

MARCOS RODRÍGUEZ

MAURICIO ROSSO

INSTITUTO UNIVERSITARIO AERONÁUTICO



PROYECTO DE GRADO

ProMiDa

Proceso de Migración de Datos

Córdoba, Mayo del 2013

DEDICATORIA

A nuestros familiares y amigos, por el apoyo que nos brindaron y la ayuda que nos ofrecieron.

AGRADECIMIENTOS

El trabajo desarrollado en esta tesis fue concretado gracias a nuestro esfuerzo, y sobre todo por el apoyo incondicional que muchas personas nos entregaron durante su desarrollo.

Queremos agradecer a nuestros padres, quienes nos apoyaron durante todo el transcurso de nuestros estudios y permitieron que fuese posible estudiar una carrera profesional.

Agradecemos a Romina y Silvina, nuestras compañeras de vida, por entregarnos la palabra adecuada en el momento oportuno y por enseñarnos a disfrutar cada momento de la vida sin importar las dificultades.

Agradecemos a Mariano García Mattio, Glenda Del Giudice, Mariela Della Mea, y Julio Gutierrez, por su asesoramiento, cooperación y predisposición a ayudarnos cada vez que los hemos necesitado.

DECLARACIÓN DE DERECHOS DE AUTOR

Este trabajo es desarrollado por los autores permitiendo a la Universidad y estudiantes la reproducción total o parcial de su contenido, posibilitando la utilización del mismo como base de consulta para fines académicos.

RESUMEN

Ante la necesidad que tiene el Instituto Universitario Aeronáutico de implementar un nuevo sistema de gestión académica que permita mejorar el proceso actual y a su vez responder a los estándares de generación de información solicitados por el Ministerio de Educación, surge el problema de resolver la forma de traspasar los datos que están en el sistema actual al nuevo sistema (SIU – Guaraní3).

Este trabajo presenta una solución al problema de la Migración de Datos del Sistema de Gestión Académica del I.U.A (Instituto Universitario Aeronáutico), que debe evolucionar para adaptarse a nuevos requisitos.

De acuerdo a la literatura y la experiencia de profesionales, el diseño e implementación de ETL (Extract, Transform and Load) constituye el 80% de los esfuerzos de un proyecto de Migración de Datos; dado que es extremadamente complejo, tiende al fracaso, y además consume mucho tiempo con un costo significativo en recursos humanos y financieros.

En el presente Trabajo Final de Grado, se describen y ponen en práctica los elementos esenciales para el desarrollo de procesos ETL insertos en el contexto de una Migración de Datos, con el propósito de apoyar a los profesionales a disminuir la probabilidad de errores, tiempos y costos, que resultan ser muy comunes en este tipo de proyectos. Asimismo, se utiliza una herramienta Open Source que permite implementar los procesos respectivos, seleccionada a partir de una investigación y comparación entre herramientas ETL aplicables.

Con los resultados obtenidos se espera contribuir en la forma de trasvasar la información de la base de datos inicial a la final, ofreciendo generar de manera automática los planes de migración de los datos y toda la documentación necesaria para documentar el cambio, agilizando en gran medida el tiempo necesario para adaptarse a los cambios.

INDICE

INTRODUCCIÓN.....	8
1.1 ANTECEDENTES	8
1.2. SITUACIÓN PROBLEMÁTICA	9
1.2.1. Descripción de los Principales Problemas.....	9
1.3. PROBLEMA.....	9
1.4. OBJETO DE ESTUDIO Y CAMPO DE ACCIÓN	10
1.5. OBJETIVOS DEL PROYECTO.....	10
1.5.1. Objetivo general	10
1.5.2. Objetivos específicos:	10
1.6. IDEA A DEFENDER.....	11
1.7. DELIMITACION DEL PROYECTO	11
1.8. APORTE.....	11
1.9. MÉTODOS DE INVESTIGACIÓN	11
1.9.1. Métodos lógicos	12
1.9.2. Métodos empíricos.....	12
MARCO CONTEXTUAL	14
2.1. ENTORNO DEL OBJETO DE ESTUDIO	14
2.2. ANÁLISIS DE LOS PROBLEMAS OBSERVADOS	14
MARCO TEÓRICO.....	16
3.1. MARCO TEÓRICO DEL OBJETO DE ESTUDIO.....	16
3.1.1 Qué es la Migración de Datos?.....	16
3.1.2 Qué es la Migración Automática de Datos?	16
3.1.3 Qué es un mecanismo de migración automática de datos?.....	16
3.1.4 Aspectos de la Migración de Datos	17
3.1.5 Razones para Migrar Datos.....	17
3.2.1. Características de la Migración de Datos.....	18
3.2.2. Consideraciones para migrar datos.....	18
3.3.3. Arquitectura de la Migración de Datos	19
3.5.1 Recomendaciones para el Proceso de Migración	20
3.6.1. Conectividad con las distintas plataformas.....	23
3.7.1. Proceso ETL.....	24
4.1. MARCO TEÓRICO DEL CAMPO DE ACCIÓN	25
4.1.1. Migración de Datos	25
5.1. ETL	27
5.1.1 Proceso ETL.....	28
5.1.2 Extracción	28
5.1.3 Transformación	31
5.1.4 Carga	35
6.1. INVESTIGACION DE HERRAMIENTAS ETL PARA LA MIGRACION DE DATOS	36

6.1.1	Arquitectura del sistema. Herramienta ETL vs Desarrollos a Medida	37
6.1.2	Herramientas ETL Open Source	40
MODELO TEÓRICO		46
7.1.	PLANIFICACIÓN	46
7.2.	REQUERIMIENTOS	47
7.2.1	Requerimientos Funcionales.....	47
7.2.2	Requerimientos de Datos	47
7.2.3	Requerimientos de software.....	68
7.3.	ANÁLISIS Y DISEÑO	73
7.3.1	Análisis de la Migración de Datos.....	73
7.3.2	Diseño de la Migración de Datos.....	103
ARQUITECTURA DE LA MIGRACIÓN DE DATOS		103
CONCRECIÓN DEL MODELO		112
8.1	IMPLEMENTACION	112
8.1.1	Secuencia de Migración.....	112
8.1.2	Orden de Inserción en tablas de SIU-Guarani.....	113
8.1.3	Representación de Transformaciones y Trabajos en Kettle.....	115
PRUEBAS Y PUESTA EN MARCHA.....		193
9.1	PRUEBAS	193
9.1.1	Caso de Test 1 – Control de Cambios en tabla: “mdp_personas”	193
9.1.2	Caso de Test 2 – Control de Cantidad de Alumnos y Docentes Migrados	195
9.1.3	Caso de Test 3 – Control de Cambios en tabla: “sga_prupuestas”	197
9.1.4	Caso de Test 4 – Control de Cantidad de Carreras Migradas	199
9.1.5	Caso de Test 5 – Control de Combinación de Registros de Diferentes Tablas	201
9.1.6	Caso de Test 6 – Control de Ejecución Trabajo de Extracción de Datos	202
9.1.7	Caso de Test 7 – Control de Ejecución Trabajo Depuración y Limpieza de Datos.....	203
9.1.8	Caso de Test 8 – Control de Ejecución Trabajo Carga y Control de Cambios.....	203
9.1.9	Caso de Test 9 – Control de Ejecución Trabajo Global.....	204
9.2	PUESTA EN MARCHA.....	206
9.2.1	Infraestructura Computacional	206
9.2.2	Configuración de la Base de Datos.....	206
9.2.3	Configuración de Pentaho	206
CONCLUSIONES.....		212
10. REFERENCIAS		214

INTRODUCCIÓN

1.1 ANTECEDENTES

El IUA, organismo dependiente de la Fuerza Aérea Argentina, fue creado el 13 de diciembre de 1947, integrándose al Sistema Universitario Nacional en el año 1971. La finalidad del IUA es impartir enseñanza universitaria, realizar investigación científica y tecnológica y acciones de extensión cultural y de transferencia tecnológica.

Actualmente el IUA cuenta con un Sistema de Información que fue desarrollado por personal interno de dicha institución con el fin de mejorar y agilizar la gestión académica.

Debido al sostenido crecimiento que atraviesa el instituto en términos de infraestructura, personal académico y alumnado, y ante la inminente necesidad de responder a los estándares que exige el Ministerio de Educación, el IUA toma la decisión de integrar el consorcio SIU. El SIU (Sistema de Información Universitario), es, en realidad, un Consorcio de Universidades que desarrolla soluciones informáticas y brinda servicios para el Sistema Universitario Nacional y distintos organismos de gobierno. Su objetivo es contribuir a mejorar la gestión de las instituciones, permitiéndoles contar con información segura, íntegra y disponible. Dicho consorcio se encuentra conformado por 37 universidades entre las que se encuentra la Universidad Nacional de Córdoba.

El SIU cuenta con varios proyectos activos en la Secretaría de Políticas Universitarias (SPU), en Universidades Nacionales (públicas y privadas) y otros organismos públicos. Se han desarrollado sistemas para la gestión y para la toma de decisiones, sistemas estadísticos y herramientas de desarrollo, entre otros proyectos. Todas las Universidades Nacionales han implementado los sistemas del SIU en alguna de sus áreas. Entre dichos sistemas podemos mencionar:

SIU-Pampa, SIU-Mapuche, SIU-Pilagá, SIU-Comechingones, SIU-Toba, SIU-Wichi, SIU-Data Warehouse, SIU-Araucano, SIU-Kolla, ComDoc, SIU-Bibliotecas y SIU-Guarani.

El SIU-Guaraní es un sistema de gestión académica. Realiza la gestión de alumnos, desde la matriculación hasta el egreso, complementándose con gestión de aulas, mesas de exámenes, jurados, etc. Posee una interfaz con el sistema de estadísticas de alumnos SIU-Araucano.

El IUA, siendo miembro del consorcio decide implementar el SIU-Guaraní como sistema de gestión académica.

Uno de los desafíos a los que se debe enfrentar el Instituto Universitario Aeronáutico al momento de implementar el nuevo sistema, es el de migrar la información que existe en el actual Sistema de Gestión Académica. Gran parte del éxito del proyecto dependerá de la calidad de los datos con la cual se inicia el nuevo Software.

Cuando se piensa en migrar datos de un sistema a otro, existen varios factores que se deben tener presente en el proceso. Por ejemplo: procesos de negocio, depuración de datos, fuentes de información, equipos de trabajo, herramientas a utilizar y planes de pruebas.

1.2. SITUACIÓN PROBLEMÁTICA

Al realizar un análisis de la situación actual, se ha encontrado que el proyecto tiene actualidad y pertinencia debido a diferentes factores, que se especifican a continuación y que definen la situación problemática.

1.2.1. Descripción de los Principales Problemas

- Desconocimiento de la estructura de la base de datos destino.
- Carencia de un procedimiento formalizado para la migración de datos.
- Limitación de tiempo para cumplir con el proyecto.

1.3. PROBLEMA

Ante la necesidad que tiene el Instituto Universitario Aeronáutico de implementar un nuevo sistema de gestión académica que permita mejorar el proceso actual y a su vez responder a los estándares de generación de información solicitados por el Ministerio de

Educación, surge el problema de resolver la forma de traspasar los datos que están en el sistema actual al nuevo sistema (SIU – Guaraní3).

1.4. OBJETO DE ESTUDIO Y CAMPO DE ACCIÓN

- Aquello por estudiar y por conocer es la estructura de la base de datos actual del S.G.A (Sistema de Gestión Académica) del IUA que reside sobre un motor SQL Server 2000 y la estructura de SIU-Guarani3 que trabaja sobre una plataforma PostgreSQL 8.3, analizando la posibilidad de migración y las herramientas a utilizar para llevar a cabo el proyecto.
- El campo de acción se atribuye directamente a la integración de datos para que los mismos puedan ser transferidos de un sistema origen a un sistema destino.

1.5. OBJETIVOS DEL PROYECTO

1.5.1. Objetivo general

Realizar un proceso de migración que permita garantizar la transferencia adecuada de los datos desde la Base de Datos del Sistema Integral de Gestión Académica del Instituto Universitario Aeronáutico a la Base de Datos del Sistema SIU-Guaraní3 (integra sistemas de Facultades e Institutos Universitarios de todo el país), para contribuir con la implementación del nuevo Sistema de Información e integrar al I.U.A con las diferentes universidades que componen el SIU-Guaraní en la actualidad, conforme a lo dispuesto con la normativa del Ministerio de Educación.

1.5.2. Objetivos específicos:

- Garantizar la consistencia, integridad y seguridad necesaria de los datos que serán transferidos al sistema destino.
- Seleccionar la/s herramienta/s más adecuada/s para llevar a cabo la migración y conversión de los datos desde la base de datos de origen a la de destino.
- Seleccionar la/s metodología/s más adecuada/s para llevar a cabo la migración y conversión de los datos desde la base de datos de origen a la de destino.

- Disponer de datos históricos del sistema de información actual en el nuevo sistema.
- Crear el proceso que será utilizado para migrar datos desde el S.G.A del I.U.A a SIU-Guaraní3; este proceso al contar con una base de datos intermedia (Staging) podrá ser reutilizado para futuros proyectos cuyo objetivo sea migrar sus datos a SIU-Guaraní3.

1.6. IDEA A DEFENDER

Se pretende demostrar que la migración de los datos que actualmente se encuentran disponibles en la BD de producción del S.G.A del I.U.A a la de base de datos de SIU-Guaraní3, podrá ser desarrollada utilizando tecnologías Open Source disponibles, brindando una solución de alta performance y gran rendimiento, disminuyendo, de este modo, los costos que podrían existir en licencias.

1.7. DELIMITACION DEL PROYECTO

El alcance de este trabajo es la migración de datos que incluye: relevamiento de información, depuración de datos, extracción, transformación / mapeo de datos, carga de datos, pruebas, simulación de cargas, cargas reales, y revisión de los resultados, dentro del marco de un contexto que involucra una Base de Datos origen (en SQL Server 2000) y una Base de Datos destino (en PostgreSQL 8.3).

1.8. APORTE

La relevancia social que tendrá el proyecto se traduce en el beneficio a la comunidad educativa del Instituto Universitario Aeronáutico otorgando la posibilidad de implementar un sistema que permita mejorar la gestión de la información.

Es importante mencionar que una vez implementado el proyecto, podrá presentarse a otras instituciones afines con las mismas necesidades, considerando que podría llevarse a cabo el proceso de manera similar, utilizando ciertos métodos que pudieran ser compatibles y herramientas usadas en el proyecto original, que servirán como experiencia y referencia para futuras migraciones.

1.9. MÉTODOS DE INVESTIGACIÓN

Se utilizarán métodos de investigación empíricos y lógicos para el desarrollo del trabajo final de grado.

Por métodos empíricos se refiere a modelos de investigación basados en la experiencia y pruebas sobre el objeto de estudio, permitiendo reconocer características y relaciones esenciales de los elementos considerados en el mismo, lo que nos dará las bases para estudios descriptivos.

Los métodos lógicos son aquellos que utilizan el pensamiento en sus funciones de deducción, análisis y síntesis.

1.9.1. Métodos lógicos

Los métodos lógicos serán utilizados para:

- Analizar la situación problemática y establecer claramente el problema que se pretende solucionar.
- Relacionar adecuadamente el marco teórico a desarrollar con el objeto de estudio y el campo de acción.
- Analizar y deducir el conjunto de fenómenos y hechos que dan surgimiento al problema.
- Realizar la implementación de la solución, considerando lo generado en el modelo teórico y utilizando las tecnologías más adecuadas.

1.9.2. Métodos empíricos

Se aplicarán métodos empíricos para:

- Estudiar el entorno del problema durante la etapa de marco contextual, donde describiremos el contexto de nuestro proyecto, basándonos en información obtenida mediante observación directa, reuniones con el personal, búsquedas en Internet, entre otros.
- Efectuar el desarrollo del marco teórico y su correspondiente diagnóstico, en base a la observación directa, la información obtenida mediante libros, blogs, descripciones documentadas y opiniones de referentes en temáticas relacionadas, así como también nos apoyaremos en la información recopilada del personal afectado.

Etapa 1

MARCO CONTEXTUAL

2.1. ENTORNO DEL OBJETO DE ESTUDIO

Como se mencionó anteriormente, el objeto de estudio se concentra en 2 (dos) estructuras de base de datos diferentes, y cuyo resultado determinará la posibilidad de migración y las herramientas más adecuadas para llevar a cabo la misma. En nuestro caso el componente fundamental que define el entorno del objeto de estudio es el proceso encargado de extraer los datos desde una fuente disímil, para luego adaptarlos a la estructura de datos requerida por el negocio, lo que requiere aplicar una serie de transformaciones y operaciones de limpieza sobre los datos extraídos, para finalmente ser cargados a la estructura de destino.

2.2. ANÁLISIS DE LOS PROBLEMAS OBSERVADOS

Ante la necesidad que tiene el Instituto Universitario Aeronáutico de implementar un nuevo sistema de gestión académico que permita mejorar el proceso actual y a su vez responder a los estándares de generación de información solicitadas por el Ministerio de Educación, surge el problema de resolver la forma de traspasar los datos que están en el sistema actual al nuevo sistema.

Uno de los principales problemas que observamos al iniciar el proyecto de migración, es identificar las características del sistema fuente que posee la institución, conocer su estructura, organización, arquitectura en la que está implementada y sobre todo identificar los datos que servirán para la migración. Luego, una vez que estos han sido identificados y extraídos desde el sistema fuente se presenta la segunda problemática: nos encontramos con datos inconsistentes, sin formato, nulos, entre otros casos que se pueden dar. Esto implica una ardua tarea de limpieza de datos y un conjunto de transformaciones que deben ser aplicadas sobre los datos para dejarlos en un formato consistente que sirva para una correcta y optima transferencia al sistema destino.

Es por ello que se considera necesario evaluar diferentes alternativas tecnológicas que ayuden a garantizar la consistencia, integridad y seguridad de los datos a migrar.

Etapa 2

MARCO TEÓRICO

3.1. MARCO TEÓRICO DEL OBJETO DE ESTUDIO

3.1.1 Qué es la Migración de Datos?

Es la transferencia segura y organizada de datos entre dos sistemas de bases de datos diferentes considerando la compatibilidad de tipos de datos, hardware, requerimientos, etc.

3.1.2 Qué es la Migración Automática de Datos?

Es la transferencia de datos entre dos sistemas de bases de datos diferentes en la cual se establece un proceso o una tarea que permita ejecutar la migración de datos cada cierto tiempo determinado.

3.1.3 Qué es un mecanismo de migración automática de datos?

Un mecanismo de migración automática de datos se define como la forma o manera de migrar datos o información automáticamente desde una base de datos (origen) hacia otras plataformas heterogéneas (destinos) de acuerdo a las herramientas de migración y a las características que presente la base de datos origen.

Ejemplo:

Si se tiene dos sistemas heterogéneos (PostgreSQL y Sql Server) los mecanismos de migración serán:

- Migración de Datos de PostgreSQL a Sql Server

La forma o la manera de migrar datos desde la plataforma origen (PostgreSQL) hacia una plataforma heterogénea (Sql Server).

- Migración de Datos de Sql Server a PostgreSQL

La forma o la manera de migrar datos desde la plataforma origen (Sql Server) hacia una plataforma heterogénea (PostgreSQL).

3.1.4 Aspectos de la Migración de Datos

Es bastante común tener que migrar datos entre diferentes servidores heterogéneos ya sea por traspasos de entornos de desarrollo a producción o por tener información para detectar errores en las aplicaciones, hacer pruebas de carga, etc.

3.1.5 Razones para Migrar Datos

Son diversas las razones por las que las organizaciones necesitan migrar datos entre distintas plataformas:

3.1.5.1 Reducir costos de almacenamiento

Muchas de las organizaciones mantienen información en gestores de bases de datos que consumen una cantidad de recursos y son muy complicados a la hora de su mantenimiento. Con datos en constante movimiento, las empresas necesitan encontrar un modo mejor de manejar los datos, es decir, un modo que minimice el trabajo, los costos y las interrupciones.

3.1.5.2 Necesidad de estandarización

Las organizaciones mantienen varios sistemas operacionales que funcionan independientemente, entonces es importante unir la información y lograr estandarización de los mismos que mejoren el tratamiento de la información dentro de la organización.

3.1.5.3 Actualización de tecnología

Si una organización tiene una base de datos como por ejemplo Access y posteriormente crece de manera que hace falta un sistema gestor de bases de datos potente, lo más seguro es que se elija por SGBD como Sql Server, MySQL, Oracle , PostgreSQL, Db2, o similares.

3.1.5.4 Requerimientos de la organización

La migración de los datos es fundamental para satisfacer los nuevos requisitos de forma correcta, ya que son una parte importante del sistema de información. A partir de la información semántica recogida en los esquemas conceptuales y de las correspondencias

entre ambos es posible generar de manera automática la migración de datos que permitirá trasladar la información entre distintas bases de datos.

3.2.1. Características de la Migración de Datos

3.2.1.1. Migrar es un Proceso

La migración no puede plantearse como un producto llave en mano, pero debe establecer un final claro con resultados medibles.

3.2.1.2. Es un Proyecto Multidisciplinar

Deben establecerse todos los procedimientos y metodologías propios de los proyectos técnicos informáticos y de calidad, donde aspectos como la formación, la psicología o la gestión están contemplados.

3.2.1.3. Proceso Rentable

El criterio de rentabilidad debe ser prioritario en todos los casos (empresas, administraciones y particulares).

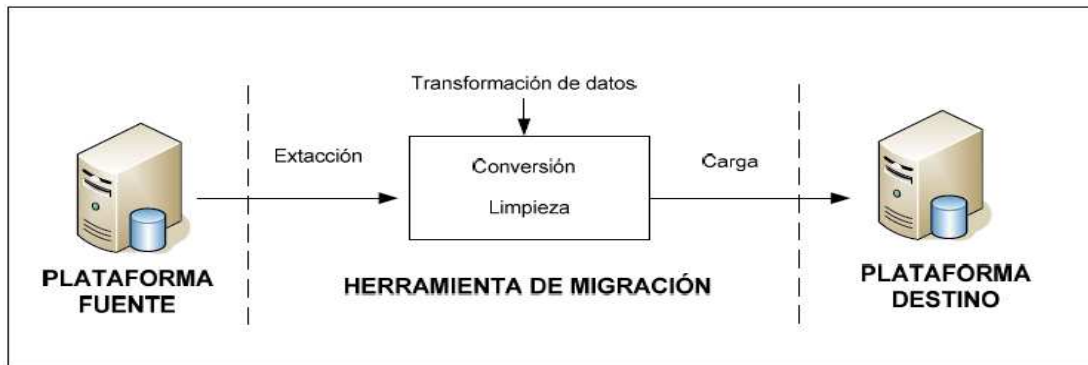
3.2.2. Consideraciones para migrar datos

- Toda la migración tiene que tener en cuenta muy especialmente, como ya se señaló, las maneras que tenga cada base de datos de guardar la información, es decir, del formato de sus tipos de datos.
- Tenemos que contar siempre con la posible necesidad de transformar algunos datos como pueden ser los campos booleanos, fechas, campos memo (texto con longitud indeterminada), etc., que pueden almacenarse de maneras distintas en cada uno de los sistemas gestores, origen y destino.
- En algunos casos posiblemente tengamos que realizar algún script que realice los cambios necesarios en los datos.
- El proceso de migración es una tarea muy complicada de llevarla a cabo ya que se deben tomar todas las medidas necesarias para obtener los resultados exitosos por parte de las organizaciones interesadas en realizar migraciones de datos.

- Cuando se piensa en migrar datos de un sistema a otro, no es sólo realizar programas que permitan efectuar la migración, existen otros factores que se deben tener presente en el proceso de migración de datos. Por ejemplo: Procesos de negocio, limpieza de datos, fuentes de información, equipos de trabajo, herramientas a utilizar, planes de pruebas, etc.
- Si no se realiza una planificación y un análisis respectivo de lo que se quiere migrar, a donde se quiere migrar y que se espera de la migración de datos esta puede no tener éxito dentro de una organización.
- No obstante, la posibilidad de una migración de datos puede resultar abrumadora, debido a incompatibilidades entre los sistemas de almacenamiento y las tecnologías de servidor, además de la exigencia de que los datos y aplicaciones estén disponibles las 24 horas, todos los días.

3.3.3. Arquitectura de la Migración de Datos

La arquitectura básica de la migración de datos se define con tres elementos principales como se muestra en la siguiente figura:



3.3.3.1. Plataforma Fuente

Es el sitio donde se almacena la información principal que se será analizada para su respectiva migración hacia otra base de datos.

3.3.3.2. Plataforma Destino

Es el sitio al cual se irán los datos de la base de datos fuente después de su transformación y limpieza respectiva.

3.3.3.3. Herramienta de Migración

Esta herramienta permitirá llevar a cabo la migración de datos entre plataformas, permitirá la extracción, la transformación y la carga de los datos desde la base de datos fuente hacia la base de datos destino de acuerdo a los requerimientos establecidos en una organización.

3.4.1. Proceso de la Migración de Datos

El proceso de la migración puede ser bastante complejo y, como hay tantas bases de datos distintas, difícil de dar una receta que funcione en todos los casos. Además, aparte de la dificultad de transferir la información entre los dos sistemas gestores de base de datos, también nos influirá mucho en la complejidad del problema el tipo de los datos de las tablas que estamos utilizando. Por ejemplo, las fechas, los campos numéricos con decimales o los boléanos pueden dar problemas al pasar de un sistema a otro porque pueden almacenarse de maneras distintas o, en el caso de los números, con una precisión distinta.

Para realizar la migración de datos, es muy importante establecer en forma temprana, los datos del sistema origen que se requieren en el sistema destino (heterogéneo). También es necesario, conocer a los dueños de la información y los criterios de depuración que tendrán los datos seleccionados, para asegurar una óptima calidad de ellos.

3.5.1 Recomendaciones para el Proceso de Migración

La migración de datos es una de las etapas más importantes en la implementación de un nuevo sistema y también puede resultar una de las más críticas. Por este motivo, se trata de un proceso que requiere de una planificación y de un equipo designado con roles, tiempos y tareas definidas.

Para lograr que esta etapa transcurra de la manera más eficaz posible es vital no perder datos en el camino. Resguardar la integridad de la información facilita que los usuarios confíen en el nuevo sistema y lo incorporen más fácilmente.

A continuación, presentamos una síntesis de las fases del proceso junto con los aspectos generales a tener en cuenta al momento de realizar la migración de datos a un nuevo sistema.

3.5.1.1 Capacitación en el modelo de datos del sistema

Resulta necesario conocer el diseño del modelo de datos del sistema. De esta manera, se podrá conocer cómo se almacenan los datos en los distintos circuitos que administra el sistema.

3.5.1.2 Analizar datos de origen

El análisis debe incluir cuáles son los datos a migrar al nuevo sistema, e implica revisar su validez y consistencia, como así también el nuevo formato y los campos obligatorios, entre los aspectos más destacados.

En las organizaciones donde hay algún sistema en funcionamiento se deberá realizar un análisis de los datos existentes. La realidad indica que suele haber inconsistencias en los datos disponibles para migrar (por ejemplo: actas y legajos duplicados, materias con más de una nota por alumno). Esto puede deberse a las sucesivas migraciones que sufrieron los datos en el tiempo, sumado a las distintas tecnologías utilizadas para su almacenamiento (por ejemplo: uso de archivos DBF que no permiten validaciones de integridad entre tablas, etc.).

Es importante mencionar que durante esta etapa es muy probable que sea necesario realizar una depuración de los datos existentes en los sistemas en funcionamiento. La demora de este trabajo dependerá de la cantidad de problemas detectados y de su complejidad.

3.5.1.3 Planificar migración de datos

Es muy probable que no todos los datos que el nuevo sistema necesita para funcionar existan en el sistema actual. Por lo tanto, una parte importante de la migración consiste en definir cómo se va a generar esta información teniendo en cuenta la realidad de la institución donde se está implementando.

En la planificación se tiene que definir el conjunto de tareas a realizar. Este listado debe contar con un responsable, tiempos de entrega y un entregable (producto final). Estas tareas pueden ser tanto automáticas como manuales.

Dentro de las tareas automáticas se encuentra el desarrollo de algoritmos de programación para la adaptación, transformación y compilación de datos que serán aportados desde los otros sistemas al nuevo.

Las tareas manuales consisten en el armado de archivos de migración realizados a partir de documentos u otra información existente que no se encuentre automatizada y se quiera incluir en la migración.

3.5.1.4 Migrar datos

El siguiente paso es la ejecución de los procesos en la base de datos. La migración debe llevarse a cabo por módulos. Cada módulo afecta a un conjunto de tablas que no pueden ser migradas por separado. También es preciso tener en cuenta el orden en que deben ser migrados estos módulos. Para cada módulo se definen requisitos previos que deben cumplirse al intentar la migración (módulos previos, parámetros del sistema, datos ingresados, etc.).

3.5.1.5 Controlar integridad y calidad de los datos migrados

Luego de ejecutar los procesos de migración es preciso controlar los resultados de salida que estos devolvieron. Este control incluye la verificación de que los datos se hayan procesado en su totalidad y sin errores u omisiones. Deben controlarse los datos con los existentes en el sistema anterior o con los registros escritos (por ejemplo: legajos de personal, historias académicas, entre otros).

3.5.1.6 Aceptar migración

Una vez que fueron realizadas las verificaciones al proceso, el mismo tiene que ser aceptado para empezar a trabajar con el nuevo sistema. Esta aceptación del trabajo debe ser decidida por personas ajenas al proceso de migración que se realizó.

Para concluir, se debe garantizar como medida de seguridad, que una vez finalizada la migración se guarde copia de la base de datos en un lugar seguro y sellado para prevenir problemas futuros. Esto permite distinguir entre los datos que se incorporaron al sistema mediante importación y los que fueron generados a partir del uso del mismo.

3.6.1. Conectividad con las distintas plataformas

Cuando nos encontramos en un determinado sistema gestor de base de datos lo que es importante es establecer las conexiones con las distintas bases de datos para lo cual se presentan muchas opciones como:

3.6.1.1. ODBC

Son las siglas de Open DataBase Connectivity, un estándar de acceso a bases de datos desarrollado por Microsoft Corporation, el objetivo de ODBC es hacer posible el acceder a cualquier dato de cualquier aplicación, sin importar qué Sistema Gestor de Bases de Datos (DBMS por sus siglas en Ingles) almacene los datos, ODBC logra esto al insertar una capa intermedia llamada manejador de Bases de Datos, entre la aplicación y el DBMS, el propósito de esta capa es traducir las consultas de datos de la aplicación en comandos que el DBMS entienda. Para que esto funcione tanto la aplicación como el DBMS deben ser compatibles con ODBC, esto es que la aplicación debe ser capaz de producir comandos ODBC y el DBMS debe ser capaz de responder a ellos.

3.6.1.2. OLE DB

OLE DB (algunas veces escrito como OLEDB u OLE-DB) es la sigla de Object Linking and Embedding for Databases ("Incrustación y enlace de objetos para bases de datos") y es una tecnología desarrollada por Microsoft usada para tener acceso a diferentes fuentes de información, o bases de datos, de manera uniforme.

OLE DB permite separar los datos de la aplicación que los requiere. Esto se hizo así ya que diferentes aplicaciones requieren acceso a diferentes tipos y almacenes de datos, y no necesariamente desean conocer cómo tener acceso a cierta funcionalidad con métodos de tecnologías específicas.

3.6.1.3. JDBC

Es el acrónimo de Java Database Connectivity, un API (Interfaz de Programación de Aplicaciones) que permite la ejecución de operaciones sobre bases de datos desde el lenguaje de programación Java (Java es una plataforma de software desarrollada por Sun Microsystems, de tal manera que los programas creados en ella puedan ejecutarse sin cambios en diferentes tipos de arquitecturas y dispositivos computacionales) independientemente del sistema de operación 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.

3.7.1. Proceso ETL

ETL (del inglés, extraer, transformar y cargar) es un proceso de consolidación de datos que implica la extracción de datos a partir de diversas fuentes, su transformación con el fin de ajustarse a las necesidades del negocio y su carga en un único repositorio, como puede ser un almacén de datos.

4.1. MARCO TEÓRICO DEL CAMPO DE ACCIÓN

4.1.1. Migración de Datos

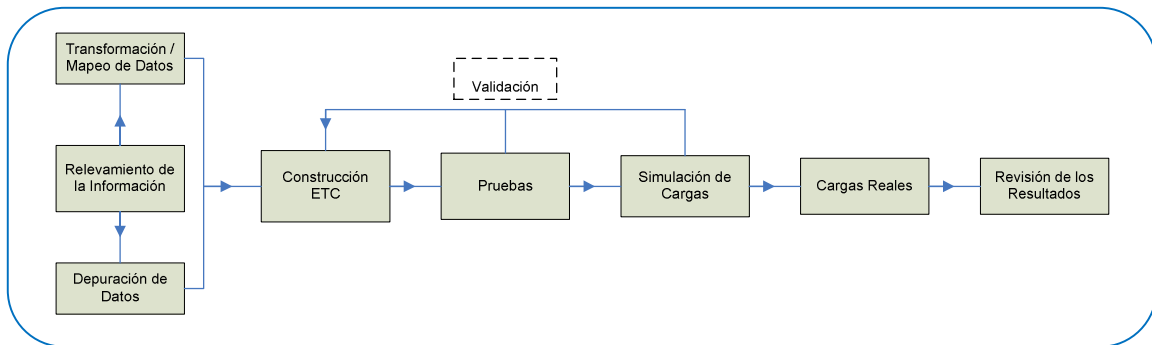
La migración de los datos es fundamental para satisfacer los nuevos requisitos de forma correcta, ya que son una parte importante del sistema de información. A partir de la información semántica recogida en los esquemas conceptuales y de las correspondencias entre ambos es posible generar de manera automática los planes y mecanismos de migración que permitirán trasladar la información entre las bases de datos, una vez que hayan sido validados por un analista. Posteriormente, se ejecutarán dichos planes en las bases de datos para adaptar la información a los nuevos requisitos.

Otro de los aspectos para abordar la migración de su tecnología de base de datos puede ser desde razones políticas y estratégicas que inclinan la balanza hacia uno u otro proveedor a razones tecnológicas generalmente asociadas a lograr mayor integración o mejores prestaciones en los sistemas operacionales dentro de una organización.

Curiosamente, lo más simple en una migración de base de datos es, precisamente, migrar los datos. Salvo situaciones puntuales de diferencias tecnológicas según las cuales una base no soporte algún tipo de dato de la otra base, la migración de los datos puede llegar a ser lo más simple del proceso. Basta con la creación del nuevo esquema, el mapeo de tipos de datos y el soporte de scripts que automaticen el traspaso.

Para realizar la migración de datos, es muy importante establecer en forma temprana, los datos que se requieren en el nuevo sistema y la ubicación de los datos en el antiguo sistema. También es necesario, conocer el(los) dueño(s) de la información y los criterios de depuración que tendrán los datos seleccionados, para asegurar una óptima calidad de ellos.

A continuación, se presentan las etapas que deben llevarse a cabo para la migración de los datos:



Etapas de Migración de Datos

Relevamiento de Información: Esta etapa contempla la definición de los datos que requiere el nuevo sistema y la identificación de la fuente de origen.

Transformación/Mapeo de datos: Esta etapa, contempla la definición de los diccionarios con las reglas de transformación y el mapeo de campos entre las fuentes de origen y destino.

Depuración de Datos: Esta etapa contempla la depuración de los datos en las fuentes de origen.

Construcción ETC: En esta etapa, se realiza la construcción de los extractores, transformadores y cargadores de datos.

- *Extracción de datos*

La extracción es el proceso de análisis y recopilación de los datos desde la base de datos fuente que se quieren migrar hacia la base de datos destino.

- *Transformación de datos*

Los tipos de datos, esquemas y modelos conceptuales de la base de datos fuente son transformados, convertidos y limpiados de tal forma que serán transferidos hacia la base de datos destino de acuerdo a los requerimientos de una organización.

- *Carga o integración de datos*

Este proceso permite cargar e integrar los datos debidamente transformados y limpiados de la base de datos origen en la base de datos destino de acuerdo a los requerimientos establecidos en la organización.

Pruebas: En esta etapa, se realizan las pruebas funcionales de la migración de datos, determinando que la información sea traspasada con éxito al nuevo sistema.

Simulación de Cargas: En esta etapa, se pretende identificar los inconvenientes que se podrían tener en las cargas reales, a través de la simulación del proceso real.

Cargas Reales: En esta etapa, se realiza la migración real de los datos desde el antiguo sistema al nuevo sistema.

Revisión de Resultados: En esta etapa, se realiza la revisión final de la información cargada al nuevo sistema.

5.1. ETL

Gran parte de los esfuerzos de la migración están centrados en las tareas de ETL, dado que estas se encargan de manipular los datos que servirán para la transferencia de los mismos al sistema destino, objetivo principal de nuestro proyecto.

Las herramientas ETL poseen una gran cantidad de funciones que facilitan los esfuerzos empleados en las tareas de ETL, proporcionando por ejemplo, una amplia conectividad con diversos sistemas fuentes para facilitar el proceso de extracción, además poseen un entorno gráfico intuitivo para desarrollar las rutinas ETL con una amplia gama de transformaciones predefinidas y funciones de limpieza de datos. Por otro lado, la mayoría de las herramientas ETL ofrecen variadas posibilidades para mejorar el rendimiento de los procesos, incorporando entre estas, el procesamiento paralelo, la segmentación de datos, distribución de rutinas, entre otras.

En la actualidad existen productos ETL que van más allá de la arquitectura básica, ya que incorporan nuevos componentes tales como limpieza de datos, perfiles de datos, captura avanzada de datos cambiantes, y una serie de nuevas fuentes y destinos de datos. Estos nuevos

componentes conforman una arquitectura de datos mucho más poderosa que la mencionada, y convierten a las herramientas ETL en productos de nueva generación denominados “plataformas de integración de datos”.

A continuación se explicará en detalle como involucraremos el proceso ETL en nuestro proyecto y como se aportará valor al mismo con las funcionalidades de las plataformas de integración de datos.

5.1.1 Proceso ETL

El proceso ETL será el encargado de realizar dos tareas bien definidas: la carga inicial y las actualizaciones y mantenimiento periódico en el Sistema Destino (SIU-Guaraní3). La carga inicial, se refiere precisamente a la primera carga de datos que se le realizará al Sistema Destino. Por lo general, esta tarea consume un tiempo bastante considerable, ya que se deben insertar registros que han sido generados, durante varios años. Además, esta etapa actúa como una caja negra, ya que se definen los diseños y desarrollos ETL durante el análisis, pero una vez que la carga inicial se ha realizado, recién se pueden ver los “datos obtenidos”. Por tanto, en este punto es donde se efectúa la retroalimentación entre los usuarios y los responsables del proyecto que luego de varias iteraciones, cumplirán con el objetivo de tener un “Migrador” consistente que cumpla con las necesidades del negocio.

La carga inicial generalmente se realiza solo una vez y al inicio del proyecto. Posteriormente solo se realizan las actualizaciones y mantenimientos periódicos, los cuales mueven pequeños volúmenes de datos, y su frecuencia está dada en función a los requerimientos del usuario y al nivel de granularidad que se tendrá. El objetivo de esta tarea será añadir a PostgreSQL solo aquellos datos que han sido actualizados o modificados desde la última carga.

A continuación abordaremos todos los aspectos que se deberemos considerar en cada etapa del proceso ETL, referentes a la carga inicial y a las actualizaciones y mantenimientos periódicos.

5.1.2 Extracción

Una vez que hayamos establecido los requerimientos del negocio y se haya diseñado un modelo conceptual que describe los procesos ETL a realizar, se procederá a extraer los

datos necesarios desde el sistema de origen (Sistema de Gestión Académica) que posee el I.U.A.

Antes de realizar la extracción de datos, será muy importante analizar el sistema de origen que se utilizará para extraer la información que requiere la Institución, identificando la estructura de datos, organización y disponibilidad.

Un requerimiento importante que exigiremos a la tarea de extracción, es que esta cause un impacto mínimo en los sistemas de origen.

Lo mencionado anteriormente está asociado a que los datos se encuentran en un sistema que soporta el día a día de la Institución, por lo que deberemos establecer la frecuencia de acceso al sistema fuente con el objetivo de no entorpecer la operación diaria del negocio o más bien, reducir el rendimiento del sistema operacional.

La frecuencia con la que se extraerán los datos desde el sistema fuente, es otro factor importante que también debe ser considerado en esta etapa.

Para nuestro proyecto en particular, será muy importante tener una latencia de datos baja, por lo que se realizarán extracciones incrementales. Esto quiere decir que solo se extraerán aquellos datos que han cambiado en el sistema de origen, y para ello emplearemos algún tipo de técnica CDC (Change Data Capture).

Es de resaltar que las extracciones incrementales se realizaran constantemente durante la etapa de actualización y mantenimiento, por lo que no se debe olvidar que primero existirá una extracción inicial, que se encargará de extraer todos los datos desde el sistema de origen de una sola vez.

La mayoría de las veces no es efectivo procesar inmediatamente los datos extraídos, ya que generalmente se trata de grandes cantidades de datos, y al ser procesados inmediatamente podría interrumpir o incluso paralizar el sistema operacional. Por dicho motivo en el proyecto en cuestión utilizaremos un área stage. De esta forma, los datos serán extraídos de manera rápida y eficaz sin interrumpir la operación diaria del negocio.

El área stage, corresponde a una base de datos en donde la información puede ser almacenada en tablas auxiliares o tablas temporales. De esta forma, los datos extraídos se almacenarán en

tablas que posteriormente serán procesadas acorde a los requerimientos del negocio para finalmente ser cargadas en la base de datos destino. Por tanto, el área stage constituirá un buffer intermedio entre el sistema operacional (SGA origen) y SIU-Guaraní3 (sistema destino). Cabe aclarar que la misma será creada en PostgreSQL.

A continuación se describen 4 (cuatro) puntos para lo cual nos será de gran utilidad disponer de un área de staging:

- **Mínimo impacto sobre el origen:** para conseguir una máxima escalabilidad y para afectar lo mínimo posible al rendimiento de los servidores transaccionales que van a ser nuestros orígenes de datos, debemos consumir de ellos los mínimos recursos posibles, y centrar el consumo de dichos recursos en el servidor que se encarga de realizar los procesos ETL.
- **Trazabilidad del dato:** es muy importante saber dónde, cuándo y quién ha realizado cada cambio. Para ello deberemos, además de realizar los propios procesos de extracción y de obtención de diferencias entre el origen y el destino, de ir almacenando todos los cambios que nos encontremos, para, en un momento dado, poder seguir la pista de cualquier dato y de los cambios que ha sufrido.
- **Generación de tablas Delta:** siempre es una alternativa mucho más óptima tener una tabla delta, que tenga registradas las operaciones de inserción, actualización y borrado que se han ido produciendo en el origen, y aplicar esos cambios en el destino. Este proceso es mucho más eficiente, y por supuesto, consume muchos menos recursos en origen y destino, que hacer una lectura de dicho origen, compararlo con el destino, y obtener de ahí las diferencias para seguidamente aplicarlas al destino.
- **Limpieza de datos:** otro punto que no debemos olvidar en ningún momento, es que en este tipo de procesos, el objetivo no es solamente traer los datos al destino y hacer ciertas transformaciones. Es muy importante llevar a cabo una serie de tareas de limpieza de datos y detección de incoherencias.

5.1.2.1 Change Data Capture

Para las Bases de Datos, Change Data Capture (CDC) es un conjunto de patrones de diseño que se utilizan para determinar los datos que han cambiado, de modo que se puedan tomar medidas sobre ellos.

En el entorno de la Migración de Datos, la técnica de CDC es muy utilizada para realizar las extracciones incrementales, ya que permiten capturar los cambios que se producen en el sistema fuente, de tal manera que pueden ser almacenados y prorrogados al sistema destino para que se mantenga la consistencia con el sistema de origen y se disminuya la latencia de datos.

En nuestro proyecto se utilizará la técnica CDC por indicadores de estado, donde utilizaremos un indicador de estado cuando se actualizan (por UPDATE, DELETE o INSERT) los datos de una determinada tabla sobre la que están definidos. Por ejemplo, se guardara en un log de archivo plano el indicador de “new”, “changed”, o “deleted”, dependiendo el tipo de operación efectuada y pudiendo referenciar a una o más filas sobre las cuales haya existido una actualización de datos. Cabe destacar que el log registrará información que ha sido transferida al sistema destino y el indicador de estado podrá visualizarse dentro de una columna adicional en el mismo log. En aquellos casos en los que no se registraran actualizaciones de datos el indicador de estado tomará el valor “identical” en todas las filas de la tabla analizada.

5.1.3 Transformación

Como se mencionó en el punto anterior, los datos extraídos desde el sistema operacional serán traspasados a un “área stage”. En ésta los datos serán limpiados y transformados, sin interrumpir el sistema operacional (origen).

En las etapas iniciales del Proyecto de Migración se analizarán los requerimientos del negocio, es decir, cuál es la información que desea analizar la institución para cumplir con sus objetivos estratégicos. Para esto se estudiarán las distintas fuentes que posee la institución de modo de identificar la información que cumple con los requerimientos solicitados, y posteriormente se realizará la “asignación de origen a destino” o también llamado “mapeo lógico”.

El mapeo lógico podrá describir los elementos de origen, elementos de destino y las transformaciones que deberán ser aplicadas entre ellos para cumplir con el esquema de destino. Esta información actúa como los metadatos que nos servirán para elaborar las rutinas de transformación y limpieza correspondientes.

Cabe mencionar que durante el estudio de las fuentes, en la mayoría de los casos se encuentran sistemas de origen poco normalizados y con muchas discrepancias, por lo que resultará indispensable realizar un examen completo de la validez de los datos durante nuestro análisis. A este proceso se le conoce como Perfilamiento de datos (Data Profiling).

