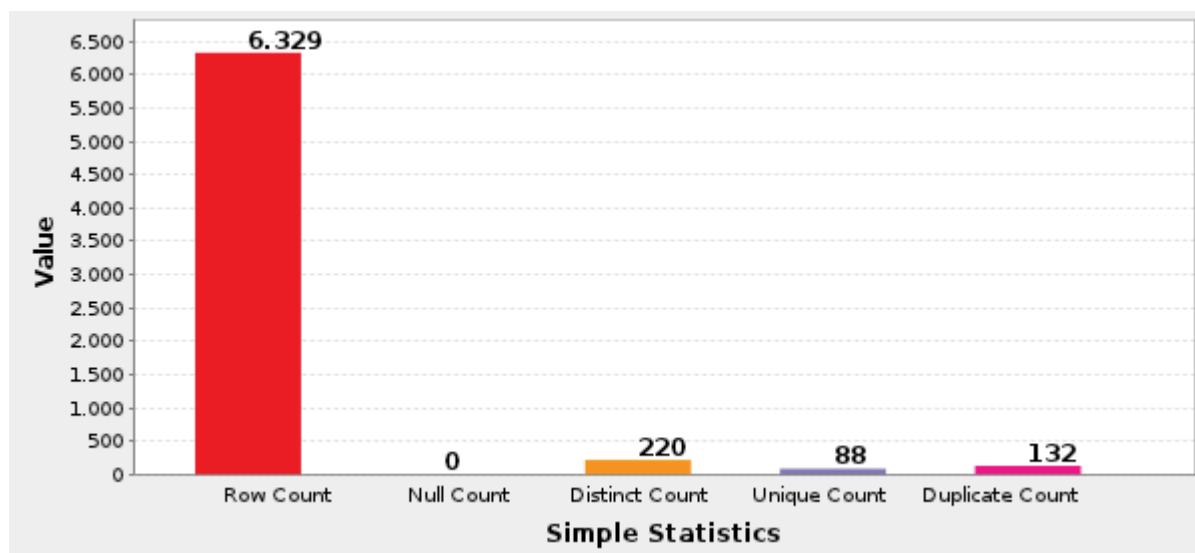


Fig 36. Campo `id_mision` de la tabla temporal `tb_dviaje` del SGCI.

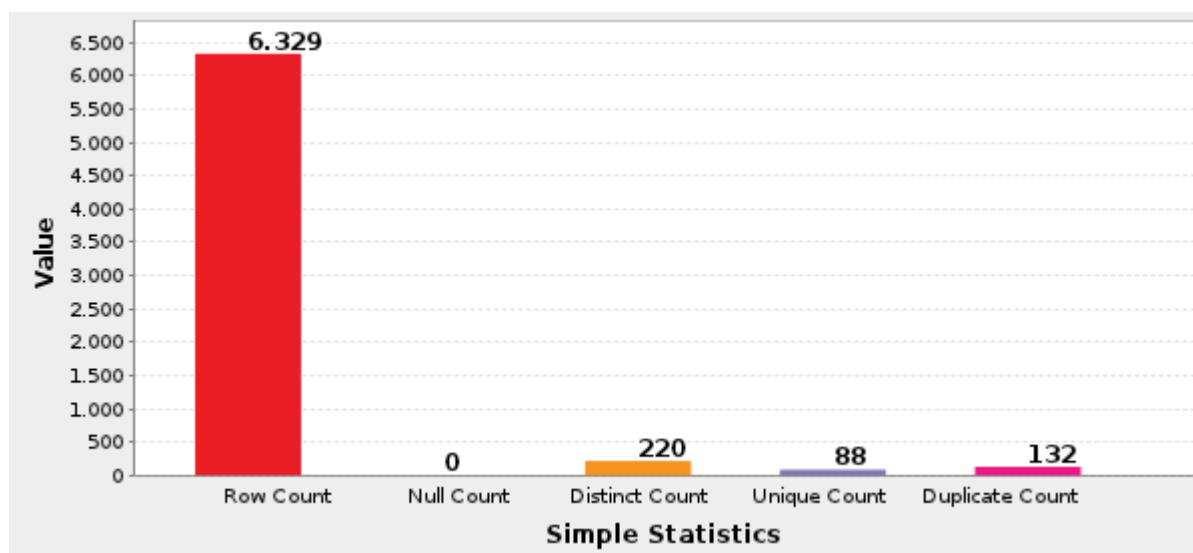


Fig 37. Campo `id_mision` de la tabla del negocio `tb_dviaje` del SGCI.

Como se puede observar, en las figuras 38 y 39 se muestra el comportamiento del campo fecha\_salida tanto en la tabla temporal tb\_dviaje del SGCI como en la tabla del negocio tb\_dviaje del mismo sistema. El hecho de que los indicadores de comparación coinciden totalmente, demuestra que el proceso de migración se realizó satisfactoriamente.

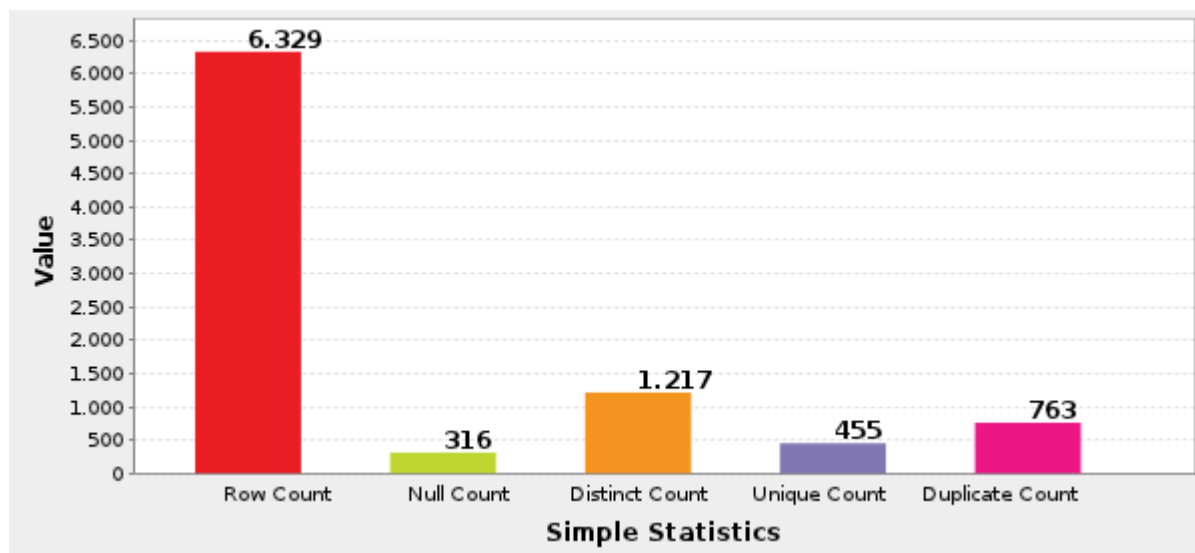


Fig 38. Campo fecha\_salida de la tabla temporal tb\_dviaje del SGCI.

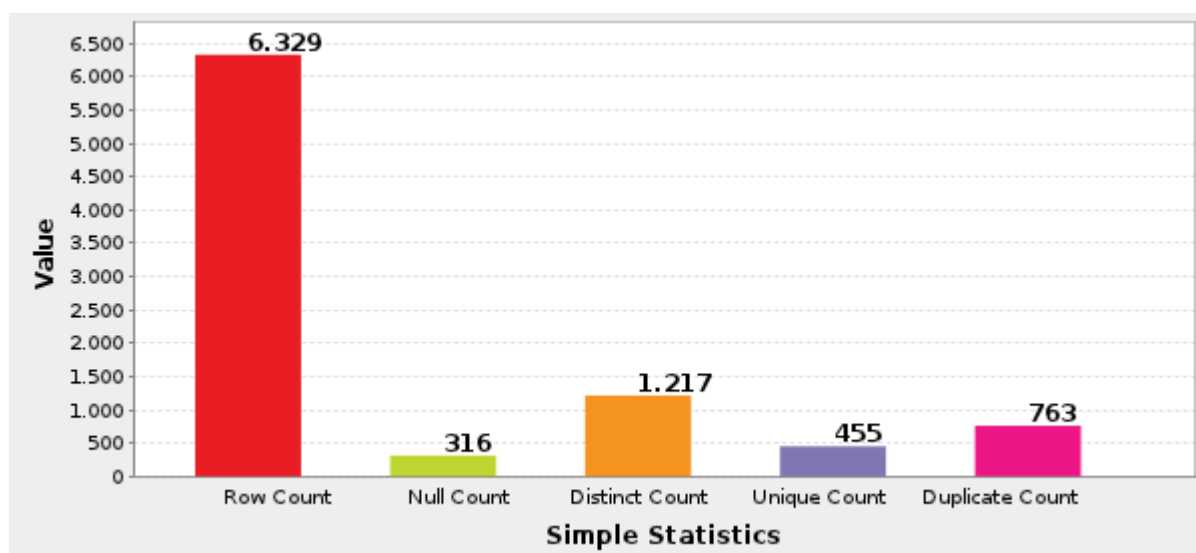


Fig 39. Campo fecha\_salida de la tabla del negocio tb\_dviaje del SGCI.

Con el perfilado de datos realizado se pudieron examinar los errores que presentaban, lo cual brindó la posibilidad de corregirlos para que fuesen cargados con la calidad requerida hacia el SGCI, y así lograr una migración satisfactoria.

Para consultar el resto de los resultados obtenidos durante el perfilado de datos, dirigirse a los Anexos 2-7.

### 3.3.2 Pruebas de aceptación

Las pruebas de aceptación deben ser realizadas por el cliente. Se utilizan para verificar que los datos en el sistema destino, cumplan con las especificaciones dadas, y confirmar que los usuarios principales del sistema no encuentren problemas en la migración de los datos realizada y lo puedan utilizar para realizar las funciones y tareas para las que fue diseñado. Con esto se puede determinar si los datos pueden ser usados en un ambiente de producción.

Para este tipo de prueba es importante mencionar que el sistema ya está en despliegue y está siendo utilizado sin ningún problema por el cliente, quien pudo comprobar que los grupos de datos (pasaportes, visas, habilitaciones, viajes y datos personales) que existían en documentos de tipo Excel y en el DRI, fueron trasladados hacia el SGCI, por tanto, el desarrollo de esta prueba fue satisfactorio, quedando como constancia el acta de aceptación del cliente. (Ver Anexo 8)

Datos	Origen/Cantidad	Destino/Cantidad
Viajes	Excel(6329)	6329
Pasaportes	Excel(4362)	4362
Visas	DRI(2929)	2929
Habilitaciones	DRI(7075)	7075
Datos personales	DRI(3707)	3707

Fig 40. Cantidad de datos del origen y cantidad de datos en el destino.

### 3.4 Conclusiones parciales

Después de validar la propuesta de solución se puede arribar a que los resultados fueron los esperados, siendo satisfactorio el proceso de migración de los datos hacia el SGCI, contando con la calidad requerida, y demostrando que todos los datos coinciden en su totalidad.

### **Conclusiones**

A partir de los objetivos planteados y el trabajo realizado en esta investigación, se arribó a las siguientes conclusiones:

- El estudio de procedimientos y conceptos permitió una mejor comprensión del proceso de migración de datos, logrando de esta forma examinar los datos y elaborar una estrategia para dar cumplimiento al problema planteado.
- Con la puesta en marcha de la estrategia para la migración de los datos se obtuvo en el SGCI datos limpios, consistentes y centralizados.
- Con la aplicación de las pruebas para la comprobación de la calidad e integridad de los datos se pudo realizar reportes, proporcionándole al cliente información sobre la veracidad y la calidad de la migración realizada.

### **Recomendaciones**

- Tener en cuenta la estrategia de migración elaborada en el presente trabajo de diploma para futuros procesos de migración de datos que tengan características similares a la problemática que se le da solución.
- Continuar realizando investigaciones referentes al tema de las migraciones de datos, así como las herramientas que existen y surgen al paso de los días, vinculadas a estos procesos.