Muchas herramientas ETL poseen dentro de sus productos la opción de Perfilamiento y Calidad de los datos, permitiendo realizar análisis sobre los esquemas de base de datos, las tablas y las columnas de las distintas fuentes. De esta forma se podrá obtener información estadística sobre las definiciones de una tabla, el número de registros y esquemas de bases de datos. También se podrá obtener información más específica para cada campo, tales como: cantidad de valores nulos, longitud mínima, longitud máxima, cantidad de valores duplicados, entre otros casos. Además nos permitirá indicar manualmente patrones de validación sobre los campos ya sea con el uso de expresiones regulares como con patrones SQL.

Una fuente de datos “limpia” requerirá una transformación mínima, mientras que una fuente de datos “sucias” requiere transformaciones de datos más complejas. Por dicho motivo es muy recomendable realizar una Limpieza de datos (Data Cleaning) de la fuente de origen antes de realizar las transformaciones asociadas al mapeo lógico.

Dicho lo anterior, la primera etapa del proceso de transformación que llevaremos a cabo corresponde a la limpieza de datos, cuyo objetivo es realizar distintos tipos de acciones contra el mayor número de datos detectados como “sucios” (Dirty Data).

Los datos sucios pueden ser:

- **Datos Inconsistentes:** se producen principalmente porque el modelo de datos no está normalizado correctamente.

- **Datos Incorrectos:** se producen principalmente por errores humanos, pero también se puede producir por modelos de datos poco robustos o errores en la transmisión de los datos.
- **Datos Incompletos:** en esta categoría se encuentran los “datos nulos” (Missing Data), y corresponden básicamente a atributos sin valor ya sea porque no era necesario registrarlo o por errores humanos que no fueron considerados en el modelo de datos.

Podemos dividir en distintas etapas las acciones realizadas por la Limpieza de datos:

- **Estandarizar:** este proceso aplica rutinas de conversión para transformar valores en formatos definidos (y consistentes) aplicando procedimientos de estandarización de acuerdo a los requerimientos definidos por el negocio.
- **Relacionar:** este proceso busca y relaciona los valores de los registros, corrigiéndolos y estandarizándolos, permitiendo además detectar y eliminar duplicados. Por ejemplo, identificando nombres y direcciones similares.
- **Consolidar:** este proceso analiza e identifica relaciones entre registros asociados a un mismo contexto y los junta en una sola representación.

Las acciones mencionadas para la Limpieza de Datos, son las que típicamente se realizan sobre la fuente de datos que posee datos sucios en un contexto de Migración, pudiendo existir más casos dependiendo de la fuente de origen y la calidad de los datos que posee la institución. Estas acciones vienen predefinidas en todas las herramientas ETL que poseen la funcionalidad de “Limpieza de datos”, teniendo que especificar algunos parámetros para su ejecución automática.

Un punto muy importante que se debe tener en cuenta en esta etapa, es identificar el por qué de la anomalía o “dato sucio”, ya que muchos datos deberán ser limpiados constantemente durante el proceso de transformación, debido a que las anomalías detectadas son constantes. En otros casos se pueden encontrar anomalías particulares que no requerirán un tratamiento constante, dado que es poco probable o imposible que se vuelva a repetir.

No obstante, en otros casos puede implicar seleccionar solo ciertas columnas, agregar códigos, obtener nuevos valores calculados, unir datos, dividir una columna en varias, entre otros. Las transformaciones más comunes e importantes corresponden a:

- **Establecer valores predeterminados para los datos nulos:** generalmente existen muchos datos nulos en el sistema operacional. Sin embargo, muchos de estos siguen siendo de interés para el negocio, por tanto deben establecerse valores predeterminados como por ejemplo “-1” o “Sin Dato”, de esta forma se asegura que los valores de interés igualmente serán cargados en el esquema de destino.
- **Usar conversiones y combinaciones para generar nuevos campos:** estas operaciones son muy utilizadas en el proceso de transformación, específicamente para generar indicadores de negocio que serán almacenados en las distintas tablas.

La etapa de transformación y el proceso ETL en general, deberá contar con una buena “gestión de errores” para así mejorar y facilitar el mantenimiento del proceso ETL, ya que su utilización permitirá identificar rápidamente los errores producidos en el sistema. A su vez, permitirá disminuir los tiempos empleados en su búsqueda y darles solución de manera oportuna.

Para la gestión de errores, se utilizarán logs de ejecución en cada flujo de trabajo, almacenando parámetros de ejecución tales como fecha, tiempo, éxito o fracaso, entre otros parámetros. Asimismo, se deberá tener una tabla de errores donde se registren tanto los errores como sus causas. Con esto nos referimos al mensaje de error tanto como la línea que ha generado el error.

Finalmente será importante “validar los datos” que serán cargados, certificando que estos sean válidos y no contengan errores que lleven a tomar decisiones basadas en datos malos. Esto generalmente se realizará en el último componente de un flujo de trabajo antes de realizar la respectiva carga utilizando bloques try/catch para su consiguiente acción. Por tanto, si los datos están correctos se entregarán a la siguiente etapa (carga). Si los datos son erróneos se ejecutarán políticas de tratamiento de excepciones, tales como rechazar el registro completo o dar al campo erróneo un valor nulo o centinela.

El proceso de transformación estará compuesto de distintos “flujos de trabajo o JOB”, quienes describirán las condiciones, los procedimientos y el orden en que serán procesados los distintos flujos de datos para cargar correctamente las tablas.

Podemos concluir que la tarea de transformación será la más compleja del proceso ETL, ya que se encargará de convertir aquellos datos inconsistentes en un conjunto de datos compatibles y congruentes, para que puedan ser cargados en la BD Destino (PostgreSQL). Estas acciones se llevarán a cabo definiendo estándares, aplicando operaciones de limpieza y transformaciones pertinentes a los datos, de modo que se cumpla con el objetivo de tener un Sistema Destino con datos integrados.

5.1.4 Carga

Las transformaciones y operaciones de limpieza que serán realizadas en la etapa anterior (transformación), dejarán los datos listos para ser cargados en el Sistema Destino.

Recordemos que el proceso de carga se separará en dos tipos: carga total y carga incremental. La primera se refiere a la carga de todos los datos que posee la empresa y la segunda se refiere a las actualizaciones que se realizan al Sistema Destino. Las actualizaciones pueden referirse solo a los datos que han cambiado desde la última carga o a todos los datos correspondientes de un periodo determinado.

Las cargas incrementales se utilizarán principalmente para disminuir la latencia de datos y para gestionar datos históricos en el Sistema Destino.

Sin embargo, para cumplir con estos requisitos se deberá tener en cuenta varios aspectos para que la carga pueda efectuarse de manera correcta y el Sistema Destino tenga datos consistentes.

La primera consideración corresponde a la frecuencia de la carga, la cual dependerá de los requerimientos del negocio y el nivel de granularidad que maneja el Sistema Destino. Recordemos que el nivel de granularidad representa el nivel de detalle al que se desea almacenar la información. En algunos casos pueden ser días como en otros casos pueden ser meses.

La segunda consideración, es identificar y evaluar en qué momento se sincronizan los datos extraídos desde el sistema de origen.

Para esto es importante recordar que el acceso al sistema de origen depende de la disponibilidad que estos tengan (durante la noche, en días específicos, etc.), por lo que resulta complicado sincronizar los datos cuando el sistema fuente posee distintas restricciones de acceso.

Ante esta problemática, el uso de un área stage podrá facilitar mucho esta tarea, ya que permitirá almacenar los datos extraídos en tablas de paso, lo que facilitará establecer un procedimiento para identificar en qué momento se sincronizan los datos en dichas tablas sin sobrecargar el sistema de origen.

Teniendo en cuenta las consideraciones mencionadas, la carga de datos al igual que la extracción de datos, puede ser una tarea sencilla si es que se utilizan herramientas ETL, ya que estas poseen una amplia conectividad con diversos esquemas de destino, permitiendo realizar cargas de manera fácil y sin necesidad de crear rutinas especializadas.

Podemos concluir entonces, que el proceso de carga será el encargado de poblar las tablas que se encuentran en el Sistema Destino, permitiendo de esta forma que los usuarios de negocio puedan tener datos consistentes y actualizados.

6.1. INVESTIGACION DE HERRAMIENTAS ETL PARA LA MIGRACION DE DATOS

En la actualidad existen mecanismos que hacen que la información de una base de datos se migre o se transfiera a otras bases de datos.

La migración de datos de diferentes tecnologías de Bases de Datos es una tarea difícil para los DBA sin el uso de una herramienta externa ETL.

Un proceso ETL, a grandes rasgos, consta de un conjunto de procedimientos para obtener los datos desde un origen predefinido, realizar los cambios y aplicar las operaciones necesarias y finalmente almacenar estos datos tratados en el destino elegido.

Las herramientas ETL son quizás el instrumento más importante en una caja de herramientas BI. Ya sea construida o comprada, una buena herramienta ETL en las manos de un desarrollador experimentado puede acelerar el despliegue del proyecto, reducir al mínimo el impacto de los cambios y mitigar los riesgos involucrados en estos proyectos.

Es necesario que las herramientas ETL posean capacidades robustas para realizar perfilamiento de datos, limpieza de datos y gestión de metadatos a nivel global. Asimismo, que aumenten el rendimiento y desempeño para manejar grandes volúmenes de datos y reducir los tiempos de proceso por lotes. En lugar de actualizar el sistema destino desde cero, se desea que las herramientas ETL capturen y actualicen los cambios que han ocurrido en el sistema de origen desde la última carga.

Las herramientas ETL también deben ser capaces de manejar una variedad más amplia de sistemas fuentes de datos, incluyendo Web, XML y las aplicaciones empaquetadas. Además, deben proveer un conjunto de transformaciones más complejas y reutilizables, como también ofrecer una mejor administración para aumentar la fiabilidad y la velocidad de los desarrolladores.

Para la obtención de los resultados realizaremos un estudio de diferentes alternativas que nos ayudará a demostrar cuál es la mejor opción para el desarrollo del caso práctico de esta tesis.

6.1.1 Arquitectura del sistema. Herramienta ETL vs Desarrollos a Medida

Una de las decisiones más importantes que se va a tomar cuando se construyen los procesos ETL es la elección de una arquitectura. Además, es una decisión que hay que tomar en un momento inicial del proyecto y que puede condicionarlo en muchos aspectos. Básicamente, podemos plantear dos escenarios o arquitecturas diferentes:

Herramienta ETL: uso de herramientas diseñadas para la construcción de un sistema ETL. Están especializadas en este ámbito y suelen incluir alguna o todas de las siguientes características:

- Conectividad / capacidades de Adaptación (con soporte a orígenes y destinos de datos): habilidad para conectar con un amplio rango de tipos de estructura de datos, que incluyen

bases de datos relacionales y no relacionales, variados formatos de ficheros, XML, aplicaciones ERP, CRM o SCM, formatos de mensajes estándar (EDI, SWIFT o HL7), colas de mensajes, e-mails, websites, repositorios de contenido o herramientas de ofimática.

- Capacidades de entrega de datos: habilidad para proporcionar datos a otras aplicaciones, procesos o bases de datos en varias formas, con capacidades para programación de procesos batch, en tiempo real o mediante lanzamiento de eventos.
- Capacidades de transformación de datos: habilidad para la transformación de los datos, desde transformaciones básicas (conversión de tipos, manipulación de cadenas o cálculos simples), transformaciones intermedias (agregaciones, sumalizaciones, lookups) hasta transformaciones complejas como análisis de texto en formato libre o texto enriquecido.
- Capacidades de Metadatos y Modelado de Datos: recuperación de los modelos de datos desde los orígenes de datos o aplicaciones, creación y mantenimiento de modelos de datos, mapeo de modelo físico a lógico, repositorio de metadatos abierto (con posibilidad de interactuar con otras herramientas), sincronización de los cambios en los metadatos en los distintos componentes de la herramienta, documentación, etc.
- Capacidades de diseño y entorno de desarrollo: representación gráfica de los objetos del repositorio, modelos de datos y flujos de datos, soporte para test y debugging, capacidades para trabajo en equipo, gestión de workflows de los procesos de desarrollo, etc.
- Capacidades de gestión de datos (calidad de datos, data profiling, etc.).
- Adaptación a las diferentes plataformas hardware y sistemas operativos existentes: Mainframes (IBM Z/OS), AS/400, HP Tandem, Unix, Wintel, Linux, Servidores Virtualizados, etc.
- Las operaciones y capacidades de administración: habilidades para gestión, monitorización y control de los procesos de integración de datos, como gestión de errores, recolección de estadísticas de ejecución, controles de seguridad, etc.

- La arquitectura y la integración: grado de compactación, consistencia e interoperabilidad de los diferentes componentes que forman la herramienta de integración de datos (con un deseable mínimo número de productos, un único repositorio, un entorno de desarrollo común, interoperabilidad con otras herramientas o vía API), etc.
- Capacidades de Arquitectura Orientada a Servicios de cliente (SOA).

Desarrollos a medida: los procesos ETL se desarrollan utilizando desarrollos a medida en un lenguaje de programación específico. Algunas de las ventajas de utilizar los lenguajes de programación pueden ser:

- Herramientas de Debug: que permiten verificar procesos y datos.
- Reutilización de código ya existente. Diseño de código consistente a través de las características de los lenguajes orientados a objetos.
- Uso de recursos humanos internos especializados en el lenguaje.
- Flexibilidad a la hora de construir los procesos y abordar cualquier tipo de tarea que ofrecen los lenguajes de programación.

La elección de un enfoque u otro va a determinar de una forma muy profunda el diseño del sistema ETL. Una herramienta ETL tiene muchísimas ventajas, tal y como hemos visto al enumerar sus características, y su continua evolución hace que cada día se plantee menos la opción de los desarrollos a medida.

Para especificar con más profundidad las ventajas y desventajas de las opciones propuestas, se seleccionarán 4 (cuatro) criterios relevantes y se hará una comparación de los mismos usando la siguiente escala de valores: “Alto, Medio y Bajo”.

En base a los resultados obtenidos de dicha comparación se tomará la decisión de utilizar una herramienta existente en el mercado o desarrollar una propia para el proyecto.

Criterios:

1. Costos de desarrollo del proyecto.

2. Tiempos de ejecución del proyecto.
3. Flexibilidad y adaptación de la herramienta al proyecto.
4. Efectividad para desarrollar las actividades planificadas.

	Herramienta desarrollada a medida	Herramienta ETL
Costos	Altos: Costos de desarrollo de la herramienta + Costos de ejecución del proyecto	Bajos: Solo costos de ejecución del proyecto
Tiempos	Altos: Tiempo de desarrollo de la herramienta + tiempo de ejecución del proyecto	Bajos: Solo tiempo de ejecución del proyecto
Flexibilidad y adaptación	Alta: La herramienta se desarrolla a medida y ajusta para que permita llevar a cabo específicamente las tareas planificadas, cubriendo ampliamente todas las necesidades del proyecto	Media: La herramienta existe en el mercado y fue desarrollada para cumplir tareas que se adaptan a proyectos en general, cubriendo una vasta gama de necesidades, las cuales pueden o no ser aplicadas en el presente TFG
Efectividad	Alta: La herramienta desarrollada cumple con gran efectividad las tareas planificadas del proyecto	Alta: Las herramientas existentes hoy en día en el mercado cumplen con gran efectividad las tareas planificadas del proyecto

Como podemos observar, con una herramienta ETL se disminuyen considerablemente los costos y tiempos de implementación del proyecto con respecto a una herramienta desarrollada a medida, y si bien, la flexibilidad y adaptación son mayores con esta última opción, la efectividad que se obtiene para llevar a cabo el trabajo es la misma con ambas propuestas. Debido a esto, y por todo lo mencionado anteriormente, se ha decidido implementar una herramienta ETL existente en el mercado para el desarrollo del Trabajo Final de Grado.

6.1.2 Herramientas ETL Open Source

Las características de herramientas ETL de software de código abierto nos permiten tener ventajas con respecto al software propietario y estas son:

- Reducción de Costos.

- Flexibilidad en la adaptación de los productos.
- Independencias de Proveedores.
- Empezar una solución con bajos costos y con gran escalabilidad.
- Las fuentes de Open Source viene de muchos orígenes, ya sea de desarrolladores individuales, asociados en grupos, pertenecientes a una empresa, pertenecientes a universidades u organismos públicos.

Las soluciones BI completas en el mercado son muchas, pero Pentaho y SpagoBI son las tecnologías que tienen como fortaleza su herramienta de ETL. Kettle (de Pentaho) y Talend (de SpagoBI) cuentan con software robusto y muy completo que soportan gran variedad de conexiones con diferentes fuentes de datos.

Como ejemplos de soluciones completas están: Pentaho y SpagoBI.

Pentaho: es una de las más completas de todas las soluciones, integra toda una serie de componentes Open Source muy utilizados y fiables: Servidor OLAP Mondrian, JPivot, Kettle ETL, Enhydra Shark, Quartz, Weka, JFreeReport, JFreeChart, JBoss Portal, entre otros.

SpagoBI: es una solución completa de Inteligencia de Negocio que incluye desde la extracción de los datos hasta la minería, pasando por análisis y la generación de reportes. Utiliza componentes muy similares a los de Pentaho.

Comparación entre Pentaho y SpagoBI.

Herramientas\Plataformas	Pentaho	SpagoBI
Servidor de Aplicaciones	JBOSS y Tomcat	JBOSS y Tomcat
Motor de reportes	JFreeReport y BIRT	JasperReport y BIRT
OLAP	Mondrian y JPivot	Mondrian y JPivot
Minería de Datos	WEKA	WEKA
ETL	Kettle	Talend
Planificador	Quartz	Quartz
Posibilidad de extensión	Sí	Sí

Pentaho y SpagoBI utilizan componentes muy similares, con la diferencia de la herramienta de Extracción, transformación y carga: Pentaho incluye Kettle y SpagoBI incluye Talend.

6.1.2.1 Comparación Multicriterio por Scoring

1. Meta General del Problema: seleccionar la mejor herramienta de ETL Open Source.

2. Alternativas.

- Talend: Permite hacer transformaciones y mapeos complejos. Posee un entorno gráfico aunque las transformaciones deben ser programadas íntegramente (Perl), no es muy intuitivo. Soporta conexión mediante JDBC. Es multiplataforma (basado en Java e integrable con Eclipse)
- Kettle: Permite realizar una amplia variedad de transformaciones y posee soporte para el trabajo con almacenes de datos. Posee un entorno gráfico muy intuitivo. Permite el monitoreo de las tareas ejecutadas. Las tareas son ejecutadas en hilos diferentes y su prioridad puede ser administrada. Puede establecer conexión mediante: JDBC, ODBC y JNDI, con alrededor de 25 tipos de gestores de bases de datos. Permite acceder a ficheros: CSV, XML, de texto personalizado, Excel, etc. Permite conexión con Servicios Web. Multiplataforma (basado en Java)

3. Criterios.

- Soporte de conexiones con diversas fuentes de datos.
- Interfáz gráfica.
- Nivel de desarrollo de la herramienta.

Asignación de una ponderación para cada criterio, mediante una escala de 4 (cuatro) puntos:

Ponderaciones:

1=Poco importante

2=Medianamente importante

3=Importante

4=Muy importante

CRITERIOS	PONDERACIÓN (Wi)
1. Soporte de conexiones	4
2. Interfáz gráfica	3
3. Nivel de desarrollo de la herramienta	4

4. Establecer el ranking de satisfacción que la herramienta ofrece al usuario, para cada alternativa, empleando una escala de 4 puntos.

1=Baja

2=Moderada

3=Alta

4=Muy alta

CRITERIOS	TALEND (ri1)	Kettle (ri2)
1. Soporte de conexiones	3	4
2. Interfáz gráfica	2	3
3. Nivel de desarrollo de la herramienta	3	4

5. Calculo de la ponderación para cada alternativa.

CRITERIOS	Ponderacion (Wi)	TALEND (ri1)	Kettle (ri2)
1. Soporte de conexiones	4	3	4
2. Interfáz gráfica	3	2	3
3. Nivel de desarrollo de la herramienta	4	3	4
Score s(j)		30	41

Formula general de cálculo de ponderación.

$S(j) = \text{sumatoria de } (W_i * r_i)$

Ponderación de Talend

Sumatoria de $(r_{i1} * W_i)$

$S(r_{i1}) = (4*3) + (3*2) + (4*3)$

$S(r_{i1}) = 30$

Ponderación de Kettle

Sumatoria de $(r_{i2} * W_i)$

$S(r_{i2}) = (4*4) + (3*3) + (4*4)$

$S(r_{i2}) = 41$

La herramienta Kettle obtiene la ponderación más alta, siendo la mejor alternativa existente en el mercado. Se decide utilizar Pentaho ya que como plataforma tiene una mayor robustez y versatilidad que incluye todos los componentes requeridos.

Etapa 3

MODELO TEÓRICO

7.1. PLANIFICACIÓN

El diagrama gantt de la **figura 7.1** presenta las actividades que se proponen. A continuación se lista cuales son los objetivos que buscan y cuáles son los resultados esperados.

Tareas que se desempeñan para el desarrollo de la migración	Marzo				Abril				Mayo				Junio				Julio			
	Sem 1	Sem 2	Sem 3	Sem 4	Sem 1	Sem 2	Sem 3	Sem 4	Sem 1	Sem 2	Sem 3	Sem 4	Sem 1	Sem 2	Sem 3	Sem 4	Sem 1	Sem 2	Sem 3	Sem 4
Relevamiento de Información	█																			
Limpieza de Datos					█															
Transformación/Mapeo de datos									█											
Carga de Datos													█							
Pruebas																	█			
Revisión de Resultados																			█	

Figura 7.1

7.2. REQUERIMIENTOS

Existen diferentes elementos que van a afectar en cómo va ser o como vamos a construir nuestro sistema ETL. Los más importantes son los requerimientos. La elección de uno o varios procesos de negocio, las tablas que vamos a analizar, su granularidad, etc., van a determinar cosas tan dispares como los orígenes de datos que vamos a tener que utilizar, la forma de procesar la información, la complejidad de los procesos, etc. Esto nos va a hacer dar cuenta de lo importante que son dichos requerimientos y su correcta definición en todas las tareas que realizaremos a continuación, incluyendo la definición de la arquitectura de nuestros procesos ETL. Además el estudio del sistema origen deberá permitir añadir nuevas funcionalidades que pueden no haber sido tenidas en cuenta antes.

En este punto lo que se pretende conseguir es detallar los requisitos que serán implementados en la migración. Además se busca obtener una base a partir de la cual serán construidas y desarrolladas las siguientes etapas del proyecto.

7.2.1 Requerimientos Funcionales

- *Migración de los datos relacionados con la Persona.*
Se considera información de tipo: Personal, Geográfica, Profesional, y Académica asociada con la persona.
- *Migración de los datos relacionados con la Carrera.*
Se considera información general de la Carrera, Inscripción a Carrera, Modalidad, y Planes.
- *Migración de los datos relacionados con el Cursado.*
Se considera información general de la Materia, Inscripción a Materia, y Exámenes.

Los principales usuarios que hacen uso de la información antes mencionada, son los secretarios académicos, alumnos y docentes del Instituto Universitario Aeronáutico.

7.2.2 Requerimientos de Datos

Los datos solicitados para dar soporte a los requerimientos de información descriptos anteriormente se obtienen de la Base de Datos del Sistema de Gestión Académica del IUA,

que reside en un Motor de Base de Datos SQL Server 2000. Dicho sistema, permite la gestión de alumnos registrando y administrando todas las actividades académicas de la universidad, desde que los alumnos ingresan como aspirantes hasta que obtienen el diploma. Fue concebido para administrar la gestión de alumnos en forma segura, con la finalidad de obtener información consistente para los niveles operativos y directivos.

A continuación detallamos las principales tablas, atributos y tipos de datos que serán sometidos a un proceso de mapeo y depuración, para su posterior transferencia al sistema destino:

1. *Tabla ALUMNO:*

ALUMNO	
Atributo	Tipo de Dato
NRO_DOC_ALU	char (10)
TIPO_DOC_ALU	char (3)
NACION_ALU	char (25)
SEXO_ALU	char (1)
ESTADO_CIVIL	varchar (20)
APELLIDO_ALU	char (30)
NOMBRE_ALU	char (30)
FNAC_ALU	datetime
COD_AREA_TE_PART	char (10)
TE_PART_ALU	char (30)
CORREO_EL_ALU	char (50)
MAIL_INSTITUCIONAL	char (70)
CODIGO_ALU	char (10)
BARRIO_ALU	char (40)
CALLE_ALU	char (100)
NRO_ALU	char (5)
DPTO_ALU	char (10)
PISO_ALU	char (3)
D_LOC_ALU	char (40)

Los atributos de la tabla ALUMNO se definen de la siguiente manera:

- a. **NRO_DOC_ALU:** número de documento del alumno. Es clave compuesta en esta tabla conjuntamente con el TIPO_DOC_ALU.
- b. **TIPO_DOC_ALU:** tipo de documento del alumno. Es clave compuesta en esta tabla conjuntamente con el NRO_DOC_ALU.

- c. **NACION_ALU:** nacionalidad del alumno.
- d. **SEXO_ALU:** sexo del alumno.
- e. **ESTADO_CIVIL:** estado civil del alumno.
- f. **APELLIDO_ALU:** apellido del alumno.
- g. **NOMBRE_ALU:** nombre del alumno.
- h. **FNAC_ALU:** fecha de nacimiento del alumno.
- i. **COD_AREA_TE_PART:** código de área de teléfono particular del alumno.
- j. **TE_PART_ALU:** número de teléfono particular del alumno.
- k. **CORREO_EL_ALU:** dirección de correo electrónico particular del alumno.
- l. **MAIL_INSTITUCIONAL:** dirección de correo electrónico provisto por el IUA para el alumno.
- m. **CODIGO_ALU:** código postal que corresponde a la zona donde reside el alumno.
- n. **BARRIO_ALU:** nombre del barrio donde reside alumno.
- o. **CALLE_ALU:** nombre de la calle donde reside el alumno.
- p. **NRO_ALU:** número de la calle donde reside el alumno.
- q. **DPTO_ALU:** letra o número del departamento donde habita el alumno.
- r. **PISO_ALU:** número de piso del departamento donde habita el alumno.
- s. **D_LOC_ALU:** nombre de la localidad donde reside el alumno.

2. *Tabla DATOS_TRABAJO:*

DATOS_TRABAJO	
Atributo	Tipo de Dato
NRO_DOC_ALU	char (10)

TIPO_DOC_ALU	char (3)
CARGO	char (40)
EMPRESA	char (40)
RUBRO	char (40)

Los atributos de la tabla DATOS_TRABAJO se definen de la siguiente manera:

- CARGO:** nombre del cargo laboral en el que se desempeña el alumno.
- EMPRESA:** nombre de la empresa en la que se desempeña el alumno.
- RUBRO:** nombre del rubro de la empresa en la que se desempeña el alumno.

*Se utilizan las mismas claves compuestas definidas en la tabla ALUMNO.

3. Tabla ESTUDIOS_NIVEL_MEDIO:

ESTUDIOS_NIVEL_MEDIO	
Atributo	Tipo de Dato
NRO_DOC_ALU	char (10)
TIPO_DOC_ALU	char (3)
TITULO_NIVEL_MEDIO	varchar (60)
ESPECIALIZACION	varchar (250)
INSTITUCION_NIVEL_MEDIO	varchar (100)
FECHA_ESTUDIO_COMPLETO	datetime
ESTUDIO_COMPLETO	char (1)
ANIO_EGRESO	integer

Los atributos de la tabla ESTUDIOS_NIVEL_MEDIO se definen de la siguiente manera:

- TITULO_NIVEL_MEDIO:** nombre del título que corresponde a los estudios de nivel medio del alumno.
- ESPECIALIZACION:** nombre de la especialización con la que se recibió el alumno en sus estudios de nivel medio.
- INSTITUCION_NIVEL_MEDIO:** nombre del establecimiento secundario (colegio secundario) al que asistió el alumno.
- FECHA_ESTUDIO_COMPLETO:** fecha de egreso del alumno de su estudio secundario.
- ESTUDIO_COMPLETO:** indicador que señala si el alumno dispone o no de estudios de nivel medio completos.

f. **ANIO_EGRESO:** año de egreso del alumno de su estudio secundario.

*Se utilizan las mismas claves compuestas definidas en la tabla ALUMNO.

4. *Tabla TITULO_NIVEL_MEDIO:*

TITULO_NIVEL_MEDIO	
Atributo	Tipo de Dato
TITULO_NIVEL_MEDIO	varchar (60)

La tabla TITULO_NIVEL_MEDIO cuenta con un solo atributo (**TITULO_NIVEL_MEDIO**), que es clave primaria en dicha tabla y fue justificado en la tabla ESTUDIOS_NIVEL_MEDIO, siendo clave foránea en esta última.

5. *Tabla TIPO_DOCUMENTO:*

TIPO_DOCUMENTO	
Atributo	Tipo de Datos
TIPO_DOC_ALU	char (3)
DESCRIPCION_TIPO_DOCUMENTO	varchar (40)

El atributo de la tabla TIPO_DOCUMENTO se define de la siguiente manera:

a. **DESCRIPCION_TIPO_DOCUMENTO:** nombre del documento de acuerdo a su tipo.

*Se utiliza el atributo TIPO_DOC_ALU como clave primaria, el cual ha sido justificado en la tabla ALUMNO.

6. *Tabla TRAMITES_ALUMNO:*

TRAMITES_ALUMNO	
Atributo	Tipo de Datos
NUMERO_TRAMITE	numeric (18,0)
TRAMITE	char (60)
DETALLE	varchar (120)
FECHA_TRAMITE	datetime
LEGAJO	numeric (18,0)

Los atributos de la tabla TRAMITES_ALUMNO se definen de la siguiente manera:

a. **NUMERO_TRAMITE:** identificador correlativo que indica el número del trámite efectuado por el alumno. Es clave primaria en la tabla señalada.

b. **TRAMITE:** nombre del trámite realizado por el alumno.

- c. **DETALLE:** descripción del trámite realizado por el alumno.
- d. **FECHA_TRAMITE:** fecha del trámite realizado por el alumno.
- e. **LEGAJO:** número de legajo que corresponde al alumno.

7. *Tabla TIPO_TRAMITE:*

TIPO_TRAMITE	
Atributo	Tipo de Datos
TRAMITE	char (60)
ID_TIPO_TRAMITE	numeric (18,0)

Los atributos de la tabla TIPO_TRAMITE se definen de la siguiente manera:

- a. **ID_TIPO_TRAMITE:** identificador correlativo que indica el número de tipo de trámite.

*El atributo **TRAMITE** se utiliza como clave primaria en la tabla TIPO_TRAMITE, el cual ha sido justificado en la tabla TRAMITES_ALUMNO, siendo clave foránea en esta última.

8. *Tabla FICHA_INSCRIPCION:*

FICHA_INSCRIPCION	
Atributo	Tipo de Datos
LEGAJO	numeric (18,0)
NRO_DOC_ALU	char (10)
TIPO_DOC_ALU	char (3)
SEDE	char (40)
DEPENDENCIA	varchar (60)
FACULTAD	varchar (60)
MODALIDAD	char (1)
COD_CAR	char (3)
PLAN_CAR	char (4)
ANIO_ACADEMICO	integer
FECHA	datetime
FECHA_INGRESO	datetime
ESTADO_ALUMNO	char (1)
FECHA_REINSCRIPCION	datetime

Los atributos de la tabla FICHA_INSCRIPCION se definen de la siguiente manera:

- a. **SEDE:** nombre de la Sede del IUA.
- b. **DEPENDENCIA:** nombre de la Dependencia del IUA.

- c. **FACULTAD:** nombre de la Unidad Académica del IUA.
- d. **MODALIDAD:** indicador que corresponde al tipo de modalidad de cursado.
- e. **COD_CAR:** código de Carrera a la que se inscribió el Alumno. Es clave foránea en esta tabla y clave compuesta de la tabla CARRERAS conjuntamente con el atributo PLAN_CAR.
- f. **PLAN_CAR:** código del Plan de Carrera a la que se inscribió el Alumno. Es clave foránea en esta tabla y clave compuesta de la tabla CARRERAS conjuntamente con el atributo COD_CAR.
- g. **ANIO_ACADEMICO:** año en el que el alumno se inscribió a una Carrera.
- h. **FECHA:** fecha de inscripción.
- i. **FECHA_INGRESO:** fecha de alta para el cursado de la carrera.
- j. **ESTADO_ALUMNO:** indicador que señala la condición académica (estado) del alumno dentro de la institución.
- k. **FECHA_REINSCRIPCION:** fecha en la que el alumno se reinscribió para un nuevo cursado.

*Se utiliza el atributo **LEGAJO** como clave primaria, el mismo fue justificado en la tabla TRAMITES_ALUMNO. Asimismo, los atributos **NRO_DOC_ALU** y **TIPO_DOC_ALU** han sido justificados en la tabla ALUMNO.

9. Tabla SEDE:

SEDE	
Atributo	Tipo de Dato
SEDE	char (40)
CODIGO_SEDE	char (3)

El atributo de la tabla SEDE se define de la siguiente manera:

- a. **CODIGO_SEDE:** código que corresponde a la Sede en la cual se inscribió el alumno.

*El atributo **SEDE** se utiliza como clave primaria en la tabla señalada, el mismo fue justificado en la tabla FICHA_INSCRIPCION, siendo clave foránea en esta última.

10. Tabla *DEPENDENCIA*:

DEPENDENCIA	
Atributo	Tipo de Dato
DEPENDENCIA	varchar (60)
TIPO_DEPENDENCIA	char (1)

El atributo de la tabla *DEPENDENCIA* se define de la siguiente manera:

- a. **TIPO_DEPENDENCIA**: indicador que discrimina el tipo de Dependencia.

*El atributo **DEPENDENCIA** se utiliza como clave primaria en la tabla señalada, el mismo fue justificado en la tabla *FICHA_INSCRIPCION*, siendo clave foránea en esta última.

11. Tabla *MODALIDAD*:

MODALIDAD	
Atributo	Tipo de Dato
MODALIDAD	char (1)
DESCRIPCION	varchar (30)
UNIDAD_ACADEMICA	varchar (60)

Los atributos de la tabla *MODALIDAD* se definen de la siguiente manera:

- a. **DESCRIPCION**: nombre de la Modalidad.
- b. **UNIDAD_ACADEMICA**: nombre de la Unidad Académica a la que hace referencia una Modalidad determinada.

*El atributo **MODALIDAD** fue justificado en la tabla *FICHA_INSCRIPCION*.

12. Tabla *UNIDAD_ACADEMICA*:

UNIDAD_ACADEMICA	
Atributo	Tipo de Dato
UNIDAD_ACADEMICA	varchar (60)

Esta tabla cuenta con un solo atributo (**UNIDAD_ACADEMICA**) que es clave primaria en la misma. El atributo mencionado fue justificado en la Tabla *MODALIDAD*, siendo clave foránea en esta última.

13. Tabla *CARRERAS*:

CARRERAS	
Atributo	Tipo de Dato
COD_CAR	char (3)

PLAN_CAR	char (4)
NOM_CAR	char (50)
CUATRIMESTRES_PREGRADO	integer
CUATRIMESTRES_GRADO	integer
ESTADO	char (1)
CARRERA_CURSO	varchar (20)
VIGENCIA_MATRICULA	char (1)
CANTIDAD_ELECTIVAS	integer
CANTIDAD_MATERIAS_GRADO	integer
CANTIDAD_MATERIAS_PREGRADO	integer

Los atributos de la tabla CARRERAS se definen de la siguiente manera:

- a. **NOM_CAR:** nombre de la Carrera.
- b. **CUATRIMESTRES_PREGRADO:** número de cuatrimestres que componen una carrera de Pregrado.
- c. **CUATRIMESTRES_GRADO:** numero de cuatrimestres que componen una carrera de Grado.
- d. **ESTADO:** indicador que señala si la Carrera se encuentra activa o dada de baja.
- e. **CARRERA_CURSO:** descripción que indica si es Carrera o Curso.
- f. **VIGENCIA_MATRICULA:** indicador que señala si está vigente o no la matriculación para la Carrera.
- g. **CANTIDAD_ELECTIVAS:** número de materias electivas que dispone la Carrera.
- h. **CANTIDAD_MATERIAS_GRADO:** número de materias de grado que dispone la Carrera.
- i. **CANTIDAD_MATERIAS_PREGRADO:** número de materias de pregrado que dispone la Carrera.

*Los atributos **COD_CAR** y **PLAN_CAR** se utilizan como claves primarias en la tabla CARRERAS. Ambos atributos fueron justificados en la tabla FICHA_INSCRIPCION, siendo claves foráneas en esta última.

14. Tabla UBICACIÓN_FISICA:

UBICACION_FISICA	
Atributo	Tipo de Dato

UBICACION	varchar (60)
------------------	---------------------

Esta tabla cuenta con un solo atributo (**UBICACION**) definido como clave primaria, y corresponde al nombre del edificio en el que se desarrolla algún tipo de actividad que hace a la gestión académica. Dicho atributo es clave foránea en la tabla AULA, que se mencionará en esta sección más adelante.

15. *Tabla ASIGNA_HORARIO_MATERIA:*

ASIGNA_HORARIO_MATERIA	
Atributo	Tipo de Dato
id_AHM	numeric (18,0)
COD_CAR	char (3)
PLAN_CAR	char (4)
CODIGO_DE_MATERIA	char (5)
DIVISION	char (6)
NUMERO_AULA	integer
FECHA_COMIENZO_GRUPO	datetime
FECHA_FIN_GRUPO	datetime
LEGAJO_PERSONAL	numeric (18,0)
CAPACIDAD_CURSO	integer

Los atributos de la tabla ASIGNA_HORARIO_MATERIA se definen de la siguiente manera:

- a. **id_AHM:** identificador numérico correlativo generado cada vez que se le asigna un horario a una materia. Es clave primaria en la tabla señalada.
- b. **CODIGO_DE_MATERIA:** código que identifica a una materia en particular. Es clave compuesta con los atributos **COD_CAR** y **PLAN_CAR**.
- c. **FECHA_COMIENZO_GRUPO:** fecha de inicio de dictado de una materia en particular.
- d. **FECHA_FIN_GRUPO:** fecha de fin de dictado de una materia en particular.
- e. **NUMERO_AULA:** número que identifica un aula determinada. Es clave primaria de la tabla AULA y clave foránea en la tabla ASIGNA_HORARIO_MATERIA.
- f. **LEGAJO_PERSONAL:** número de legajo correspondiente al docente. Es clave primaria de la tabla DOCENTE y clave foránea en la tabla ASIGNA_HORARIO_MATERIA.

- g. **DIVISION:** nombre de la comisión que incluye a un grupo de alumnos que han cursado una materia en particular. Es clave compuesta con los atributos **COD_CAR** y **PLAN_CAR**.
- h. **CAPACIDAD_CURSO:** número del cupo de alumnos que se asignó para el cursado de una materia en particular.

*Los atributos **COD_CAR** y **PLAN_CAR** fueron justificados en la tabla FICHA_INSCRIPCION, siendo claves foráneas con los atributos **CODIGO_DE_MATERIA** y **DIVISION** en la tabla ASIGNA_HORARIO_MATERIA.

16. *Tabla DIA_HORA:*

DIA_HORA	
Atributo	Tipo de Dato
ID_MODULO	numeric (18,0)
id_AHM	numeric (18,0)
COD_CAR	char (3)
PLAN_CAR	char (4)
CODIGO_DE_MATERIA	char (5)
ACTIVIDAD_MODULO	char (50)
FECHA_MODULO	datetime
HORA_FIN	datetime
HORA_INICIO	datetime
NUMERO_AULA	integer
DIVISION	char (6)

Los atributos de la tabla DIA_HORA se definen de la siguiente manera:

- a. **ID_MODULO:** identificador numérico correlativo generado cada vez que se le asigna un día y un horario a una materia en particular. Es clave primaria en la tabla señalada.
- b. **ACTIVIDAD_MODULO:** nombre de la actividad que se desarrolla en un día y horario fijado.
- c. **FECHA_MODULO:** fecha en la que se fijó la realización de una actividad.
- d. **HORA_FIN:** hora en que finalizó una actividad.
- e. **HORA_INICIO:** hora en que inició una actividad.

*Los atributos **id_AHM**, **CODIGO_DE_MATERIA**, **NUMERO_AULA**, y **DIVISION** fueron justificados en la tabla ASIGNA_HORARIO_MATERIA, siendo **id_AHM** y **NUMERO_AULA** claves foráneas en la tabla

DIA_HORA. Asimismo, los atributos **COD_CAR** y **PLAN_CAR** fueron justificados en la tabla FICHA_INSCRIPCION, siendo también claves foráneas en la tabla DIA_HORA.

17. *Tabla TIPO_HORA:*

TIPO_HORA	
Atributo	Tipo de Dato
ID_TIPO_HORA	integer
DESCRIPCION	varchar (255)
TIPO_HORA	varchar (50)

Los atributos de la tabla TIPO_HORA se definen de la siguiente manera:

- a. **ID_TIPO_HORA:** identificador numérico correlativo que identifica el tipo de clase dictada. Es clave primaria en la tabla señalada.
- b. **DESCRIPCION:** detalle de la clase dictada.
- c. **TIPO_HORA:** nombre de la clase dictada.

18. *Tabla INSCRIPCION_EN_MATERI:*

INSCRIPCION_EN_MATERI	
Atributo	Tipo de Dato
NUMERO_TRANSACCION	numeric (18,0)
LEGAJO	numeric (18,0)
COD_CAR	char (3)
PLAN_CAR	char (4)
CODIGO_DE_MATERIA	char (5)
LEGAJO_PERSONAL	numeric (18,0)
FECHA_INSCRIP_EN_MATERIA	datetime
DIVISION_SEDE	varchar (10)

Los atributos de la tabla INSCRIPCION_EN_MATERI se definen de la siguiente manera:

- a. **NUMERO_TRANSACCION:** número que se crea automáticamente asignándose a cada inscripción de materia. Es clave primaria en la tabla señalada.
- b. **FECHA_INSCRIP_EN_MATERIA:** fecha de inscripción a una materia.
- c. **DIVISION_SEDE:** código que indica la comisión que se abrió para el cursado de una materia.

*El atributo **LEGAJO** ha sido justificado en la tabla **TRAMITES_ALUMNOS** y los atributos **COD_CAR** y **PLAN_CAR** fueron justificados en la tabla **FICHA_INSCRIPCION**, asimismo el atributo

CODIGO_DE_MATERIA fue justificado en la tabla **ASIGNA_HORARIO_MATERIA** al igual que **LEGAJO_PERSONAL**.

LEGAJO, **COD_CAR**, **PLAN_CAR**, **CODIGO_DE_MATERIA** y **LEGAJO_PERSONAL** son claves foráneas en la tabla **INSCRIPCION_EN_MATERI**.

19. *Tabla MATERIA:*

MATERIA	
Atributo	Tipo de Dato
CODIGO_DE_MATERIA	char (5)
COD_CAR	char (3)
PLAN_CAR	char (4)
NOMBRE_DE_MATERIA	char (50)
TIPO_MATERIA	numeric (18,0)
CUATRIMESTRE_DE_MATERIA	integer
Cursar	char (1)
rendir_libre	char (1)
EN_ACTA	char (1)

Los atributos de la tabla **MATERIA** se definen de la siguiente manera:

- d. **NOMBRE_DE_MATERIA:** nombre de la materia.
- e. **TIPO_MATERIA:** número que señala el tipo de materia, por ej: 1 (Curricular), 2 (No Curricular), 3 (Electiva), etc.
- a. **CUATRIMESTRE_DE_MATERIA:** número que indica el cuatrimestre en el cuál se dicta la materia.
- b. **cursar:** indicador que señala si la materia requiere ser cursada o no.
- c. **rendir_libre:** indicador que señala si la materia puede rendirse libre o no.
- d. **EN_ACTA:** indicador que señala si la materia se encuentra incluida en Acta (S), no incluida en Acta (N), o Anulada (X).

*El atributo **CODIGO_DE_MATERIA** fue justificado en la tabla **ASIGNA_HORARIO_MATERIA**, mientras que **COD_CAR** y **PLAN_CAR** fueron justificados en la tabla **FICHA_INSCRIPCION**. Estos tres atributos conjuntamente se definen como claves compuestas de la tabla **MATERIA**.

20. *Tabla TIPO_MATERIA:*

TIPO_MATERIA	
Atributo	Tipo de Dato

TIPO_MATERIA	numeric (18,0)
NOMBRE_TIPO	varchar (50)
PROMEDIA	char (1)

Los atributos de la tabla TIPO_MATERIA se definen de la siguiente manera:

- a. **NOMBRE_TIPO:** nombre que indica el tipo de materia.
- b. **PROMEDIA:** indicador que señala si la materia es promediable o no.

*El atributo **TIPO_MATERIA** es clave primaria en la tabla TIPO_MATERIA, y fue justificado en la tabla MATERIA, siendo clave foránea en esta última.

21. Tabla SISTEMA_CALIF:

SISTEMA_CALIF	
Atributo	Tipo de Dato
ID_SIST_CALIF	numeric (18,0)
TIPO_MATERIA	numeric (18,0)
VALOR_REAL	float (8)
APROBADO	varchar (50)

Los atributos de la tabla SISTEMA_CALIF se definen de la siguiente manera:

- a. **ID_SIST_CALIF:** identificador numérico correlativo que identifica el sistema de calificación para un tipo de materia. Es clave primaria en la tabla señalada.
- b. **VALOR_REAL:** valor numérico que indica la nota que puede contener un tipo de materia.
- c. **APROBADO:** descripción que indica la condición de Aprobado, No Aprobado, o Ausente, dependiendo del Valor Real aplicado.

*El atributo **TIPO_MATERIA** fue justificado en la tabla MATERIA, siendo clave foránea en la tabla SISTEMA_CALIF.

22. Tabla EVALUACION

EVALUACION	
Atributo	Tipo de Dato
LEGAJO	numeric (18,0)
COD_CAR	char (4)
PLAN_CAR	char (3)
CODIGO_DE_MATERIA	char (5)
CALIFICACION_EVALUACION	float (8)
FECHA_GRABACION	datetime

TIPO_DE_ACTIVIDAD	char (12)
-------------------	-----------

Los atributos de la tabla EVALUACION se definen de la siguiente manera:

- a. **CALIFICACION_EVALUACION:** valor que permite calificar la evaluación de una actividad.
- b. **FECHA_GRABACION:** fecha del período académico que se toma la evaluación.
- c. **TIPO_DE_ACTIVIDAD:** descripción del tipo de actividad que se está evaluando (ej. Actividad Obligatoria, Parcial, Equivalencia).

*El atributo **LEGAJO** ha sido justificado en la tabla TRAMITES_ALUMNOS y los atributos **COD_CAR** y **PLAN_CAR** fueron justificados en la tabla FICHA_INSCRIPCION; asimismo el atributo **CODIGO_DE_MATERIA** fue justificado en la tabla ASIGNA_HORARIO_MATERIA. Todos los mencionados son clave foránea en la tabla EVALUACION.

23. Tabla AULA:

AULA	
Atributo	Tipo de Dato
NUMERO_AULA	integer
CAPACIDAD	integer
UBICACION	varchar (60)

Los atributos de la tabla AULA se definen de la siguiente manera:

- a. **NUMERO_AULA:** número que identifica el espacio donde se dicta una materia. Es clave primaria en la tabla señalada.
- b. **CAPACIDAD:** número que indica la cantidad de alumnos que pueden ocupar un espacio durante el dictado de una materia.
- c. **UBICACIÓN:** nombre del edificio donde residen las aulas o espacios que son ocupados por una determinada cantidad de alumnos.

24. Tabla MESA_EXAMEN:

MESA_EXAMEN	
Atributo	Tipo de Dato
COD_CAR	char (3)
PLAN_CAR	char (4)
CODIGO_DE_MATERIA	char (5)

FECHA_EXAMEN_FINAL	Datetime
ID_MESA	numeric (18,0)

Los atributos de la tabla MESA_EXAMEN se definen de la siguiente manera:

- a. **FECHA_EXAMEN_FINAL**: fecha propuesta para el Examen Final. Este atributo se utiliza en la tabla señalada como clave compuesta conjuntamente con los atributos **COD_CAR**, **PLAN_CAR** y **CODIGO_MATERIA**.
- b. **ID_MESA**: número que identifica una Mesa de Examen.

*El atributo **CODIGO_DE_MATERIA** fue justificado en la tabla ASIGNA_HORARIO_MATERIA, mientras que **COD_CAR** y **PLAN_CAR** fueron justificados en la tabla FICHA_INSCRIPCION.

25. *Tabla INSCRIPCION_EN_EXAMEN:*

INSCRIPCION_EN_EXAMEN	
Atributo	Tipo de Dato
LEGAJO	numeric (18,0)
COD_CAR	char (3)
PLAN_CAR	char (4)
CODIGO_DE_MATERIA	char (5)
FEC_HOR_GRAB	Datetime
FECHA_EXAMEN	Datetime
TIPO_EXAMEN	char (15)
CALIFICACION_EXAMEN	float (8)
NUMERO_ACTA	Integer
LIBRO	Integer
FOLIO	Integer

Los atributos de la tabla INSCRIPCION_EN_EXAMEN se definen de la siguiente manera:

- a. **FECHA_HORA_GRAB**: fecha y hora en que se registró la inscripción a un examen.
- b. **FECHA_EXAMEN**: fecha del período académico que se toma el examen.
- c. **TIPO_EXAMEN**: descripción que indica el tipo de examen (ej. Ingreso, Final).
- d. **CALIFICACION_EXAMEN**: valor que permite calificar la evaluación de un examen.
- e. **NUMERO_ACTA**: número que identifica el Acta sobre la cual quedo registrado el examen.

f. **LIBRO:** número que identifica el Libro que contiene el Acta sobre la cual quedo registrado el examen.

g. **FOLIO:** número que identifica la hoja del Acta sobre la cual quedo registrado el examen.

*El atributo **CODIGO_DE_MATERIA** fue justificado en la tabla ASIGNA_HORARIO_MATERIA, mientras que el atributo **LEGAJO** ha sido justificado en la tabla TRAMITES_ALUMNOS. Asimismo, los atributos **COD_CAR** y **PLAN_CAR** fueron justificados en la tabla FICHA_INSCRIPCION. Estos cuatro atributos son claves foráneas en la tabla INSCRIPCION_EN_EXAMEN.

26. *Tabla DOCENTE:*

DOCENTE	
Atributo	Tipo de Dato
LEGAJO_PERSONAL	numeric (18,0)
TIPO_DOC_DOCENTE	char (3)
SEXO_DOCENTE	char (1)
APELLIDO_DOCENTE	char (25)
NOMBRE_DDOCENTE	char (30)
FECHA_DE_NACIMIENTO	Datetime
CODIGO_TITULO	numeric (18,0)
COD_AREA_TE_PART	char (10)
TE_CELULAR	char (30)
TELEFONO_	char (30)
MAIL_PARTICULAR	varchar (50)
NUM_DOC_DOCENTE	char (10)
CODIGO_POSTAL	char (18)
BARRIO	char (25)
DOMICILIO_PARTICULAR	char (30)
NUMERO_DOM	Integer
DPTO	char (5)
PISO	char (3)

Los atributos de la tabla DOCENTE se definen de la siguiente manera:

a. **TIPO_DOC_DOCENTE:** nomenclatura que indica el tipo de documento del docente.

b. **SEXO_DOCENTE:** letra que indica el sexo del docente de manera abreviada.

c. **APELLIDO_DOCENTE:** descripción del apellido del docente.

d. **NOMBRE_DDOCENTE:** descripción del nombre del docente.

- e. **FECHA_DE_NACIMIENTO:** fecha de nacimiento del docente.
- f. **CODIGO_TITULO:** identificador numérico que indica el título que tiene el docente.
- g. **COD_AREA_TE_PART:** código de área de teléfono particular del docente.
- h. **TE_CELULAR:** número de teléfono celular del docente.
- i. **TELEFONO_:** número de teléfono particular del docente.
- j. **MAIL_PARTICULAR:** dirección de correo electrónico particular del docente.
- k. **NUM_DOC_DOCENTE:** número de documento del docente.
- l. **CODIGO_POSTAL:** código postal que corresponde a la zona donde reside el docente.
- m. **BARRIO:** nombre del barrio donde reside el docente.
- n. **DOMICILIO_PARTICULAR:** nombre de la calle que corresponde al domicilio particular donde reside el docente.
- o. **NUMERO_DOM:** número de la calle que corresponde al domicilio donde reside el docente.
- p. **DPTO:** letra o número que corresponde al departamento donde habita el docente.
- q. **PISO:** número del piso que corresponde al departamento donde habita el docente.

*Se utiliza el atributo **LEGAJO_PERSONAL** como clave primaria, el cual ha sido justificado en la tabla ASIGNA_HORARIO_MATERIA, siendo clave foránea en esta última.

27. *Tabla DOCENTE_FUNCION:*

DOCENTE_FUNCION	
Atributo	Tipo de Dato
LEGAJO_PERSONAL	numeric (18,0)
DEPENDENCIA	varchar (60)
CARGO	varchar (60)
UNIDAD_ACADEMICA	varchar (60)
E_MAIL_IUA	varchar (50)

Los atributos de la tabla **DOCENTE_FUNCION** se definen de la siguiente manera:

- a. **E_MAIL_IUA**: dirección de correo electrónico provista por el IUA para el docente.
- b. **CARGO**: función que desempeña el docente dentro de la Institución.

*El atributo **CARGO** es clave compuesta de la tabla **DOCENTE_FUNCION** conjuntamente con los atributos **LEGAJO_PERSONAL** y **DEPENDENCIA**. El atributo **LEGAJO_PERSONAL** ha sido justificado en la tabla **ASIGNA_HORARIO_MATERIA**, mientras que el atributo **DEPENDENCIA** ha sido justificado en la tabla **FICHA_INSCRIPCION**. Asimismo, el atributo **UNIDAD_ACADEMICA** ha sido justificado en la tabla **MODALIDAD**.

28. *Tabla CARGO:*

CARGO	
Atributo	Tipo de Dato
CARGO	varchar (60)

Esta tabla cuenta con un solo atributo (**CARGO**) que es clave primaria en la misma y fue justificado en la tabla **DOCENTE_FUNCION**, siendo clave compuesta en esta última.

29. *Tabla EGRESADO:*

EGRESADO	
Atributo	Tipo de Dato
LEGAJO	numeric (18,0)
id_nivel_titulo	integer
FECHA_APROBACION_TFI	datetime
FECHA_INGRESO	datetime

Los atributos de la tabla **EGRESADO** se definen de la siguiente manera:

- a. **id_nivel_titulo**: identificador numérico que señala si el título es de Pregrado o de Grado.
El mismo es clave compuesta conjuntamente con el atributo **LEGAJO**.
- b. **FECHA_APROBACION_TFI**: fecha de aprobación del Trabajo Final.
- c. **FECHA_INGRESO**: fecha de ingreso del alumno.

*El atributo **LEGAJO** ha sido justificado en la tabla **TRAMITES_ALUMNO**. El mismo es clave compuesta conjuntamente con **id_nivel_titulo** en la tabla **EGRESADO**.

30. Tabla TITULO:

TITULO	
Atributo	Tipo de Dato
CODIGO_TITULO	numeric (18,0)
TITULO	varchar (100)

Los atributos de la tabla TITULO se definen de la siguiente manera:

- a. **TITULO:** descripción del nombre del título que tiene el docente.

*Se utiliza el atributo **CODIGO_TITULO** como clave primaria en la tabla TITULO y ha sido justificado en la tabla DOCENTE, siendo clave foránea en esta última.

31. Tabla nivel_titulacion:

nivel_titulacion	
Atributo	Tipo de Dato
id_nivel_titulo	integer
Descripcion	varchar (100)

Los atributos de la tabla nivel_titulacion se definen de la siguiente manera:

- a. **descripción:** detalle del nombre que indica el nivel del título.

*El atributo **id_nivel_titulo** es clave primaria en la tabla nivel_titulacion y ha sido justificado en la tabla EGRESADO, siendo clave compuesta en esta última.

32. Tabla titulación:

Titulacion	
Atributo	Tipo de Dato
id_titulacion	integer
id_nivel_titulo	integer
nombre_oficial	varchar (255)

Los atributos de la tabla titulacion se definen de la siguiente manera:

- a. **id_titulacion:** identificador numérico generado correlativamente que permite identificar el nombre oficial de un título. Es clave primaria en la tabla señalada.
- b. **nombre_oficial:** descripción que indica el nombre oficial de un título.

*El atributo **id_nivel_titulo** ha sido justificado en la tabla EGRESADO, siendo clave foránea en la tabla titulacion.

33. Tabla PAIS:

PAIS	
Atributo	Tipo de Dato
COD_PAIS	integer
PAIS	varchar (25)

Los atributos de la tabla PAIS se definen de la siguiente manera:

- COD_PAIS:** número que se genera de manera correlativa y que identifica a un País en particular. Es clave primaria en la tabla señalada.
- PAIS:** descripción que indica el nombre de un País en particular.

34. Tabla PROVINCIA:

PROVINCIA	
Atributo	Tipo de Dato
COD_PROVINCIA	numeric (18,0)
PROVINCIA	char (40)
COD_PAIS	integer

Los atributos de la tabla PROVINCIA se definen de la siguiente manera:

- COD_PROVINCIA:** número que se genera de manera correlativa y que identifica a una Provincia en particular. Es clave primaria en la tabla señalada.
- PROVINCIA:** descripción que indica el nombre de una Provincia en particular.

*El atributo COD_PAIS ha sido justificado en la tabla PAIS, siendo clave foránea en la tabla PROVINCIA.

35. Tabla LOCALIDAD:

LOCALIDAD	
Atributo	Tipo de Dato
COD_LOCALIDAD	numeric (18,0)
LOCALIDAD	char (40)
COD_PROVINCIA	numeric (18,0)

Los atributos de la tabla LOCALIDAD se definen de la siguiente manera:

- COD_LOCALIDAD:** número que se genera de manera correlativa y que identifica a una Localidad en particular. Es clave primaria en la tabla señalada.

b. **LOCALIDAD:** descripción que indica el nombre de una Localidad.

*El atributo **COD_PROVINCIA** ha sido justificado en la tabla PROVINCIA, siendo clave foránea en la tabla LOCALIDAD.

36. Tabla PERFIL:

PERFIL	
Atributo	Tipo de Dato
CODIGO_PERFIL	integer
USUARIO	varchar (20)
SEDE	char (40)

Los atributos de la tabla PERFIL se definen de la siguiente manera:

a. **CODIGO_PERFIL:** código numérico que identifica el perfil de un usuario. Es clave compuesta en la tabla señalada conjuntamente con el atributo **LEGAJO_PERSONAL**.

b. **USUARIO:** descripción que indica el nombre del usuario.

*El atributo **SEDE** ha sido justificado en la tabla SEDE, siendo clave foránea en la tabla PERFIL.

37. Tabla TIPO_PERFIL:

TIPO_PERFIL	
Atributo	Tipo de Dato
CODIGO_PERFIL	integer
PERFIL	varchar (40)

Los atributos de la tabla TIPO_PERFIL se definen de la siguiente manera:

a. **PERFIL:** descripción del tipo de perfil que tiene asignado un usuario.

*Se utiliza el atributo **CODIGO_PERFIL** como clave primaria en la tabla **TIPO_PERFIL** y ha sido justificado en la tabla **PERFIL**, siendo clave compuesta en esta última.

7.2.3 Requerimientos de software.

7.2.3.1 Pentaho

A continuación se describen los módulos utilizados en el presente trabajo final de grado.

Pentaho Data Integration (PDI)

Pentaho Data Integration es una herramienta Open Source que permite implementar los procesos ETL. También se la conoce como Kettle, acrónimo recursivo de: "Kettle Extraction, Transformation, Transportation, and Load Environment ").

PDI se caracteriza por: no tener costes de licencia, ser multiplataforma (se puede utilizar en windows, macintosh, Linux), tener un entorno gráfico de desarrollo, ser fácil de instalar y configurar.

Las soluciones de PDI se basan en dos diferentes tipos de objetos:

- Transformaciones: Una transformación es una colección de pasos. Cada paso representa una operación particular sobre los datos. Los pasos se conectan entre sí a través de saltos y trabajan de manera simultánea y asincrónica.
- Trabajos: Un trabajo presenta una orientación a la tarea y a los datos. Usualmente un Trabajo se compone de una o más Transformaciones, que serán ejecutadas secuencialmente.

Básicamente PDI incluye cuatro herramientas:

- *Spoon*: para diseñar transformaciones y trabajos utilizando un entorno gráfico.
- *Pan*: para ejecutar transformaciones mediante líneas de comando.
- *Kitchen*: para ejecutar trabajos mediante líneas de comando.
- *Carte*: es un server liviano para ejecutar transformaciones y trabajos en host remotos.

Instalación

Para instalar PDI es necesario:

- Instalar Java Runtime Environment (JRE) versión 1.5 o superior.
- Descargar Kettle, desde:
<http://sourceforge.net/projects/pentaho/files/Data%20Integration/4.4.0-stable/>
- Descomprimir el archivo descargado, en un directorio de su elección.

- En ambientes tipo Unix (Solaris, GNU/Linux, MacOS, por ejemplo), se debe hacer que los scripts en shell sean ejecutables.

7.2.3.2 Verificación del funcionamiento de Pentaho

- *Verificación de las Consideraciones Generales del Producto*

Hemos verificado que:

- El producto funciona en las plataformas actuales de la Institución y si es compatible con las herramientas y utilidades existentes.
 - El producto es compatible con todos los procesos ETL (extracción, transformación y carga) y proporciona un soporte completo para el ciclo de vida de desarrollo, prueba e implementación.
 - La herramienta proporciona un desempeño adecuado y compatible con las transformaciones que necesita la organización.
- *Verificación de las Características de diseño*

Hemos verificado que la herramienta ofrece:

- Un entorno de desarrollo grafico integrado.
- Diagramas interactivos de flujos de trabajo para representar los procesos, tareas y datos.
- Una interfaz grafica “drag and drop” para asignar datos desde la fuente al destino.
- Reutilización de contenedores y objetos personalizados en múltiples flujos de trabajo y proyectos.
- Un procedimiento o entorno de desarrollo orientado a objetos.
- Un lenguaje de scripting para escribir objetos y rutinas personalizadas de transformación.
- Salidas para llamar a códigos o rutinas externas sobre la misma o diferente plataforma.
- Control de versiones.

- Un registro de auditoría que sigue los cambios a todos los procesos, trabajos, tareas, y objetos.

- *Verificación de las Características de Transformación*

Hemos verificado que la herramienta ofrece:

- Una amplia variedad de objetos base de transformación.
- Soporte para múltiples tipos de transformación y funciones de asignación.
- La capacidad de crear archivos temporales, si es necesario para las transformaciones complejas.
- La capacidad para realizar procesos recursivos o bucles.

- *Verificación de las Características de Calidad de Datos*

Hemos verificado que la herramienta ofrece:

- Herramientas de perfilamiento y limpieza de datos.
- Tareas de limpieza que se integran en los flujos de trabajo y diagramas visuales.
- Generación automática y personalizada de reglas que detectan y arreglan los defectos en los datos.

- *Verificación de las Características de Rendimiento*

Hemos verificado que la herramienta ofrece:

- Una manera flexible y de alto rendimiento para procesar y transformar los datos.
- Alta velocidad de procesamiento usando múltiples flujos de trabajo concurrentes, varios procesadores de hardware, sistemas de hebras y servidores de cluster.

- *Verificación de las Características de Extracción y Captura*

Hemos verificado que la herramienta ofrece:

- Capacidad para programar las extracciones por tiempo, intervalo o evento.
- Una amplia variedad de adaptadores para extraer datos desde diversos tipos de fuentes.

- Un marco de desarrollo que permita crear adaptadores de datos personalizados.
 - La selección, asignación y fusión de los registros desde múltiples ficheros fuentes.
 - Facilidad para capturar los cambios en los archivos de origen.
- *Verificación de las Características de Carga y Actualización*

Hemos verificado que la herramienta ofrece:

- Cargar datos en diferentes tipos de sistemas de destino.
 - Cargar datos en diferentes tipos de sistemas de destino en paralelo:
 - Soportar particiones de datos en el destino.
 - Actualizar y añadir datos en la base de datos de destino.
 - Actualizar incrementalmente los datos de destino.
 - Soportar servicios de carga de alta velocidad.
 - Quitar los índices de integridad referencial e indexaciones, en caso de ser necesario.
- *Verificación de las Características de Operación y Administración*

Hemos verificado que la herramienta ofrece:

- Una consola visual para administrar y monitorear los procesos ETL.
- La capacidad de validar trabajos antes de ejecutarlos.
- Un control gráfico robusto que muestra en tiempo real:
 - La posibilidad de reiniciar desde los puntos de control sin perder los datos.
 - Notificaciones de errores y estadísticas de los trabajos.
 - Capacidad para gestionar los registros rechazados.
 - Informes administrativos útiles y fáciles de entender.

7.3. ANALISIS Y DISEÑO

Esta etapa corresponde al estudio de la forma en que los datos serán transferidos de SQL Server a PostgreSQL, para asentar las bases de futuros desarrollos e implementaciones sobre el sistema SIU-Guaraní3. Con los modelos de planificación acordados, empezaremos con la implementación de la Etapa de Análisis.

7.3.1 Análisis de la Migración de Datos

7.3.1.1 Consideraciones generales sobre las plataformas involucradas.

El Sistema de Gestión Académica del I.U.A dispone de una Base de Datos relacional, denominada: *Base_Distancia*, que se levanta desde un Motor SQL Server 2000 y corre sobre un S.O Windows 2003 Server.

El SIU-Guaraní3 se conecta con una Base de Datos relacional, denominada *SIU-Guarani3*, que se levanta desde un Servidor PostgreSQL 8.3 a partir de una plataforma de S.O Linux.

Para poder efectuar una correcta determinación de equivalencias y realizar un adecuado mapeo de datos será muy importante familiarizarnos con algunos conceptos definidos en SIU-Guarani3, los cuales se declaran a continuación:

❖ **ACTIVIDADES:** es la mínima actividad a la que puede aspirar a realizar un alumno. Estas actividades pueden ser:

- *Materias*
- *Seminarios*
- *Actividades extracurriculares*
- *Cursos*
- *Requisitos (por ejemplo: nivel de inglés, certificado de salud,...)*
- *etc.*

También pueden representar actividades realizadas en otras instituciones.

❖ **ACCIÓN:** la acción es un concepto que generaliza algunas características para las operaciones que realicen una misma acción. Un usuario solo ejecutará acciones a partir del uso de las operaciones disponibles.

Una acción es la ejecución de una tarea sobre una o más entidades. Por ejemplo:

- *Inscribir a Cursar: sobre una Comisión*
- *Inscribir a Propuesta: sobre una Propuesta*
- *Cambiar de plan: sobre los Planes origen y destino*
- *Inscribir a Rendir: sobre una Mesa de Examen*
- *Generar Acta de Cursado: sobre una Comisión*

❖ **ACTA:** en Guarani3 un acta es el documento que certifica los Resultados de Evaluación de una Evaluación de una Entidad de Gestión.

❖ **ALUMNO:** un alumno es una persona que cursó o está cursando una propuesta formativa en la institución.

❖ **AÑOS ACADÉMICOS:** el año académico es la unidad de medida mayor que contiene al Calendario Académico de la Institución. Hay un solo año académico activo para la Institución.

❖ **ASPIRANTE:** un aspirante es una persona que se ha preinscripto a una Propuesta Formativa en la Institución.

❖ **CLASE:** una clase es la unidad mínima de cursada correspondiente a una Banda Horaria de una comisión, identificada por un horario de inicio y finalización, en un espacio de la institución en una fecha.

❖ **COMISIÓN:** es una Entidad de Gestión definida para que los alumnos se inscriban a Cursar / Promocionar una actividad.

❖ **DOCENTE:** en Guarani 3, un docente es una Persona al que se le agregan los atributos vinculados al ejercicio de la docencia en la institución.

- ❖ **EDIFICIO:** edificación que contiene aulas y/o oficinas o dependencias destinados a realizar tareas de gestión administrativa, académica, de extensión y de investigación. Todo edificio tiene definida una Ubicación.
- ❖ **ELEMENTO:** un Elemento es un componente de una propuesta. Los Elementos pueden ser: Módulos, Actividades y Subactividades.
- ❖ **ESCALA DE NOTAS:** es una sucesión ordenada de valores distintos que permite calificar las evaluaciones de las actividades de un alumno dentro de la institución.
- ❖ **ESPACIO:** lugar identificable en el cual se realizan actividades educativas, administrativas, recreativas, etc. Todo espacio tiene asociado un Edificio.
- ❖ **EVALUACIÓN:** una Evaluación es una Entidad que se define para gestionar la evaluación de una actividad académica. Es un concepto que representa en forma genérica una evaluación que puede ser utilizada en una mesa, comisión, subcomisión (Entidades de Gestión).
- ❖ **ENTIDAD:** una Entidad identifica a todo componente lógico al cual se le pueden definir Requisitos para realizar determinadas acciones y/o Formas de Aprobación. Una Entidad es una generalización de Módulos, Actividades, Planes, Responsables Académicas, etc.
- ❖ **IDENTIFICACIÓN PERSONA:** cada persona registrada en el sistema se lo identifica por un tipo de documento, número de documento y país del documento. Las personas pueden tener registrado más de un documento que lo identifique, pero solo uno de ellos será el documento principal. Este documento principal es el que se muestra en todos los listados del sistema. El tipo de documento es el que indica el orden de prioridad para definir qué documento de la persona es el documento principal.

Tipos de Documentos: Documento Nacional de Identidad, Doc. Nac. de Identidad Temporal, Cedula de Identidad, CUIL/CUIT, Libreta de Enrolamiento, Libreta Cívica, Pasaporte.
- ❖ **INSCRIPCIÓN:** una Inscripción es el acto administrativo que registra la voluntad de un alumno de:

- a) *cumplir una Instancia (cursar, promocionar, rendir, etc.) en una actividad.*
- b) *rendir una Evaluación definida para una Actividad en una Entidad de Gestión.*

- ❖ **INSTITUCIÓN:** una institución es un establecimiento u organismo que realiza una labor de tipo social, educativa, cultural, de investigación, gubernamental, etc. La Universidad es una Institución. Dentro de este concepto se incluyen organismos estatales, privados y empresas.
- ❖ **LLAMADOS DE EXAMEN:** un llamado es un Período que está dentro de un Turno de Examen y cuenta con uno o varios Períodos de Inscripción. Cada llamado tendrá asociadas una o varias Mesas de Examen.
- ❖ **MESA DE EXAMEN:** es una Entidad de Gestión definida para que los alumnos se inscriban a rendir examen de una Actividad.
- ❖ **MODALIDAD DE CURSADA:** forma en que el Alumno puede realizar las actividades de una Propuesta. Las modalidades de cursadas pueden ser: Presencial, Semi Presencial, A distancia. A cada Plan de Estudios de una propuesta se le define un conjunto de modalidades. Cada una de las actividades del plan de estudios puede tomar la/las modalidad/des definidas en ese conjunto.
- ❖ **PERÍODO GENÉRICO:** es una tipificación que se aplica a los Períodos y se conservan para los diferentes Años Académicos. Ejemplos: Primer Cuatrimestre, Segundo Cuatrimestre, Primer Ingreso.
- ❖ **PERÍODO LECTIVO:** es el Período definido en un Año Académico de un tipo de Período Lectivo Genérico en el cual se desarrolla la cursada de una actividad.
- ❖ **PLAN DE ESTUDIOS:** un plan de estudios es el diseño curricular concreto de una determinada Propuesta Formativa, cuya superación da derecho a la obtención de uno o varios Certificados. Una propuesta formativa puede tener uno o más planes de estudios y cada uno de estos planes a su vez puede tener una o más versiones del mismo. Un plan de estudios puede diferenciarse de otro plan dentro de una misma propuesta curricular por diferentes causas, como ser:

- *La modalidad de cursada (presencial, a distancia, mixto).*
- *Las actividades que la componen.*
- *Las certificaciones que otorga.*
- *etc.*

A cada Plan de Estudios se le deberá definir las modalidades de cursadas de sus actividades.

Un plan puede tener diferente cantidad y tipos de certificados. Así por ejemplo, un plan de estudios puede tener más de un título de grado o más de un título intermedio.

Podrá haber más de un plan de estudios en estado Activo Vigente dentro de una propuesta curricular. Esto puede estar dado ya que podría haber un plan de estudios en modalidad Presencial y otro a Distancia.

Una versión de un plan de estudios identifica la composición (componentes y reglas) del mismo en un momento particular.

❖ ***PROPUESTA FORMATIVA:*** es un agrupamiento de actividades y módulos organizado por reglas dentro de un plan de estudios que conforman la oferta educativa de la Institución. Toda oferta educativa debería ser cargada como Propuesta Formativa. Una propuesta formativa puede ser:

- *Una carrera.*
- *Un ciclo básico, si el mismo se oferta independientemente de una carrera.*
- *Un conjunto de actividades que se ofrecen a la comunidad.*
- *etc.*

Tipos de Propuestas Formativas:

- *Pregrado*
- *Grado*
- *Posgrado*
- *Curso de ingreso*
- *Ciclo Básico o Común*
- *Postítulo*

- *Terciario*
 - *Vocacional*
 - *Extracurricular*
- ❖ **RESPONSABLE ACADEMICA:** el concepto de Responsable Académica abarca cualquier "área" dentro de la Institución que defina Propuestas, defina Elementos y realice Convenios Académicos con otras Instituciones o Responsables Académicas.
- ❖ **TURNO DE EXAMEN:** es un período dentro del Año Académico dentro del cual se definen uno o varios Llamados dentro de cada cual se definen una o varias Mesas de Examen.
- ❖ **UBICACIÓN:** el concepto Ubicación representa en el modelo de dominio tanto al concepto de Sede como a los de Campus y Campus Virtual. Cada Institución podrá aplicar el concepto Ubicación según su realidad y necesidad. Se denomina Sede a un lugar geográfico.
- Se denomina Campus a un conjunto de edificios y terrenos. Este concepto se podría extender a la educación a distancia, definiendo "campus virtuales". Un campus físico está asociado a una única sede, mientras que un campus virtual podría estar asociado a más de una. En una sede pueden existir uno o más campus.

7.3.1.2 Determinación de Equivalencias y Mapeo de Datos

Antes de iniciar con la tarea de mapeo de datos es importante aclarar los siguientes conceptos relacionados con las tablas de SIU-Guarani3:

- Las tablas que comienzan con las iniciales **mdp** se traducen a información relacionada con datos personales.
- Las tablas que comienzan con las iniciales **sga** se traducen a información relacionada con la gestión académica.
- Las tablas que comienzan con las iniciales **mug** se traducen a información relacionada con ubicación geográfica.

Cabe mencionar que para cada representación gráfica se incluirán los atributos a migrar que pueden ser claves primarias o atributos que son propios de la tabla correspondiente, no así aquellos atributos que son claves foráneas, ya que los mismos estarán incluidos en la representación gráfica de las tablas donde sean claves primarias.

La correspondencia entre atributos y tablas se expone de la siguiente manera:

1. Mapeo de atributos de la Tabla Alumnos con sus equivalentes en SIU-GUARANI:

Tabla Origen: ALUMNO	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
NRO_DOC_ALU	char (10)	nro_documento (varchar (15))	mdp_personas_documentos
TIPO_DOC_ALU	char (3)	desc_abreviada (varchar (3))	mdp_tipo_documento
APELLIDO_ALU	char (30)	apellido (varchar (60))	mdp_personas
NOMBRE_ALU	char (30)	nombres (varchar (60))	mdp_personas
BARRIO_ALU	char (40)	periodo_lectivo_barrio (varchar (60))	mdp_datos_personales
CALLE_ALU	char (100)	periodo_lectivo_calle (varchar (50))	mdp_datos_personales
CED_IDEN_ALU	char (15)	nro_documento (varchar (15))	mdp_personas_documentos
COD_AREA_TE_PART	char (10)	telefono_codigo_area (varchar (6))	mdp_personas_contactos
CODIGO_ALU	char (10)	periodo_lectivo_codigo_postal (varchar (15))	mdp_datos_personales
CORREO_EL_ALU	char (50)	email (varchar (100))	mdp_personas_contactos
MAIL_INSTITUCIONAL	char (70)	email (varchar (100))	mdp_personas_contactos
DPTO_ALU	char (10)	periodo_lectivo_departamento (varchar (5))	mdp_datos_personales
ESTADO_CIVIL	varchar (20)	descripcion (varchar (40))	mdp_estados_civiles
FNAC_ALU	datetime	fecha_nacimiento (date (13))	mdp_personas
NACION_ALU	char (25)	descripcion (varchar (100))	mdp_nacionalidades
NRO_ALU	char (5)	periodo_lectivo_numero (varchar (10))	mdp_datos_personales
PISO_ALU	char (3)	periodo_lectivo_piso (varchar (3))	mdp_datos_personales
SEXO_ALU	char (1)	descripcion (varchar (20))	mdp_personasexo
TE_PART_ALU	char (30)	telefono_numero (varchar (20))	mdp_personas_contactos
D_LOC_ALU	char (40)	nombre (varchar (100))	mug_localidades

El diagrama de la **Figura 1** representa las tablas involucradas en la migración de los datos de la tabla ALUMNO a la estructura de SIU-Guarani3:

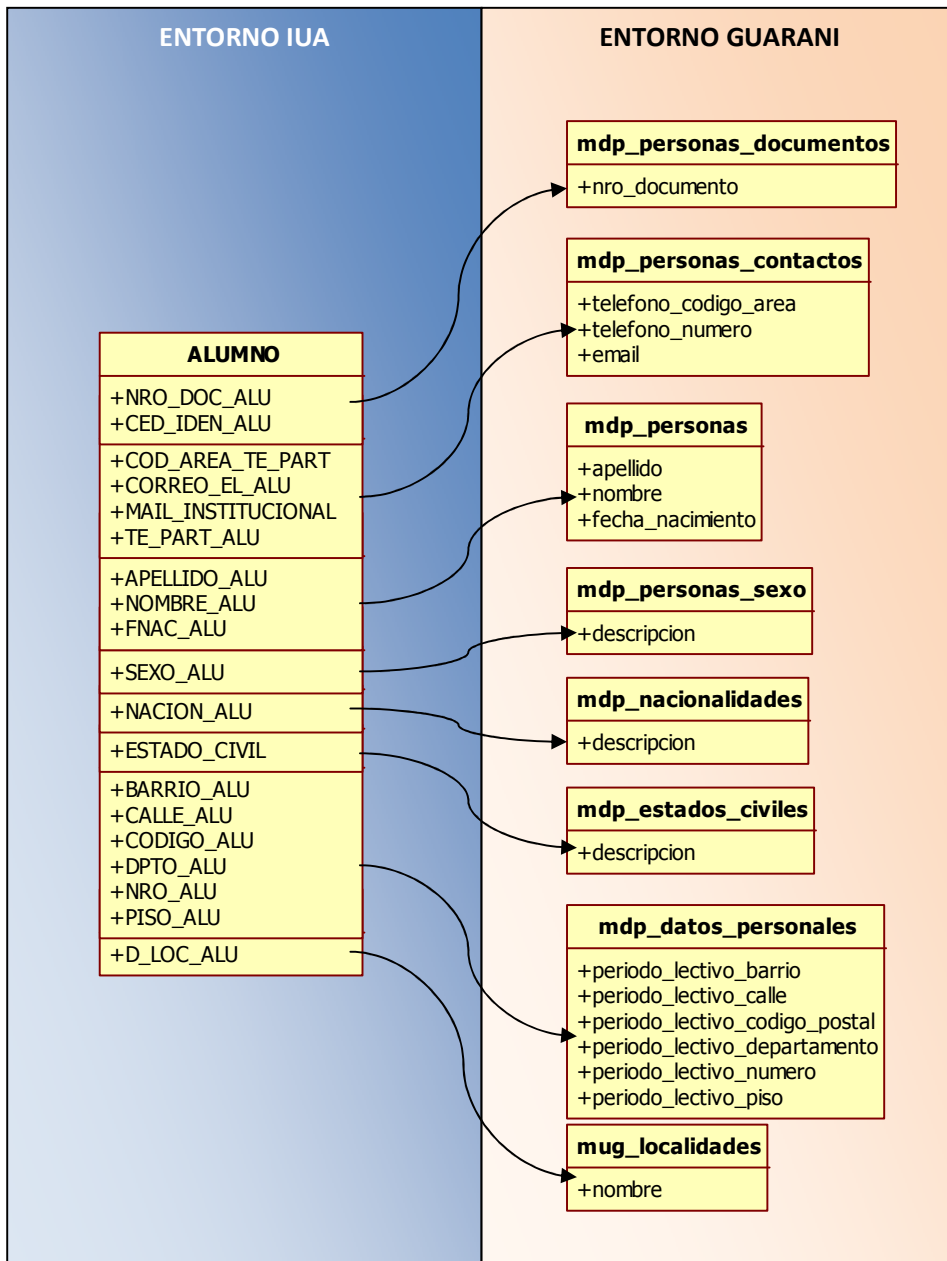


Figura 1

2. Mapeo de atributos de la Tabla DATOS_TRABAJO con sus equivalentes en SIU-GUARANI:

Tabla Origen: DATOS_TRABAJO	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
NRO_DOC_ALU	char (10)	nro_documento (varchar (15))	mdp_personas_documentos
TIPO_DOC_ALU	char (3)	desc_abreviada (varchar (3))	mdp_tipo_documento
CARGO	char (40)	cargo (varchar (255))	mdp_antecedentes_profesionales

EMPRESA	char (40)	empresa (varchar (255))	mdp_antecedentes_profesionales
RUBRO	char (40)	area (varchar (255))	mdp_antecedentes_profesionales

El diagrama de la **Figura 2** representa las tablas involucradas en la migración de los datos de la tabla DATOS_TRABAJO a la estructura de SIU-Guarani:

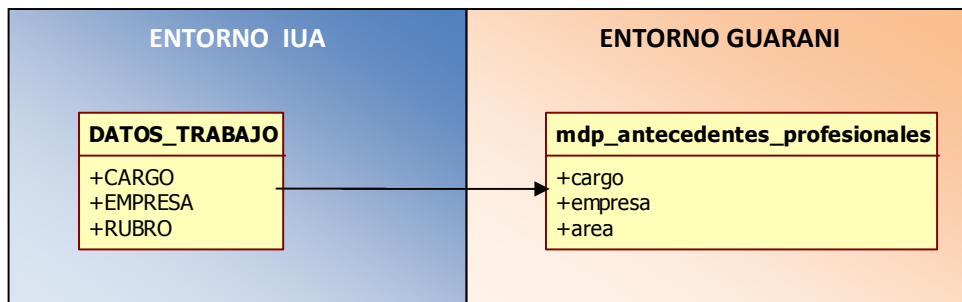


Figura 2

3. Mapeo de atributos de la Tabla ESTUDIOS_NIVEL_MEDIO con sus equivalentes en SIU-GUARANI:

Tabla Origen ESTUDIOS_NIVEL_MEDIO	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
NRO_DOC_ALU	char (10)	nro_documento (varchar (15))	mdp_personas_documentos
TIPO_DOC_ALU	char (3)	desc_abreviada (varchar (3))	mdp_tipo_documento
TITULO_NIVEL_MEDIO	varchar (60)	nombre (varchar (60))	mdp_titulos
ANIO_EGRESO	integer	anio_egreso (int2 (5))	mdp_datos_estudios
ESPECIALIZACION	varchar (250)	nombre (varchar (255))	mdp_orientacion_recibida
ESTUDIO_COMPLETO	char (1)	estado (bpchar (1))	mdp_datos_estudios
INSTITUCION_NIVEL_MEDIO	varchar (100)	nombre (varchar (100))	sga_colegios_secundarios
FECHA_ESTUDIO_COMPLETO	datetime	fecha_egreso (date (13))	mdp_datos_estudios

El diagrama de la **Figura 3** representa las tablas involucradas en la migración de los datos de la tabla ESTUDIOS_NIVEL_MEDIO a la estructura de SIU-Guarani:

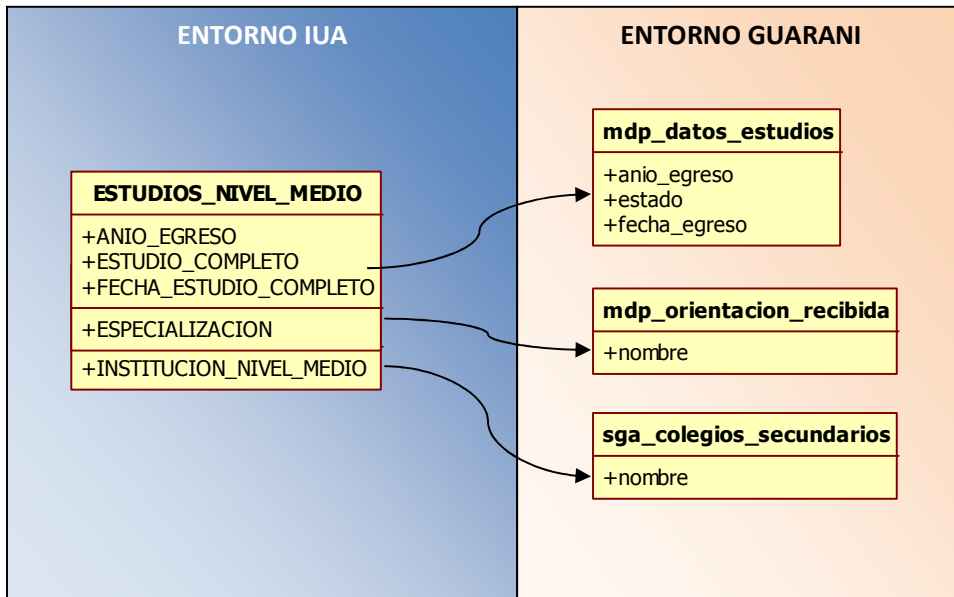


Figura 3

4. Mapeo de atributo de la Tabla TITULO_NIVEL_MEDIO con su equivalente en SIU-GUARANI:

Tabla Origen TITULO_NIVEL_MEDIO	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
TITULO_NIVEL_MEDIO	varchar (60)	nombre (varchar (60))	mdp_titulos

El diagrama de la **Figura 4** representa las tablas involucradas en la migración de los datos de la tabla TITULO_NIVEL_MEDIO a la estructura de SIU-Guarani:

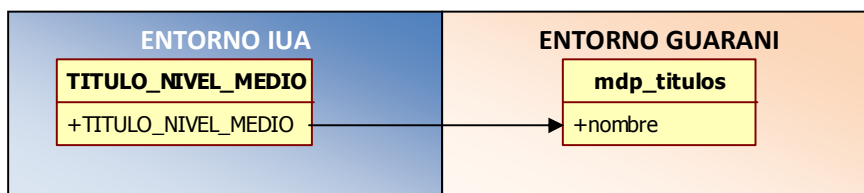


Figura 4

5. Mapeo de atributos de la Tabla TIPO_DOCUMENTO con sus equivalentes en SIU-GUARANI:

Tabla Origen TIPO_DOCUMENTO	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
TIPO_DOC_ALU	char (3)	desc_abreviada (varchar (3))	mdp_tipo_documento

DESCRIPCION_TIPO_DOCUMENTO	varchar (40)	descripcion (varchar (50))	mdp_tipo_documento
----------------------------	--------------	----------------------------	--------------------

El diagrama de la **Figura 5** representa las tablas involucradas en la migración de los datos de la tabla TIPO_DOCUMENTO a la estructura de SIU-Guarani:

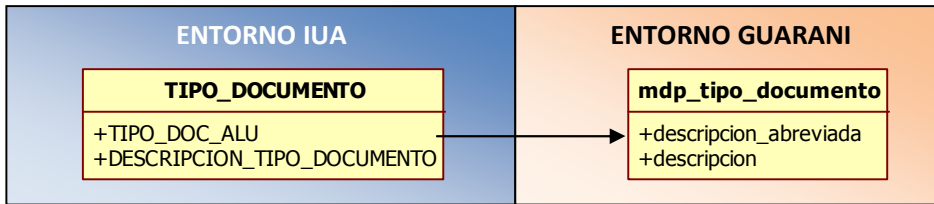


Figura 5

6. Mapeo de atributos de la Tabla TRAMITES_ALUMNO con sus equivalentes en SIU-GUARANI:

Tabla Origen TRAMITES_ALUMNO	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
TRAMITE	char (60)	nombre (varchar (50))	sga_acciones
DETALLE	varchar (120)	descripcion (varchar (255))	sga_acciones
FECHA_TRAMITE	datetime	fecha_presentacion (date (13))	sga_requisitos_presentados
LEGAJO	numeric (18,0)	legajo (varchar (20))	sga_alumnos

El diagrama de la **Figura 6** representa las tablas involucradas en la migración de los datos de la tabla TRAMITES_ALUMNO a la estructura de SIU-Guarani:

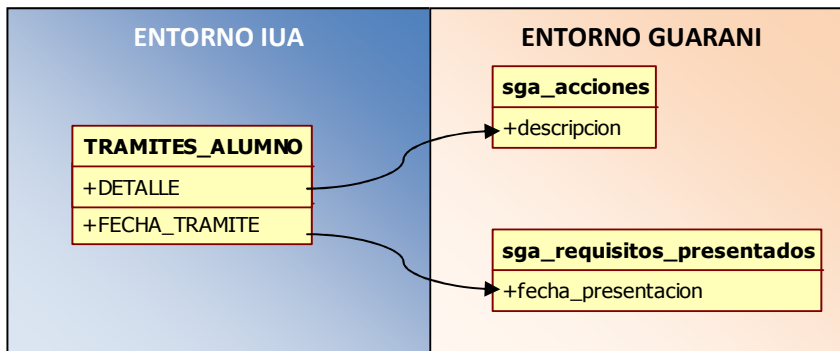


Figura 6

7. Mapeo de atributos de la Tabla TIPO_TRAMITE con sus equivalentes en SIU-GUARANI:

Tabla Origen TIPO_TRAMITE	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
ID_TIPO_TRAMITE	numeric (18,0)	accion (serial (10))	sga_acciones

TRAMITE	char (60)	nombre (varchar (50))	sga_acciones
---------	-----------	-----------------------	--------------

El diagrama de la **Figura 7** representa las tablas involucradas en la migración de los datos de la tabla TIPO_TRAMITE a la estructura de SIU-Guarani:

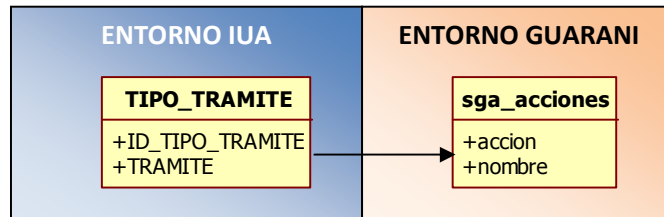


Figura 7

8. Mapeo de atributos de la Tabla FICHA_INSCRIPCION con sus equivalentes en SIU-GUARANI:

Tabla Origen FICHA_INSCRIPCION	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
LEGAJO	numeric (18,0)	legajo (varchar (20))	sga_alumnos
NRO_DOC_ALU	char (10)	nro_documento (varchar (15))	mdp_personas_documentos
COD_CAR	char (3)	codigo (varchar(10))	sga_propuestas
PLAN_CAR	char (4)	nombre (varchar(100))	sga_planes
DEPENDENCIA	varchar (60)	nombre (varchar (200))	sga_responsables_academicas
ANIO_ACADEMICO	integer	anio_academico (numeric (131089))	sga_anios_academicos
ESTADO_ALUMNO	char (1)	estado (bpchar (1))	sga_inscripciones_estados
FACULTAD	varchar (60)	nombre (varchar (255))	sga_instituciones
FECHA	datetime	fecha_inscripcion (date (13))	sga_propuestas_aspira
FECHA_INGRESO	datetime	fecha_alta (timestampz (35,6))	sga_propuestas_aspira
FECHA_REINSCRIPCION	datetime	fecha_reinscripcion (timestampz (35,6))	sga_reinscripciones
MODALIDAD	char (1)	modalidad (bpchar (1))	sga_modalidad_cursada
SEDE	char (40)	nombre (varchar(100))	sga_ubicaciones

*En la tabla indicada el atributo FACULTAD es el equivalente del atributo UNIDAD_ACADEMICA representado en la figura 12.