### Referencias bibliográficas

- Alma Carrillo, Freddy Cervantes, Miguel Gutiérrez. 2010.** SQL Server. [En línea] 2010. [Citado el: 21 de Noviembre de 2011.]  
<http://www.uaem.mx/posgrado/mcruz/cursos/miic/sql.pdf>.
- Álvarez, Miguel Angel. 2009.** desarrolloweb.com. *CodeIgniter*. [En línea] 23 de Noviembre de 2009. [Citado el: 14 de Enero de 2012.]  
<http://www.desarrolloweb.com/articulos/codeigniter.html>.
- . **2007.** desarrolloweb.com. *Qué es PHP*. [En línea] 9 de Mayo de 2007. [Citado el: 15 de Enero de 2012.] <http://www.desarrolloweb.com/articulos/392.php>.
- Arias, A.A. 2007.** *Sistema Integrado de Gestión Estadística (SIGE). Rol Diseñador de la Base de Datos*. La Habana : Universidad de las Ciencias Informáticas, 2007.
- Ciberaula. 2010.** Introducción a Apache. [En línea] 2010. [Citado el: 16 de Enero de 2012.] [http://www.ecured.cu/index.php/Servidor\\_web](http://www.ecured.cu/index.php/Servidor_web).
- Consuelo Inés González Lau, Carolina Aristizábal Moreno. 2007.** *Estado del arte de pruebas de bases de datos para la migración y validación de datos*. Medellín : Universidad EAFIT, Tesis, 2007.
- Cruz, Felipe Nieves. 2006.** *Estrategia del Mundo Contemporáneo*. México : Hitek Patrick, 2006.
- Cutro, Alfonso. 2009.** Dataprix. [En línea] 24 de Julio de 2009. [Citado el: 25 de Noviembre de 2011.] <http://www.dataprix.com/empresa/productos/talend-open-data-solutions>.
- Date, C. J. 2001.** *Introducción a los sistemas de bases de datos*. México : Editorial Mexicana, 2001. Vol. Séptima Edición.
- Dirección de Cooperación Internacional. 2010.** *Funciones de la DCI*. La Habana : s.n., 2010.
- EcuRed. 2011.** Lenguaje de programación. [En línea] [Citado el: 16 de Enero de 2012.] [http://www.ecured.cu/index.php/Lenguaje\\_de\\_programaci%C3%B3n](http://www.ecured.cu/index.php/Lenguaje_de_programaci%C3%B3n).
- . **2011.** Servidor web. [En línea] [Citado el: 4 de Enero de 2012.] [http://www.ecured.cu/index.php/Servidor\\_web](http://www.ecured.cu/index.php/Servidor_web).
- Gayosso, Jazmín Hernández. 2010.** *Desarrollo y adecuación de aplicaciones web para la administración del capital humano en empresas públicas y privadas*. México : Instituto Politécnico Nacional, 2010.
- González, Carlos D. 2012.** Curso Base de Datos PostgreSQL, SQL avanzado y PHP. [En línea] Febrero de 2012. [Citado el: 12 de Febrero de 2012.] <http://www.usabilidadweb.com.ar/postgre.php>.
- Henry Mintzberg, James Brian Quinn, John Voyer. 1997.** *El proceso estratégico: Conceptos, contextos y casos*. México : Prentice Hall Hispanoamericana, 1997.

- Herrera, Alvaro Alejandro Jaque. 2007.** MailxMail.com. *Migración de datos*. [En línea] 24 de Octubre de 2007. [Citado el: 26 de Enero de 2012.] <http://www.mailxmail.com/curso-migracion-datos/migracion-datos>.
- IBM. 2008.** Softek TDMF Unix. [En línea] 2008. [Citado el: 4 de Diciembre de 2011.] <http://www-05.ibm.com/services/es/sds/0905050200.html>.
- Innovative Systems, Inc. 2005.** Migración de datos. [En línea] 2005. [Citado el: 24 de Noviembre de 2011.] [http://espanol.innovativesystems.com/soluciones\\_de\\_negocios/migracion\\_de\\_datos/index.php](http://espanol.innovativesystems.com/soluciones_de_negocios/migracion_de_datos/index.php).
- Javier Lamas Collejo, Yasser Argelio Fuentes. 2010.** *Migración de la información del Sistema de Gestión Académica, Akademos, al Sistema de Gestión Académica de Pregrado*. La Habana : Universidad de las Ciencias Informáticas, Tesis, 2010.
- Kettle. 2007.** Xperimentos. *Kettle*. [En línea] 2007. [Citado el: 3 de Diciembre de 2011.] <http://www.xperimentos.com/2007/01/30/kettle-240/>.
- Leffingwell, D. Widrig. 2006.** *Managning Software Requeriments*. s.l. : Wesley Pearson Education, 2006.
- LinuxZone. 2010.** Ubuntu: descripción de Ubuntu, descarga, características de Ubuntu. [En línea] 2010. [Citado el: 12 de Enero de 2010.] <http://www.linuxzone.es/distribuciones-principales/ubuntu/>.
- Lozano, Viktor. 2008.** Manual de CodeIgniter. [En línea] 2008. [Citado el: 14 de Enero de 2012.] 2012.] <http://es.scribd.com/doc/50379909/14/Caracteristicas-de-CodeIgniter>.
- Lucía E. Domínguez Abreu, María del Carmen Paderni López, Otniel Barrera Palenzuela, Denis Derivet Thureauux. 2011.** *Migración integrada de datos para el Sistema Nacional de Salud*. La Habana : UCIENCIA, 2011.
- Marcos Ortíz Valmaseda, Susany Rodríguez. 2009.** *O2PMigration: Herramienta para la migración de base de datos Oracle a PostgreSQL*. La Habana : Universidad de las Ciencias Informáticas, Tesis, 2009.
- Martínez, Rafael. 2010.** PostgreSQL-es. *Sobre PostgreSQL*. [En línea] 2 de Octubre de 2010. [Citado el: 12 de Enero de 2012.] [http://www.postgresql.org.es/sobre\\_postgresql](http://www.postgresql.org.es/sobre_postgresql).
- Mato, Rosa María García. 1999.** *Diseño de base de datos*. 1999.
- Netbeans. 2012.** Bienvenido a NetBeans y [www.netbeans.org](http://www.netbeans.org). [En línea] 2012. [Citado el: 15 de Enero de 2012.] [http://netbeans.org/index\\_es.html](http://netbeans.org/index_es.html).
- PgAdmin. 2010.** Pgadmin PostgreSQL Tools. [En línea] 2010. [Citado el: 15 de Enero de 2012.] <http://www.pgadmin.org/>.

- PHP. 2012.** ¿Qué se puede hacer con PHP? [En línea] 11 de Mayo de 2012. [Citado el: 17 de Mayo de 2012.] <http://www.php.net/manual/es/intro-whatcando.php>.
- Pressman, Roger S. 2010.** *Ingeniería del Software: Un enfoque práctico. Quinta edición.* s.l. : Mcgraw Hill Publication, 2010.
- QData. 2012.** Talend Data Quality. [En línea] 8 de Febrero de 2012. [Citado el: 15 de Marzo de 2012.] <http://qdata.sisorg.com.mx/dqtalend.html>.
- Scribd. 2011.** PostgreSQL. [En línea] 2011. [Citado el: 13 de Enero de 2012.] <http://es.scribd.com/doc/36570462/postgreSQL-investigacion>.
- SwisSQL. 2009.** SwisSQL-Data Migration Tool. [En línea] 2009. [Citado el: 3 de Diciembre de 2011.] <http://www.swissql.com/products/datamigration/data-migration.html>.
- Talend Open Studio. 2010.** talend.com. [En línea] 2010. [Citado el: 3 de Diciembre de 2011.] <http://es.talend.com/products-data-integration/talend-open-studio.php>.
- Tamargo, Msc. Lic. Lourdes Cereza. 2011.** Primeros pasos para adentrarnos en el mundo de los Data Warehouse. [En línea] Abril-Junio de 2011. [Citado el: 10 de Marzo de 2012.] [http://www.betsime.disaic.cu/secciones/tec\\_ja\\_04.htm](http://www.betsime.disaic.cu/secciones/tec_ja_04.htm).
- Visual Paradigm. 2009.** [En línea] 2009. [Citado el: 25 de Enero de 2012.] <http://www.visual-paradigm.com>.
- Vivar, Xiomara Carolina López. 2003.** *Sistemas de bases de datos activas: Disparadores y reglas.* Guatemala : Universidad de San Carlos de Guatemala, 2003.
- Zaragoza, Arthur. 2010.** Migración de datos. *Técnicas de migración de datos.* [En línea] 2010. [Citado el: 25 de Febrero de 2012.] <http://es.scribd.com/doc/51334306/Migracion-de-datos>.