El diagrama de la **Figura 8** representa las tablas involucradas en la migración de los datos de la tabla FICHA_INSCRIPCION a la estructura de SIU-Guarani:

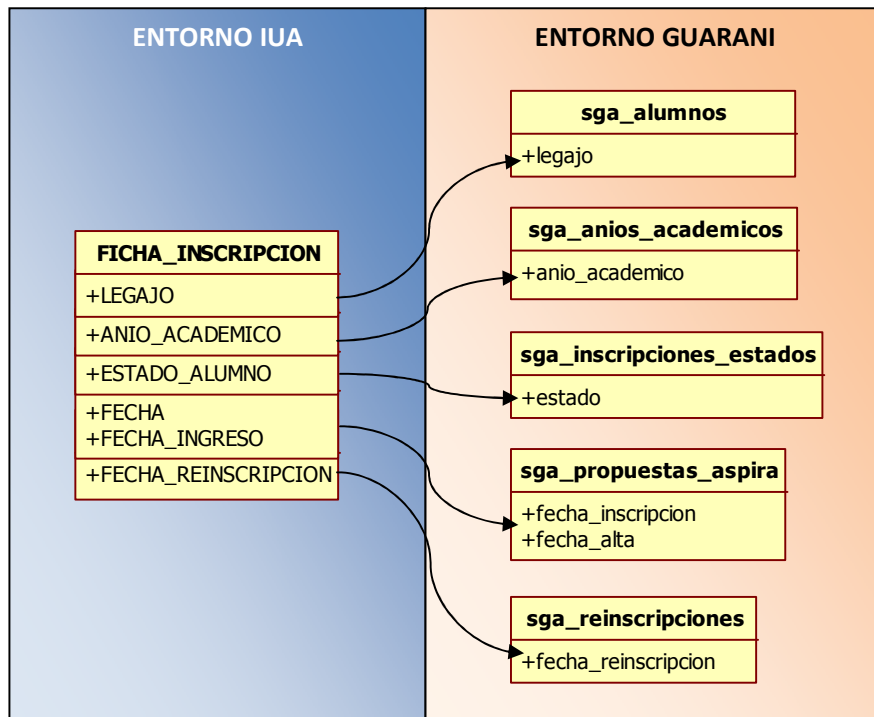


Figura 8

9. Mapeo de atributos de la Tabla SEDE con sus equivalentes en SIU-GUARANI:

Tabla Origen: SEDE	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
SEDE	char (40)	nombre (varchar (100))	sga_ubicaciones
CODIGO_SEDE	char (3)	ubicación (serial (10))	sga_ubicaciones

El diagrama de la **Figura 9** representa las tablas involucradas en la migración de los datos de la tabla SEDE a la estructura de SIU-Guarani:

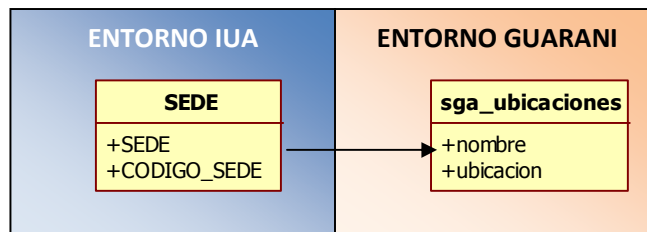


Figura 9

10. Mapeo de atributo de la Tabla *DEPENDENCIA* con su equivalente en SIU-GUARANI:

Tabla Origen: <i>DEPENDENCIA</i>	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
<i>DEPENDENCIA</i>	varchar (60)	nombre (varchar (200))	sga_responsables_academicas
<i>TIPO_DEPENDENCIA</i>	char (1)	responsable_academica_tipo (int4(10))	sga_responsables_academicas

El diagrama de la **Figura 10** representa las tablas involucradas en la migración de los datos de la tabla *DEPENDENCIA* a la estructura de SIU-Guarani:

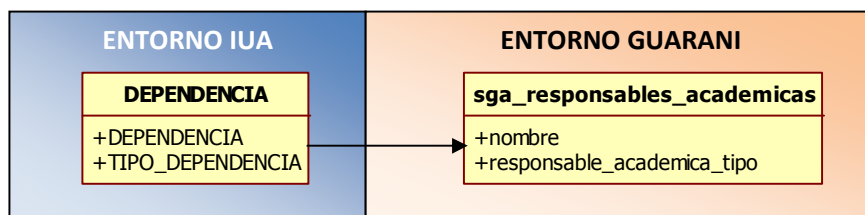


Figura 10

11. Mapeo de atributos de la Tabla *MODALIDAD* con sus equivalentes en SIU-GUARANI:

Tabla Origen <i>MODALIDAD</i>	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
<i>DESCRIPCION</i>	varchar (30)	nombre (varchar (50))	sga_modalidad_cursada
<i>MODALIDAD</i>	char (1)	modalidad (bpchar (1))	sga_modalidad_cursada
<i>UNIDAD_ACADEMICA</i>	varchar (60)	nombre (varchar (255))	sga_instituciones

El diagrama de la **Figura 11** representa las tablas involucradas en la migración de los datos de la tabla *MODALIDAD* a la estructura de SIU-Guarani:

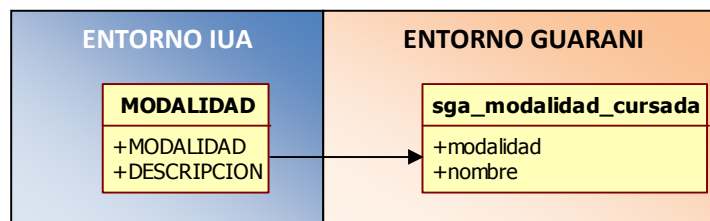


Figura 11

12. Mapeo de atributo de la Tabla *UNIDAD_ACADEMICA* con su equivalente en SIU-GUARANI:

Tabla Origen UNIDAD_ACADEMICA	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
UNIDAD_ACADEMICA	varchar (60)	nombre (varchar (255))	sga_instituciones

El diagrama de la **Figura 12** representa las tablas involucradas en la migración de los datos de la tabla UNIDAD_ACADEMICA a la estructura de SIU-Guarani:

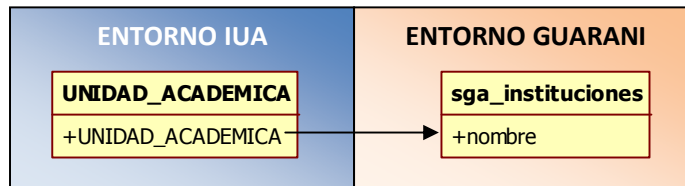


Figura 12

13. Mapeo de atributos de la Tabla CARRERAS con sus equivalentes en SIU-GUARANI:

Tabla Origen CARRERAS	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
COD_CAR	char (3)	codigo (varchar(10))	sga_propuestas
PLAN_CAR	char (4)	nombre (varchar(100))	sga_planes
CANTIDAD_ELECTIVAS	integer	cnt_optativas (int2 (5))	sga_planes_versiones
CANTIDAD_MATERIAS_GRADO	integer	cnt_materias (int2(5))	sga_planes_versiones
CANTIDAD_MATERIAS_PREGRADO	integer		
CARRERA_CURSO	varchar (20)	descripcion (varchar (100))	sga_propuestas_tipos
CUATRIMESTRES_GRADO	integer	duracion_en_anios (int4(10)) duracion_en_meses (int4(10))	sga_planes
CUATRIMESTRES_PREGRADO	integer	duracion_en_anios (int4(10)) duracion_en_meses (int4(10))	sga_planes
ESTADO	char (1)	estado (bpchar (1))	sga_propuestas_estados
NOM_CAR	char (50)	nombre (varchar (255))	sga_propuestas
VIGENCIA_MATRICULA	char (1)	permite_matricular (bpchar (1))	sga_propuestas_tipos

El diagrama de la **Figura 13** representa las tablas involucradas en la migración de los datos de la tabla CARRERAS a la estructura de SIU-Guarani:

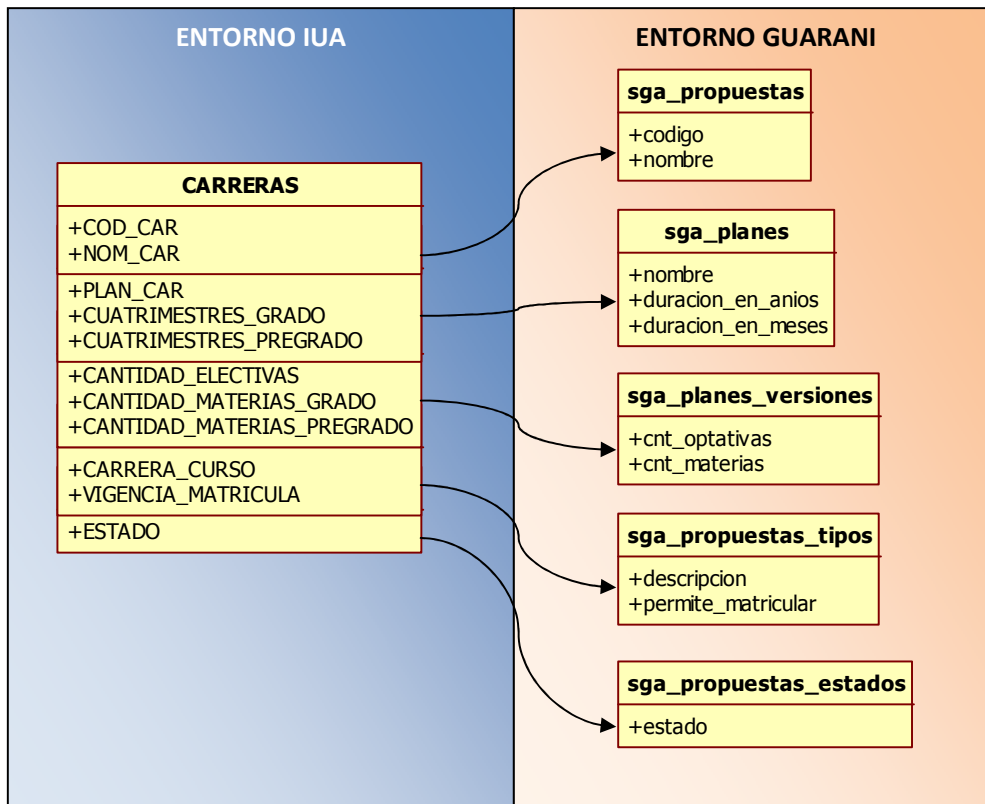


Figura 13

14. Mapeo de atributo de la Tabla UBICACIÓN_FISICA con su equivalente en SIU-GUARANI:

Tabla Origen: UBICACION_FISICA	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
UBICACIÓN	varchar (60)	nombre (varchar (100))	sga_edificaciones

El diagrama de la **Figura 14** representa las tablas involucradas en la migración de los datos de la tabla UBICACIÓN_FISICA a la estructura de SIU-Guarani:

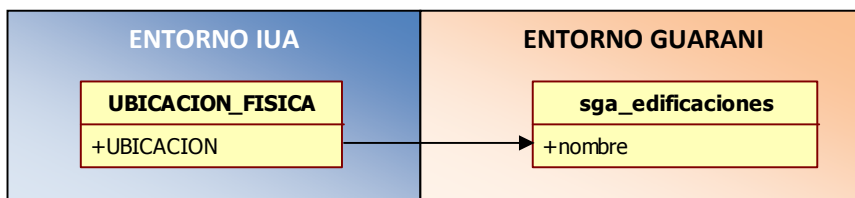


Figura 14

15. Mapeo de atributos de la Tabla ASIGNA_HORARIO_MATERIA con sus equivalentes en SIU-GUARANI:

Tabla Origen: ASIGNA_HORARIO_MATERIA	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
id_AHM	numeric (18,0)	periodo_lectivo (serial (10))	sga_periodos_lectivos
DIVISION	char (6)	nombre (varchar (100))	sga_comisiones
CAPACIDAD_CURSO	integer	cupo (int2(5))	sga_comisiones
LEGAJO_PERSONAL	numeric (18,0)	legajo (varchar (15))	sga_docentes
CODIGO_DE_MATERIA	char (5)	código (varchar(10))	sga_elementos
NUMERO_AULA	integer	nombre (varchar (60))	sga_espacios
FECHA_COMIENZO_GRUPO	datetime	fecha_inicio_dictado (date (13))	sga_periodos_lectivos
FECHA_FIN_GRUPO	datetime	fecha_fin_dictado date (13))	sga_periodos_lectivos
PLAN_CAR	char (4)	nombre (varchar(100))	sga_planes
COD_CAR	char (3)	codigo (varchar(10))	sga_propuestas

El diagrama de la **Figura 15** representa las tablas involucradas en la migración de los datos de la tabla ASIGNA_HORARIO_MATERIA a la estructura de SIU-Guarani:

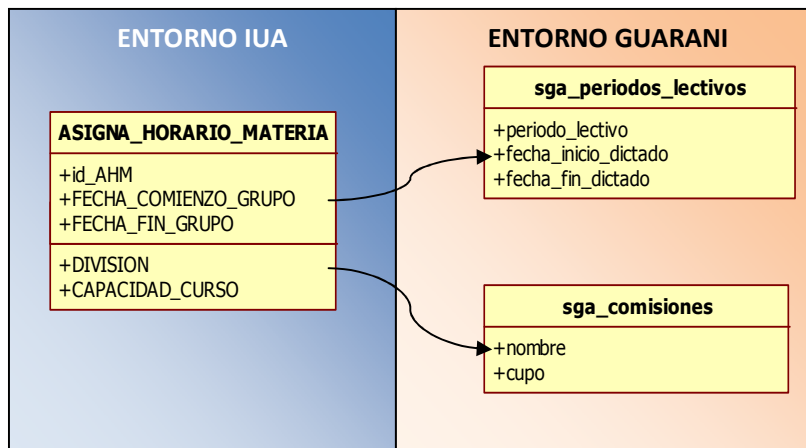


Figura 15

16. Mapeo de atributos de la Tabla DIA_HORA con sus equivalentes en SIU-GUARANI:

Tabla Origen: DIA_HORA	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
ID_MODULO	numeric (18,0)	prestamo_tipo (serial (10))	sga_prestamos_tipos
DIVISION	char (6)	nombre (varchar (100))	sga_comisiones
id_AHM	numeric (18,0)	periodo_lectivo (serial (10))	sga_periodos_lectivos
CODIGO_DE_MATERIA	char (5)	codigo (varchar(10))	sga_elementos
NUMERO_AULA	integer	nombre (varchar (60))	sga_espacios
PLAN_CAR	char (4)	nombre (varchar(100))	sga_planes

FECHA_MODULO	datetime	fecha (date (13))	sga_prestamos
HORA_FIN	datetime	hora_finalizacion (time(15,6))	sga_prestamos
HORA_INICIO	datetime	hora_inicio (time(15,6))	sga_prestamos
ACTIVIDAD_MODULO	char (50)	nombre (varchar (50))	sga_prestamos_tipos
COD_CAR	char (3)	codigo (varchar(10))	sga_propuestas

El diagrama de la **Figura 16** representa las tablas involucradas en la migración de los datos de la tabla DIA_HORA a la estructura de SIU-Guarani:

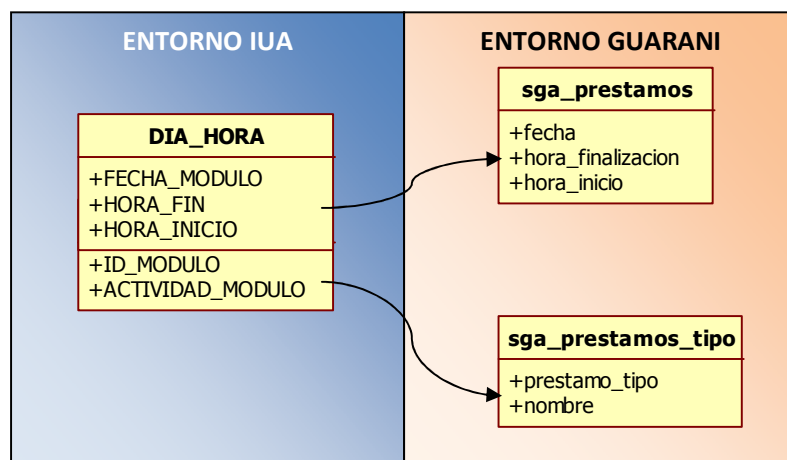


Figura 16

17. Mapeo de atributos de la Tabla TIPO_HORA con sus equivalentes en SIU-GUARANI:

Tabla Origen: TIPO_HORA	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
ID_TIPO_HORA	integer	tipo_clase (integer)	sga_clases_tipos
DESCRIPCION	varchar (255)	descripcion (varchar (255))	sga_clases_tipos
TIPO_HORA	varchar (50)	nombre (varchar (50))	sga_clases_tipos

El diagrama de la **Figura 17** representa las tablas involucradas en la migración de los datos de la tabla TIPO_HORA a la estructura de SIU-Guarani:

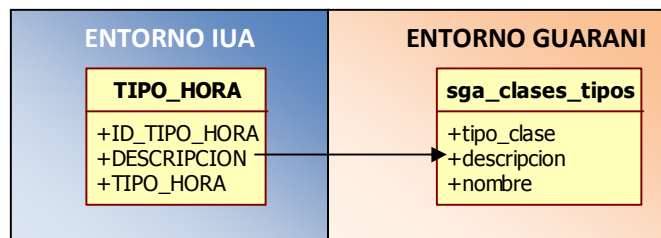


Figura 17

18. Mapeo de atributos de la Tabla INSCRIPCION_EN_MATERI con sus equivalentes en SIU-GUARANI:

Tabla Origen: INSCRIPCION_EN_MATERI	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
NUMERO_TRANSACCION	numeric (18,0)	inscripcion (integer)	sga_insc_cursada
LEGAJO	numeric (18,0)	legajo (varchar (20))	sga_alumnos
COD_CAR	char (3)	codigo (varchar(10))	sga_propuestas
PLAN_CAR	char (4)	nombre (varchar(100))	sga_planes
CODIGO_DE_MATERIA	char (5)	codigo (varchar(10))	sga_elementos
LEGAJO_PERSONAL	numeric (18,0)	legajo (varchar (15))	sga_docentes
FECHA_INSCRIP_EN_MATERIA	datetime	fecha_inscripcion (timestampz (35,6))	sga_insc_cursada
DIVISION_SEDE	(varchar (10))	comisión (integer)	sga_insc_cursada

El diagrama de la **Figura 18** representa las tablas involucradas en la migración de los datos de la tabla INSCRIPCION_EN_MATERI a la estructura de SIU-Guarani:

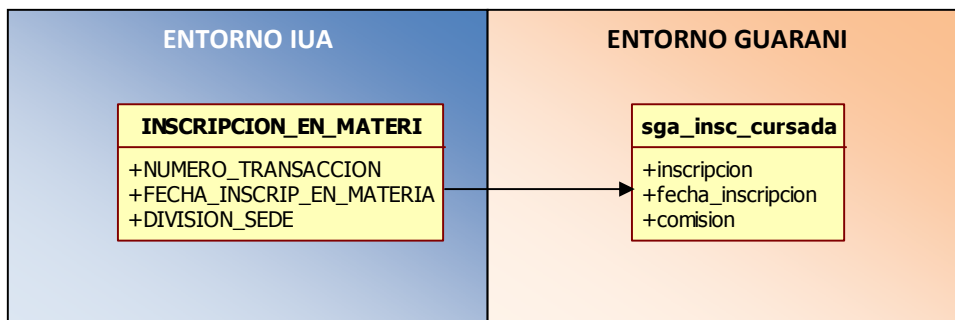


Figura 18

19. Mapeo de atributos de la Tabla MATERIA con sus equivalentes en SIU-GUARANI:

Tabla Origen MATERIAS	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
CODIGO_DE_MATERIA	char (5)	código (varchar(10))	sga_elementos
PLAN_CAR	char (4)	nombre (varchar(100))	sga_planes
COD_CAR	char (3)	codigo (varchar(10))	sga_propuestas
EN_ACTA	char (1)	estado (bpchar (1))	sga_actas_estados
NOMBRE_DE_MATERIA	char (50)	nombre (varchar (255))	sga_elementos
Cursar	char (1)	requiere_cursada (bpchar(1))	sga_elementos_atrib
rendir_libre	char (1)	permite_rendir_libre (bpchar(1))	sga_elementos_atrib
ID_TIPO_MATERIA	numeric (18,0)	entidad_subtipo (serial (10))	sga_g3entidades_subtipos
CUATRIMESTRE_DE_MATERIA	integer	periodo_generico (serial (10))	sga_periodos_genericos

El diagrama de la **Figura 19** representa las tablas involucradas en la migración de los datos de la tabla MATERIA a la estructura de SIU-Guarani:

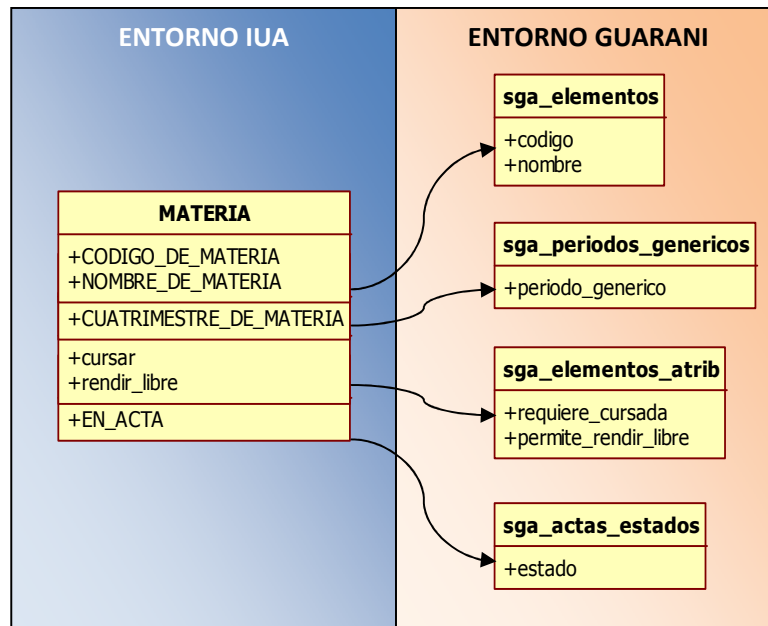


Figura 19

20. Mapeo de atributos de la Tabla TIPO_MATERIA con sus equivalentes en SIU-GUARANI:

Tabla Origen TIPO_MATERIA	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
ID_TIPO_MATERIA	numeric (18,0)	entidad_subtipo (serial (10))	sga_g3entidades_subtipos
NOMBRE_TIPO	varchar (50)	nombre (varchar (100))	sga_g3entidades_subtipos
PROMEDIA	char (1)	promediable (char (1))	sga_elementos_atrib

El diagrama de la **Figura 20** representa las tablas involucradas en la migración de los datos de la tabla TIPO_MATERIA a la estructura de SIU-Guarani:

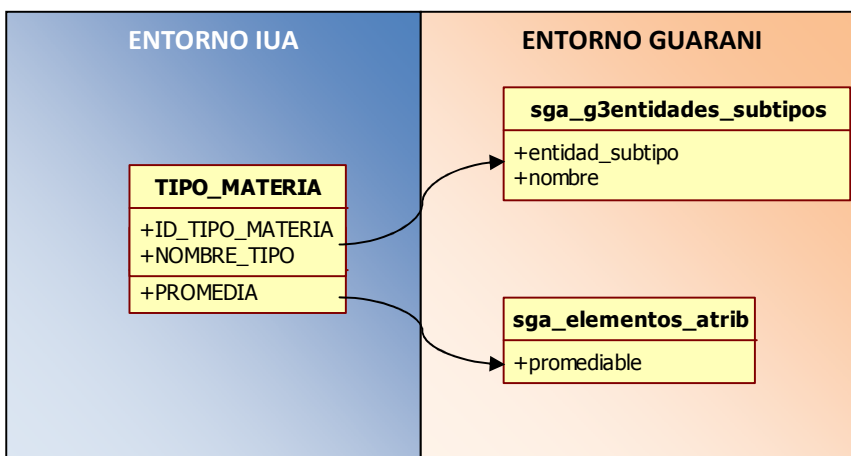


Figura 20

21. Mapeo de atributos de la Tabla *SISTEMA_CALIF* con sus equivalentes en SIU-GUARANI:

Tabla Origen: SISTEMA_CALIF	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
ID_SIST_CALIF	numeric (18,0)	escala_nota (serial (10))	sga_escalas_notas
VALOR_REAL	float (8)	valor_numerico (numeric (10,4))	sga_escalas_notas_det
APROBADO	varchar (50)	descripcion (varchar (50))	sga_escalas_notas_resultado
ID_TIPO_MATERIA	numeric (18,0)	entidad_subtipo (serial (10))	sga_g3entidades_subtipos

El diagrama de la **Figura 21** representa las tablas involucradas en la migración de los datos de la tabla TIPO_MATERIA a la estructura de SIU-Guarani:

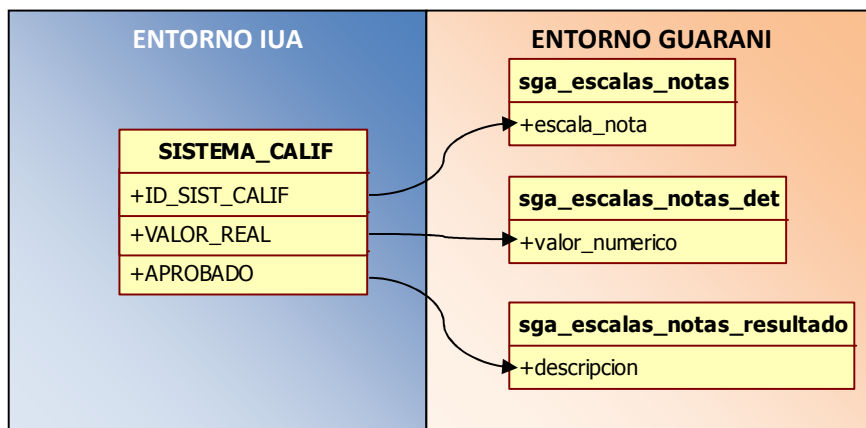


Figura 21

22. Mapeo de atributos de la Tabla *EVALUACION* con sus equivalentes en SIU-GUARANI:

Tabla Origen: EVALUACION	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
LEGAJO	numeric (18,0)	legajo (varchar (20))	sga_alumnos
COD_CAR	char (4)	nombre (varchar(10))	sga_planes
PLAN_CAR	char (3)	codigo (varchar(100))	sga_propuestas
CODIGO_DE_MATERIA	char (5)	codigo (varchar(10))	sga_elementos
CALIFICACION_EVALUACION	float (8)	nota_cursada (varchar(10))	sga_eval_detalle_cursadas
FECHA_GRABACION	datetime	fecha_regular (date (13))	sga_eval_detalle_cursadas
TIPO_DE_ACTIVIDAD	char (12)	nombre (varchar(100))	sga_evaluaciones_tipos

El diagrama de la **Figura 22** representa las tablas involucradas en la migración de los datos de la tabla EVALUACION a la estructura de SIU-Guarani:

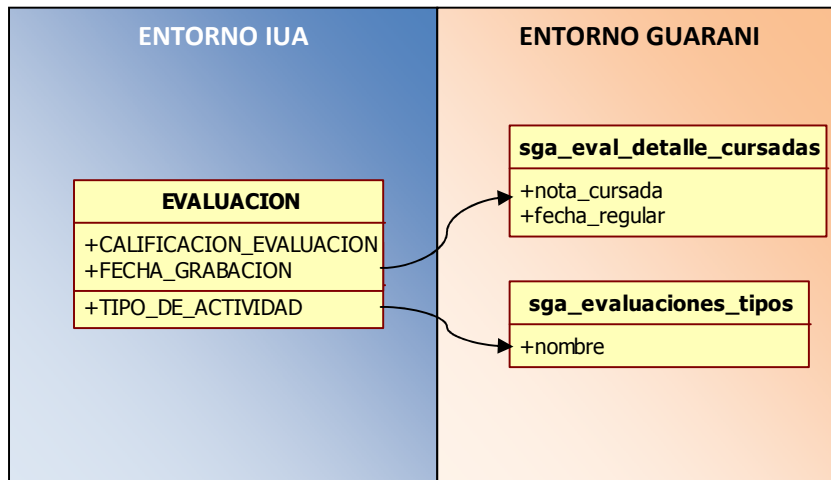


Figura 22

23. Mapeo de atributos de la Tabla AULA con sus equivalentes en SIU-GUARANI:

Tabla Origen: AULA	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
NUMERO_AULA	integer	nombre (varchar (60))	sga_espacios
UBICACIÓN	varchar (60)	nombre (varchar (100))	sga_edificaciones
CAPACIDAD	integer	capacidad (int2(5))	sga_espacios

El diagrama de la **Figura 23** representa las tablas involucradas en la migración de los datos de la tabla AULA a la estructura de SIU-Guarani:

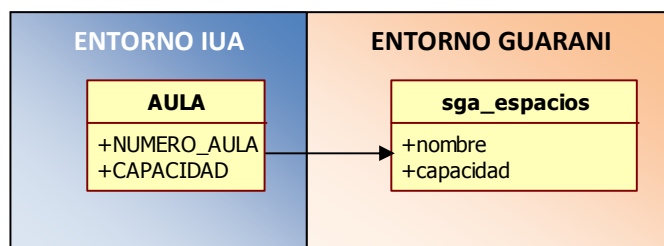


Figura 23

24. Mapeo de atributos de la Tabla MESA_EXAMEN con sus equivalentes en SIU-GUARANI:

Tabla Origen MESA_EXAMEN	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
CODIGO_DE_MATERIA	char (5)	codigo (varchar(10))	sga_elementos
PLAN_CAR	char (4)	nombre (varchar(100))	sga_planes

COD_CAR	char (3)	codigo (varchar(10))	sga_propuestas
FECHA_EXAMEN_FINAL	datetime	fecha (date(13))	sga_llamados_mesa
ID_MESA	numeric (18,0)	mesa_examen (serial (10))	sga_mesas_examen

El diagrama de la **Figura 24** representa las tablas involucradas en la migración de los datos de la tabla MESA_EXAMEN a la estructura de SIU-Guarani:

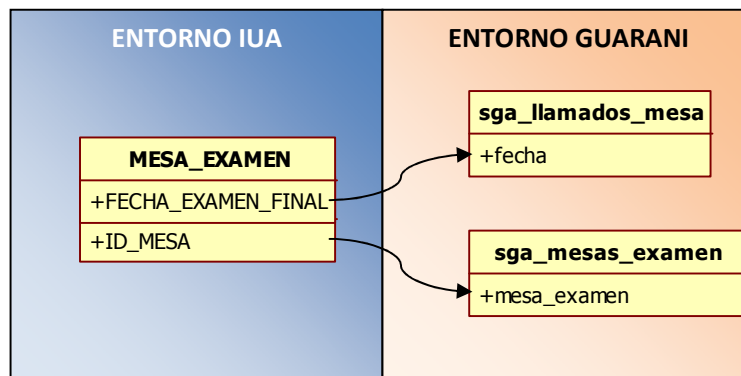


Figura 24

25. Mapeo de atributos de la Tabla INSCRIPCION_EN_EXAMEN con sus equivalentes en SIU-GUARANI:

Tabla Origen	INSCRIPCION_EN_EXAMEN	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
PLAN_CAR		char (4)	nombre (varchar(100))	sga_planes
COD_CAR		char (3)	codigo (varchar(10))	sga_propuestas
CODIGO_DE_MATERIA		char (5)	codigo (varchar(10))	sga_elementos
LEGAJO		numeric (18,0)	legajo (varchar (20))	sga_alumnos
NUMERO_ACTA		integer	nro_acta (varchar (30))	sga_actas
FOLIO		integer	folio (smallint)	sga_actas_folios
FECHA_EXAMEN		datetime	fecha (date (13))	sga_eval_detalle_examen
TIPO_EXAMEN		char (15)	nombre (varchar (100))	sga_evaluaciones_tipos
CALIFICACION_EXAMEN		float (8)	nota (varchar (10))	sga_eval_detalle_examen
LIBRO		integer	nro_libro (varchar (30))	sga_libros_actas

El diagrama de la **Figura 25** representa las tablas involucradas en la migración de los datos de la tabla INSCRIPCION_EN_EXAMEN a la estructura de SIU-Guarani:

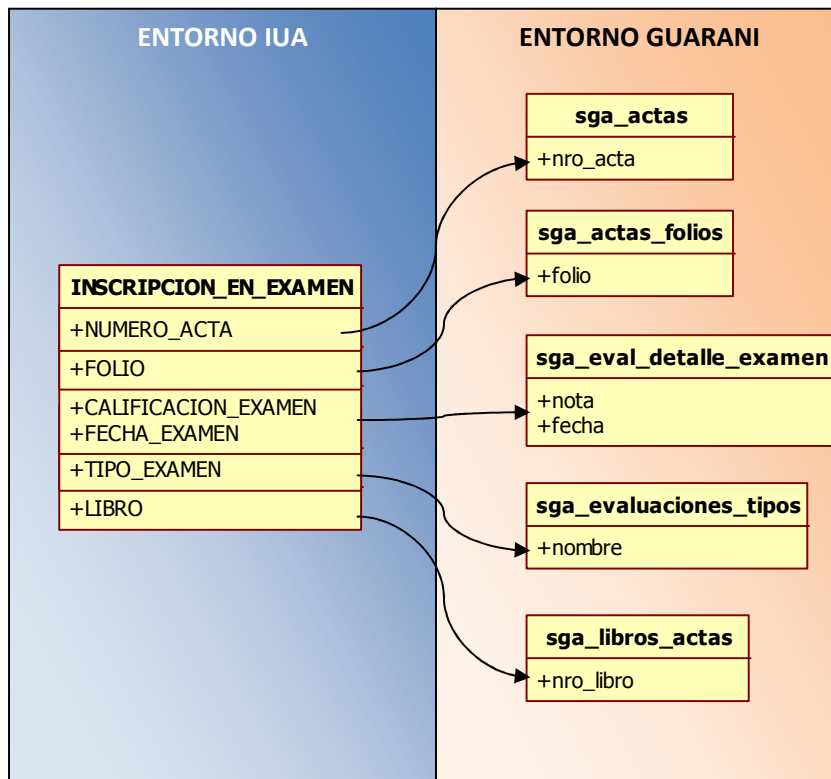


Figura 25

26. Mapeo de atributos de la Tabla DOCENTE con sus equivalentes en SIU-GUARANI:

Tabla Origen DOCENTE	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
LEGAJO_PERSONAL	numeric (18,0)	legajo (varchar (15))	sga_docentes
BARRIO	char (25)	periodo_lectivo_barrio (varchar (60))	mdp_datos_personales
CODIGO_POSTAL	char (18)	periodo_lectivo_codigo_postal (varchar (15))	mdp_datos_personales
DOMICILIO_PARTICULAR	char (30)	periodo_lectivo_calle (varchar (50))	mdp_datos_personales
DPTO	char (5)	periodo_lectivo_departamento (varchar (5))	mdp_datos_personales
NUMERO_DOM	integer	periodo_lectivo_numero (varchar (10))	mdp_datos_personales
PISO	char (3)	periodo_lectivo_piso (varchar (3))	mdp_datos_personales
SEXO_DOCENTE	char (1)	descripcion (varchar (20))	mdp_personasexo
APELLIDO_DOCENTE	char (25)	apellido (varchar (60))	mdp_personas
FECHA_DE_NACIMIENTO	datetime	fecha_nacimiento (date (13))	mdp_personas
NOMBRE_DDOCENTE	char (30)	nombres (varchar (60))	mdp_personas
COD_AREA_TE_PART	char (10)	telefono_codigo_area (varchar (6))	mdp_personas_contactos
MAIL_PARTICULAR	varchar (50)	email (varchar (100))	mdp_personas_contactos
TE_CELULAR	char (30)	telefono_numero (varchar (20))	mdp_personas_contactos
TELEFONO_	char (30)	telefono_numero (varchar (20))	mdp_personas_contactos

NUM_DOC_DOCENTE	char (10)	nro_documento (varchar (15))	mdp_personas_documentos
TIPO_DOC_DOCENTE	char (3)	desc_abreviada (varchar (3))	mdp_tipo_documento
CODIGO_TITULO	numeric (18,0)	titulo (serial (10))	mdp_titulos

El diagrama de la **Figura 26** representa las tablas involucradas en la migración de los datos de la tabla DOCENTE a la estructura de SIU-Guarani:

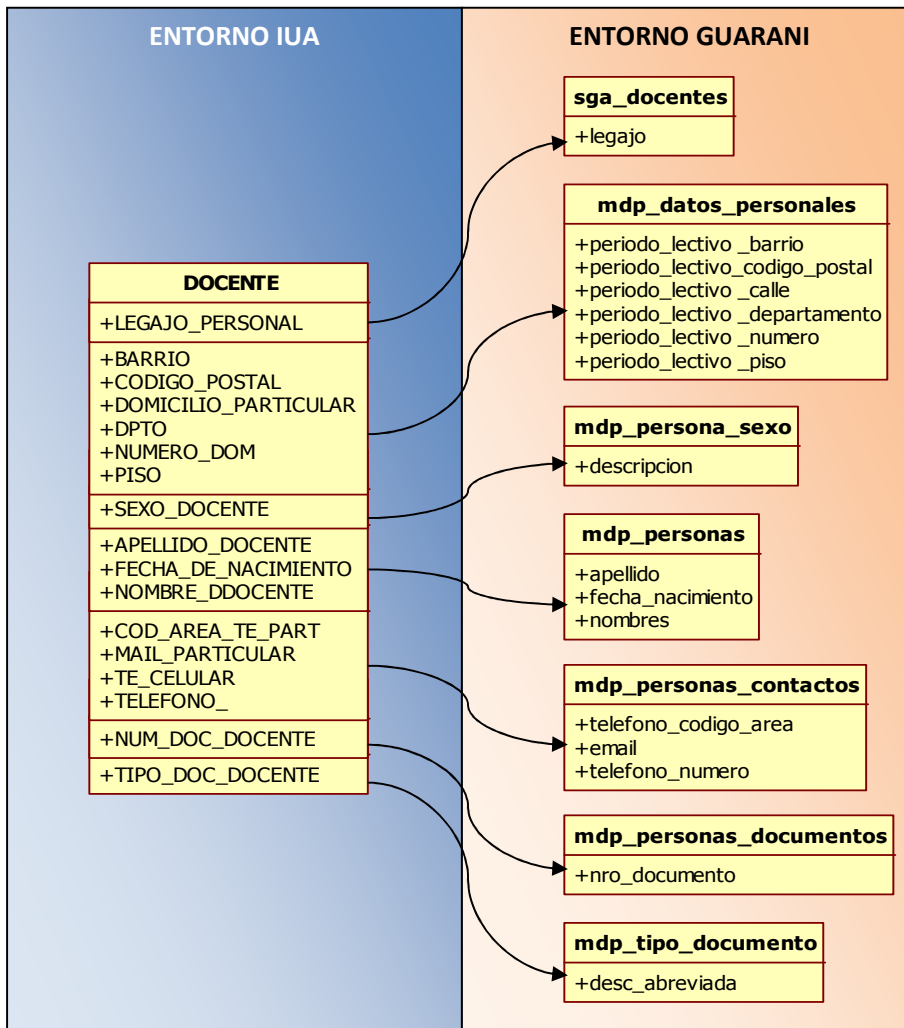


Figura 26

27. Mapeo de atributos de la Tabla *DOCENTE_FUNCION* con sus equivalentes en SIU-GUARANI:

Tabla Origen <i>DOCENTE_FUNCION</i>	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
LEGAJO_PERSONAL	numeric (18,0)	legajo (varchar (15))	sga_docentes
CARGO	varchar (60)	cargo (varchar (255))	mdp_antecedentes_docentes

DEPENDENCIA	varchar (60)	nombre (varchar (200))	sga_responsables_academicas
E_MAIL_IUA	varchar (50)	email (varchar (100))	mdp_personas_contactos
UNIDAD_ACADEMICA	varchar (60)	nombre (varchar (255))	sga_unidades_gestion

El diagrama de la **Figura 27** representa las tablas involucradas en la migración de los datos de la tabla **DOCENTE_FUNCION** a la estructura de SIU-Guarani:

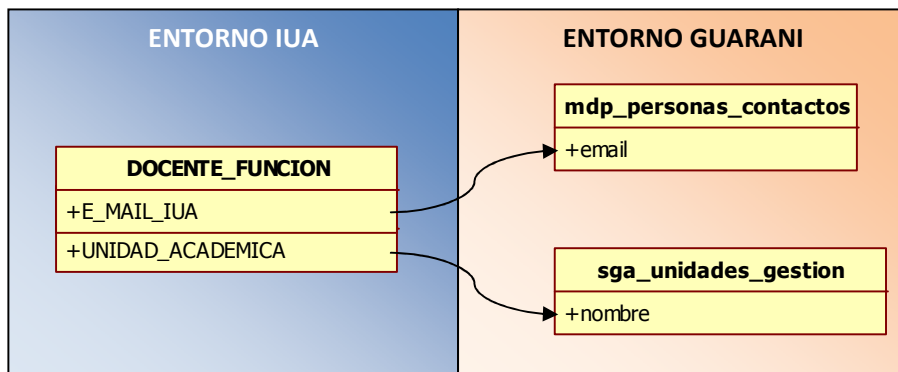


Figura 27

28. Mapeo de atributo de la Tabla **CARGO** con su equivalente en SIU-GUARANI:

Tabla Origen Cargo	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
CARGO	Varchar (60)	cargo (varchar (255))	mdp_antecedentes_docentes

El diagrama de la **Figura 28** representa las tablas involucradas en la migración de los datos de la tabla **CARGO** a la estructura de SIU-Guarani:

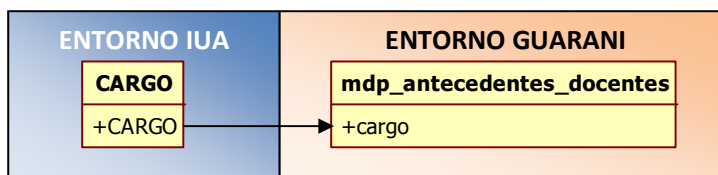


Figura 28

29. Mapeo de atributos de la Tabla **EGRESADO** con sus equivalentes en SIU-GUARANI:

Tabla Origen: EGRESADO	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
LEGAJO	numeric (18,0)	legajo (varchar (20))	sga_alumnos
id_nivel_titulo	integer	titulo_tipo (bpchar(1))	mdp_titulos_tipos
FECHA_INGRESO	datetime	fecha_ingreso (date(13))	mdp_datos_estudios

FECHA_APROBACION_TFI	datetime	fecha_egreso (date(13))	mdp_datos_estudios
----------------------	----------	-------------------------	--------------------

El diagrama de la **Figura 29** representa las tablas involucradas en la migración de los datos de la tabla EGRESADO a la estructura de SIU-Guarani:

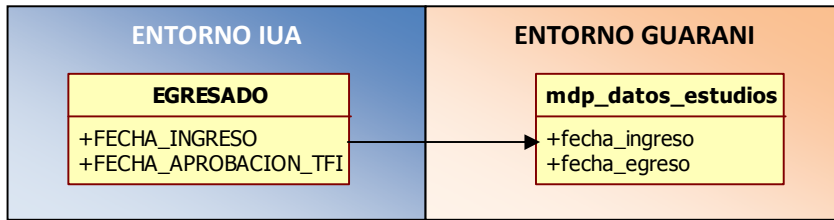


Figura 29

30. Mapeo de atributos de la Tabla TITULO con sus equivalentes en SIU-GUARANI:

Tabla Origen TITULO	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
CODIGO_TITULO	numeric (18,0)	titulo (serial (10))	mdp_titulos
TITULO	varchar (100)	nombre (varchar (255))	mdp_titulos

El diagrama de la **Figura 30** representa las tablas involucradas en la migración de los datos de la tabla TITULO a la estructura de SIU-Guarani:

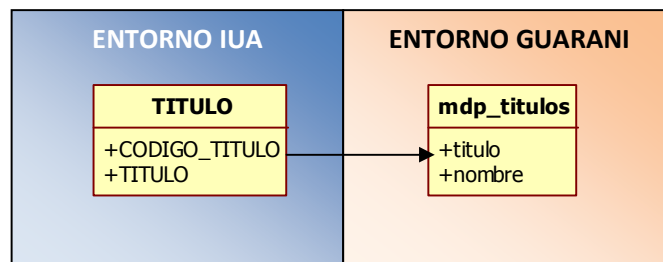


Figura 30

31. Mapeo de atributos de la Tabla nivel_titulacion con sus equivalentes en SIU-GUARANI:

Tabla Origen nivel_titulacion	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
id_nivel_titulo	integer	titulo_tipo (bpchar (1))	mdp_titulos_tipos
Descripcion	varchar (100)	nombre (varchar (255))	mdp_titulos_tipos

El diagrama de la **Figura 31** representa las tablas involucradas en la migración de los datos de la tabla nivel_titulacion a la estructura de SIU-Guarani:

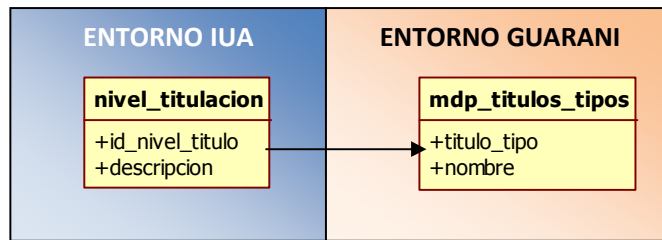


Figura 31

32. Mapeo de atributos de la Tabla titulacion con sus equivalentes en SIU-GUARANI:

Tabla Origen: titulacion	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
id_titulacion	integer	titulo (serial(10))	mdp_titulos
id_nivel_titulo	integer	titulo_tipo (bpchar(1))	mdp_titulos_tipos
nombre_oficial	varchar (255)	nombre (varchar (255))	mdp_titulos

El diagrama de la **Figura 32** representa las tablas involucradas en la migración de los datos de la tabla titulacion a la estructura de SIU-Guarani:

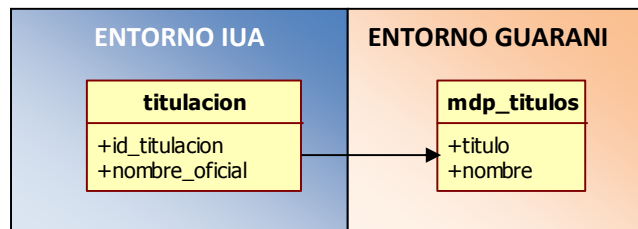


Figura 32

33. Mapeo de atributos de la Tabla PAIS con sus equivalentes en SIU-GUARANI:

Tabla Origen PAIS	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
COD_PAIS	integer	pais (int2(5))	mug_paises
PAIS	varchar (25)	nombre (varchar (60))	mug_paises

El diagrama de la **Figura 33** representa las tablas involucradas en la migración de los datos de la tabla PAIS a la estructura de SIU-Guarani:

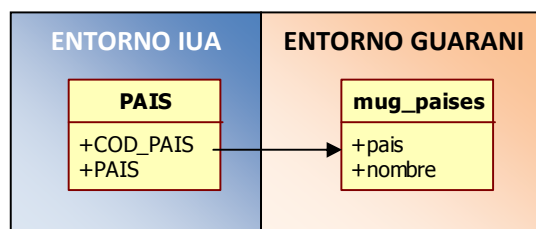


Figura 33

34. Mapeo de atributos de la Tabla *PROVINCIA* con sus equivalentes en SIU-GUARANI:

Tabla Origen <i>PROVINCIA</i>	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
COD_PROVINCIA	numeric (18,0)	provincia (int4(10))	mug_provincias
PROVINCIA	char (40)	nombre (varchar (60))	mug_provincias
COD_PAIS	integer	Pais (int2(5))	mug_paises

El diagrama de la **Figura 34** representa las tablas involucradas en la migración de los datos de la tabla *PROVINCIA* a la estructura de SIU-Guarani:

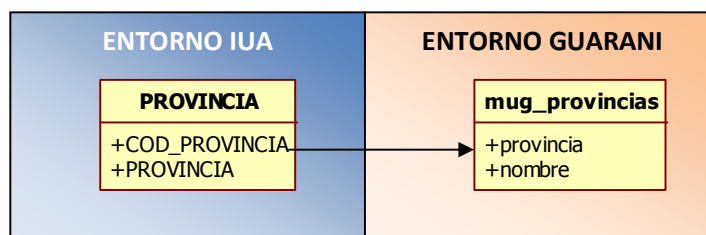


Figura 34

35. Mapeo de atributos de la Tabla *LOCALIDAD* con sus equivalentes en SIU-GUARANI:

Tabla Origen <i>LOCALIDAD</i>	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
COD_LOCALIDAD	numeric (18,0)	localidad (int4(10))	mug_localidades
LOCALIDAD	char (40)	nombre (varchar (100))	mug_localidades
COD_PROVINCIA	numeric (18,0)	provincia (int4(10))	mug_provincias

El diagrama de la **Figura 35** representa las tablas involucradas en la migración de los datos de la tabla *LOCALIDAD* a la estructura de SIU-Guarani:

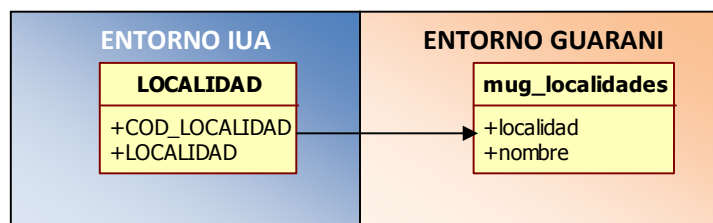


Figura 35

36. Mapeo de atributos de la Tabla *PERFIL* con sus equivalentes en SIU-GUARANI:

Tabla Origen: PERFIL	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
CODIGO_PERFIL	integer	tipo_usuario (varchar (20))	acc_tipos_usuarios
USUARIO	varchar (20)	usuario_grupo_acc (varchar (30))	acc_grupo_acc_x_tipo_usuario
SEDE	char (40)	nombre (varchar (100))	sga_ubicaciones

El diagrama de la **Figura 36** representa las tablas involucradas en la migración de los datos de la tabla *PERFIL* a la estructura de SIU-Guarani:

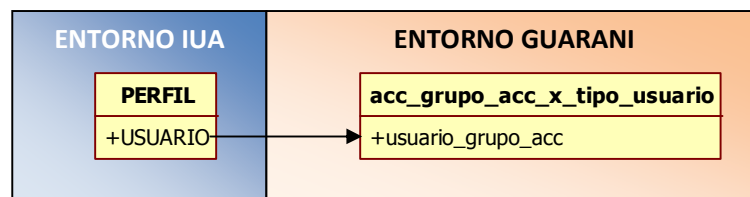


Figura 36

37. Mapeo de atributos de la Tabla *TIPO_PERFIL* con sus equivalentes en SIU-GUARANI:

Tabla Origen: TIPO_PERFIL	Data type	Atributo Destino en SIU-GUARANI	Tabla Destino en SIU-GUARANI
CODIGO_PERFIL	integer	tipo_usuario (varchar (20))	acc_tipos_usuarios
PERFIL	varchar (40)	nombre (bpchar (50))	acc_tipos_usuarios

El diagrama de la **Figura 37** representa las tablas involucradas en la migración de los datos de la tabla *TIPO_PERFIL* a la estructura de SIU-Guarani:

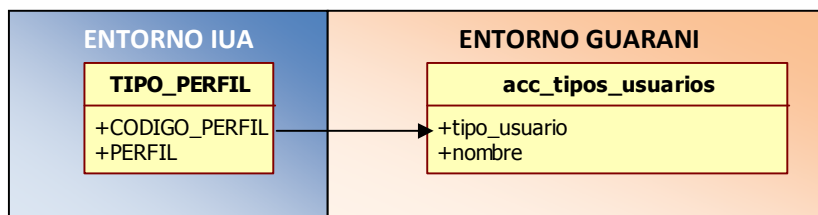


Figura 37

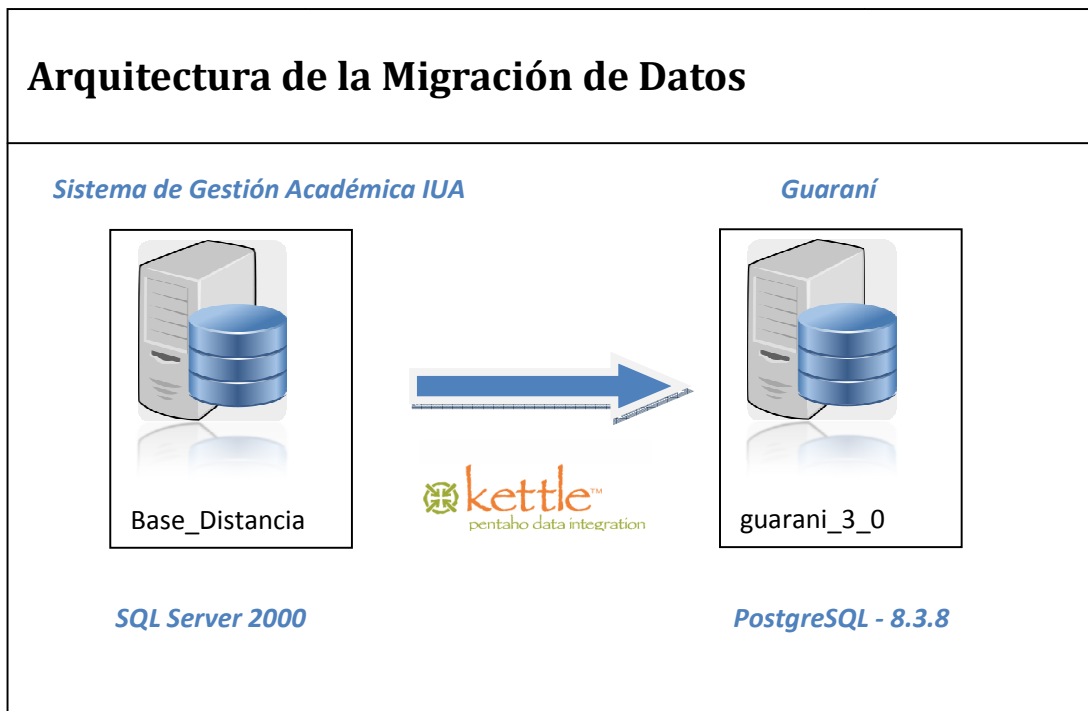
7.3.2 Diseño de la Migración de Datos

7.3.2.1. Introducción

En esta sección se presenta la arquitectura y la estructura de la aplicación para migrar los datos.

7.3.2.2 Diseño de la Arquitectura de la Migración de Datos

A continuación se observa el diagrama que representa la arquitectura de la migración de datos entre el Sistema de Gestión Académica del IUA y SIU-Guaraní:



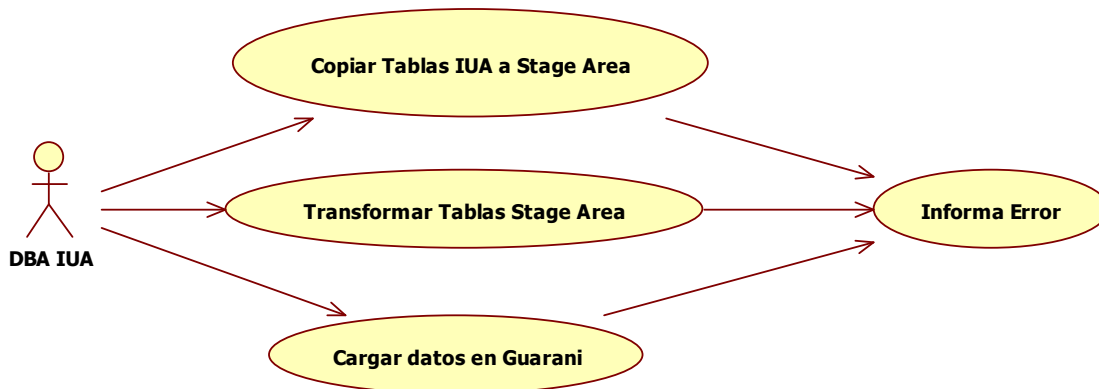
7.3.2.3 Diseño de Caso de Uso de la Migración

La definición del caso de uso describe el diseño real según una tecnología concreta de entrada y de salida, y su implementación.

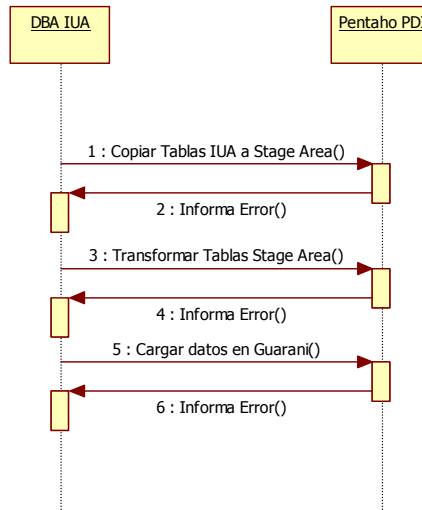
7.3.2.3.1 Diseño Caso de Uso - Migración de Datos.

Caso de Uso:	Migración de Datos
Tipo:	Esencial primario
Actor:	DBA IUA
Propósito:	Permitir que se ejecute la migración de datos desde la base de datos del Sistema de Gestión Académica del IUA (Sql Server 2000) hacia la base de datos de SIU-Guaraní (PostgreSQL - 8.3.8).
Descripción general:	La migración de datos se llevará a cabo a través de Pentaho PDI que será ejecutado mediante la manipulación del DBA del IUA.

7.3.2.3.2 Diseño Diagramas de caso de uso – Migración de Datos.



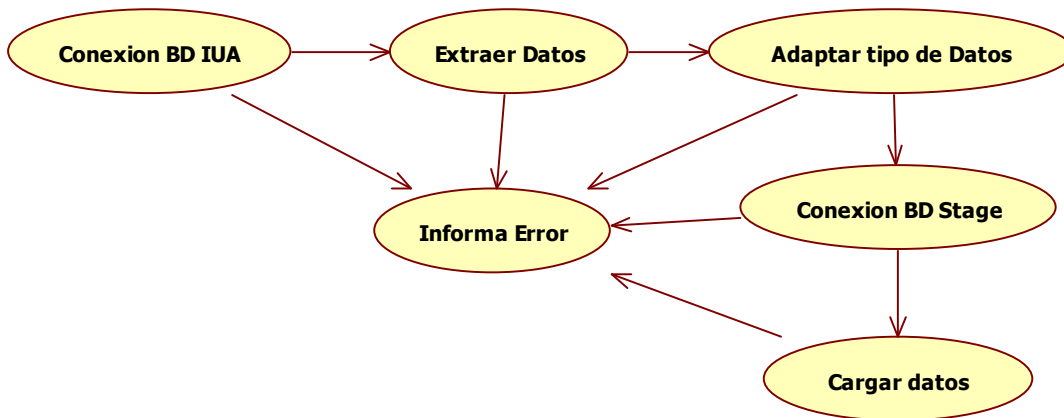
7.3.2.3.4 Diseño Diagramas de secuencia – Migración de Datos.



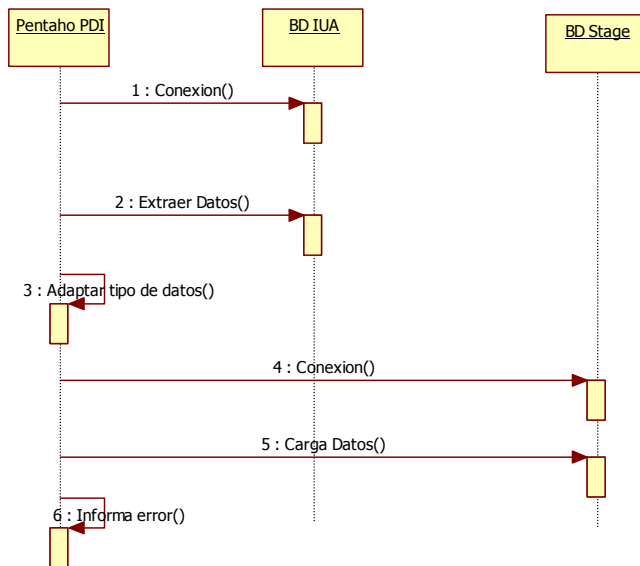
7.3.2.3.5 Diseño Caso de Uso - Copiar Tablas IUA a Stage Area.

Caso de Uso:	Copiar Tablas IUA a Stage Area.
Tipo:	Esencial primario
Actor:	
Propósito:	Duplicar las tablas de la base de datos del IUA a la Base de daos intermedia.
Descripción general:	Se copiaran las tablas de la Base de datos del IUA a la Base de datos intermedia.

7.3.2.3.6 Diseño Caso de Uso - Copiar Tablas IUA a Stage Area.



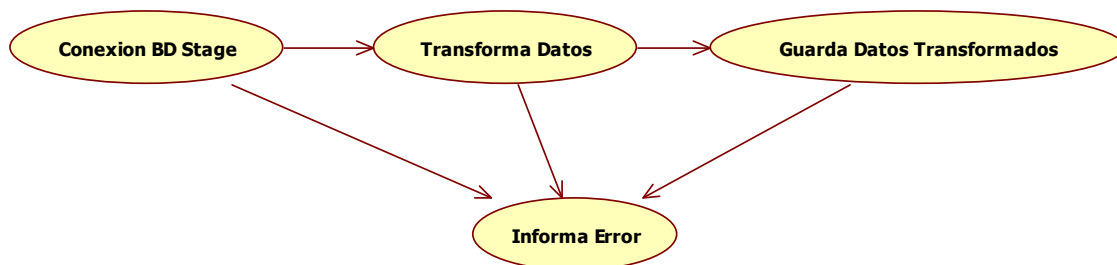
7.3.2.3.7 Diseño Diagramas de secuencia – Copiar Tablas IUA a Stage Area.



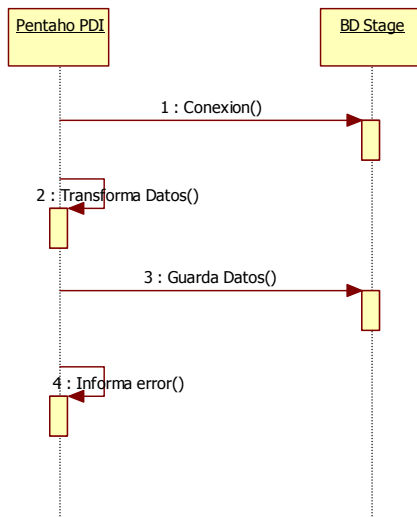
7.3.2.3.8 Diseño Caso de Uso - Transformar Tablas Stage Area.

Caso de Uso:	Transformar Tablas Stage Area.
Tipo:	Esencial primario
Actor:	
Propósito:	Transforma los Datos para adaptarlos al destino.
Descripción general:	Los datos extraídos del IUA serán transformados para poder ser cargados en Guarani.

7.3.2.3.9 Diseño Caso de Uso - Transformar Tablas Stage Area.



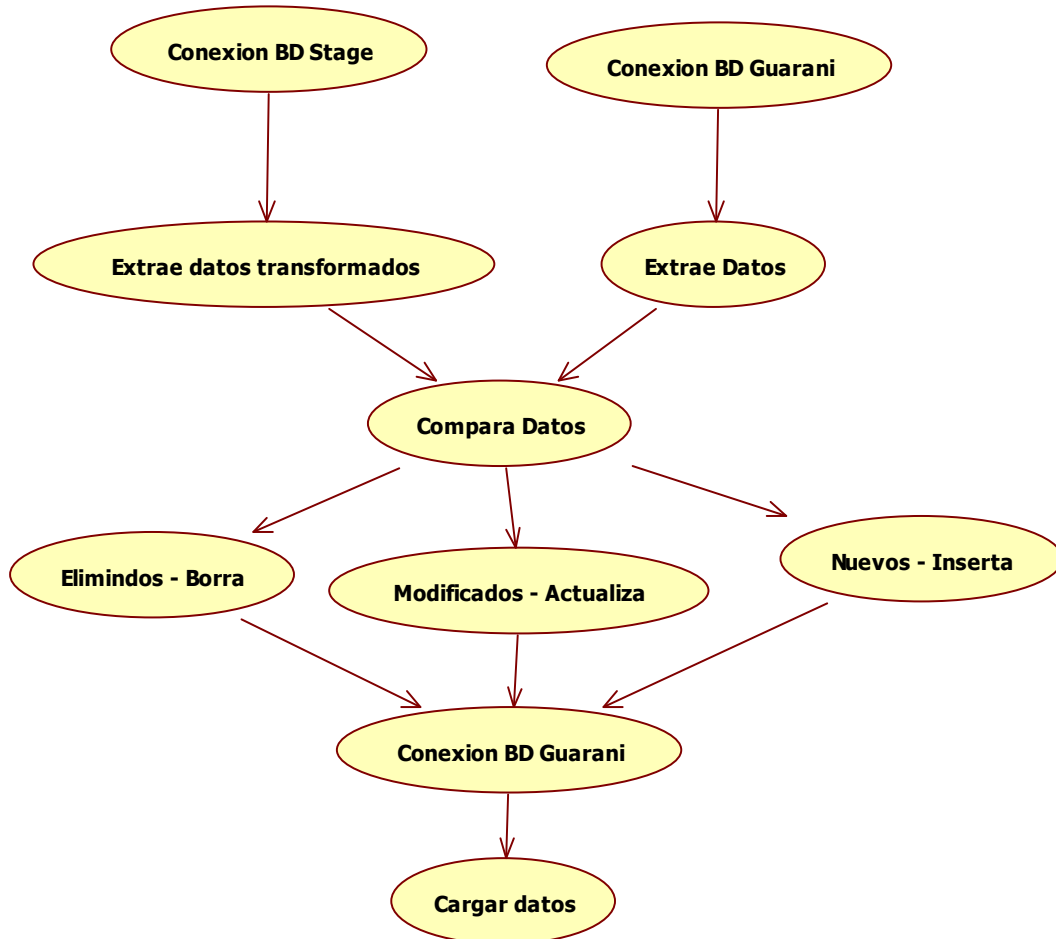
7.3.2.3.10 Diseño Diagramas de secuencia - Transformar Tablas Stage Area.



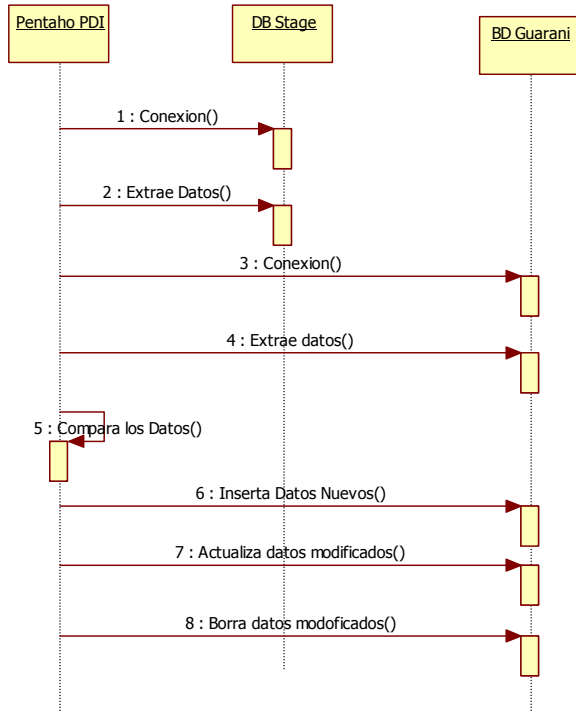
7.3.2.3.11 Diseño Caso de Uso - Cargar datos en Guarani

Caso de Uso:	Cargar datos en Guarani.
Tipo:	Esencial primario
Actor:	
Propósito:	Carga en Guarani los Datos que fueron transformados, previa verificación de datos nuevos, actualizados o borrados.
Descripción general:	Realiza la carga inicial o parcial de los datos realizando la comparación entre los datos nuevos y los previamente cargados en Guarani.

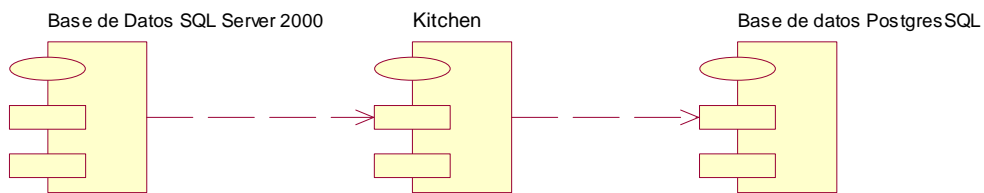
7.3.2.3.12 Diseño Diagramas de caso de uso – Cargar datos en Guarani



7.3.2.3.13 Diseño Diagramas de secuencia - Cargar datos en Guarani



7.3.2.3.14 Diseño Diagramas de componentes – Migración de Datos.



Etapa 4

CONCRECIÓN DEL MODELO

En esta etapa describiremos la introducción de la tecnología al medio, a la situación problema. Determinaremos y analizaremos los resultados obtenidos.

8.1 IMPLEMENTACION

8.1.1 Secuencia de Migración

La migración de datos se llevará a cabo siguiendo el orden de los siguientes trabajos:

Trabajo N° 1 (Extracción): se extraen los datos del Sistema de Gestión Académica del IUA y se copian a una base de datos intermedia (Área Stage) creada en PostgreSQL. El resultado de las extracciones se guardará en la base de datos intermedia, en tablas que mantienen la siguiente nomenclatura: “NombreDeTabla_Stage”

Trabajo N° 2 (Transformación): se realiza la depuración de datos en área stage. El resultado de las transformaciones se guardará en la misma base de datos intermedia, en tablas que mantienen la siguiente nomenclatura: “NombreDeTabla_Trans”.

Trabajo N° 3 (Carga): se transfieren los datos desde la base de datos intermedia a Guarani3.

Los datos que fueron transformados en el 2° Trabajo se comparan con sus equivalentes de Guarani3 y se realiza un control de cambios que aplica para los siguientes eventos:

- a. *Carga inicial de datos*
- b. *Modificación de datos cargados*
- c. *Eliminación de datos cargados*
- d. *Inserción de nuevos datos*

Para efectuar esta tarea se utiliza el tipo de técnica CDC (Change Data Capture) por indicador de estado, que permite capturar los cambios que se producen en el sistema fuente, de tal manera que pueden ser almacenados y prorrogados al sistema destino para que se mantenga la consistencia con el sistema de origen y se disminuya la latencia de datos.

Es importante aclarar que el evento de carga inicial de datos se efectúa una sola vez (primera migración). No obstante, los eventos de modificación de datos cargados, eliminación de datos cargados, así como inserción de nuevos datos podrán efectuarse siempre y cuando se haya ejecutado previamente la carga inicial de datos.

Trabajo N° 4 (Secuencia ETL): trabajo global que contiene los trabajos anteriormente mencionados, invocando a los mismos y ejecutándolos en el orden ETL que se encuentran establecidos.

8.1.2 Orden de Inserción en tablas de SIU-Guarani

SIU-Guarani3 está montado sobre un motor de base de datos relacional PostgreSQL en su versión 8.3. La integración de datos en el mismo se debe realizar siguiendo el orden establecido en la tabla (**Fig. Tabla 1**) que se expone a continuación.

Tabla de Orden de Inserción en SIU-Guarani3

Número de Insert	Nombre de Tabla de SIU-Guarani
1	sga_modalidad_cursada
2	sga_anios_academicos
3	mdp_tipo_documento
4	sga_acciones
5	mdp_nacionalidades
6	mdp_personasexo
7	mdp_titulos_tipos
8	acc_tipos_usuarios
9	mdp_estados_civiles
10	mdp_orientacion_recibida
11	sga_escalas_notas_resultado
12	sga_escalas_notas_det
13	sga_prestamos_tipos
14	sga_g3entidades_subtipos
15	mug_paises
16	sga_periodos_genericos
17	sga_propuestas_estados

18	sga_propuestas_tipos
19	mug_provincias
20	sga_elementos
21	sga_propuestas
22	mug_dptos_partidos
23	mug_localidades
24	sga_elementos_atrib
25	sga_planes
26	sga_planes_versiones
27	sga_ubicaciones
28	sga_edificaciones
29	mdp_personas
30	sga_instituciones
31	sga_espacios
32	sga_periodos_lectivos
33	sga_propuestas_aspira
34	sga_prestamos
35	sga_mesas_examen
36	sga_responsables_academicas
37	sga_alumnos
38	sga_colegios_secundarios
39	mdp_titulos
40	sga_docentes
41	sga_comisiones
42	acc_grupo_acc_x_tipo_usuario
43	sga_insc_cursada
44	mdp_antecedentes_profesionales
45	sga_reinscripciones
46	sga_requisitos_presentados
47	mdp_personas_contactos
48	mdp_antecedentes_docentes
49	mdp_personas_documentos
49	sga_escalas_notas_det
50	mdp_datos_censales
51	mdp_datos_personales
52	sga_inscripciones_estados
53	sga_actas_estados

54	sga_llamados_mesa
55	sga_evaluaciones_tipos
56	mdp_datos_estudios
57	sga_evaluaciones
58	sga_libros_actas
59	sga_eval_detalle_cursadas sga_eval_detalle_examen
60	sga_actas sga_actas_folios
61	sga_clases_tipos

Fig. Tabla 1

Es importante resaltar que el orden establecido de inserción de datos ha sido proporcionado por personal del sector de desarrollo de SIU-Guarani3. Este requerimiento fue entregado conjuntamente con la estructura de la base de datos.

8.1.3 Representación de Transformaciones y Trabajos en Kettle


En este punto ilustraremos la integración de los datos, representando las transformaciones y los trabajos que se implementarán con la herramienta Kettle.

Las transformaciones ETL se representarán siguiendo el orden de inserción que tiene la base de datos de SIU-Guarani3. Puede darse el caso que algunas tablas no necesiten ser sometidas a una instancia de transformación (depuración y limpieza), por lo que solo deberán cumplir con los pasos de extracción y carga.


Transformaciones de Extracción

A continuación se ilustran los Steps de Kettle que se utilizaran en las transformaciones de extracción de información desde la BD del IUA e inserción a la BD Stage:


Entrada de Tabla

 <p>Entrada Tabla BD IUA</p>	<p>Este objeto lee información desde una base de datos SQL Server 2000.</p> <p>Se utiliza para efectuar el SELECT que extraerá información de la BD del IUA para su traspaso a un Área Stage.</p>
---	---


Añadir Secuencia

 <p>Añadir secuencia</p>	<p>Este objeto permite obtener el valor de una secuencia que se generará en las tablas de la Base de Datos Stage.</p> <p>Lo utilizaremos para crear índices numéricos de manera ascendente. Será requerido siempre y cuando no exista en las tablas de SQL y se necesite para establecer relaciones entre tablas de PostgreSQL (Stage).</p>
---	---


Selecciona / Renombra Valores

 <p>Selecciona/Renombra valores</p>	<p>Este objeto permite seleccionar o eliminar campos de una fila. Opcionalmente, permite establecer la meta-información del campo: tipo, longitud, y precisión.</p> <p>Lo utilizaremos para modificar el tipo de dato del campo seleccionado, que se origina en una tabla de la BD del IUA y se requiere en PostgreSQL con un tipo de dato diferente. En las transformaciones de extracción solamente se usará para la tabla DEPENDENCIA.</p>
--	---


Si el Valor del Campo es Nulo

 <p>If field value is null</p>	<p>Este objeto permite establecer un valor de campo a una constante si es nulo.</p> <p>Lo utilizaremos para sustituir a los nulos que provienen de una tabla de la BD del IUA por un valor determinado y requerido en PostgreSQL. En las transformaciones de extracción solamente se usará para la tabla ESTUDIOS_NIVEL_MEDIO.</p>
---	--

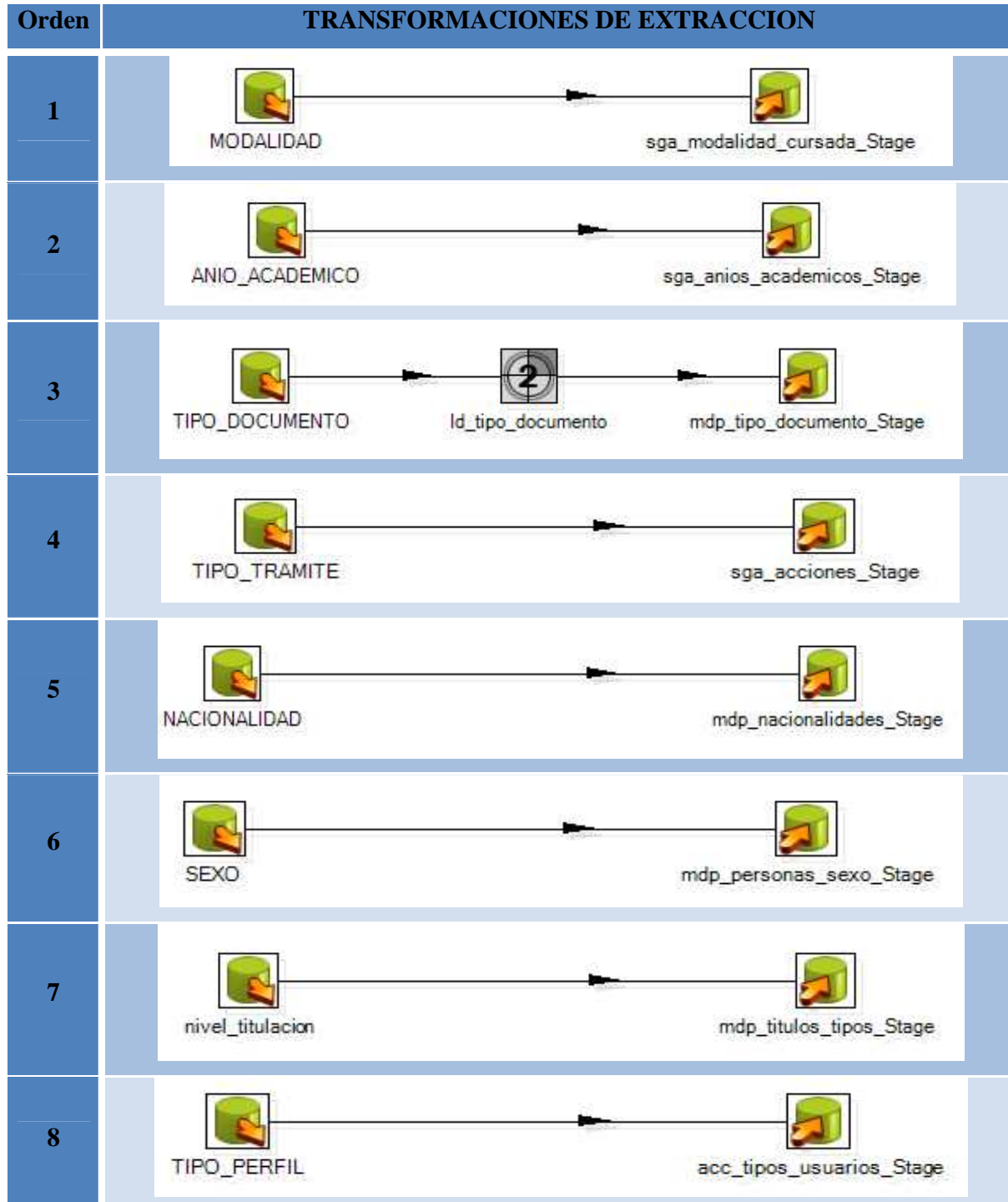
Generar Valor Aleatorio

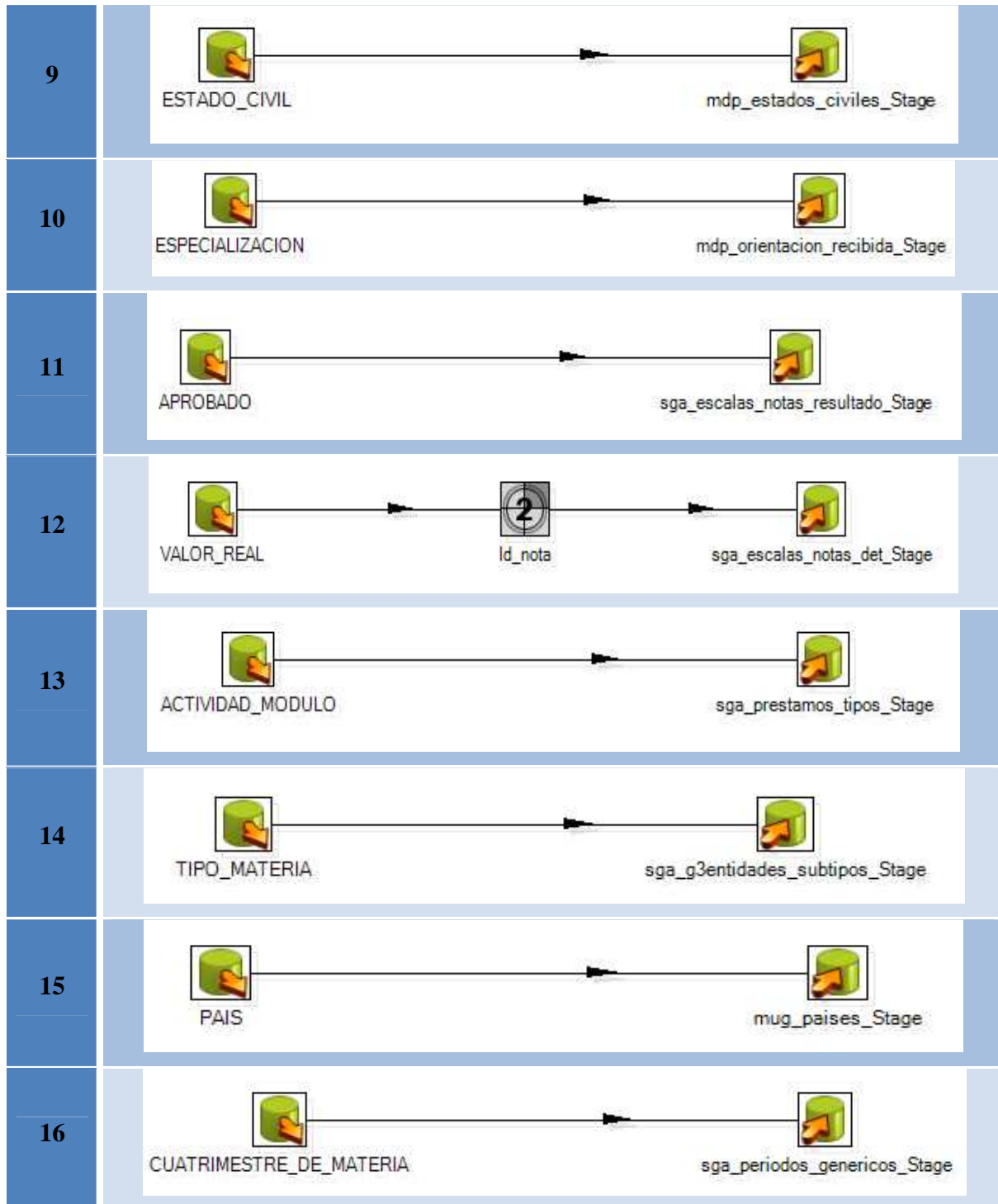
 <p>Generate random value</p>	<p>Este objeto genera números aleatorios, enteros, cadenas, y hasta identificadores universales únicos.</p> <p>Lo utilizaremos para generar una cadena aleatoria basada en un valor aleatorio de 64 bits de largo. Este paso es usado durante la extracción de datos de la tabla PERFIL, más precisamente sobre el campo USUARIO que contiene un dato genérico (usuario) presentado de tal forma por una cuestión de confidencialidad de la información. Por tal motivo, se generan valores aleatorios contenidos en el campo USUARIO para poder aplicar una distinción entre los mismos durante la migración.</p>
--	--

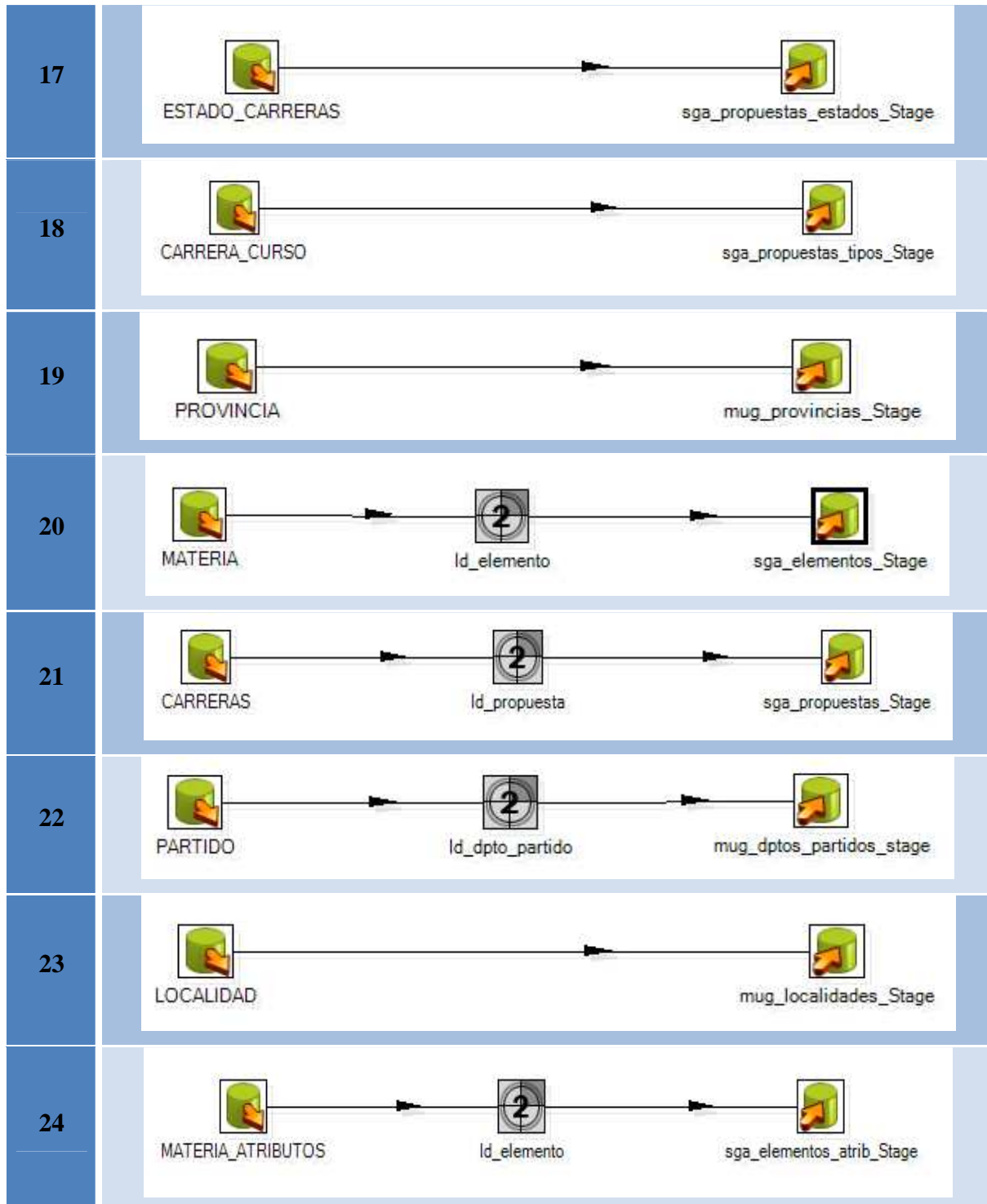
Salida de Tabla

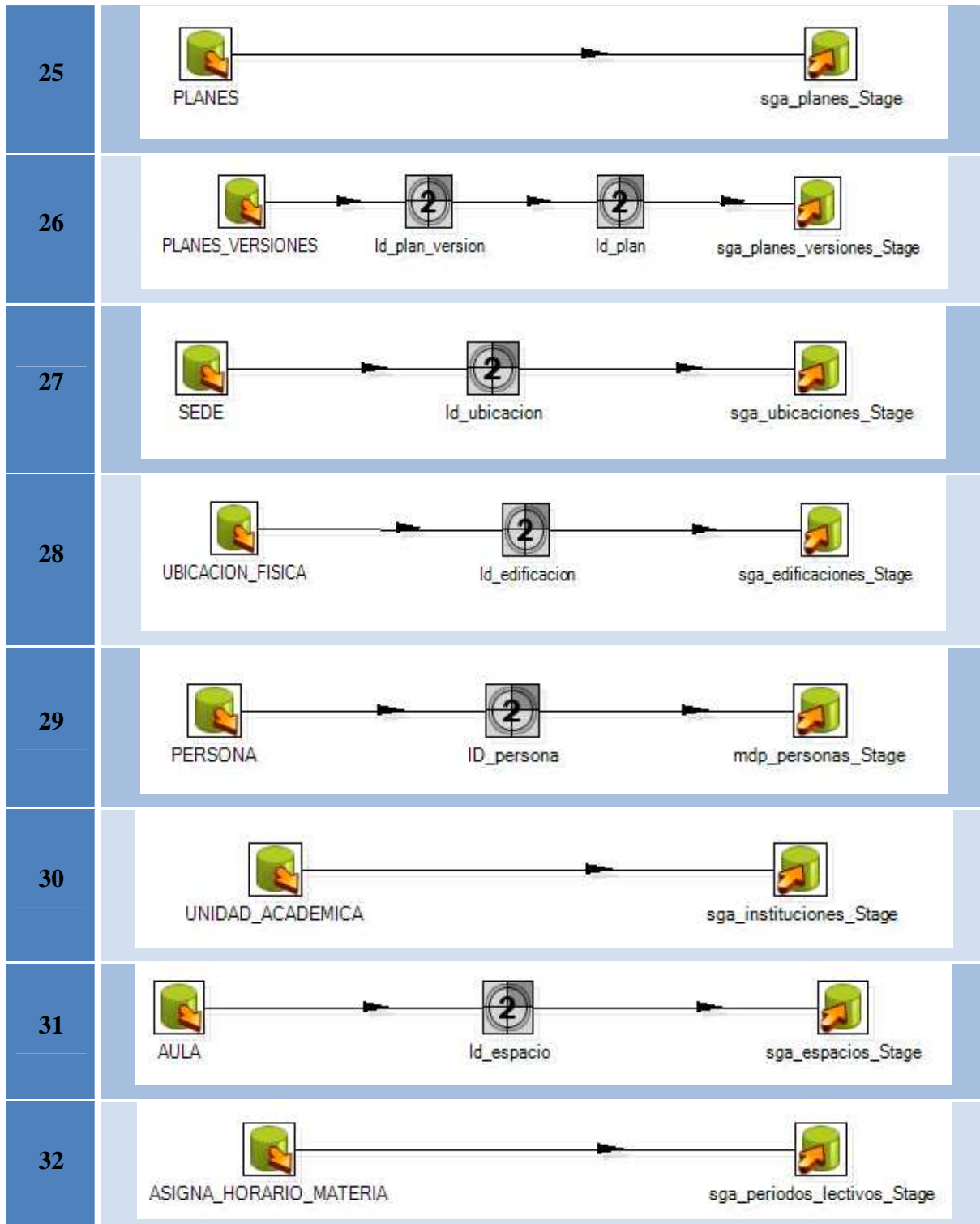
 <p>Salida Tabla Stage</p>	<p>Este objeto escribe información a una tabla de base de datos PostgreSQL.</p> <p>Se utiliza para insertar información procedente de la BD del IUA a una BD Intermedia (Área Stage). Los datos serán insertados en tablas previamente creadas en dicha Área.</p>
---	---