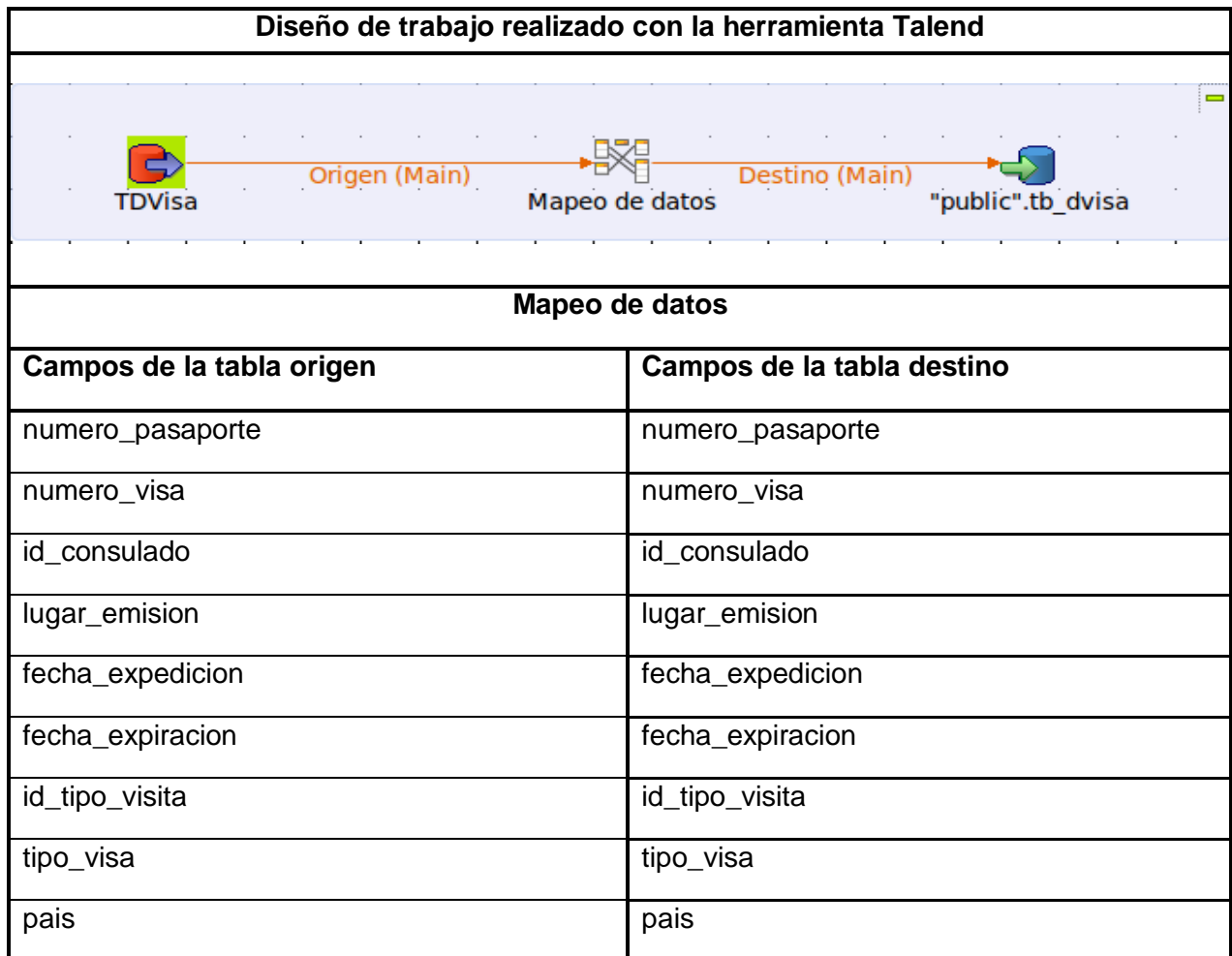
### Bibliografía consultada

- **Yunier Rodríguez Lucas, Héctor Alfredo Zúñiga Baldemira.** *Implementación de los procesos de Integración de datos de los sistemas SAIME e INTTT para el Cuerpo de Investigaciones Científicas, Penales y Criminalísticas de la República Bolivariana de Venezuela.* Universidad de las Ciencias Informáticas. Ciudad de la Habana. 2010. Tesis.
- **Pentaho.** Pentaho Open Source Business Intelligence: Kettle Project. [En línea] 2009. [Citado el 25 de Enero de 2012.] <http://kettle.pentaho.org>
- **Talend.** Talend Open Data Solutions. [En línea] 2009. [Citado el 10 de Febrero de 2012.] <http://es.talend.com/index.php>.
- Pressman, Roger S. *Ingeniería del Software: Un enfoque práctico.* Quinta edición. . s.l.: McGraw Hill Publication, 2010.
- **Keyrus.** Talend - Open Data Solutions. [En línea] 2007. [Citado el: 20 de Marzo de 2012.] <http://www.keyrus.es/showpage2.php?PAG=19&id=4>.
- **Kimball, Ralph.** *The Data Warehouse Toolkit.* s.l.: WILEY PUBLISHING, 1996.
- **Kimball, Ralph y Caserta, Joe.** *The DW ETL Toolkit Extracting, Cleaning, Conforming, and Delivered Data.* Canada: Wiley Publishing. 2004.
- **Rafael Caballero Roldán.** *Evolución de las Bases de Datos.* [Citado el: 12 de Febrero de 2012.] [www.fdi.ucm.es/profesor/ricardo/ei2/evolucionBD.pdf](http://www.fdi.ucm.es/profesor/ricardo/ei2/evolucionBD.pdf).
- **Dr. C. María Elena Guardo García.** *Revista Pedagogía Universitaria. LOS COMPONENTES DEL DISEÑO TEÓRICO DE LA INVESTIGACIÓN CIENTÍFICA. UNA REFLEXIÓN PRAXIOLÓGICA.* 2009. Vol. XIV No. 3.
- **D., Leffingwell D. y Widrig.** *Managing Software Requirements.* s.l.: Wesley Pearson Education, 2006.
- **Castaño, Paula Andrea Cárdenas.** 2005. *Estudio de la evolución de los sistemas open source de BI durante la última década.* 2005.

**Anexos**

**Anexo 1:** Descripción de los procedimientos de migración realizados.

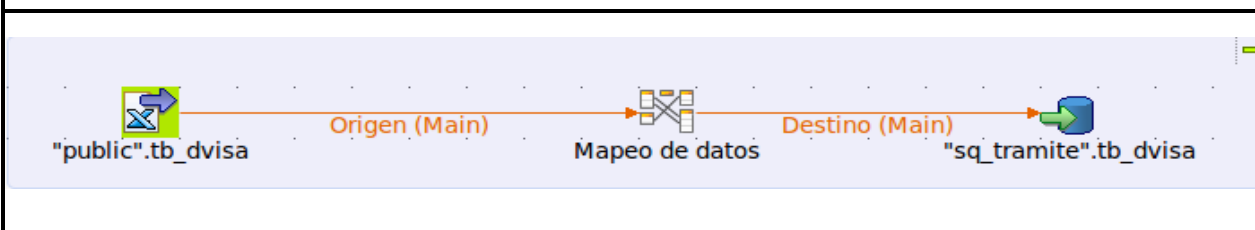
**Tabla 1:** Descripción de la solución para la migración de las visas desde el DRI hacia el SGCI.



**Descripción de la migración:** Utilizando la herramienta Talend se creó un diseño de trabajo que permitió cargar los datos desde la BD del DRI hacia la tabla temporal creada en la BD del SGCI. Luego se realizó el mapeo de los datos utilizando el componente tMap (Mapeo de datos) que brinda la facilidad de enlazar los campos entre ambas tablas, insertándose 2929 visas en este primer proceso. Después se realiza el proceso de transformación y limpieza de los datos de la siguiente manera:

- Se añadió el campo `codigo_pais` debido a que es un dato obligatorio a insertar en la tabla del negocio. Para darle el valor real al mismo se realizaron consultas en el lenguaje SQL, utilizando el campo `país` para obtener su código en el nomenclador `tb_npais` del módulo de cooperación.
- Se modificó el valor del campo `id_tipo_visa` debido a que los valores que contiene la tabla temporal son diferentes a los registrados en el nomenclador `tb_ntipo_visa` del módulo de trámites, realizando consultas que obtienen el valor real teniendo en cuenta el campo `tipo_visa` de la tabla temporal.
- Se eliminó la información que no era necesario migrar hacia el SGCI.

Una vez realizadas las transformaciones anteriores, se utiliza nuevamente la herramienta para migrar los datos hacia la tabla del negocio. Dicho proceso se muestra en la siguiente imagen:



**Mapeo de los datos**

Campos de la tabla origen	Campos de la tabla destino
numero_pasaporte	numero_pasaporte
numero_visa	numero_visa
id_consulado	id_consulado
lugar_emision	lugar_emision
fecha_expedicion	fecha_expedicion
fecha_expiracion	fecha_expiracion

id_tipo_visita	id_tipo_visita
codigo_pais	codigo_pais
<b>Observación:</b> Los campos pais y tipo_visa no se tendrán en cuenta en este proceso de migración debido a que no se registran en la tabla sq_tamite.tb_dvisa del SGCI.	
<b>Descripción:</b> Haciendo uso de la herramienta se creó un diseño de trabajo que permitió migrar los datos desde la tabla temporal hacia la tabla del negocio del SGCI, utilizando el mapeo de datos para el enlace de los campos de la tabla origen con los campos de la tabla destino.	
<b>Resultados:</b> Se obtuvo como resultado la migración de todas las visas almacenadas en la tabla temporal del SGCI.	

**Tabla 2:** Descripción de la solución para la migración de los pasaportes desde el DRI hacia el SGCI.

Diseño de trabajo realizado con la herramienta Talend	
Mapeo de datos	
Campos de la tabla origen	Campos de la tabla destino
nombres	nombre_persona
apellidos	apellidos
numero_pasaporte	numero_pasaporte
fecha_expedicion	fecha_expedicion
fecha_vencimiento	fecha_vencimiento
ubicacion	ubicacion
codigo	codigo
tipo_pasaporte	tipo_pasaporte

**Descripción de la migración:** Utilizando la herramienta Talend se creó un diseño de trabajo que permitió cargar los datos desde la BD del DRI hacia la tabla temporal creada en la BD del SGCI. Luego se realizó el mapeo de los datos utilizando el componente tMap (Mapeo de datos) que brinda la facilidad de enlazar los campos entre ambas tablas, insertándose 4362 pasaportes en este primer proceso. Después se realiza la transformación y limpieza de los datos de la siguiente manera:

- Se añadió el campo `id_tipo_pasaporte` el cual es necesario registrar en el SGCI. Para darle el valor real se realizaron consultas en el lenguaje SQL, obteniéndose el dato desde la tabla `tb_ntipo_pasaporte` del módulo de trámites teniendo en cuenta el campo `tipo_pasaporte` registrado en la tabla temporal.
- Se añadió el campo `id_ubicacion_pasaporte` y para actualizar su valor se realizaron consultas en el lenguaje SQL, seleccionando el dato real desde la tabla `tb_nubicacion_pasaporte` del módulo de trámites, teniendo en cuenta el valor del campo `ubicacion` de la tabla temporal.

**Observación:** Se encontraron errores en la información los cuales se mencionan a continuación:

- Valor fuera de rango para el campo `numero_pasaporte` (varchar mayor que 7).
- Valores nulos en campos necesarios a insertar en el SGCI.

**Tratamiento de errores encontrados:**

- Los pasaportes con valor mayor que 7 y los campos con valor nulo, fueron verificados con el cliente obteniéndose el dato válido.

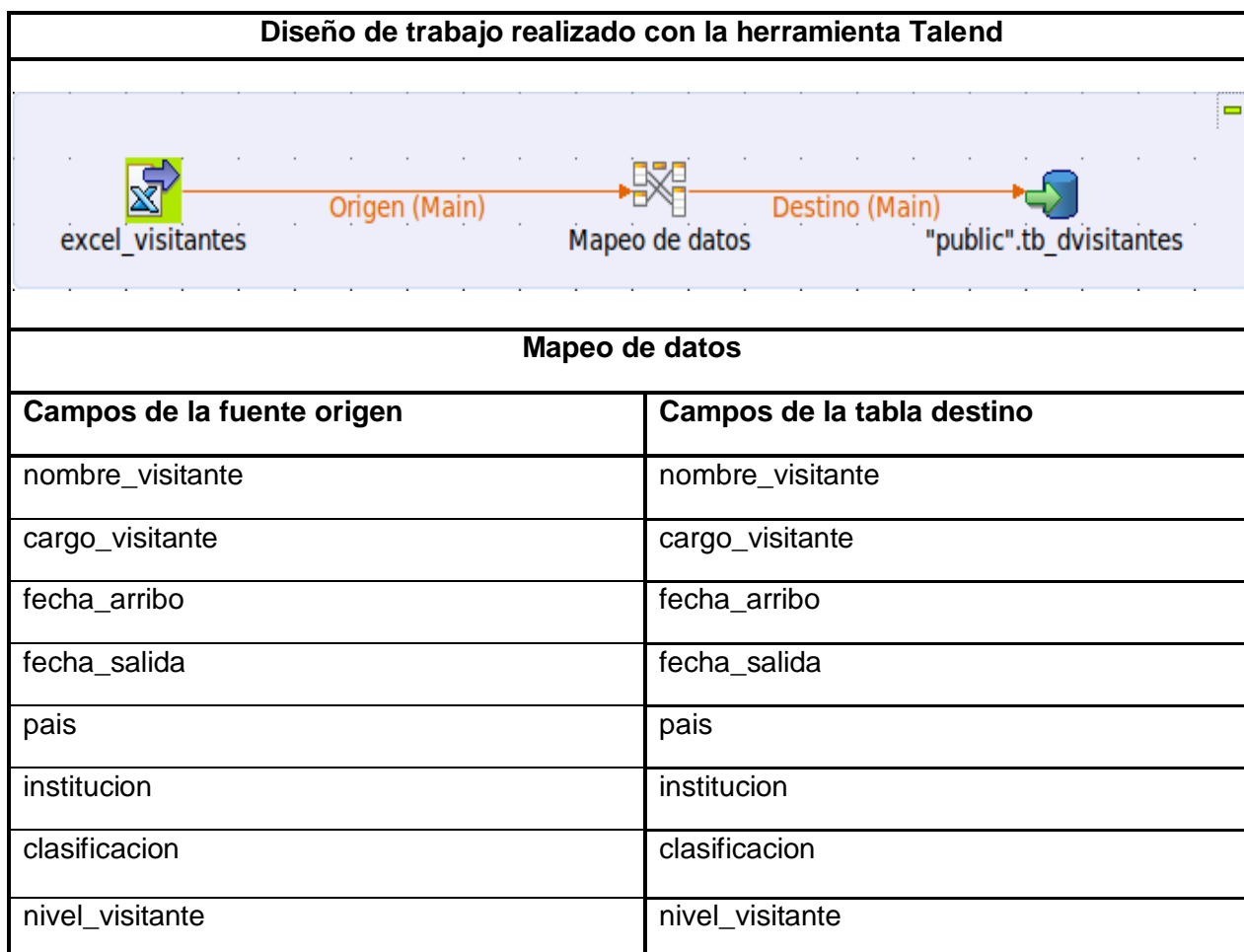
**Descripción del algoritmo implementado:** Una vez realizadas las transformaciones anteriores, se implementó un algoritmo que permitiera trasladar los datos contenidos en la tabla temporal hacia las tablas del negocio del SGCI. El mismo inicia con la búsqueda en el núcleo, de las personas referentes a los pasaportes. En caso de que ya exista, se guarda el `id_persona` y se procede entonces a insertarlo en la tabla `tb_dpasaporte` del módulo de trámites del SGCI, junto con los datos que tiene asociados. De no encontrarse la persona, entonces se imprime en el navegador Web la información de la misma para analizar al final del procedimiento si contiene algún error.

**Resultados de la primera iteración:** En la primera iteración del algoritmo implementado se obtuvo como resultado la migración de 4350 pasaportes quedando solamente 12 sin migrar. Estos últimos no fueron migrados debido a que no aparecieron en el núcleo las personas

asociadas a ellos. Se utilizaron luego los filtros con que cuenta el núcleo, realizando posibles combinaciones de búsqueda con los nombres de las personas hasta encontrar 7 de ellos. Entonces se actualizó la información en la tabla temporal. Los demás pasaportes fueron consultados con el cliente y finalmente se decidió insertar las personas no encontradas, en la tabla tb\_dpersona del núcleo.