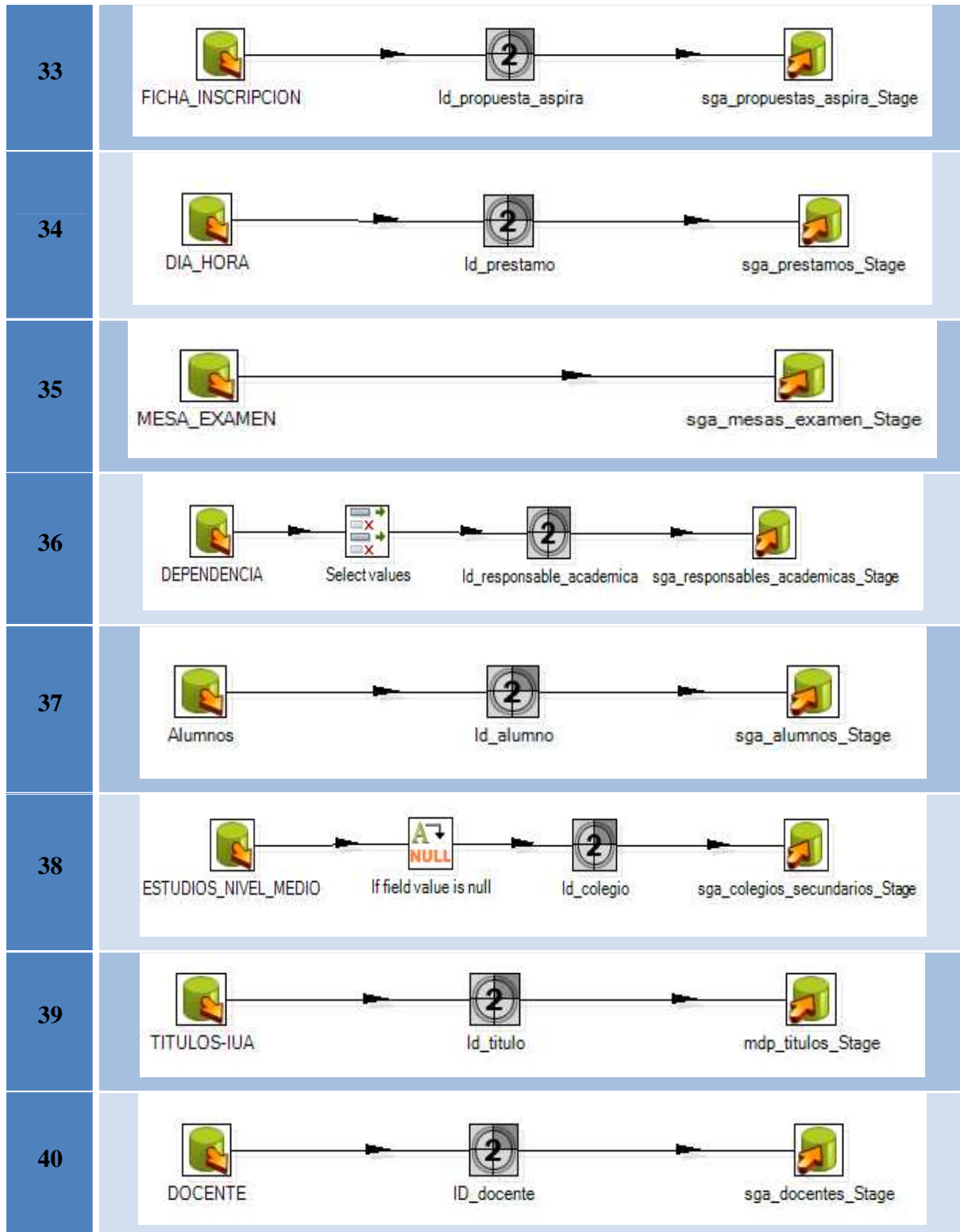
8.1.3.2 Transformaciones para Extracción de Datos





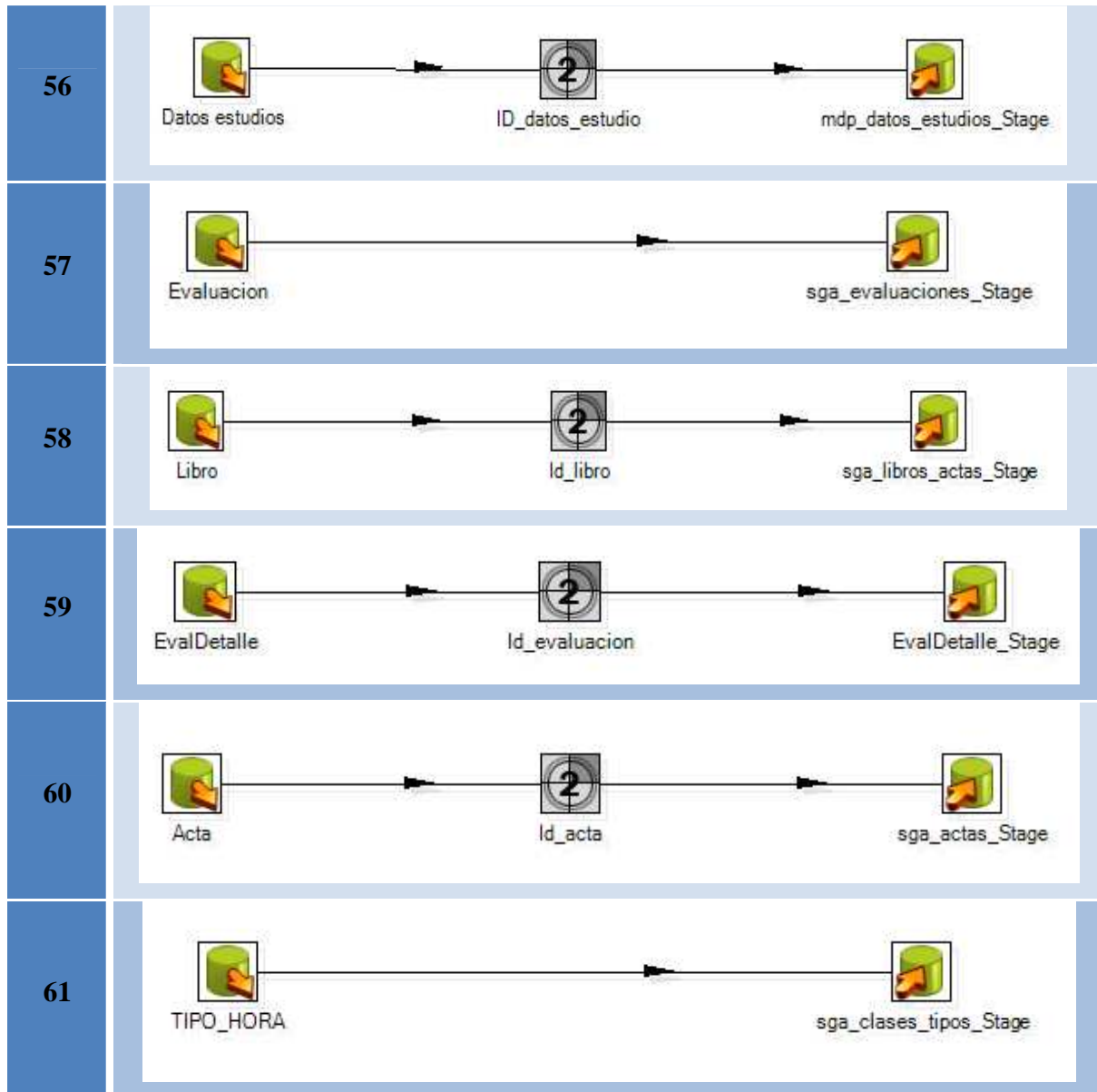






41	<p>Division → Id_comision → sga_comisiones_Stage</p>
42	<p>Perfil → Generate random value → acc_grupo_stage</p>
43	<p>fech_insc_mat → Id_inscripcion → sga_insc_cursada_Stage</p>
44	<p>DATOS_TRABAJO → Id_antecedente_profesional → mdp_antecedentes_profesionales_Stage</p>
45	<p>FECHA_REINSCRIPCION → Id_reinscripcion → sga_reinscripciones_Stage</p>
46	<p>FECHA_TRAMITE → Id_requisito_presentado → sga_requisitos_presentados_Stage</p>
47	<p>contactos → Id_persona_contacto → mdp_personas_contactos_Stage</p>

48	<p>DOCENTE_FUNCION Id_antecedente_docente mdp_antecedentes_docentes_Stage</p>
49	<p>NRO_DOC Id_documento mdp_personas_documentos_Stage</p>
50	<p>Datos Personales Id_datos_censal dato_censal_Stage</p>
51	<p>Datos Personales mdp_datos_personales_Stage</p>
52	<p>ESTADO_ALUMNO sga_inscripciones_estados_Stage</p>
53	<p>EN_ACTA sga_actas_estados_Stage</p>
54	<p>FECHA_EXAMEN_FINAL Id_llamados_mesa sga_llamados_mesa_Stage</p>
55	<p>EvalTipo sga_evaluaciones_tipos_Stage</p>



Existen tablas que fueron cargadas previamente en SIU-Guarani, dado que las mismas complementan datos existentes en la base de datos del IUA. Los ejemplos son los siguientes:

Tabla *sga_escalas_notas*:

	escala_nota integer	nombre character varying(50)
1	1	0 a 10
2	2	Aprobado-Desaprobado
3	3	Promociones
4	4	Parciales con Decimales

La tabla “*sga_escalas_notas*” fue considerada para poder complementar la carga de la tabla “*sga_escalas_notas_det*” que contiene cada uno de los resultados de notas posibles. A continuación se puede observar un ejemplo del contenido de la tabla “*sga_escalas_notas_det*” donde el campo *escala_nota* es igual a “1” definido con el nombre “0 a 10” en la tabla “*sga_escalas_notas*”.

	escala_nota integer	nota character varying(10)	descripcion character varying(100)	resultado character(1)	valor_numerico numeric(10,4)
1	1	1	NO APROBADO	3	0.0000
2	1	2	NO APROBADO	3	1.0000
3	1	3	NO APROBADO	3	2.0000
4	1	4	NO APROBADO	3	3.0000
5	1	5	APROBADO	1	4.0000
6	1	6	APROBADO	1	5.0000
7	1	7	APROBADO	1	6.0000
8	1	8	APROBADO	1	7.0000
9	1	9	APROBADO	1	8.0000
10	1	10	APROBADO	1	9.0000
11	1	11	APROBADO	1	10.0000
12	1	12	AUSENTE	2	101.0000

Es importante aclarar que el campo *resultado* se relaciona con la tabla “*sga_escalas_notas_resultado*” que se expone a continuación:

	resultado [PK] character(1)	descripcion character varying(50)
1	1	APROBADO
2	2	AUSENTE
3	3	NO APROBADO

Tabla *sga_planes_estados*:

	estado [PK] character(1)	descripcion character varying(50)
1	A	Activo no Vigente
2	B	Dado de Baja
3	N	Nuevo
4	V	Activo Vigente

La tabla “*sga_planes_estados*” fue considerada para poder complementar la carga de las tablas: “*sga_planes*” y “*sga_planes_versiones*”. Observamos que ambas tablas incluyen el campo *estado*:

Tabla *sga_planes*:

	plan [PK] integer	propuesta integer	nombre character vai	codigo character vai	duracion_en_anios integer	duracion_en_meses integer	estado character(1)
1	1	1	1996	0	1	12	B
2	2	2	1996	A5	1	14	B
3	3	3	2009	AER	0	0	A
4	4	4	2004	ALE	0	0	A
5	5	5	2004	AUD	0	0	A
6	6	6	2007	AUD	1	12	A
7	7	7	2004	C01	4	49	A
8	8	8	2004	C02	4	49	A
9	9	9	2004	C03	4	49	A
10	10	10	2004	C04	0	1	A
11	11	11	2004	C05	0	1	A
12	12	12	2004	C06	0	1	A
13	13	13	2004	C07	0	1	A
14	14	14	2008	CAD	0	6	B
15	15	15	2009	CAD	0	6	A
16	16	16	2007	CAT	0	0	A
17	17	17	1994	CC	0	0	A
18	18	18	2003	CI	0	0	B
19	19	19	2006	CI	0	1	A
20	20	20	2009	CLD	0	0	A
21	21	21	2009	CLP	0	0	A
22	22	22	2008	CNE	0	6	B
23	23	23	2006	CON	0	0	A
24	24	24	2002	CP	4	59	A
25	25	25	1997	CR	4	59	B
26	26	26	1997	CS	0	0	A
27	27	27	2007	CS	1	12	A
28	28	28	2006	DC1	0	6	B
29	29	29	2003	EAF	0	0	B
30	30	30	2005	EFW	0	0	A
31	31	31	ENI2	EN2	0	6	B
32	32	32	ENI3	EN3	0	6	A
33	33	33	ENIM	ENI	0	0	B
34	34	34	2007	ESE	1	12	A

Tabla *sga_planes_versiones*:

	plan_version [PK] integer	plan integer	cnt_materias smallint	cnt_optativa smallint	estado character(1)
1	1	1	0		B
2	2	2	0		B
3	3	3	0		A
4	4	4	0		A
5	5	5	0		A
6	6	6	7		A
7	7	7	1	0	A
8	8	8	1	0	A
9	9	9	1	0	A
10	10	10	0		A
11	11	11	0		A
12	12	12	0		A
13	13	13	0		A
14	14	14	1	0	B
15	15	15	1	0	A
16	16	16	0		A
17	17	17	6	0	A
18	18	18	1	1	B
19	19	19	1	1	A
20	20	20	0		A
21	21	21	0		A
22	22	22	1	0	B
23	23	23	0		A
24	24	24	38	0	A
25	25	25	2	0	B
26	26	26	6	0	A
27	27	27	8		A
28	28	28	1	0	B
29	29	29	1	0	B
30	30	30	0		A
31	31	31	3	0	B
32	32	32	2	0	A
33	33	33	2	2	B

Por otra parte, hemos utilizado dos tablas puente (intermedias):

1. *mug_dptos_partidos*
2. *mdp_datos_censales*.

La primera es utilizada para poder establecer la relación entre Provincias y Localidades, dado que SIU-Guarani3 utiliza *mug_dptos_partidos* para administrar los partidos o departamentos contenidos en una Provincia determinada. Asimismo, el IUA en su base de datos no considera

esa gestión y solo incluye Localidades y Provincias, por lo que la tabla señalada ha sido usada en nuestra migración para poder establecer la relación entre ambas.

La segunda es utilizada para poder establecer la relación entre Persona y Datos Personales, dado que SIU-Guarani3 utiliza *mdp_datos_censales* como una tabla intermedia que permite realizar la vinculación entre ambas. En el IUA la relación es directa, por lo que fue necesario para la migración generar una secuencia que permita cargar la tabla *mdp_datos_censales* y de esta manera poder establecer la relación entre Persona y Datos Personales.

8.1.3.3 Ejemplos de Pasos ejecutados durante la Extracción de Datos

- *Ejemplo Edición del Paso **Entrada de Tabla** “MATERIA”.*

En la **imagen 8.1.3.3.a** se observa la consulta a la tabla MATERIA en SQL Server.

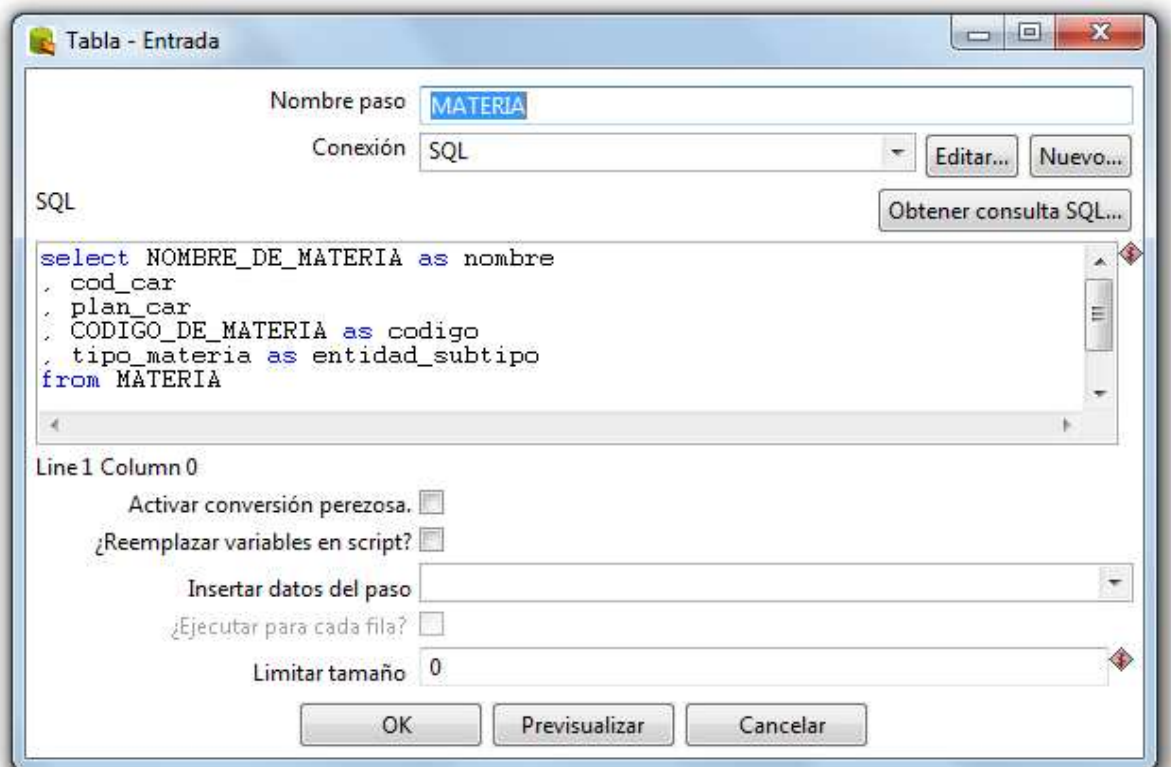


Imagen 8.1.3.3.a

- *Ejemplo Edición del Paso **Añadir Secuencia** para el índice “elemento”.*

En la **imagen 8.1.3.3.b** se observa la generación del valor numérico secuencial: “elemento”.

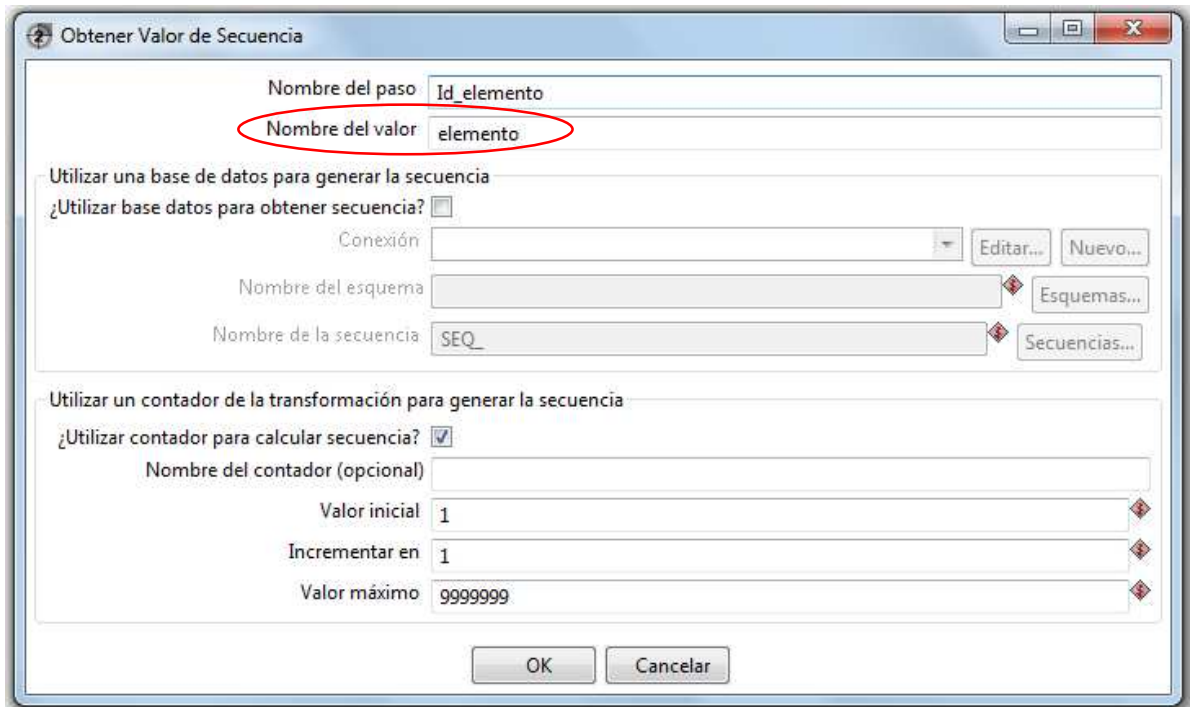


Imagen 8.1.3.3.b

- *Ejemplo Edición del Paso **Selecciona / Renombra valores** que aplica para el tipo de dato del campo “responsable_academica_tipo”.*

En la **imagen 8.1.3.3.c** se observa la modificación a Integer para el tipo de dato del campo “responsable_academica_tipo”.

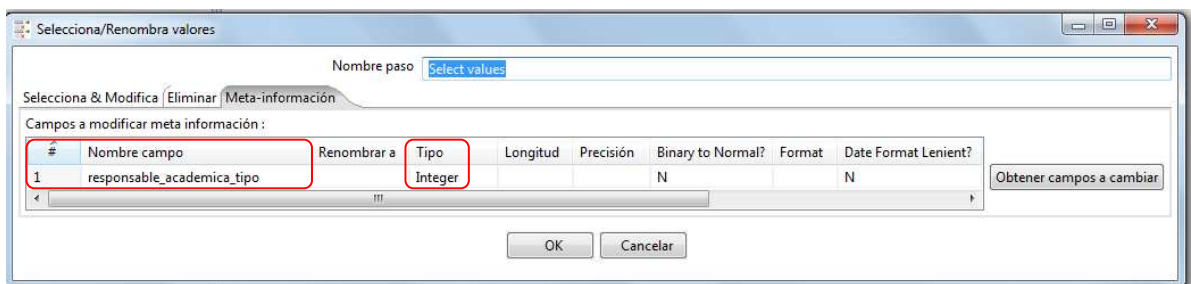


Imagen 8.1.3.3.c

- *Ejemplo Edición del Paso **Si el Valor del Campo es Nulo** que aplica para el nombre de la Institución de Nivel Medio.*

En la **imagen 8.1.3.3.d** se observa el reemplazo de los valores Null por N/D.

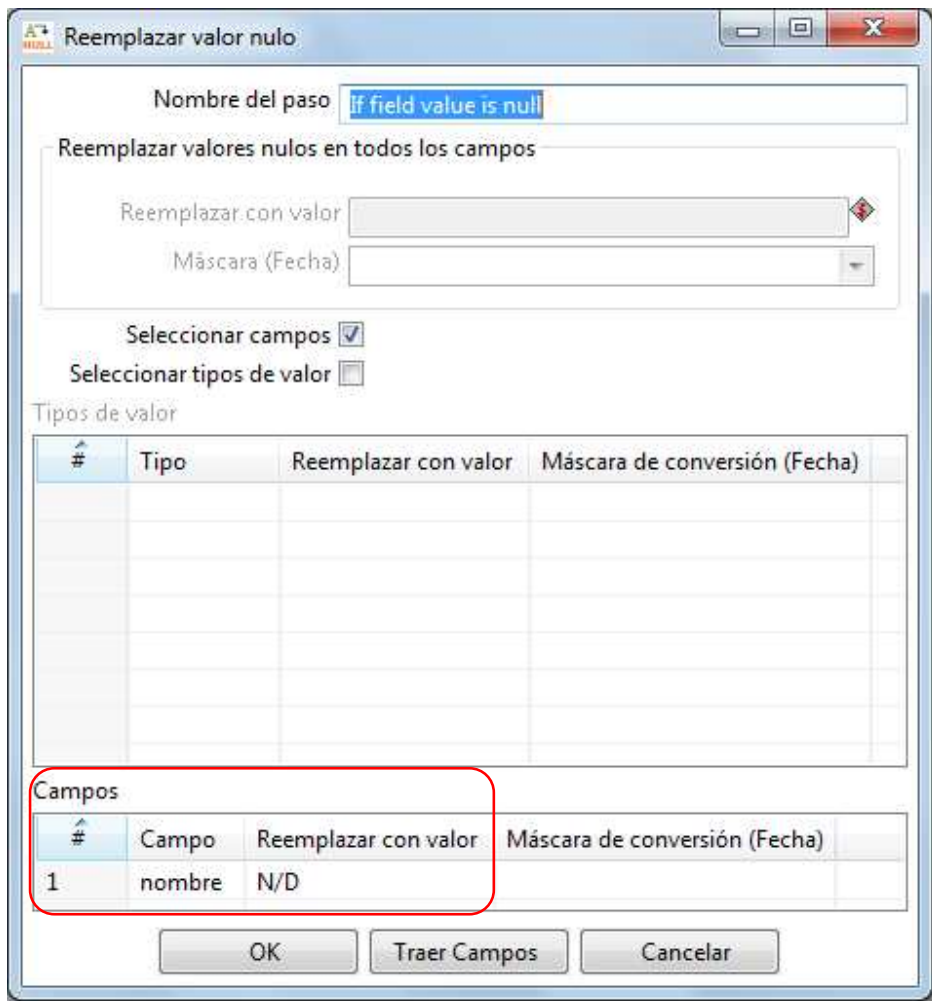


Imagen 8.1.3.3.d

- Ejemplo Edición del Paso **Generar valor aleatorio** que aplica para el campo *Usuario*.

En la **imagen 8.1.3.3.e** se observa la generación de un valor de tipo Cadena aleatoria que se asigna al campo *usuario_grupo_acc* para insertar en la tabla *acc_grupo_Stage*.

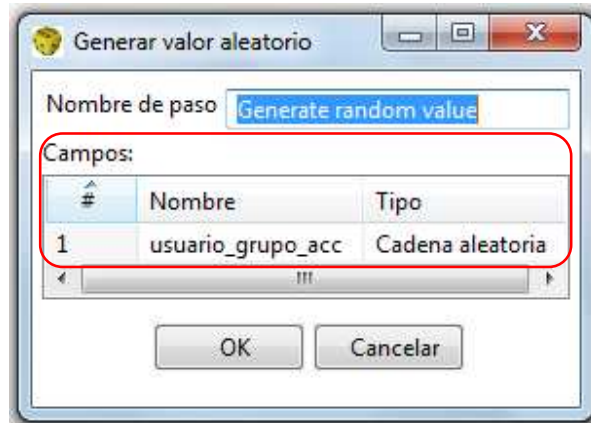


Imagen 8.1.3.3.e

- Ejemplo Edición del Paso **Salida de Tabla** “sga_propuestas_aspira_Stage”.

En la **imagen 8.1.3.3.f** se observa la conexión que se establece a la base de datos Stage donde se insertaran los datos previos a su depuración y limpieza.

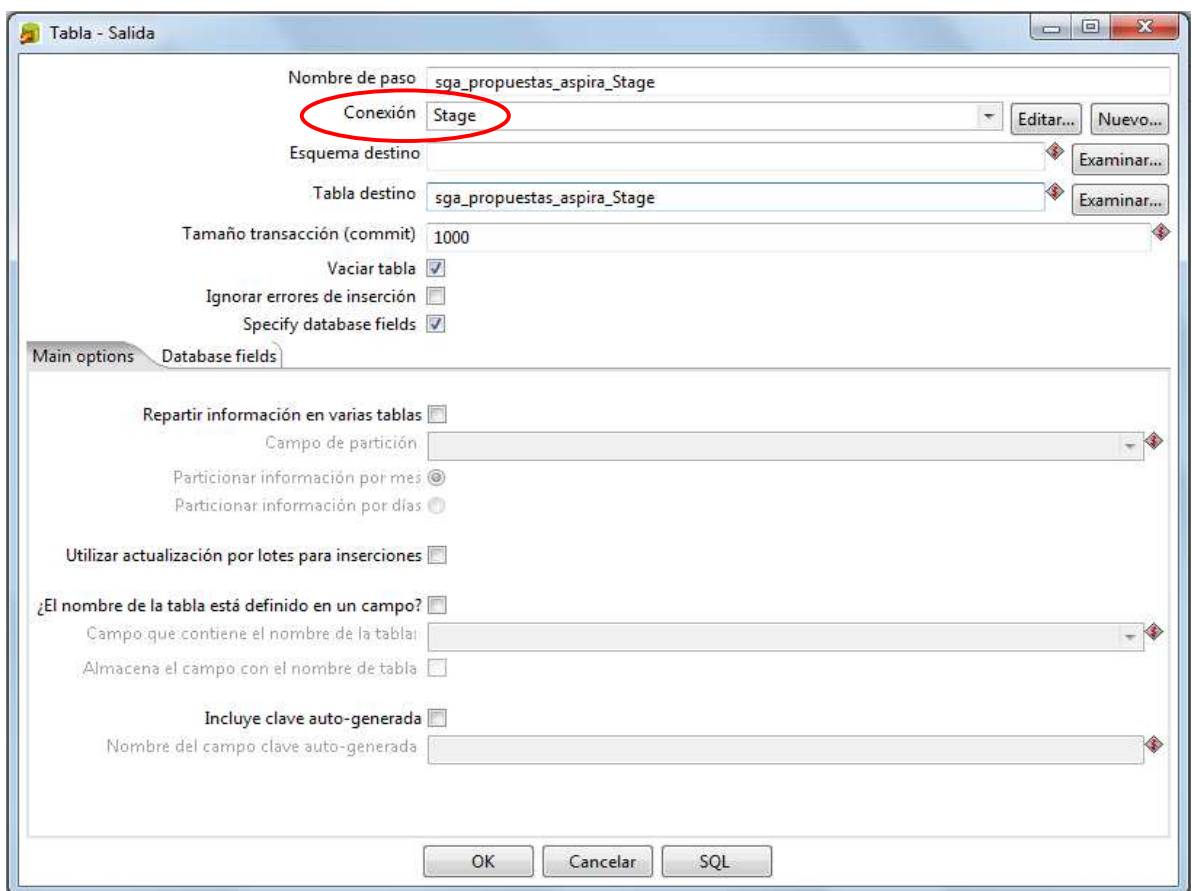


Imagen 8.1.3.3.f

En la **imagen 8.1.3.3.g** se observan los campos para insertar en la base de datos Stage.

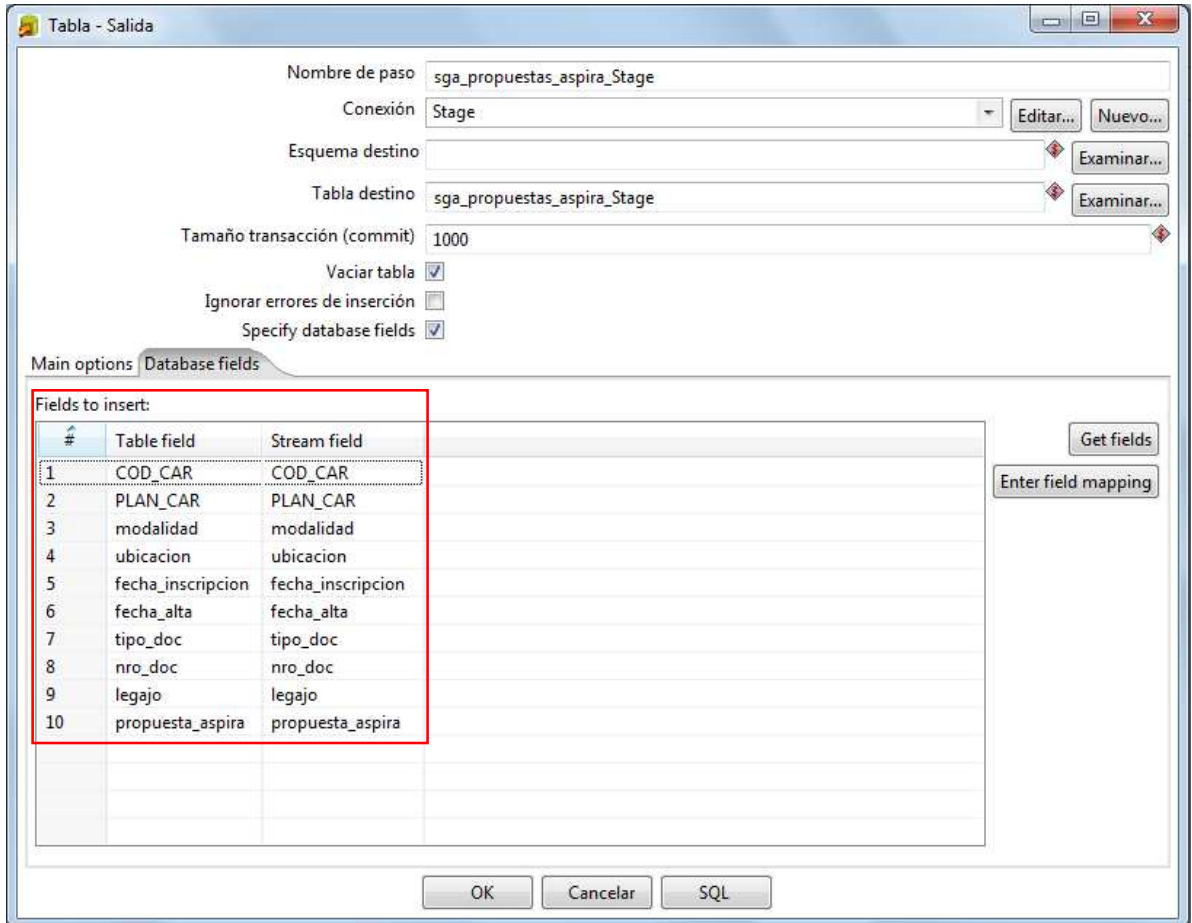


Imagen 8.1.3.3.g

Cabe aclarar que todos los ejemplos ilustrados corresponden a diferentes transformaciones de la etapa de extracción de datos.

8.1.3.4 Trabajo para Extracción de Datos

A continuación se ilustra en la **Fig. 8.1.3.4** el Trabajo planificado en Kettle que ejecuta automáticamente y en orden secuencial cada una de las transformaciones para Extracción de Datos:

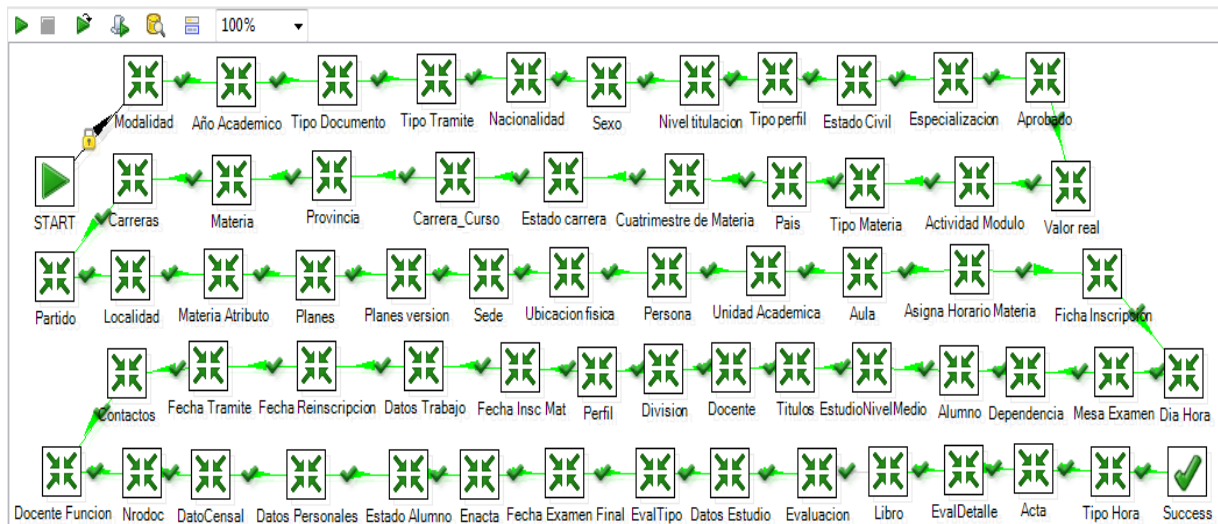



Fig. 8.1.3.4

El resultado de la ejecución del trabajo previamente ilustrado quedará demostrado en la etapa de Pruebas, en el caso de test que aplique al mismo.


Transformaciones de Depuración y Limpieza de Datos

A continuación se ilustran los Steps de Kettle que se utilizarán en las transformaciones de limpieza y depuración de datos en Stage (BD Intermedia):


Entrada de Tabla

 <p>Entrada Tabla BD Stage</p>	<p>Este objeto lee información desde una base de datos PostgreSQL.</p> <p>Se utiliza para efectuar el SELECT que extraerá información de la BD Intermedia para su posterior depuración y limpieza.</p>
---	--


Ordenar Filas

 <p>Ordenar filas</p>	<p>Este objeto ordena filas basado en valores de campos.</p> <p>Lo utilizaremos para ordenar filas de manera ascendente. Este paso es requerido durante la depuración y limpieza de datos para poder considerar en forma ordenada la información a transformar.</p>
--	---


Filas Únicas

 <p>Filas Únicas</p>	<p>Este objeto permite la eliminación de filas duplicadas y deja solo ocurrencias únicas. Funciona solo con entradas ordenadas.</p> <p>Lo utilizaremos en tablas de la BD Stage para seleccionar valores únicos. Por lo general, se aplica para aquellas tablas que no se encuentran correctamente normalizadas en la BD del IUA y deben transformarse para respetar las formas normales que requiere PostgreSQL para SIU-Guarani3.</p>
---	---


Operaciones con Cadenas

 <p>String operations</p>	<p>Este objeto permite aplicar ciertas operaciones, como el recorte, relleno y otros, al valor de una cadena.</p> <p>Lo utilizaremos para recortar los espacios en blanco que contienen algunas cadenas de datos que fueron ingresados a la BD del IUA con esas imperfecciones. Por lo general, el recorte se efectúa a la derecha de la cadena.</p>
--	--


Mapeo de Valores

 <p>Mapeo de Valores</p>	<p>Este objeto establece una conversión de valores de un campo a otro valor.</p> <p>Lo utilizaremos para efectuar el mapeo de aquellos valores que se cargaron incorrectamente en la BD del IUA. Por ejemplo, valores con errores de ortografía, abreviaturas incorrectas, nulos, y mal uso de mayúsculas y minúsculas, etc. De esta manera, la conversión a un valor estándar optimizará la integración de la información aportando un aumento en la consistencia de los datos.</p>
---	--


Añadir Secuencia

 <p>Añadir secuencia</p>	<p>Este objeto permite obtener el valor de una secuencia.</p> <p>Lo utilizaremos para crear índices numéricos de manera ascendente. Será requerido siempre y cuando no exista en las tablas de la BD Stage y se necesite para establecer relaciones entre tablas de PostgreSQL (SIU-Guarani3).</p>
---	--


Búsqueda en Flujo de Datos

 <p>Búsqueda en Flujo de Datos</p>	<p>Este objeto permite obtener valores procedentes de otros flujos en la transformación.</p> <p>Lo utilizaremos para buscar datos utilizando información procedente de otros pasos con el objetivo de añadirlos en el siguiente paso.</p>
---	---


Si el Valor del Capo es Nulo

 <p>If field value is null</p>	<p>Este objeto permite establecer un valor de campo a una constante si es nulo.</p> <p>Lo utilizaremos para sustituir a los nulos en aquellos campos que requieran de manera indispensable un valor que no sea nulo. En las transformaciones de limpieza y depuración solamente se usará para las tablas: sga_periodos_genericos_Stage, sga_planes_Stage, y sga_planes_versiones_Stage.</p>
--	---


Filtrar Filas

 <p>Filtrar filas</p>	<p>Este objeto permite filtrar filas utilizando fórmulas sencillas.</p> <p>Lo utilizaremos con el fin de dividir una sola corriente de información en diferentes corrientes más pequeñas. La información resultante del filtro aplicado se puede traducir a datos “Verdaderos” o datos “Falsos” dependiendo del objetivo de la transformación.</p>
--	--


Transformación simulada (no hace nada)

 <p>Transformación Simulada (no hace nada)</p>	<p>Pasos de este tipo no hacen nada. Es útil en ciertas situaciones donde es necesario partir flujos de información.</p> <p>Por lo general se aplicará cuando sea utilizado el paso “Filtrar Filas” que deberá conectarse al mismo y recibir de este los flujos de información que se presenten como datos “Falsos”.</p>
---	--

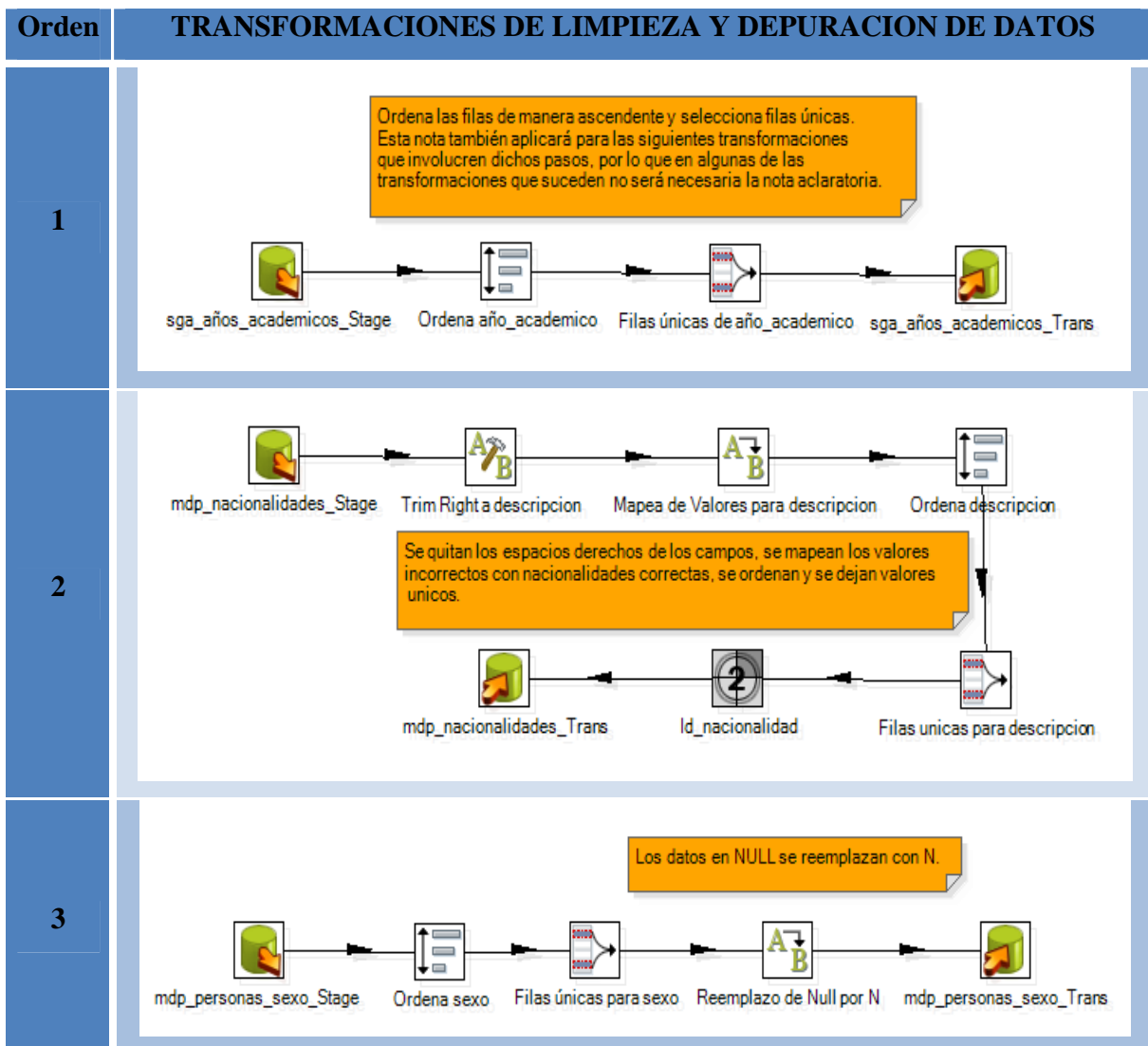
Calculadora

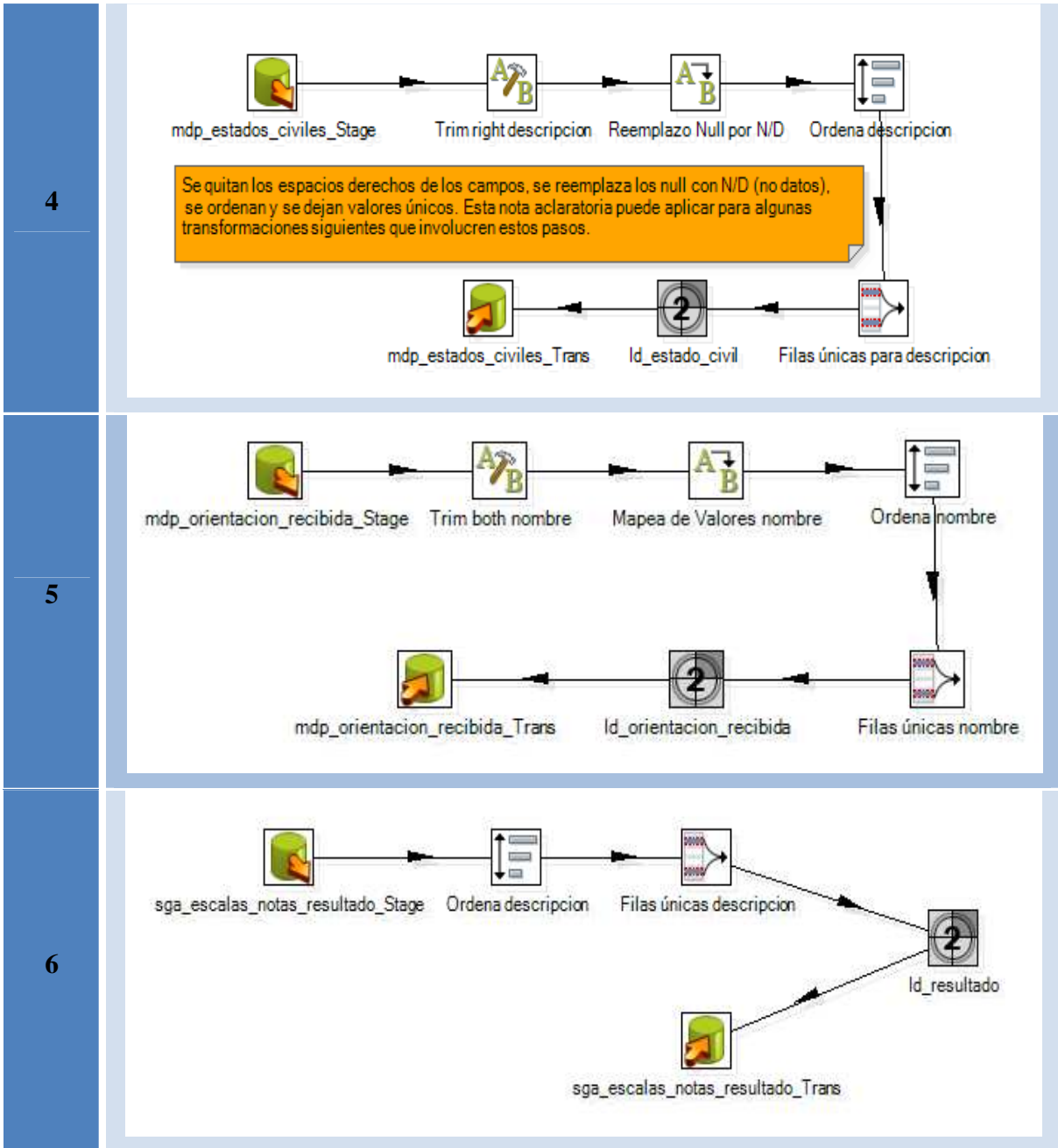
 <p>Calculadora</p>	<p>Este objeto permite crear nuevos campos a partir de cálculos sencillos.</p> <p>Lo utilizaremos cuando sea necesario crear en una tabla de la BD Stage un nuevo campo que sea resultado del cálculo entre campos ya especificados que se pasaran como argumentos con un valor determinado.</p>
--	--