**Resultados de la segunda iteración:** Luego de haber corregido los errores encontrados en los datos no migrados en la primera iteración, se realizó una segunda ejecución del algoritmo obteniéndose como resultado final, la migración de los 12 pasaportes que habían quedado anteriormente.

**Tabla 3:** Descripción de la solución para la migración de los visitantes desde documentos Excel hacia la BD de prueba.



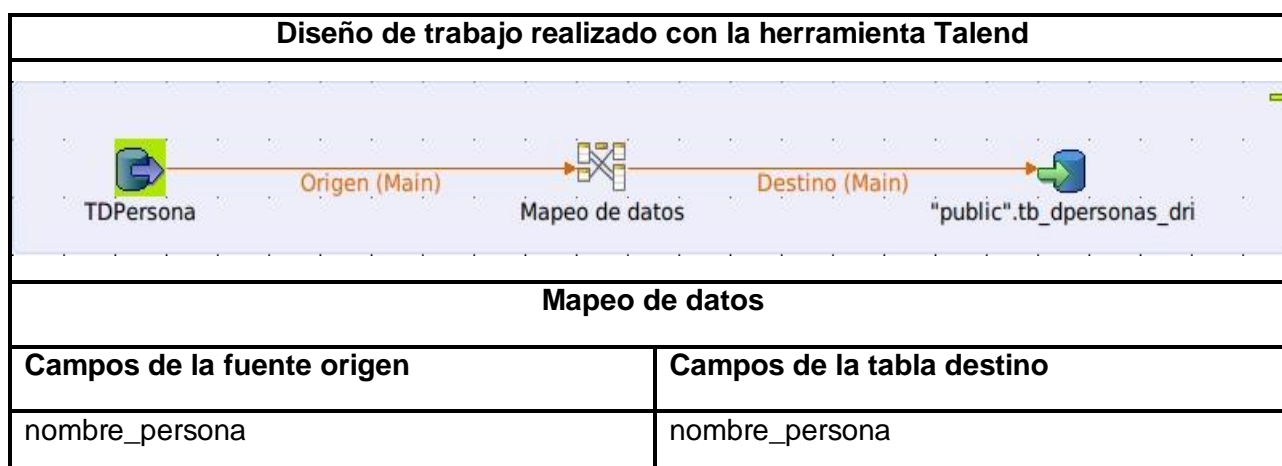
organismo_solicitante	organismo_solicitante
<p><b>Descripción de la migración:</b> Primero se organizaron los documentos que contienen los datos de los visitantes en archivos de tipo Excel. Luego utilizando la herramienta Talend se creó un diseño de trabajo que permitió cargar la información hacia la tabla temporal creada en la BD de prueba, la cual es una copia o backup de la BD del SGCI. Después se llevó a cabo el mapeo de los datos empleando el componente tMap (Mapeo de datos) que brinda la facilidad de enlazar los campos entre ambas tablas. Se insertaron 1840 visitantes en este primer proceso. Seguidamente se realiza la transformación y limpieza de los datos de la siguiente manera:</p> <ul style="list-style-type: none"> <li>• Se le añadió a la tabla temporal el campo nivel_visitante y para actualizar su valor se realizaron consultas en el lenguaje SQL, obteniendo el nivel del visitante registrado en la tabla tb_nnivel_visitante del módulo de visita asociado al cargo del visitante.</li> <li>• Se añadió además el campo id_nivel_visitante cuyo valor se infiere de nivel_visitante, obteniéndose el dato desde la tabla tb_nnivel_visitante a través de consultas en el lenguaje SQL.</li> <li>• Se eliminó la información que no era necesario migrar.</li> <li>• Finalmente se agregó el campo codigo_pais. El mismo fue actualizado obteniéndose el valor desde la tabla tb_npais del módulo de cooperación, teniendo en cuenta el país registrado en la tabla temporal para cada uno de los visitantes.</li> </ul>	
<p><b>Observación:</b> Se encontraron errores en la información los cuales se mencionan a continuación:</p> <ul style="list-style-type: none"> <li>• Valores nulos en los campos fecha_arriba y codigo_pais necesarios a insertar en el sistema final.</li> </ul>	
<p><b>Tratamiento de errores encontrados:</b></p> <ul style="list-style-type: none"> <li>• Se elaboró un documento con los visitantes que contenían los errores anteriormente expuestos y se les entregó al cliente, el cual verificó la información y se pudo entonces actualizar los datos en la tabla temporal.</li> </ul>	

**Descripción del algoritmo implementado:** Cuando estuvo lista la información entonces se implementó un algoritmo que permitiera migrarla hacia las tablas del negocio del sistema final. Su funcionamiento de manera general se describe a continuación:

Inicialmente se verifica en la tabla visitante si el nombre de la persona que se va a migrar ya ha sido registrado anteriormente, si es así entonces se obtiene el id\_visitante, si no, se inserta y de igual modo se guarda el id\_visitante con que se registró y se procede a comprobar en la tabla que almacena las visitas si existe alguna a la que el visitante esté asociado. En caso de que el resultado sea positivo se obtiene el id\_visita, de no ser así se inserta una nueva visita y se obtiene igualmente el id\_visita con que se guarda en dicha tabla. Finalmente se realiza la inserción en la tabla tb\_rvisita\_visitante del id\_visitante y el id\_visita asociado al mismo, donde queda constancia de las visitas y los visitantes que participan en las mismas. Todas estas transformaciones se realizaron en el módulo de visita.

**Resultados de la primera iteración:** Para la migración de los visitantes sólo fue necesario realizar una sola iteración del algoritmo obteniéndose como resultado el traslado de 1783 visitantes. Esto se debe a que existían nombres duplicados en la tabla temporal y en la del negocio sólo se registra este dato una vez. Sin embargo, esto no significa que existan errores en la información pues un nombre puede estar repetido siempre y cuando la fecha de arribo del visitante no sea la misma. Por lo que la evidencia de que todos los visitantes a migrar fueron registrados en el sistema, está en la tabla tb\_rvisita\_visitante que guarda el id\_visitante y el id\_visita en la que participa.

**Tabla 4:** Descripción de la solución para la migración de los datos personales desde el DRI hacia el SGCI.





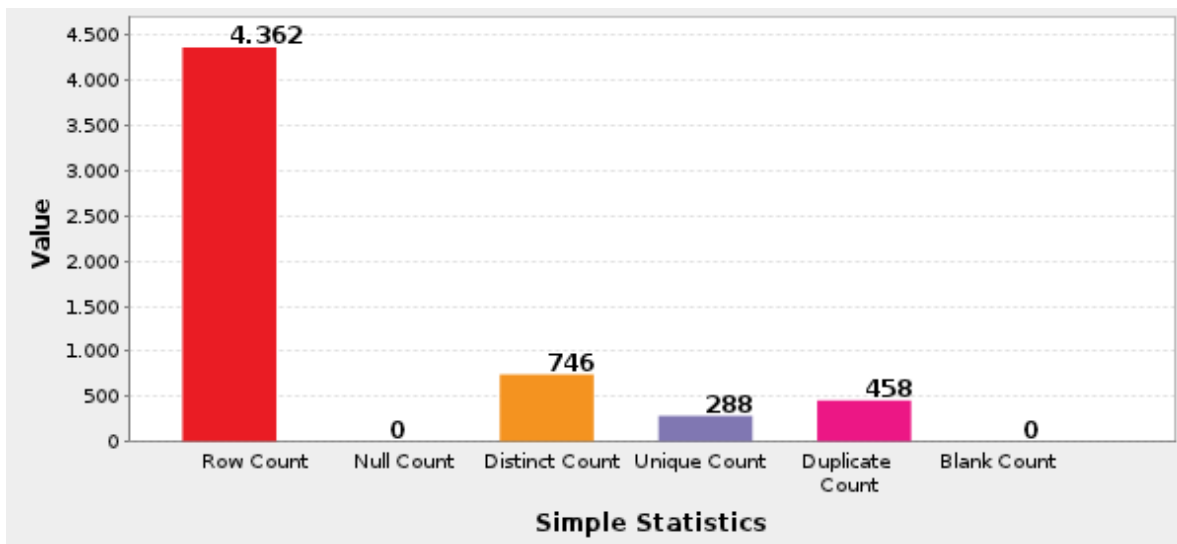
color_cabello	color_cabello
estatura	estatura
padre	nombre_padre
madre	nombre_madre
mun_nac	mun_nacimiento
mun	munic_reside
circunscripción	circunscripción
localidad	localidad
finca	finca
cpa	cpa
apto	apto
km	km
numero	numero_apto
entre	entre
entre2	entre2
carretera	carretera
calle	calle

**Descripción de la migración:** Utilizando la herramienta Talend se creó un diseño de trabajo que permitió cargar la información hacia la tabla temporal creada en la BD del SGCI. Después se llevó a cabo el mapeo de los datos empleando el componente tMap (Mapeo de datos) que brinda la facilidad de enlazar los campos entre ambas tablas. Se insertaron 3707 personas con todos los datos que recoge la tabla TDPersonas del DRI. Una vez realizado este procedimiento se llevó a cabo la transformación y limpieza de los datos:

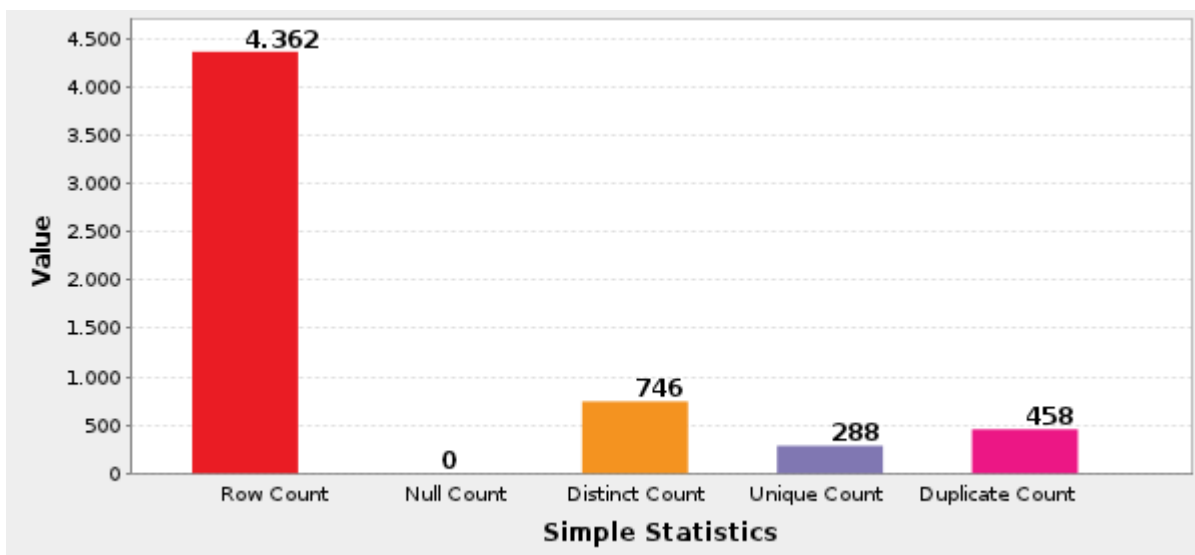
- Se le añadió a la tabla temporal el campo id\_color\_cabello, y para actualizar su valor se realizaron consultas en el lenguaje SQL, obteniéndose el dato de la tabla tb\_ncolor\_cabello perteneciente al módulo de trámite, teniendo en cuenta el campo color\_cabello de la tabla temporal.
- Se añadió además el campo id\_mun\_nac cuyo valor se infiere del campo

<p>mun_nacimiento, obteniéndose a través de consultas en el lenguaje SQL.</p> <ul style="list-style-type: none"><li>• Se añadió además el campo id_mun_reside cuyo valor se infiere del campo mun_reside, obteniéndose a través de consultas en el lenguaje SQL.</li><li>• Se eliminó la información que no era necesario migrar hacia el SGCI.</li></ul>
<p><b>Observación:</b> Se encontraron errores en la información tales como:</p> <ul style="list-style-type: none"><li>• Errores ortográficos en los nombres de las personas a migrar.</li><li>• Ausencia de datos en los nombres de las personas a migrar.</li></ul>
<p><b>Tratamiento de errores encontrados:</b></p> <ul style="list-style-type: none"><li>• Se elaboró un documento con los nombres de las personas que contenían los errores anteriormente expuestos y se les entregó al cliente, el cual verificó la información y se pudo entonces actualizar los datos en la tabla temporal.</li></ul>
<p><b>Descripción del algoritmo implementado:</b> Una vez preparada la información en la tabla temporal para ser migrada a las tablas del negocio del SGCI, se implementó un algoritmo que permitiera realizar este proceso. A continuación se describe de manera general como funciona el mismo:</p> <p>Primeramente se realiza la búsqueda en el núcleo de las personas que se desean migrar para verificar si ya han sido registradas. Si la búsqueda resulta satisfactoria entonces se guarda el id_persona de la misma y se inserta en la tabla tb_ddatos_misionero junto con los demás datos referentes a él que requiere la misma y están almacenados en la tabla temporal. Lo mismo se hace en las tablas tb_ddireccion, tb_ddireccion_nacimiento, tb_ddatos_contacto que recogen toda la información referente a los datos personales. En caso de que el resultado de la búsqueda sea negativo se imprime la información en el navegador Web para posteriormente analizarla y resolver las causas por las que no se migraron esos datos.</p>
<p><b>Resultados de la primera iteración:</b> En la primera iteración del algoritmo se obtuvo como resultado la migración de 3697 datos personales quedando un total de 10 por migrar. Luego de analizada la información se obtuvo que las personas no aparecieron en el núcleo, por lo que se consultó con el cliente y la solución al problema fue insertarlas en el mismo.</p>
<p><b>Resultados de la segunda iteración:</b> En la segunda iteración del algoritmo se obtuvo como resultado la migración de los datos personales que quedaron anteriormente, llegando entonces a la completa migración de los datos contenidos en la tabla temporal.</p>

**Anexo 2:** Gráficos de comparación de los campos que fueron migrados hacia la tabla tb\_dpaspaporte del SGCI.



**Fig 41.** Campo fecha\_vencimiento del pasaporte de la tabla TDPaspaporte del DRI.



**Fig 42.** Campo fecha\_vencimiento del pasaporte de la tabla tb\_dpaspaporte del SGCI.

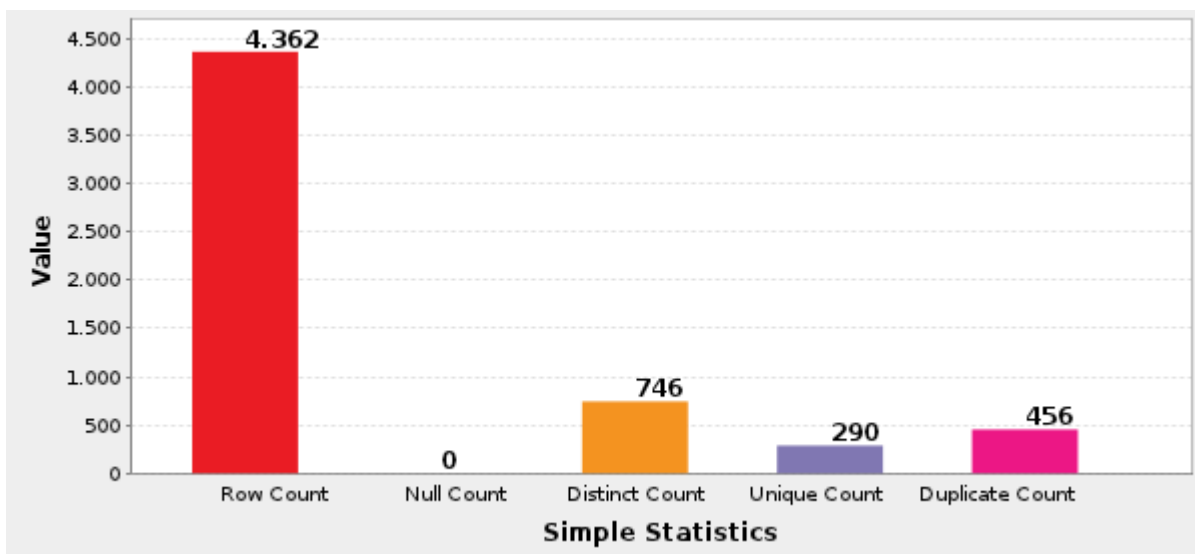


Fig 43. Campo fecha\_expedicion del pasaporte de la tabla TDPasaporte del DRI.

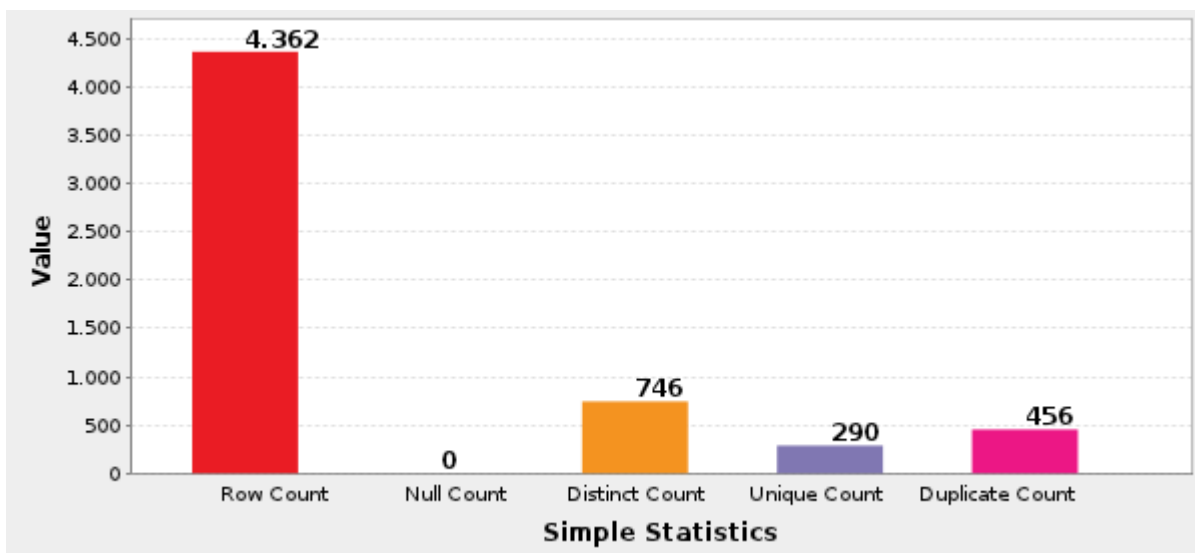
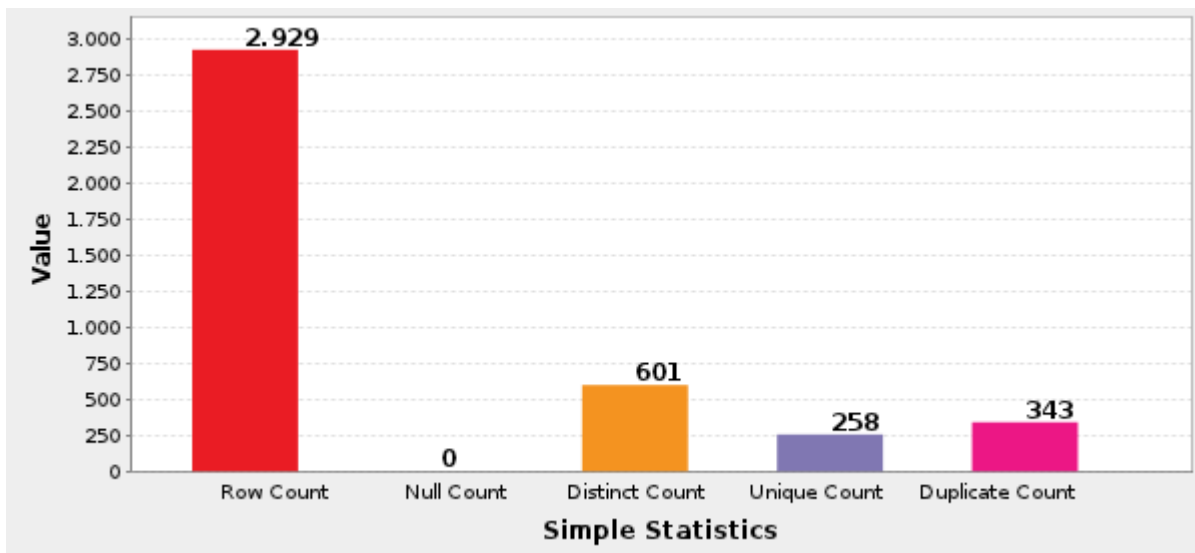
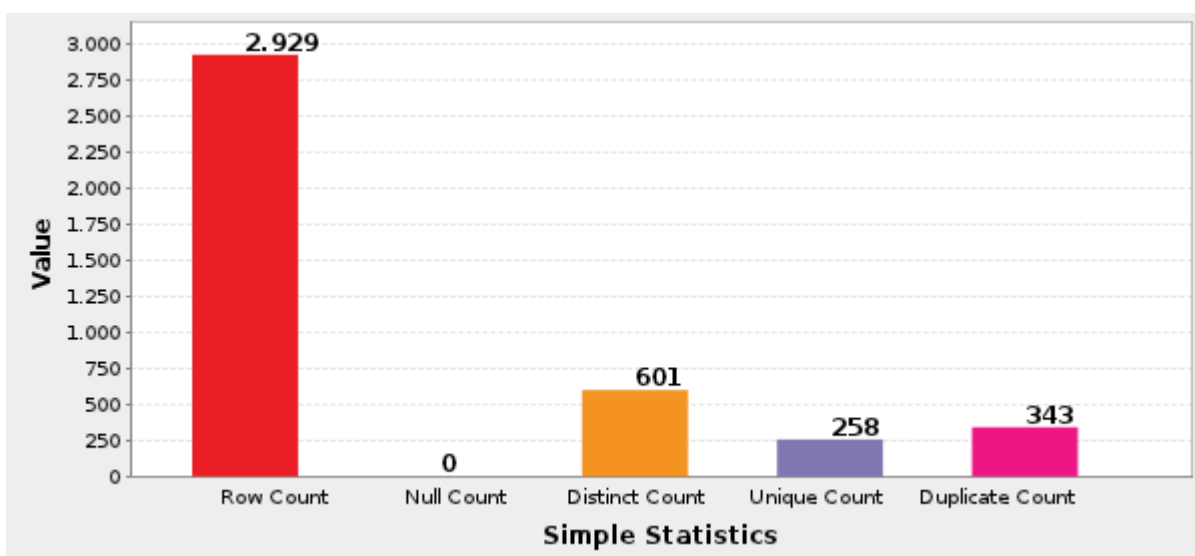


Fig 44. Campo fecha\_expedicion del pasaporte de la tabla tb\_dpasaporte del SGCI.

**Anexo 3:** Gráficos de comparación de los campos que fueron migrados hacia la tabla tb\_dviza del SGCI.



**Fig 45.** Campo fecha\_expedicion de la tabla TDViza del DRI.



**Fig 46.** Campo fecha\_expedicion de la tabla tb\_dviza del SGCI.

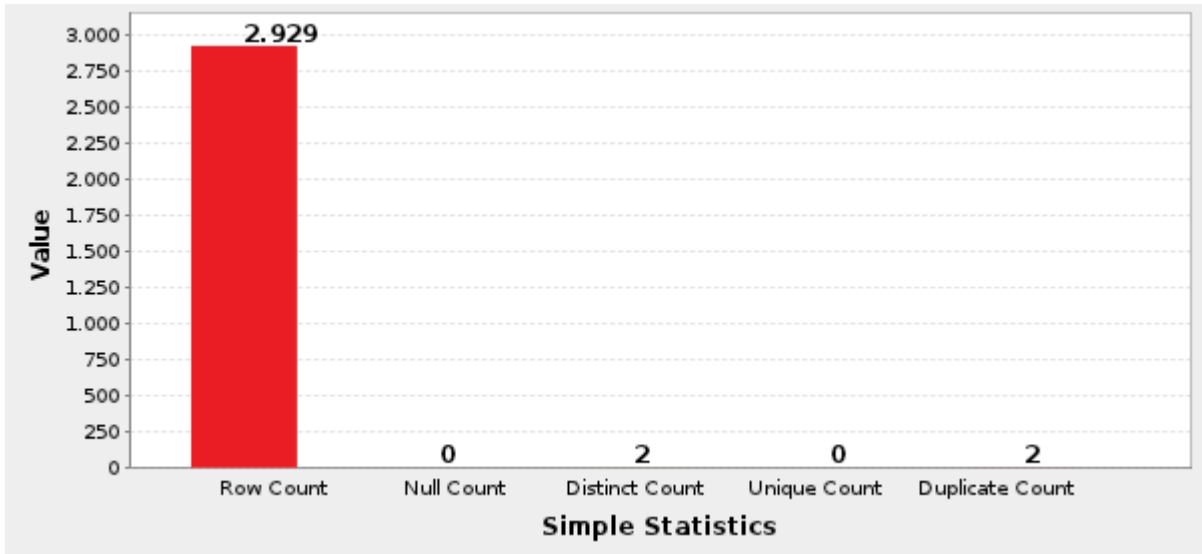


Fig 47. Campo id\_tipo\_visa de la tabla TDVisa del DRI.