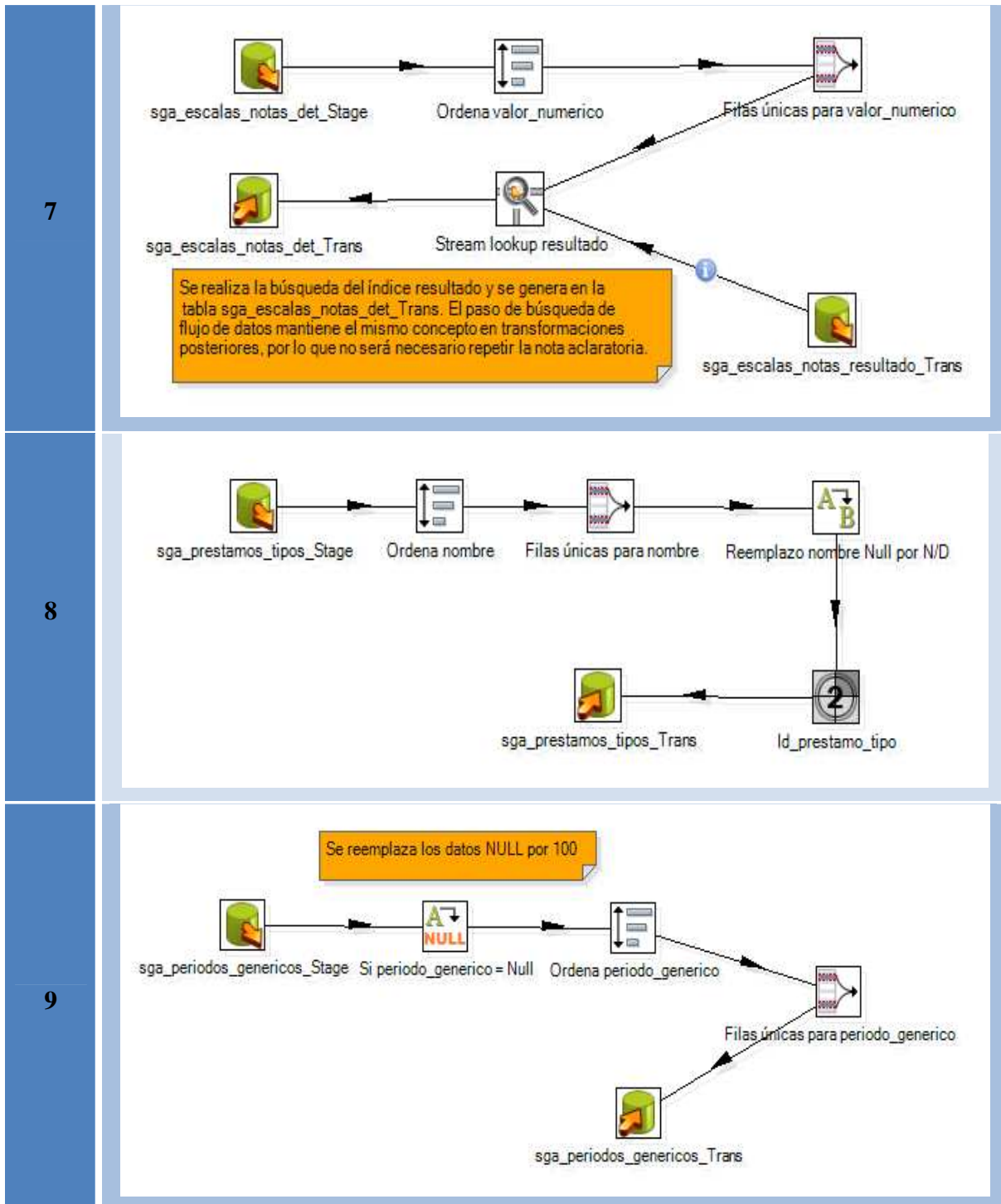
Salida de Tabla

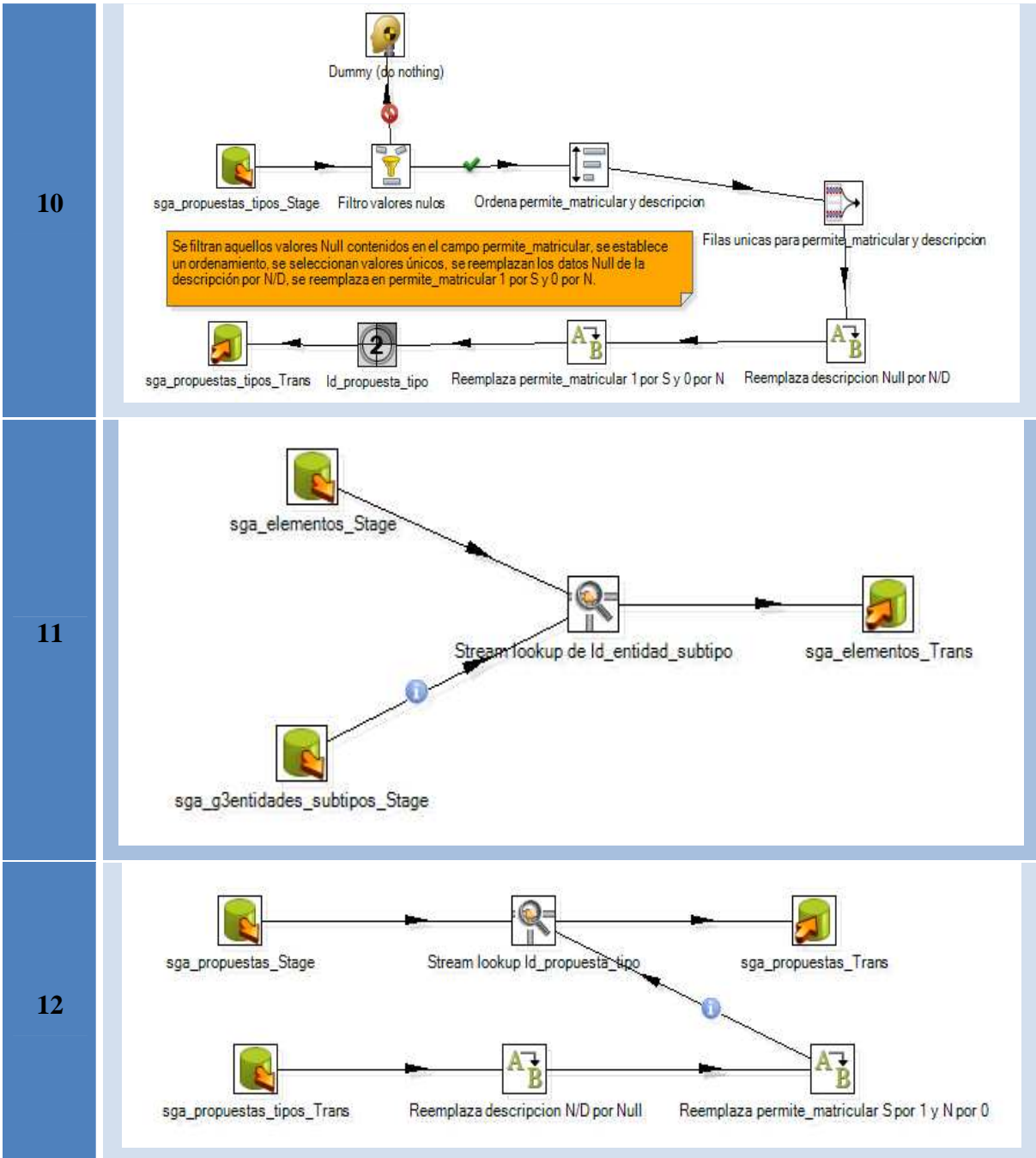
	<p>Este objeto escribe información a una tabla de base de datos PostgreSQL.</p> <p>Se utiliza para insertar información ya transformada (depurada) dentro del Área Stage. Los datos serán insertados en tablas previamente creadas en dicha Área y serán utilizados como información de Entrada de Tabla en las transformaciones de Carga.</p>
---	--

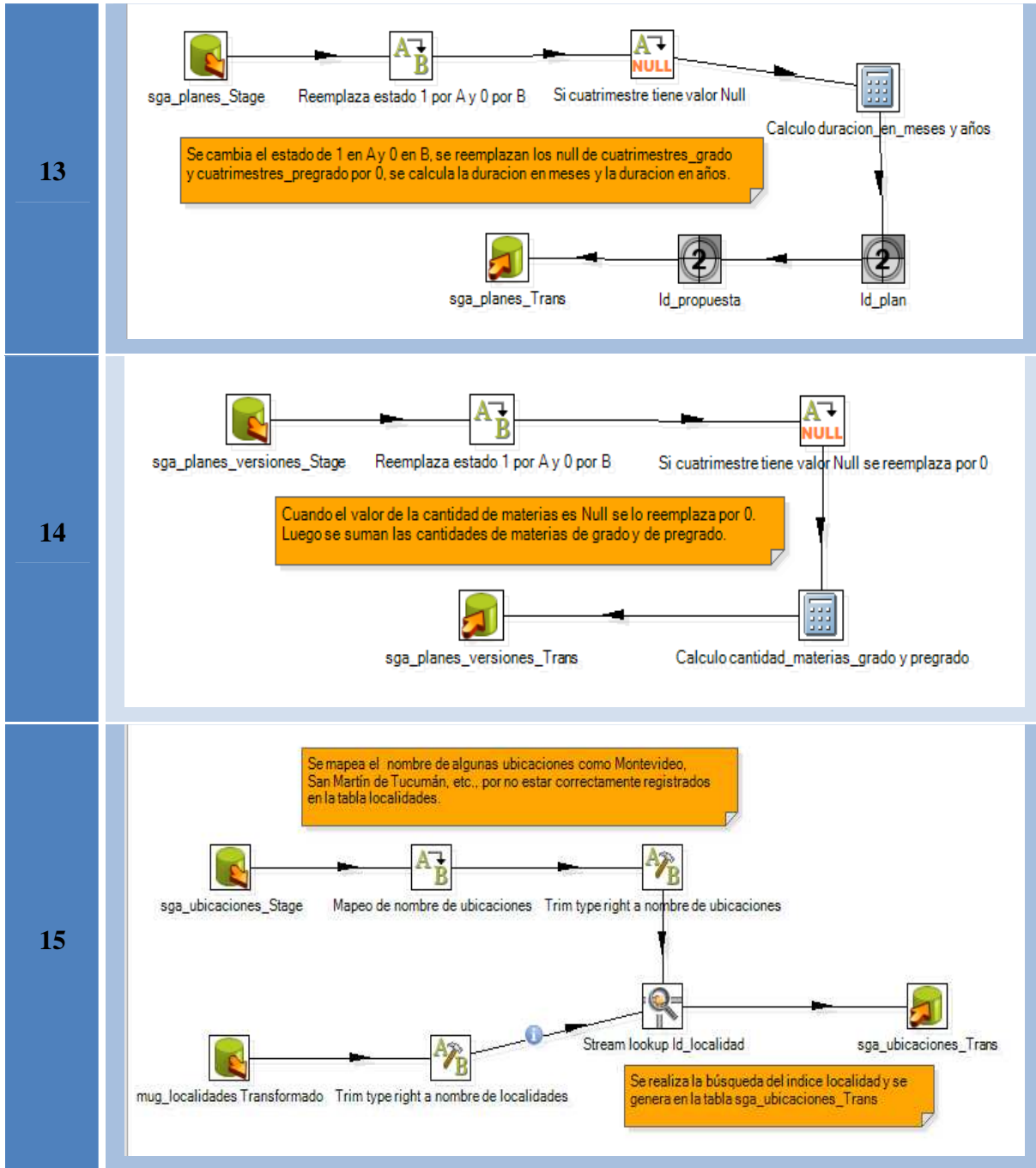
8.1.3.4 Transformaciones para Limpieza y Depuración de Datos

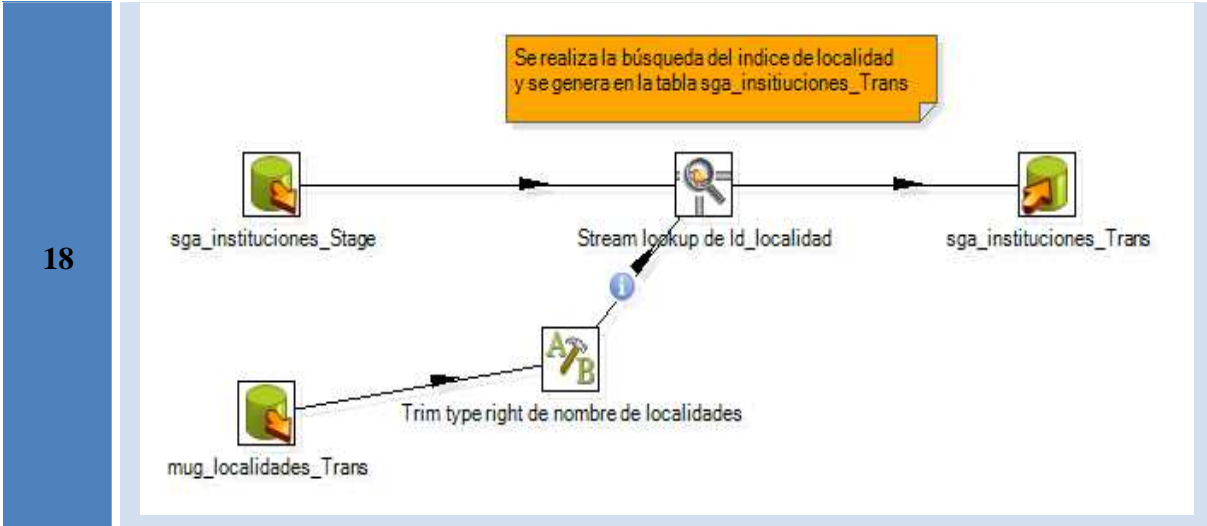
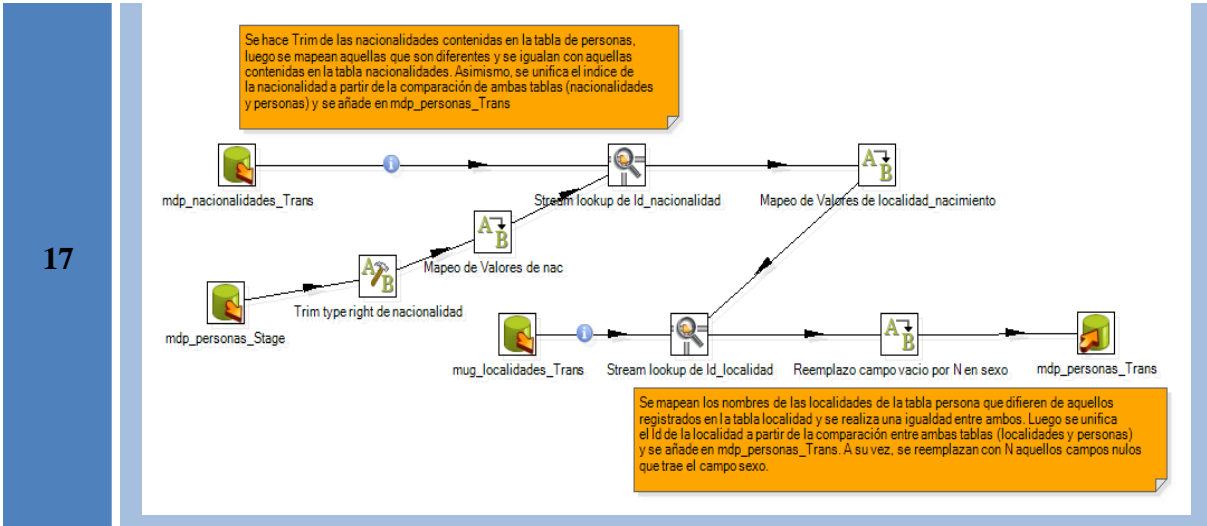
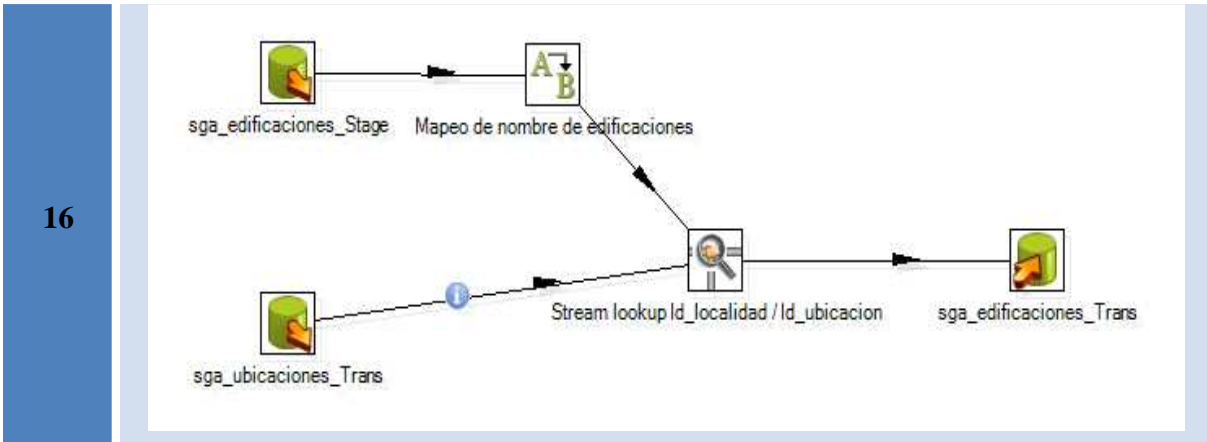


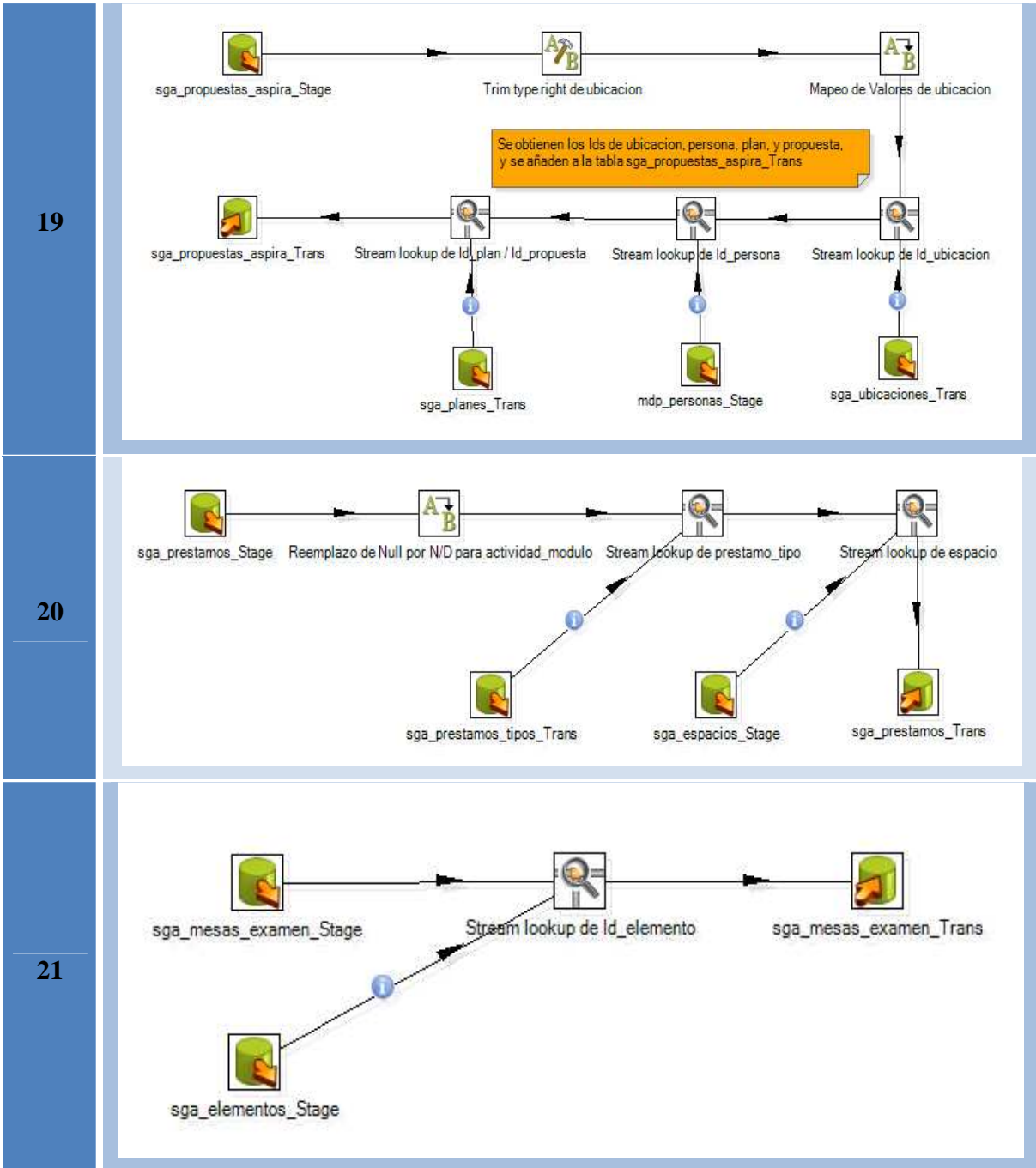


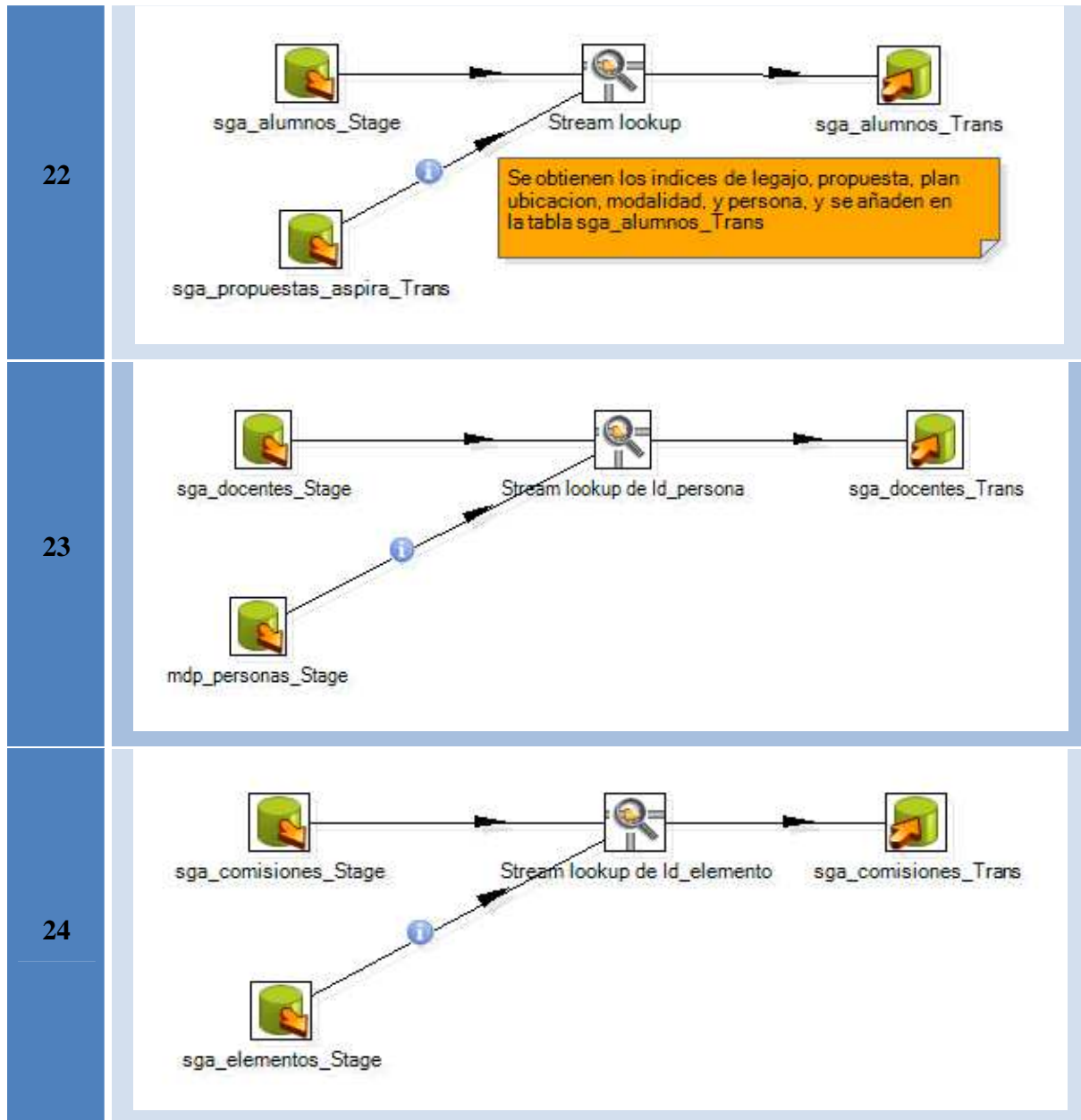


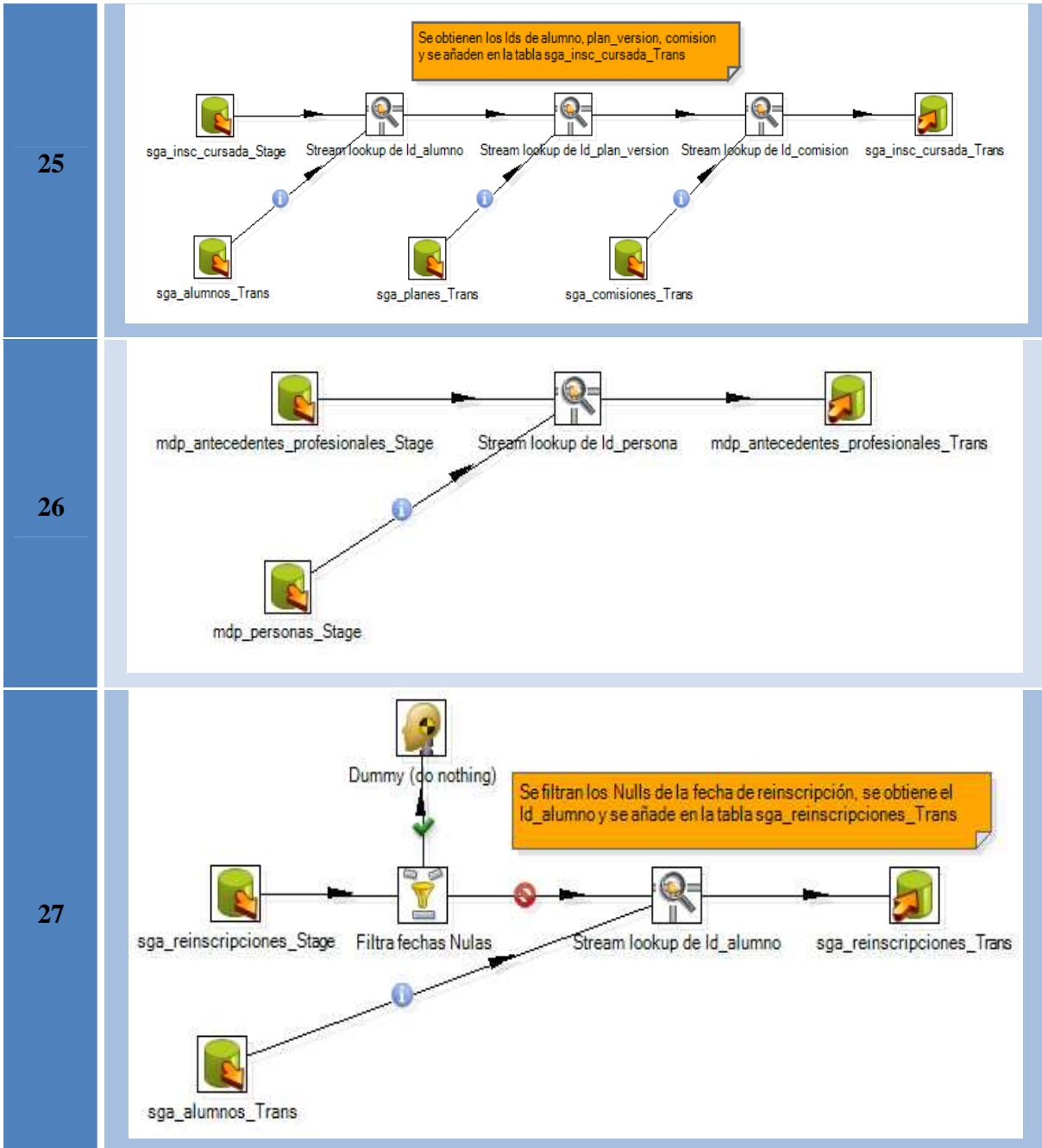


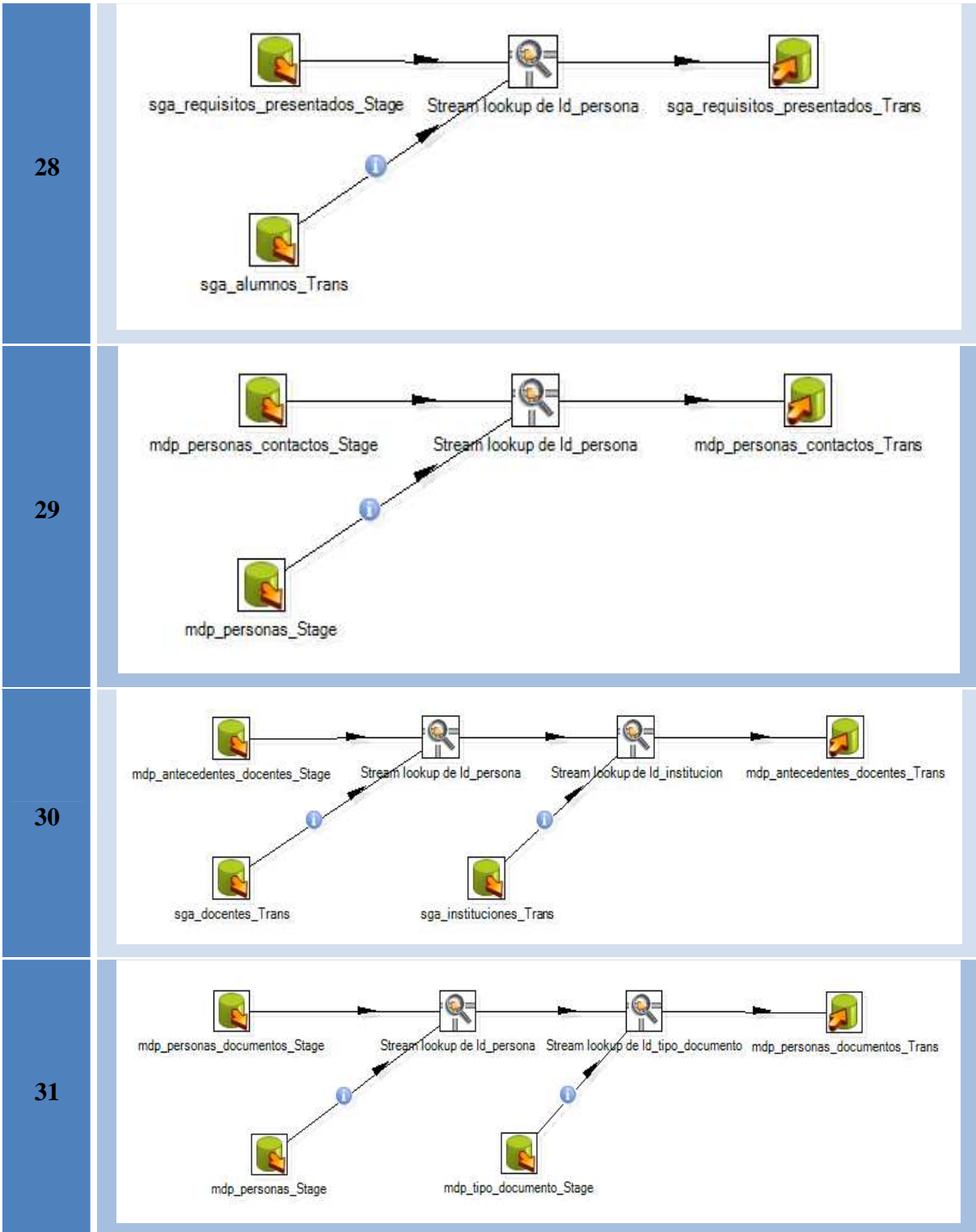


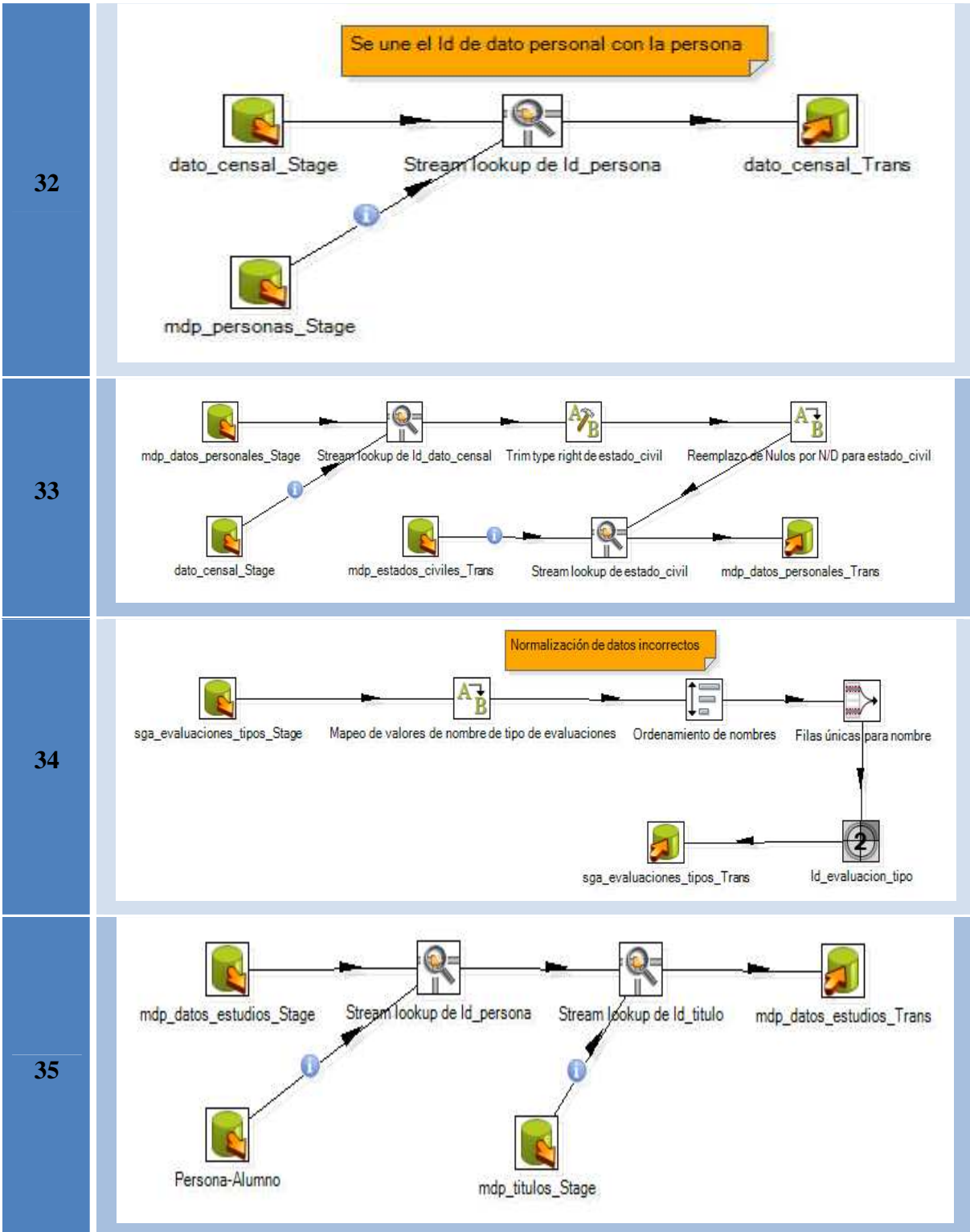


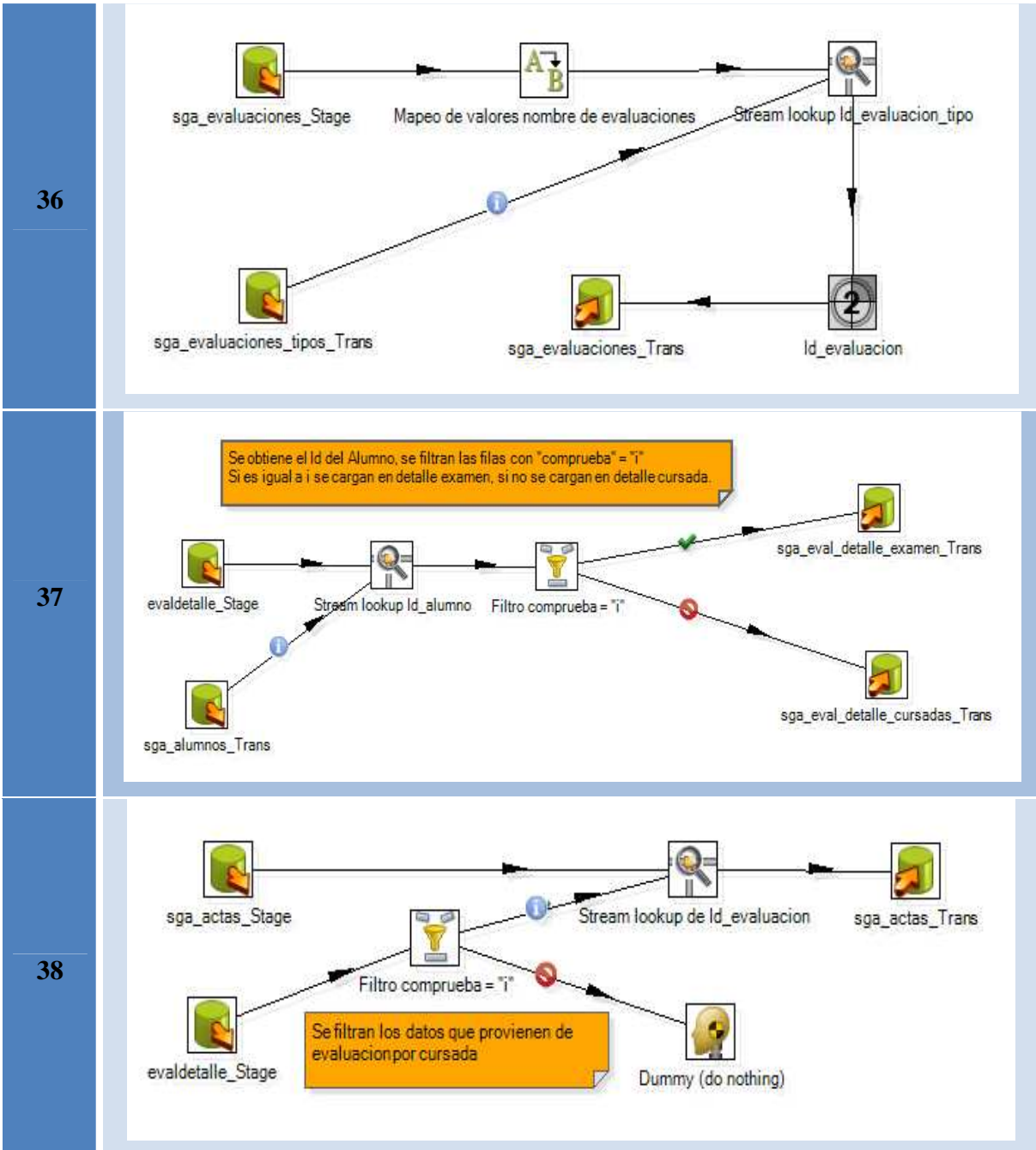












8.1.3.5 Ejemplos de Pasos ejecutados durante la Depuración y Limpieza de Datos

- *Ejemplo Edición del Paso **Entrada de Tabla** “sga_propuestas_Stage”.*

En la **imagen 8.1.3.5.a** se observa la consulta a la tabla sga_propuestas_Stage en la base de datos intermedia (Stage).