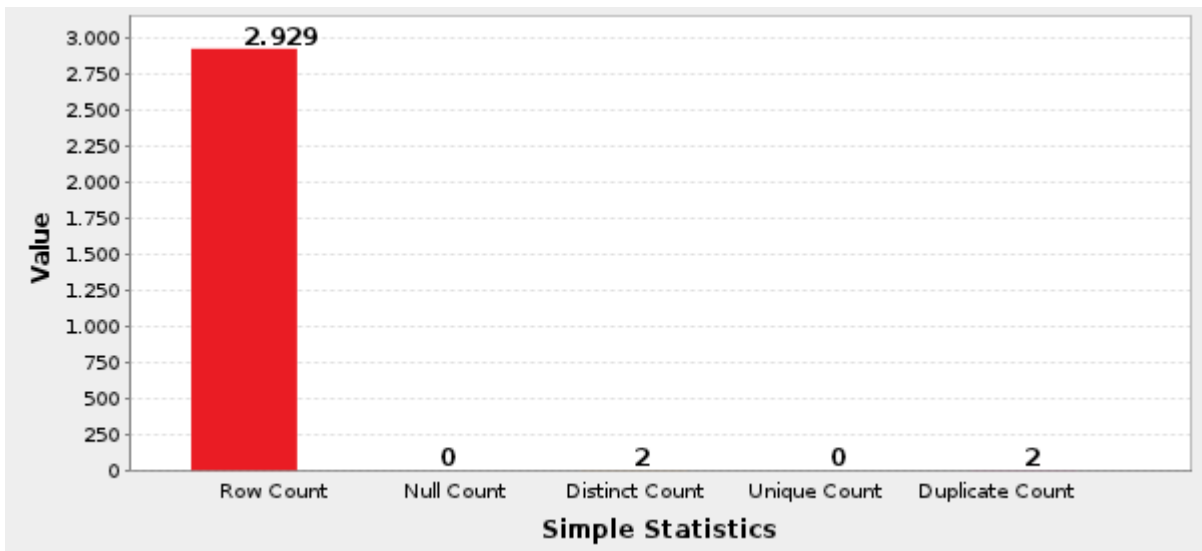
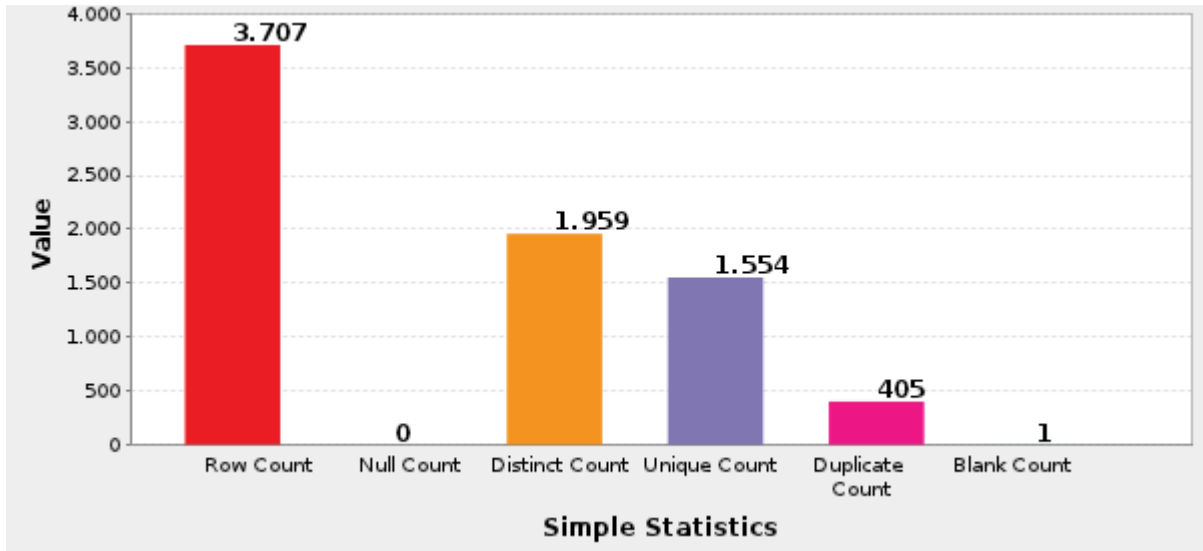
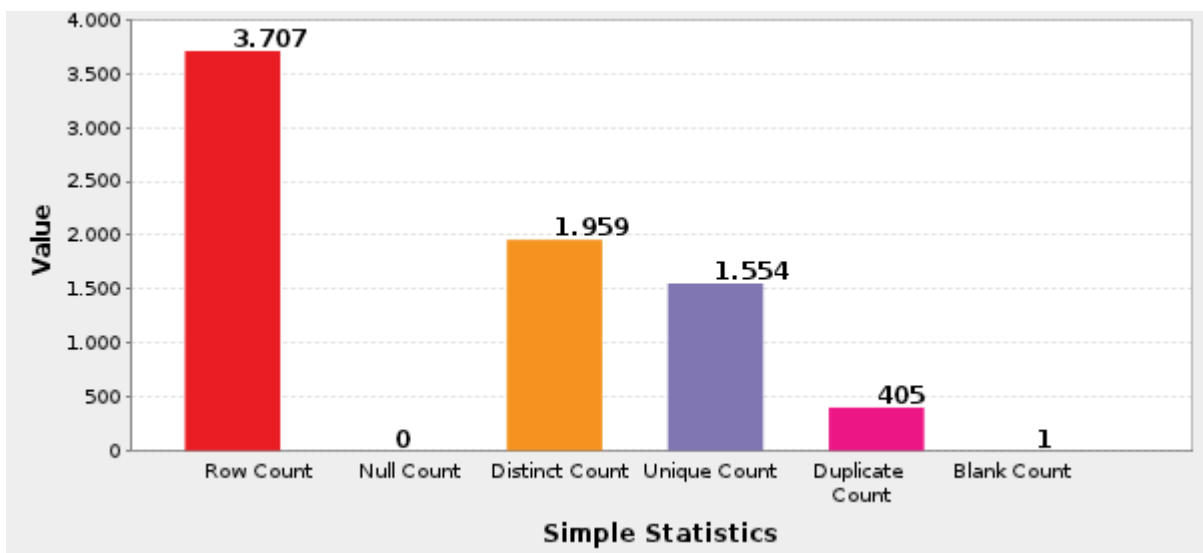


Fig 48. Campo id\_tipo\_visa de la tabla tb\_dvisa del SGCI.

**Anexo 4:** Gráficos de comparación de los campos que fueron migrados hacia la tabla tb\_ddatos\_misionero del SGCI.



**Fig 49.** Campo nombre\_madre de la tabla TDPersona del DRI.



**Fig 50.** Campo nombre\_madre de la tabla tb\_ddatos\_misionero del SGCI.

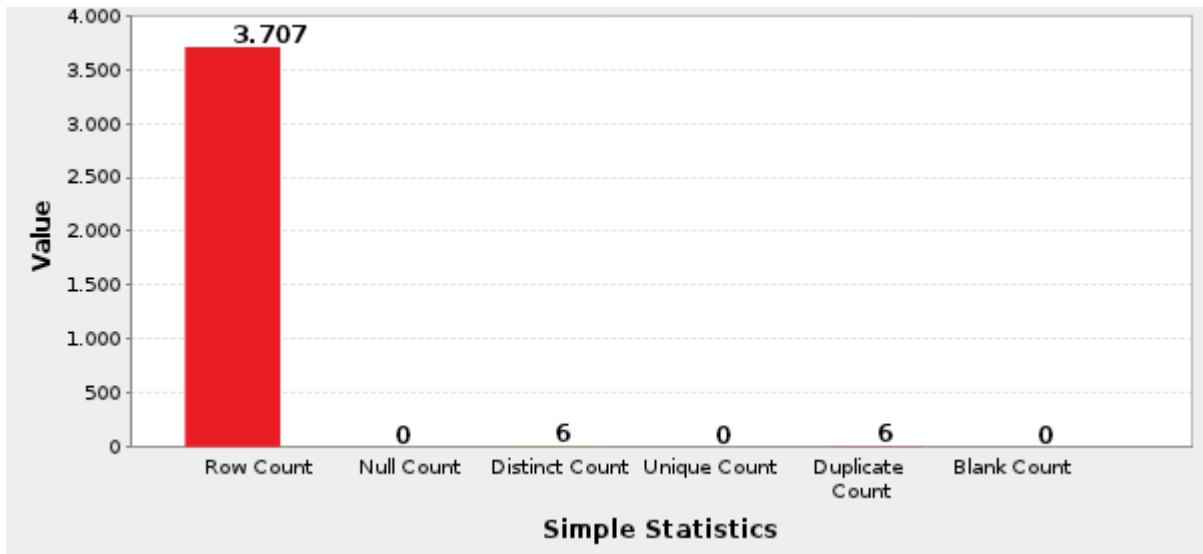


Fig 51. Campo color\_cabello de la tabla TDPersona del DRI.

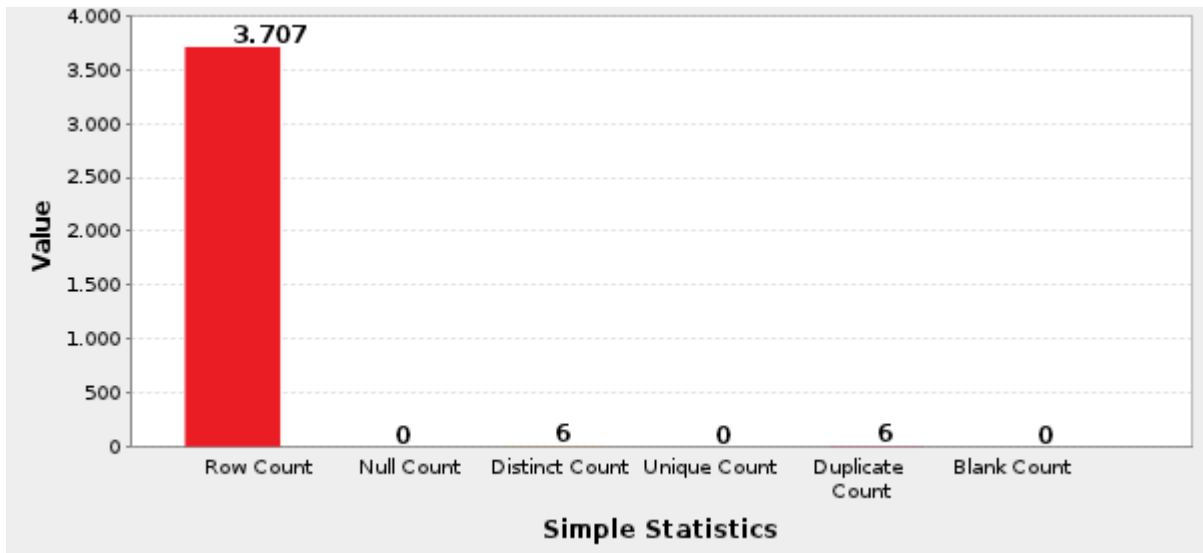


Fig 52. Campo color\_cabello de la tabla tb\_ddatos\_misionero del SGCI.



**Anexo 5:** Gráficos de comparación de los campos que fueron migrados hacia la tabla tb\_dhabilitacion del SGCI.

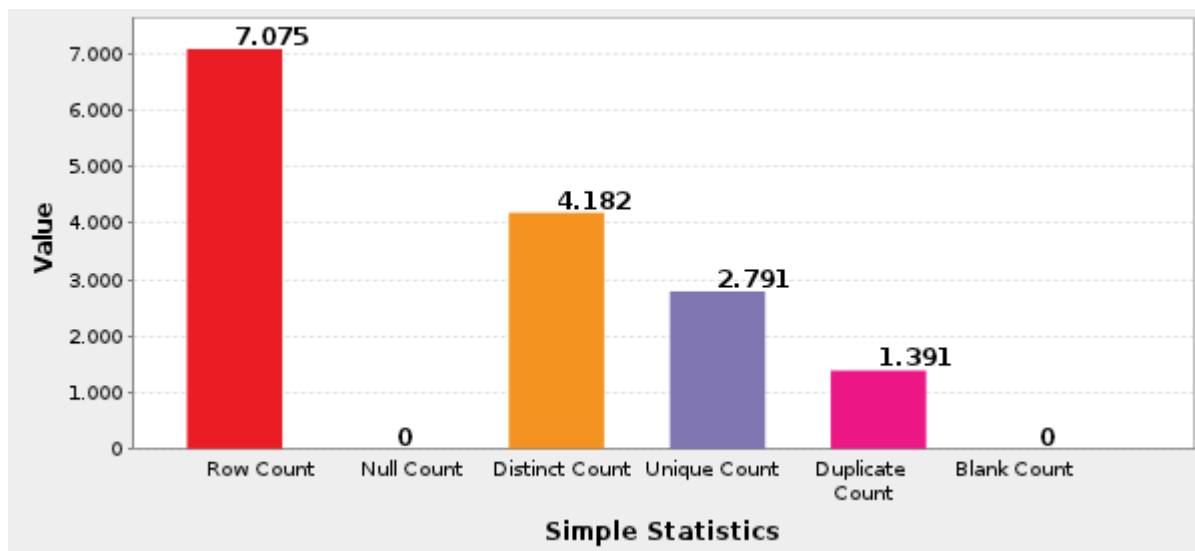


Fig 53. Campo numero\_pasaporte\_habilitacion de la tabla TDHabilitacion del DRI.

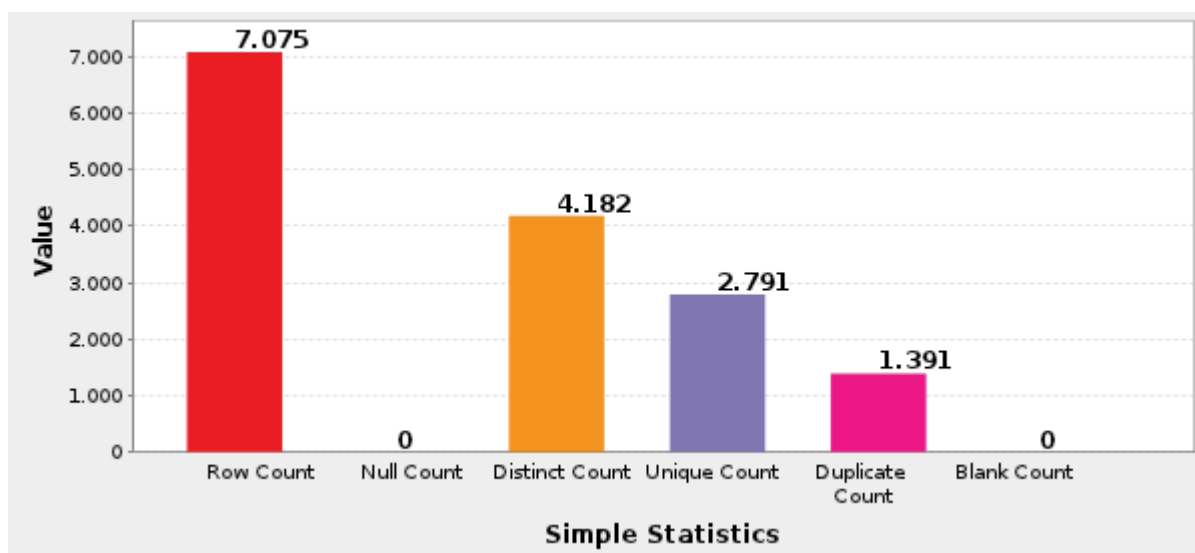


Fig 54. Campo numero\_pasaporte\_habilitacion de la tabla tb\_dhabilitacion del SGCI.

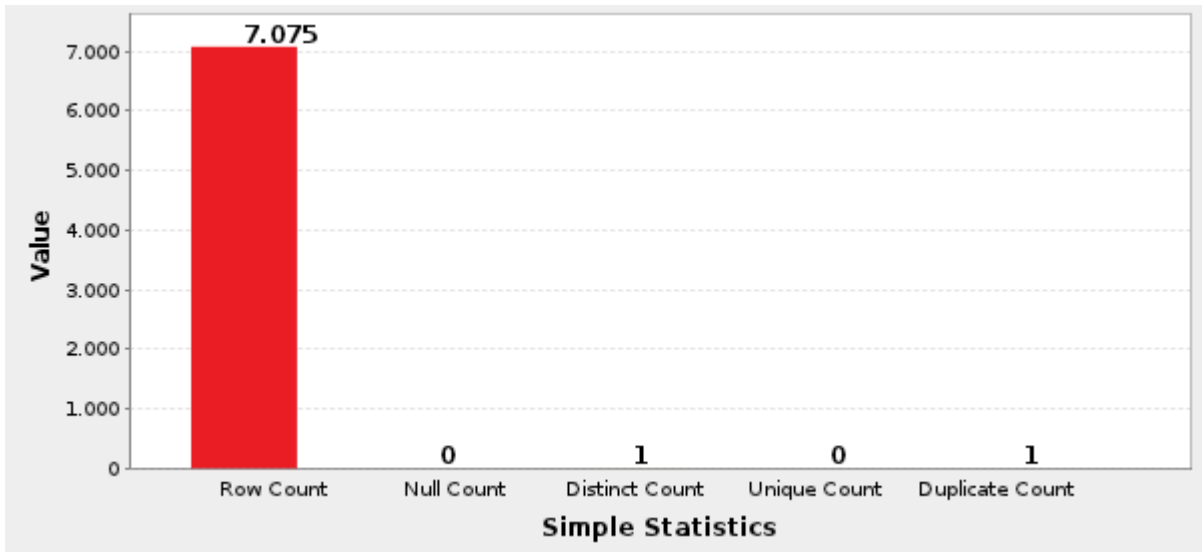


Fig 55. Campo id\_tipo\_permiso\_salida\_tramite de la tabla TDHabilitacion del DRI.

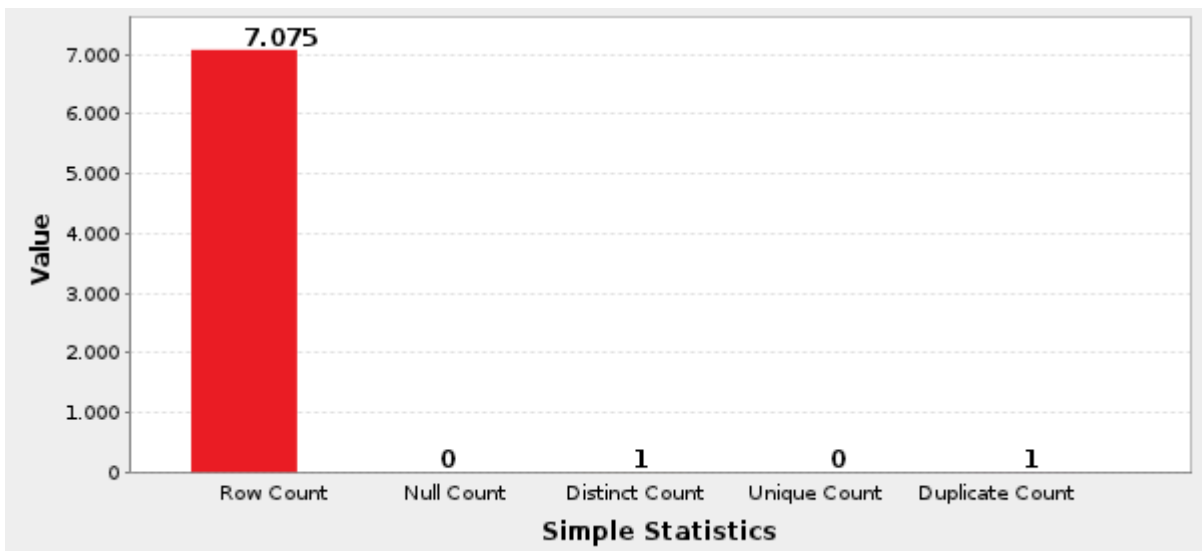


Fig 56. Campo id\_tipo\_permiso\_salida\_tramite de la tabla tb\_dhabilitacion del SGCI.

**Anexo 6:** Gráficos de comparación de los campos que fueron migrados hacia la tabla tb\_ddviaje del SGCI.

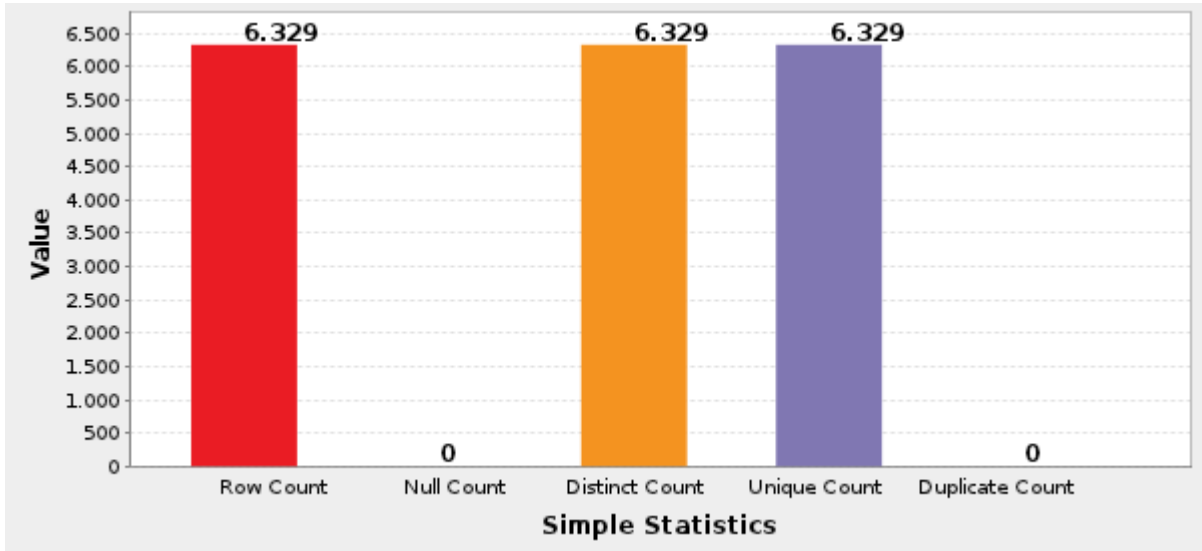


Fig 57. Campo id\_solicitud\_mision de la tabla temporal tb\_dviaje del SGCI.

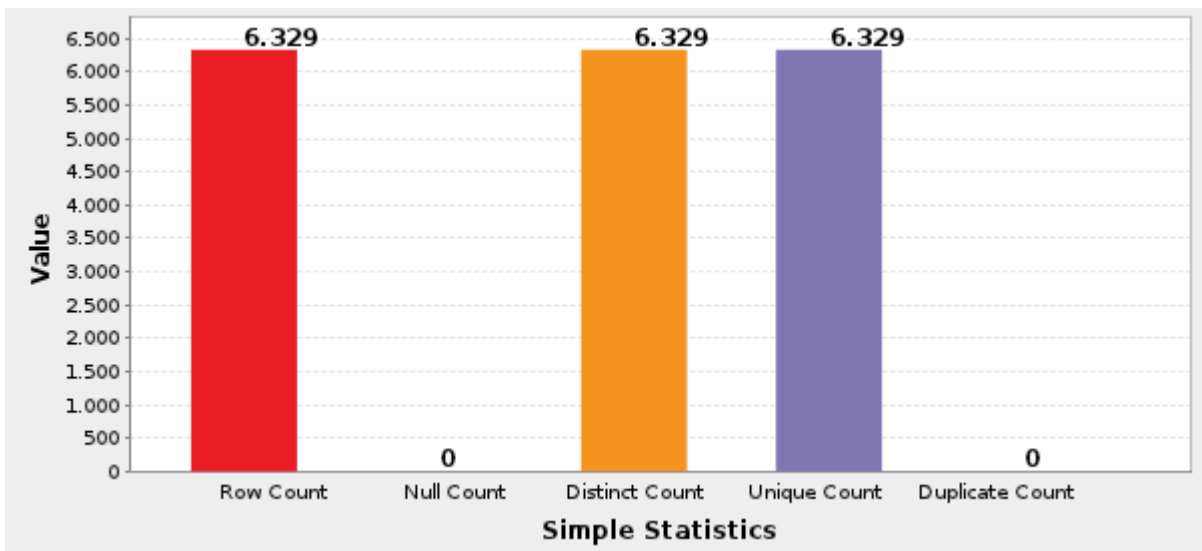


Fig 58. Campo id\_solicitud\_mision de la tabla tb\_dviaje del SGCI.

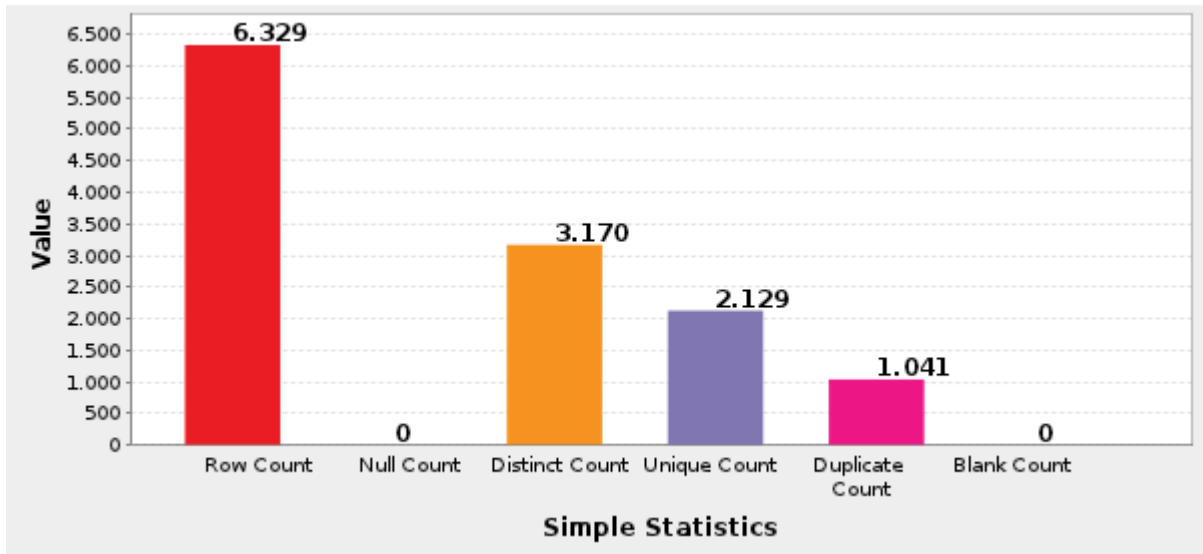


Fig 59. Campo id\_persona de la tabla temporal tb\_dviaje del SGCI.

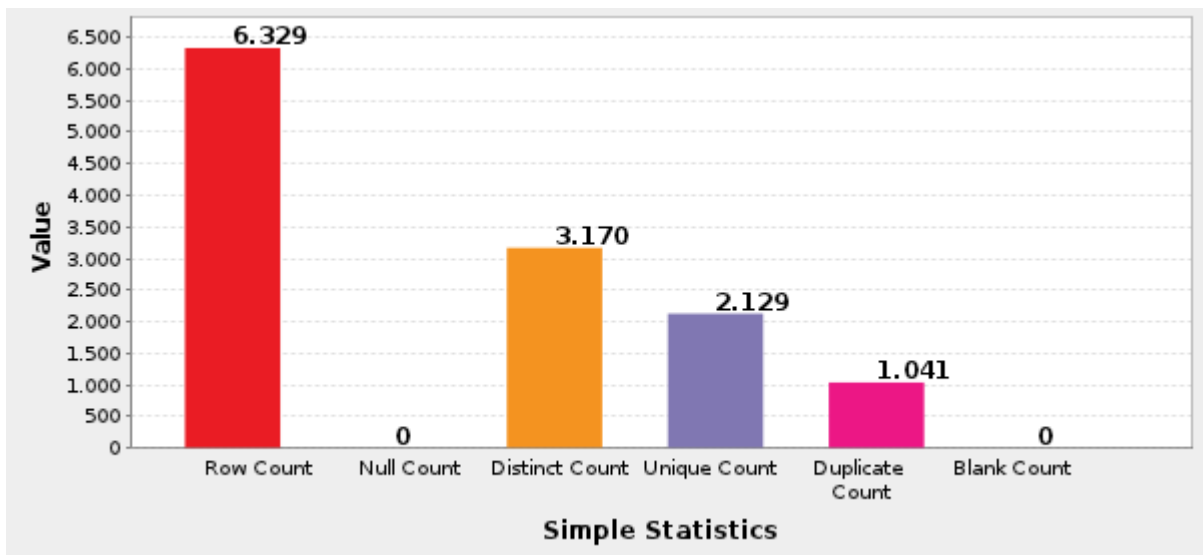


Fig 60. Campo id\_persona de la tabla tb\_dviaje del SGCI.

**Anexo 7:** Gráficos de comparación de los campos que fueron migrados hacia la tabla tb\_dvisitantes de la BD de prueba.

Como se podrá observar en las gráficas a continuación, que muestra el comportamiento del campo id\_persona tanto en la tabla temporal de visitantes como en la tabla tb\_dvisitante del negocio, no existe una igualdad absoluta atendiendo a los indicadores de comparación. Esto se debe a que en esta última tabla se registra el nombre del visitante una sola vez, por lo tanto el comportamiento de los demás campos relacionados con el visitante se ve afectado.

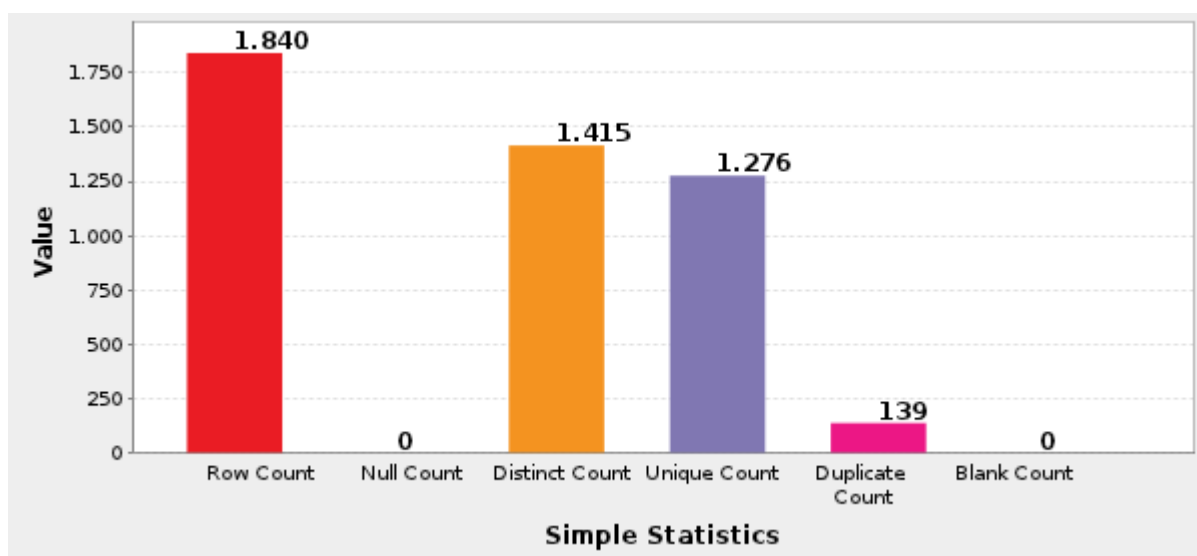


Fig 61. Campo id\_persona de la tabla temporal tb\_dvisitante de la BD de prueba.

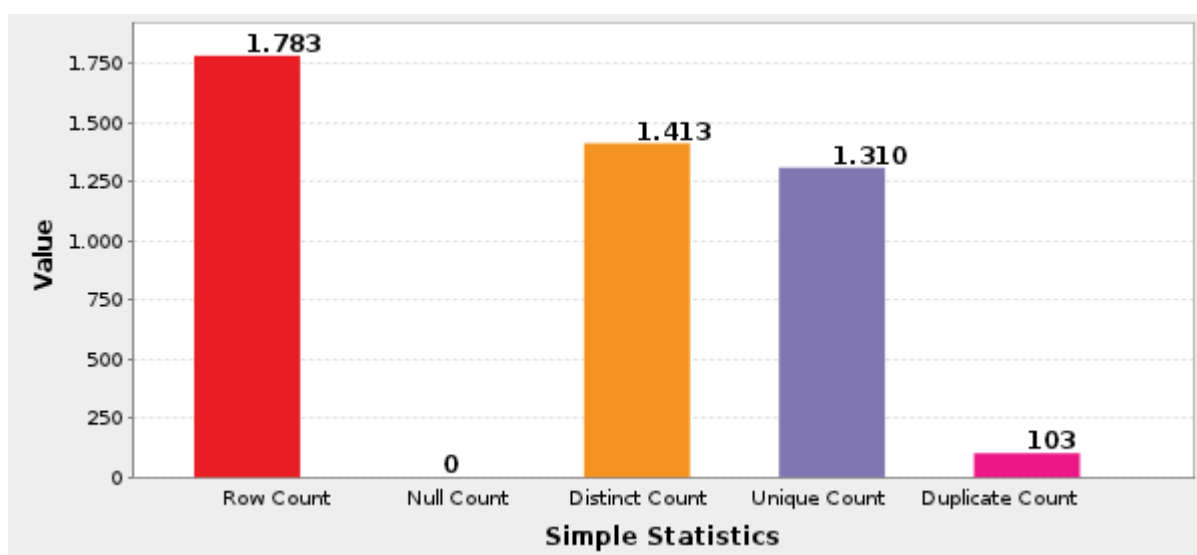



Fig 62. Campo id\_persona de la tabla del negocio tb\_dvisitante de la BD de prueba.

Anexo 8: Acta de aceptación del cliente.



## Acta de aceptación

**ACTA DE ACEPTACIÓN**





En cumplimiento del compromiso adquirido entre la Dirección de Cooperación Internacional (DCI) y el Centro de Informatización Universitaria y en función de la ejecución del Proyecto: Sistema de Gestión de Cooperación Internacional (SGCI), se certifica que fueron migradas correctamente las bases de datos que se listan a continuación:

- Historial de los datos de todos los pasaportes confeccionados en la DCI.
- Historial de los datos de todas las salidas realizadas por los misioneros.
- Historial de los datos de todas las visas otorgadas a los pasaportes.
- Historial de los datos de todas las habilitaciones otorgadas a los pasaportes.
- Historial de los datos personales de todos los misioneros.

**Beneficios:**

La migración de los datos históricos manejados por la Dirección de Cooperación Internacional hacia el Sistema de Gestión de Cooperación Internacional en la Universidad de las Ciencias Informáticas permitirá:

- Tener la información organizada y centralizada en un mismo sistema.
- Mantener la seguridad e integridad de los datos migrados.
- Garantizar que el nuevo sistema cuente con el historial de los datos de los procesos de trámites que se realizan en la universidad.

Entrega	Recibe
<p>Nombre y Apellidos: <u>Jesús González La Nuez</u></p> <p>Cargo: <u>Arquitecto</u></p> <p>Firma: </p> <p>Comentarios: _____</p>	<p>Nombre y Apellidos: <u>Beatriz Aragón Fernández</u></p> <p>Cargo: <u>Directora</u></p> <p>Firma: </p>
<p><b>Representante Parte Suministradora</b></p> <p>Nombre y Apellidos: <u>Probalis Terrero Caland</u></p> <p>Cargo: <u>Desarrollador</u></p> <p>Firma: </p>	
<p><b>Observador independiente</b></p> <p>Nombre y Apellidos: <u>Reinier Morejón Novales</u></p> <p>Cargo: <u>Profesor</u></p> <p>Firma: </p>	

Fecha: 11/06/2012

Fig 63. Acta de aceptación del cliente.