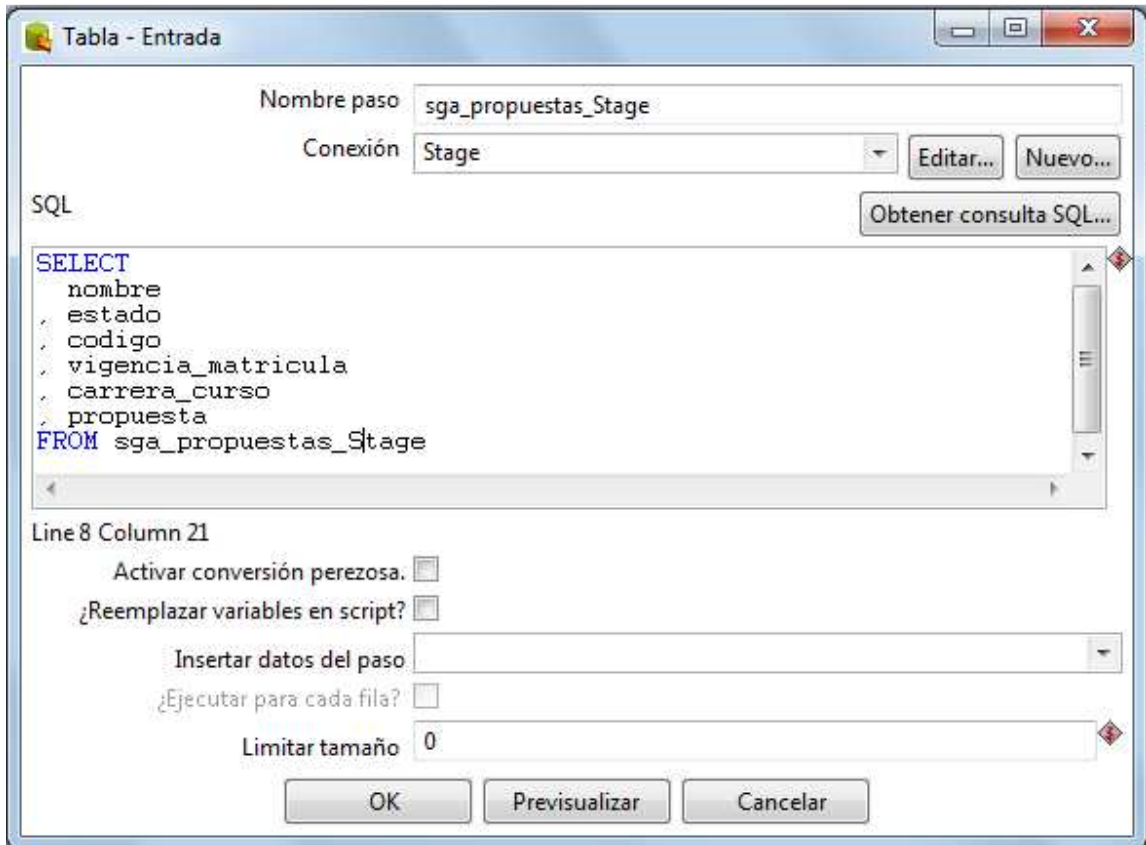


Imagen 8.1.3.5.a

- *Ejemplo Edición del Paso **Ordenar Filas**.*

En la **imagen 8.1.3.5.b** se observa el ordenamiento de filas de manera ascendente para el campo nombre de la tabla sga_prestamos_tipos_Stage.

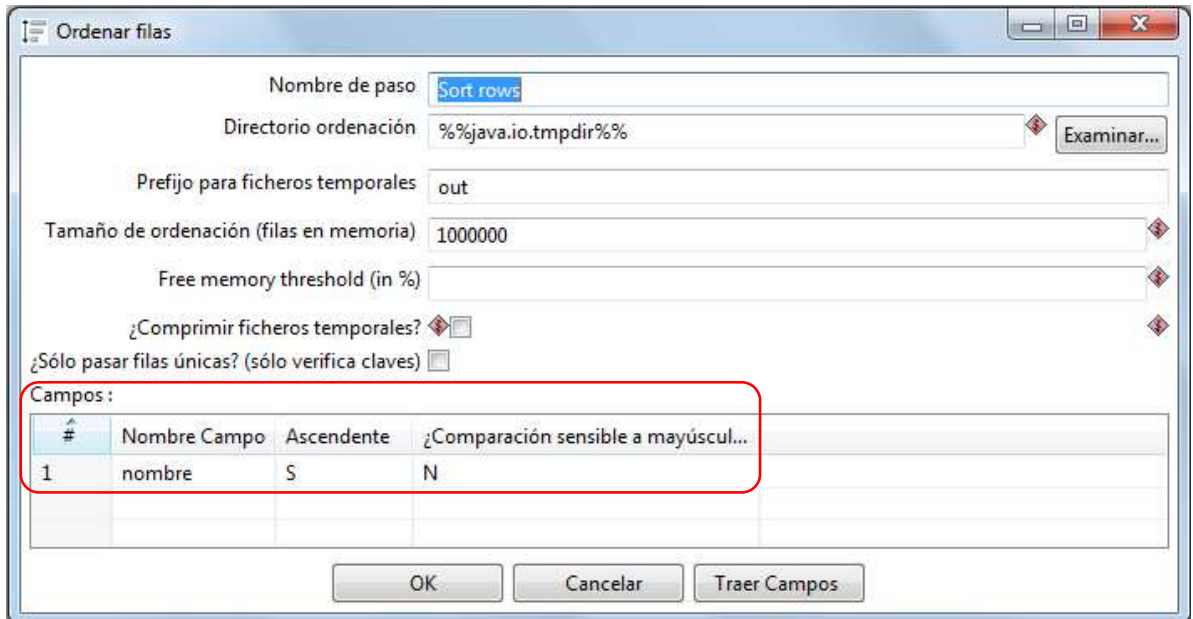


Imagen 8.1.3.5.b

- *Ejemplo Edición del Paso **Filas Únicas**.*

En la **imagen 8.1.3.5.c** se observa la aplicación de filas únicas para el campo periodo_generico de la tabla sga_periodos_genericos_Stage.

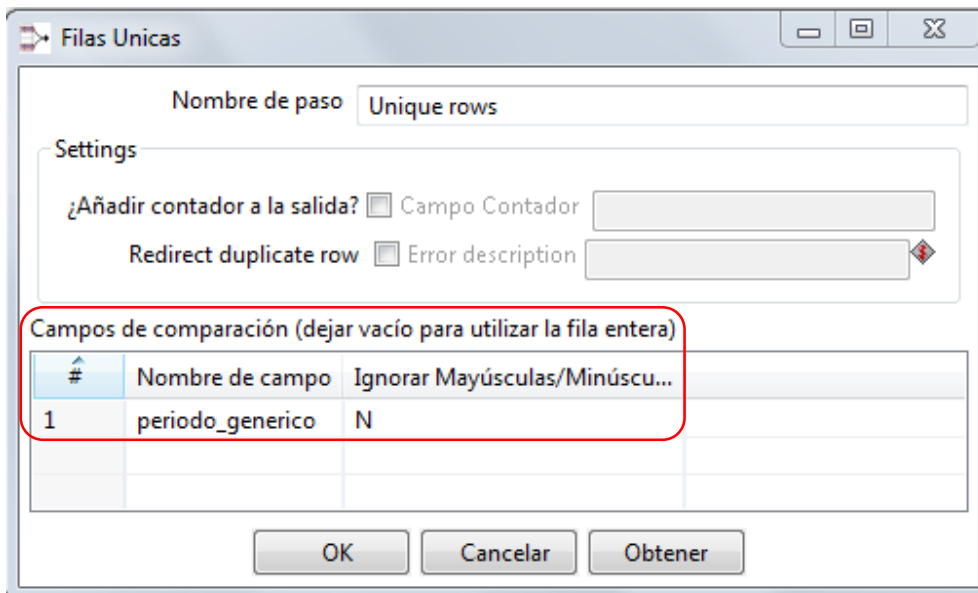


Imagen 8.1.3.5.c

- *Ejemplo Edición del Paso Operaciones con Cadenas.*

En la **imagen 8.1.3.5.d** se observa un recorte de espacios (trim type both) para el campo nombre que corresponde a la tabla mdp_orientacion_recibida_Stage.

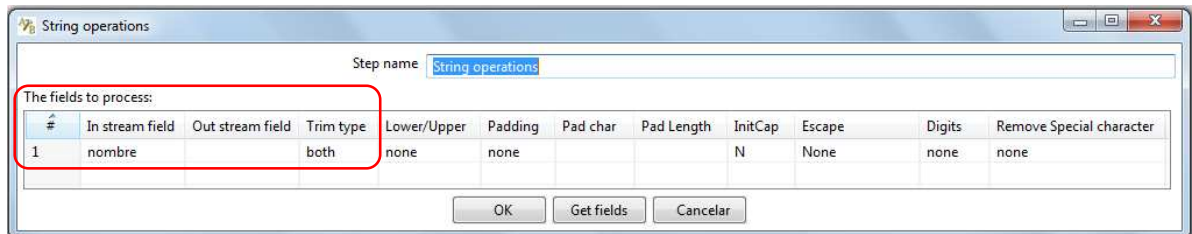


Imagen 8.1.3.5.d

- *Ejemplo Edición del Paso Mapeo de Valores.*

En la **imagen 8.1.3.5.e** se observa el mapeo para el estado de un plan de carrera, considerando valor origen y destino.

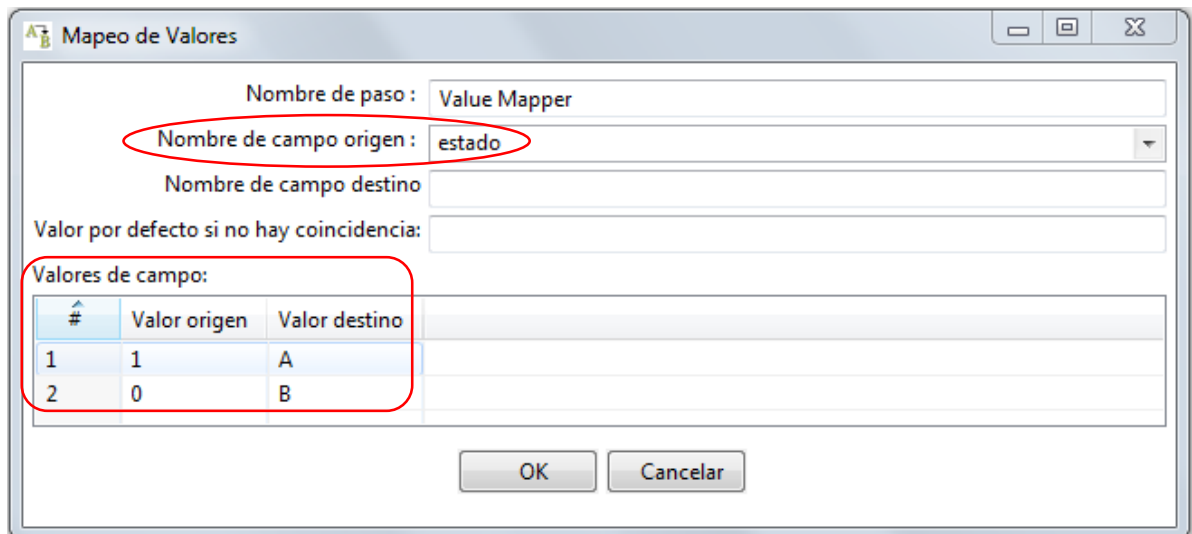


Imagen 8.1.3.5.e

- *Ejemplo Edición del Paso Añadir Secuencia para el índice "evaluacion".*

En la **imagen 8.1.3.3.f** se observa la generación del valor numérico secuencial: "evaluacion".

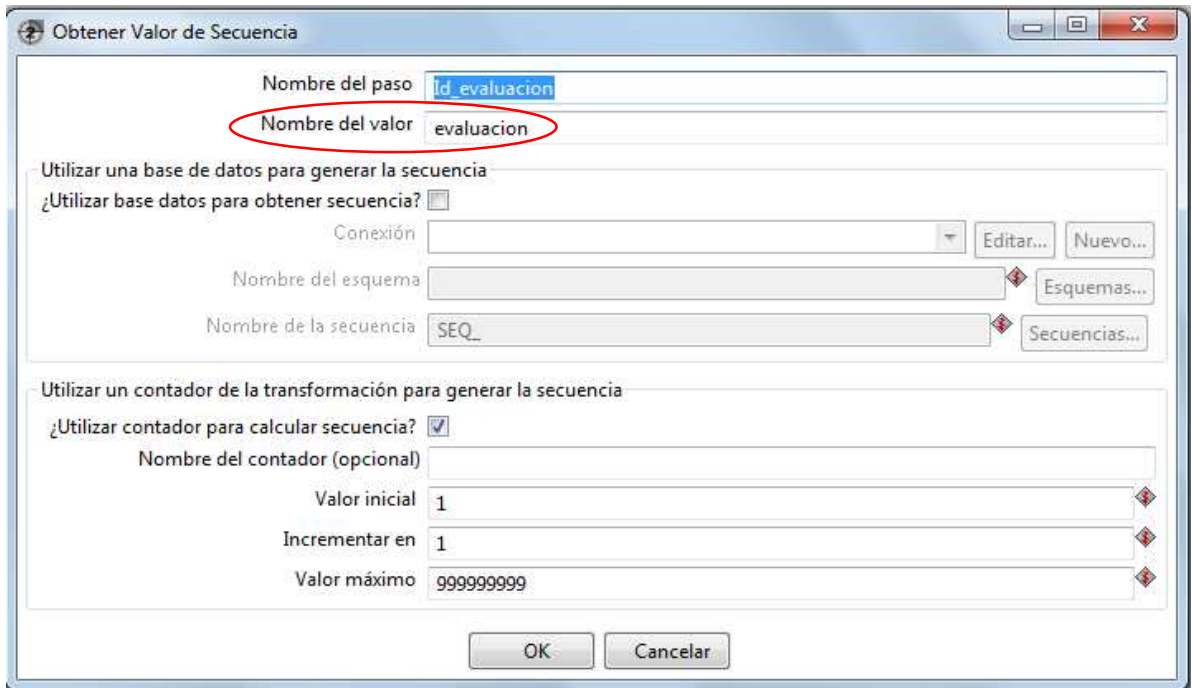


Imagen 8.1.3.5.f

- *Ejemplo Edición del Paso **Búsqueda en Flujo de Datos**.*

En la **imagen 8.1.3.3.g** se observa la devolución del campo propuesta_tipo, desde una búsqueda realizada a partir de ciertos valores.

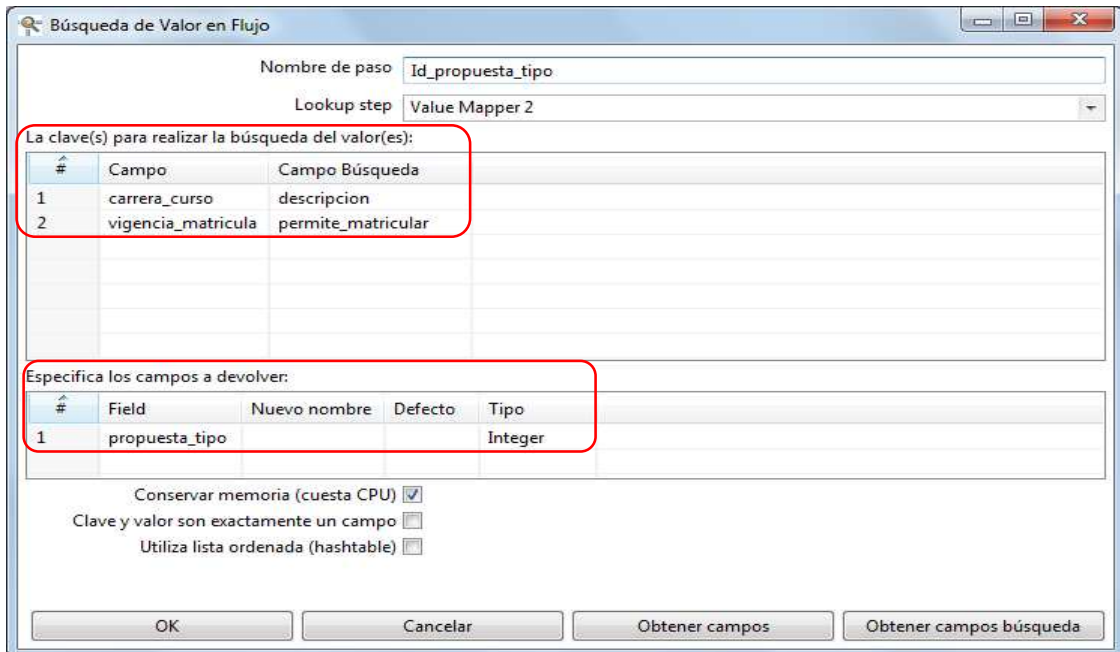


Imagen 8.1.3.5.g

- *Ejemplo Edición del Paso Si el Valor del Campo es Nulo*

En la **imagen 8.1.3.5.h** se observa el reemplazo con un valor 100 de aquellos valores nulos que trae el campo periodo_generico de la tabla sga_periodos_genericos_Stage.

Nombre del paso: If field value is null

Reemplazar valores nulos en todos los campos

Reemplazar con valor: 100

Máscara (Fecha):

Seleccionar campos:

Seleccionar tipos de valor:

Tipos de valor

#	Tipo	Reemplazar con valor	Máscara de conversión (Fecha)

Campos

#	Campo	Reemplazar con valor	Máscara de conversión (Fecha)
1	periodo_generico		

OK Traer Campos Cancelar

Imagen 8.1.3.5.h

- *Ejemplo Edición del Paso Filtrar Filas*

En la **imagen 8.1.3.5.i** se puede observar el filtrado de las fechas de reinscripción que tienen valor null. Este paso al detectar nulos no hace nada, es decir envía los datos al paso Dummy. Los valores que se identifican como no nulos siguen en la corriente de secuencia.

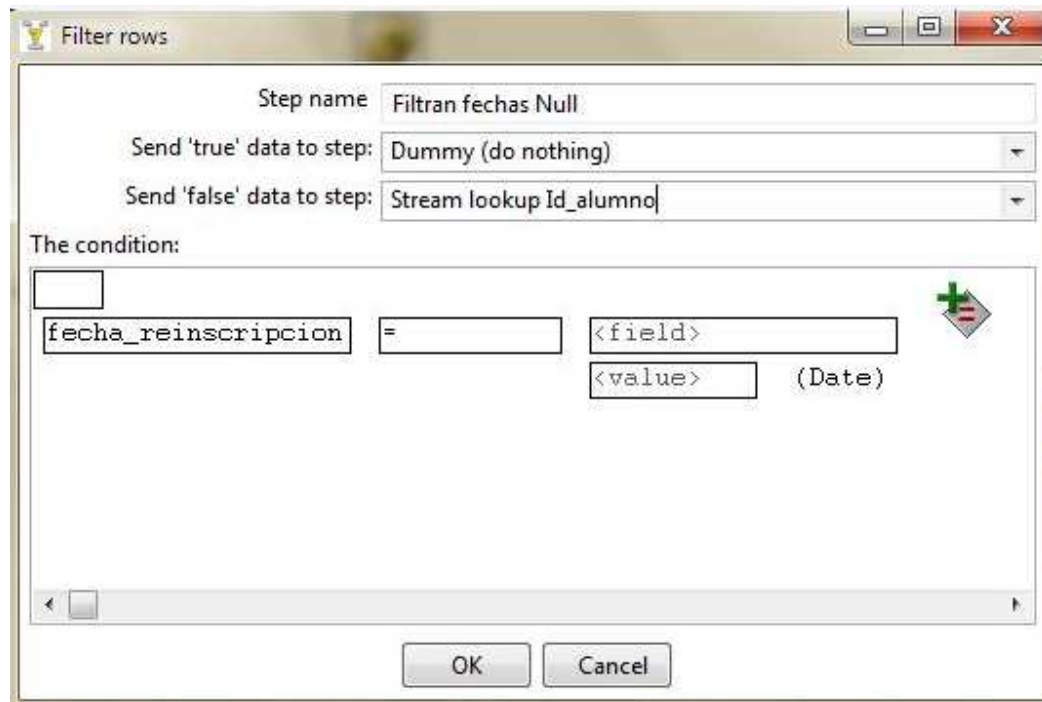


Imagen 8.1.3.5.i

- *Ejemplo Edición del Paso **Transformación Simulada** (No hace nada)*

En la **imagen 8.1.3.5.j** se observa que el paso no hace nada, solo recibe información que se presenta como flujo de datos falsos.

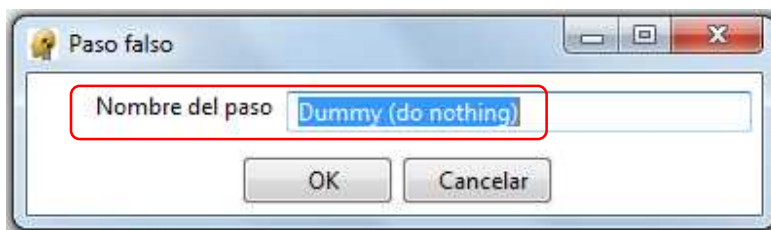


Imagen 8.1.3.5.j

- *Ejemplo Edición del Paso **Calculadora***

En la **imagen 8.1.3.5.k** se observa el resultado de un cálculo que se guarda en un nuevo campo y que surge de sumar la cantidad de materias de grado con la cantidad de materias de pregrado.

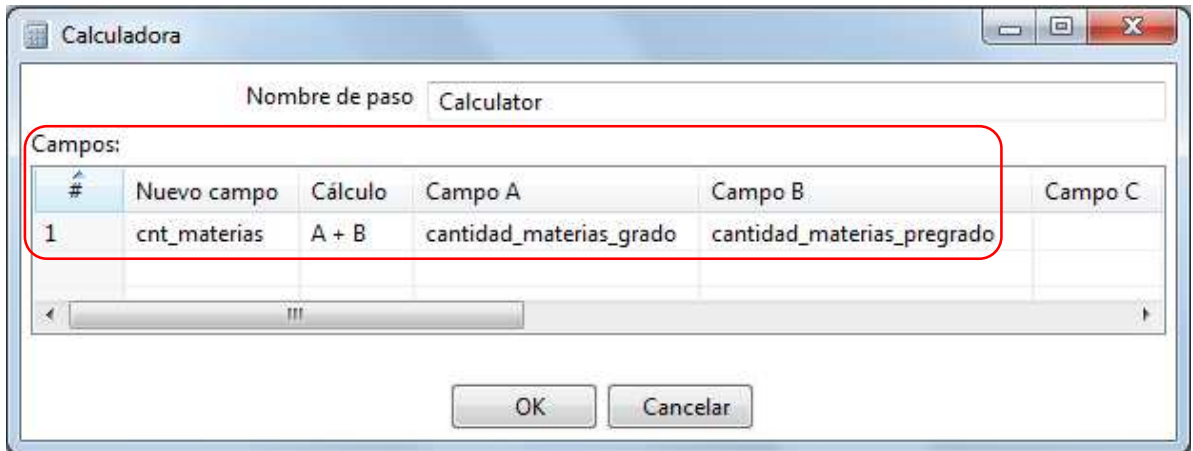


Imagen 8.1.3.5.k

- *Ejemplo Edición del Paso Salida de Tabla “sga_actas_Trans”.*

En la **imagen 8.1.3.5.l** se observa la conexión que se establece a la base de datos Stage donde se insertaran los datos ya transformados.

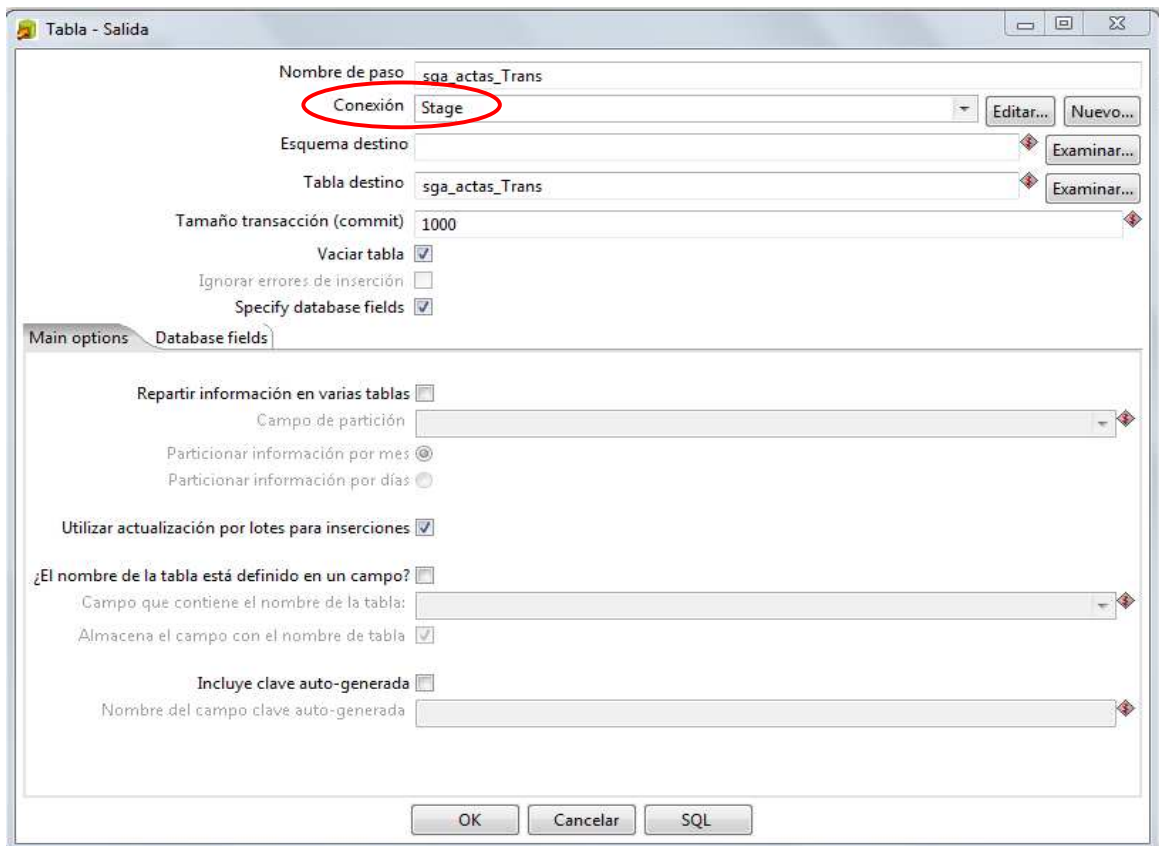


Imagen 8.1.3.5.l

En la **imagen 8.1.3.5.m** se observan los campos transformados para insertar en la base de datos Stage.

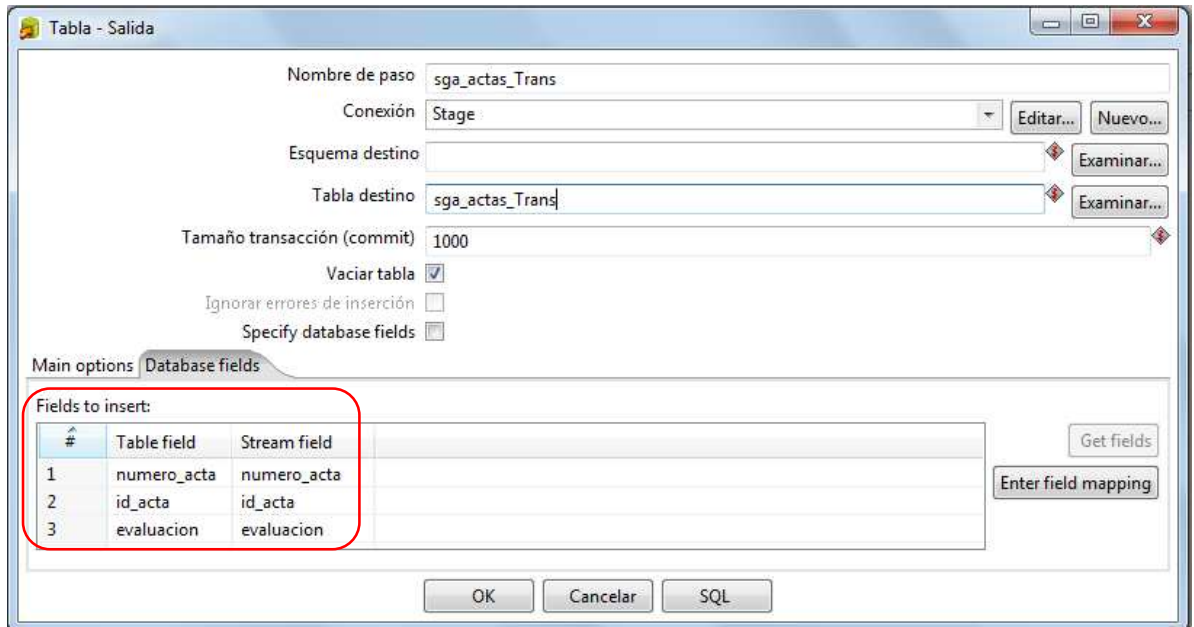


Imagen 8.1.3.5.m

Cabe aclarar que todos los ejemplos ilustrados corresponden a diferentes transformaciones de la etapa de depuración y limpieza de datos.

8.1.3.6 Trabajo para Depuración y Limpieza de Datos

A continuación se ilustra en la **Fig. 8.1.3.6** el Trabajo planificado en Kettle que ejecuta automáticamente y en orden secuencial cada una de las transformaciones para Depuración y Limpieza de Datos:

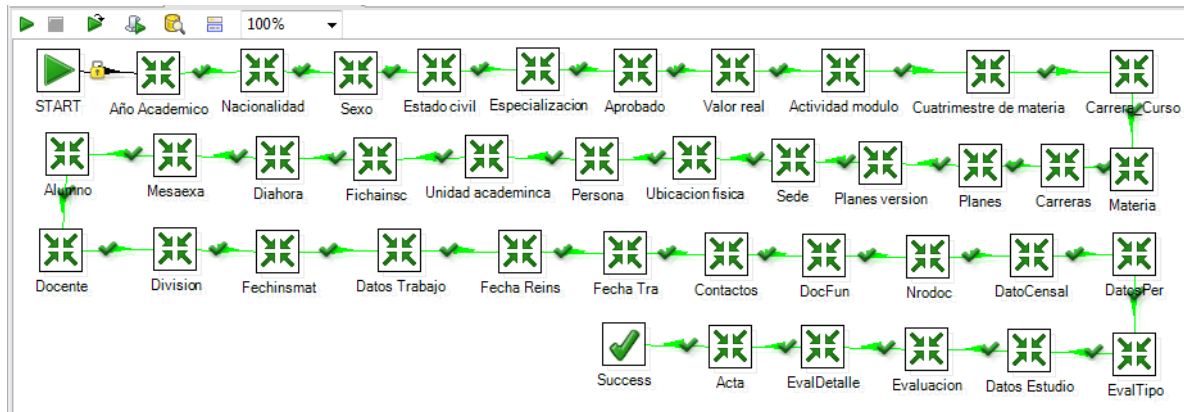



Fig. 8.1.3.6

El resultado de la ejecución del trabajo previamente ilustrado quedará demostrado en la etapa de Pruebas, en el caso de test que aplique al mismo.


Transformaciones de Carga y Control de Cambios

A continuación se ilustran los Steps de Kettle que se utilizarán en las transformaciones de Carga y Control de Cambios en PostgreSQL (SIU-Guarani3):


Entrada de Tabla

 <p>Entrada Tabla BD Stage</p>	<p>Este objeto lee información desde una base de datos PostgreSQL.</p> <p>Se utiliza para efectuar el SELECT que extraerá información de la BD Intermedia para su posterior carga y control de cambios en la BD de SIU-Guarani3.</p>
---	--


Mapeo de Valores

 <p>Mapeo de Valores</p>	<p>Este objeto establece una conversión de valores de un campo a otro valor.</p> <p>Lo utilizaremos para efectuar el mapeo de aquellos datos que necesitarán cargarse en la BD de SIU-Guarani3 con otro valor. Por ejemplo, datos que se insertaran en PostgreSQL en campos diferentes y con valores diferentes de los originales.</p>
---	--


Operaciones con Cadenas

 <p>String operations</p>	<p>Este objeto permite aplicar ciertas operaciones, como el recorte, relleno y otros, al valor de una cadena.</p> <p>Lo utilizaremos para recortar los espacios en blanco que contienen algunas cadenas de datos para que puedan cargarse correctamente en la BD de SIU-Guarani3. Por lo general, el recorte se efectúa a la derecha de la cadena y se aplicará únicamente para las tablas sga_acciones_Stage y mdp_nacionalidades_Trans.</p>
--	---


Selecciona / Renombra Valores

 <p>Selecciona/Renombra valores</p>	<p>Este objeto permite seleccionar o eliminar campos de una fila. Opcionalmente, permite establecer la meta-información del campo: tipo, longitud, y precisión.</p> <p>Lo utilizaremos para modificar el tipo de dato del campo seleccionado, que se origina en una tabla de la BD Stage y se requiere en la BD de SIU-Guarani3 con un tipo de dato diferente.</p>
--	--


Ordenar Filas

 <p>Ordenar filas</p>	<p>Este objeto ordena filas basado en valores de campos.</p> <p>Lo utilizaremos para ordenar filas de manera ascendente. Este paso es requerido durante la carga de datos para poder considerar en forma ordenada la información a transferir.</p>
--	--


Filas Únicas

 <p>Filas Unicas</p>	<p>Este objeto permite la eliminación de filas duplicadas y deja solo ocurrencias únicas. Funciona solo con entradas ordenadas.</p> <p>Lo utilizaremos en tablas de la BD Stage para seleccionar valores únicos. Por lo general, se aplica para aquellas tablas que no se encuentran correctamente normalizadas en la BD del IUA y deben transformarse para respetar las formas normales que requiere PostgreSQL para SIU-Guarani3.</p>
---	---


Si el Valor del Campo es Nulo

 <p>If field value is null</p>	<p>Este objeto permite establecer un valor de campo a una constante si es nulo.</p> <p>Lo utilizaremos para sustituir a los nulos en aquellos campos que requieran de manera indispensable un valor que no sea nulo. En las transformaciones de carga solamente se usará para las tablas: sga_responsables_academicas_Stage, sga_actas_estados_Stage y sga_libros_actas_Stage.</p>
---	--


Filtrar Filas

 <p>Filtrar filas</p>	<p>Este objeto permite filtrar filas utilizando fórmulas sencillas.</p> <p>Lo utilizaremos con el fin de dividir una sola corriente de información en diferentes corrientes más pequeñas. La información resultante del filtro aplicado se puede traducir a datos “Verdaderos” o datos “Falsos” dependiendo del objetivo de la transformación.</p>
--	--


Transformación simulada (no hace nada)

 <p>Transformación Simulada (no hace nada)</p>	<p>Pasos de este tipo no hacen nada. Es útil en ciertas situaciones donde es necesario partir flujos de información.</p> <p>Por lo general se aplicará cuando sea utilizado el paso “Filtrar Filas” que deberá conectarse al mismo y recibir de este los flujos de información que se presenten como datos “Falsos”.</p>
---	--


Fundir Filas

 <p>Fundir filas</p>	<p>Este objeto permite fundir dos flujos de filas, ordenadas por un campo clave.</p> <p>Lo utilizaremos para comparar dos flujos de información y señalar las filas que son iguales, han cambiado, se han eliminado y las nuevas, permitiendo gestionar correctamente el control de cambios.</p>
--	--


Synchronize after merge

 <p>Synchronize after merge</p>	<p>Este objeto permite realizar la operación de inserción / actualización / eliminación de una sola vez en función del valor de un campo.</p> <p>Lo utilizaremos como una salida de flujo, en cuyo paso se ejecutará la tarea de insertar / actualizar / eliminar los datos en las tablas de la BD de SIU-Guarani3.</p>
--	---

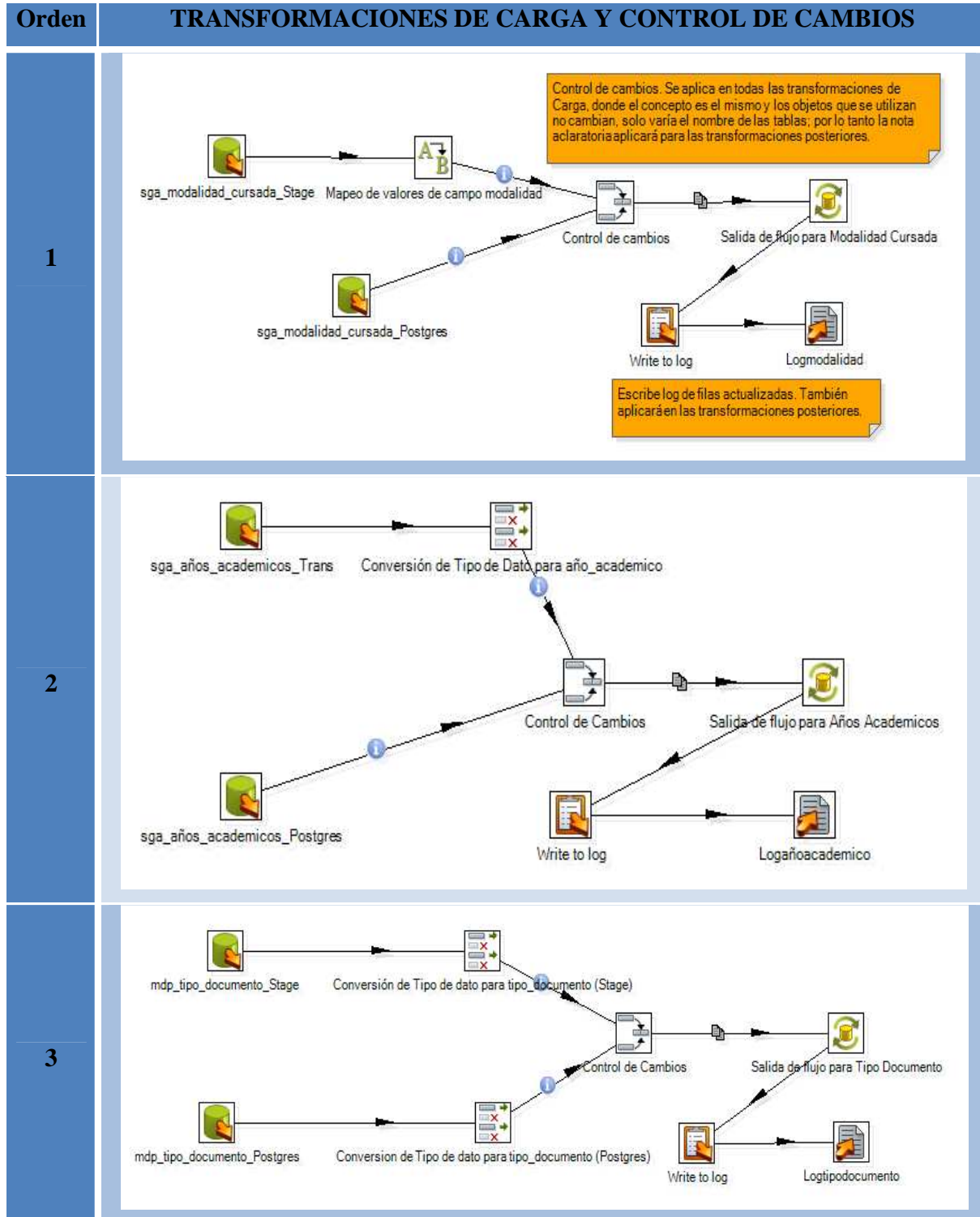
Write data to log

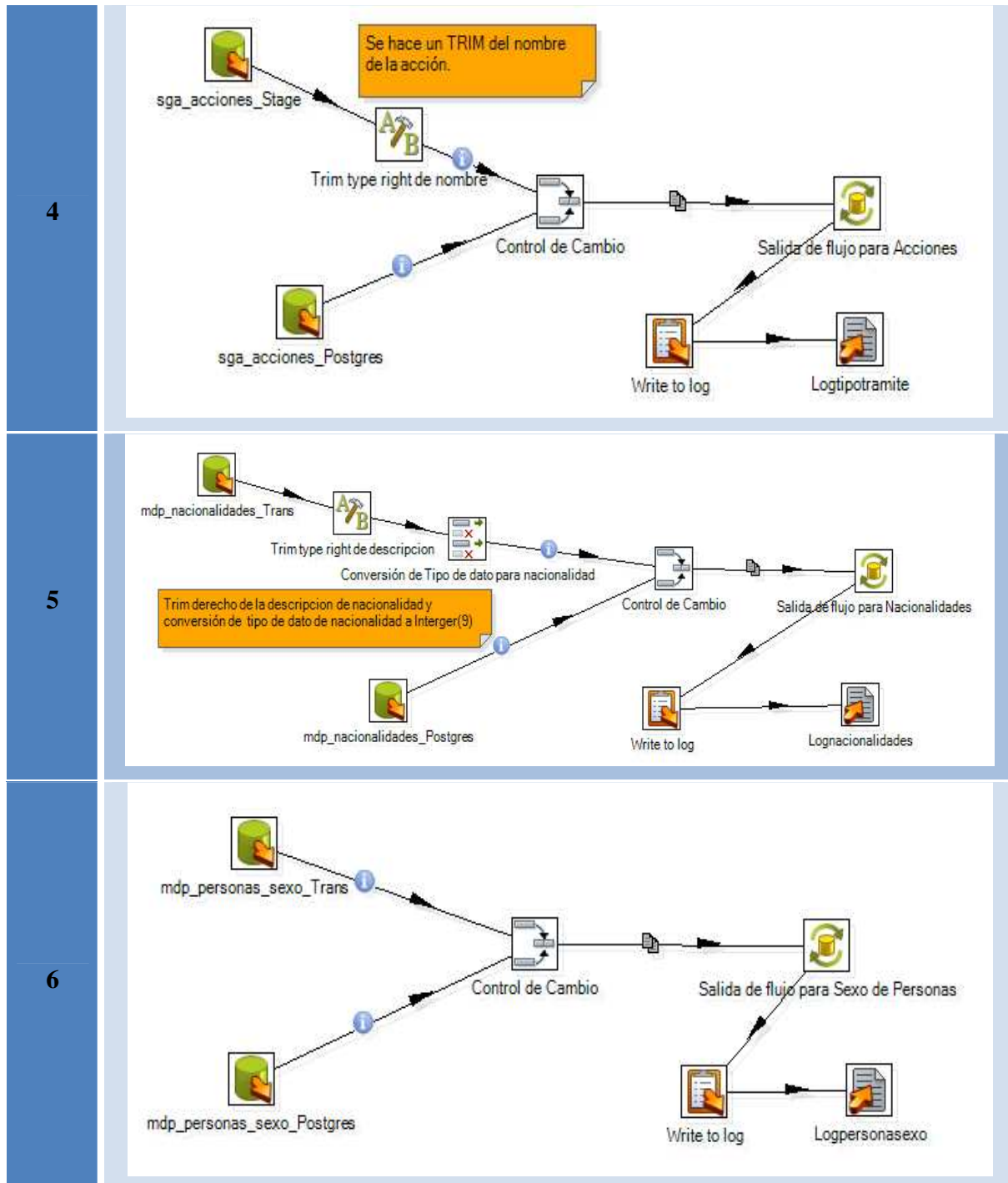
 <p>Write to log</p>	<p>Este objeto permite escribir los datos migrados a un log.</p> <p>Lo utilizaremos para escribir los datos de carga a un log que se podrá visualizar en un archivo de formato plano a través de un objeto de Salida de Fichero de Texto.</p>
---	---

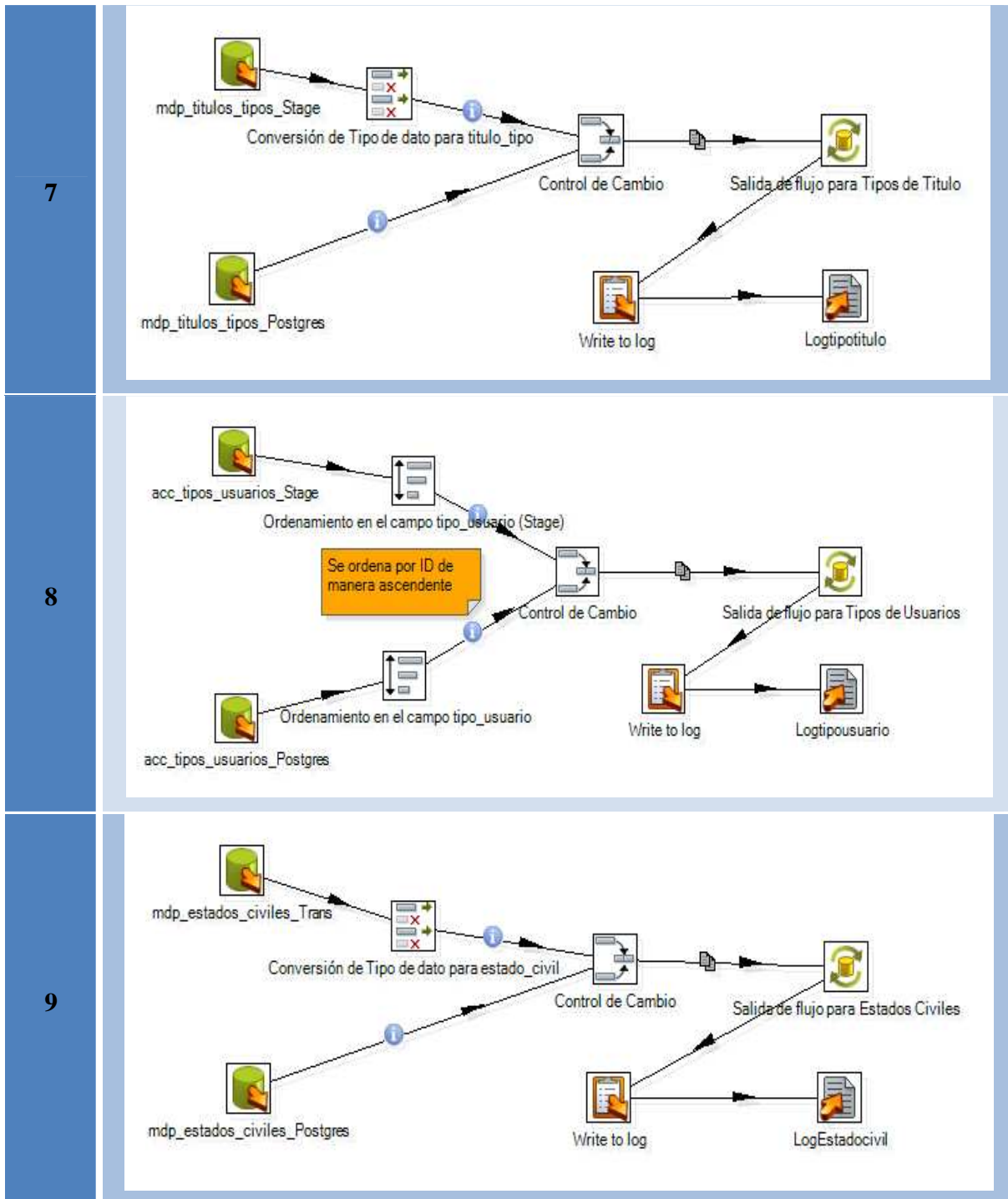
Salida Fichero de Texto

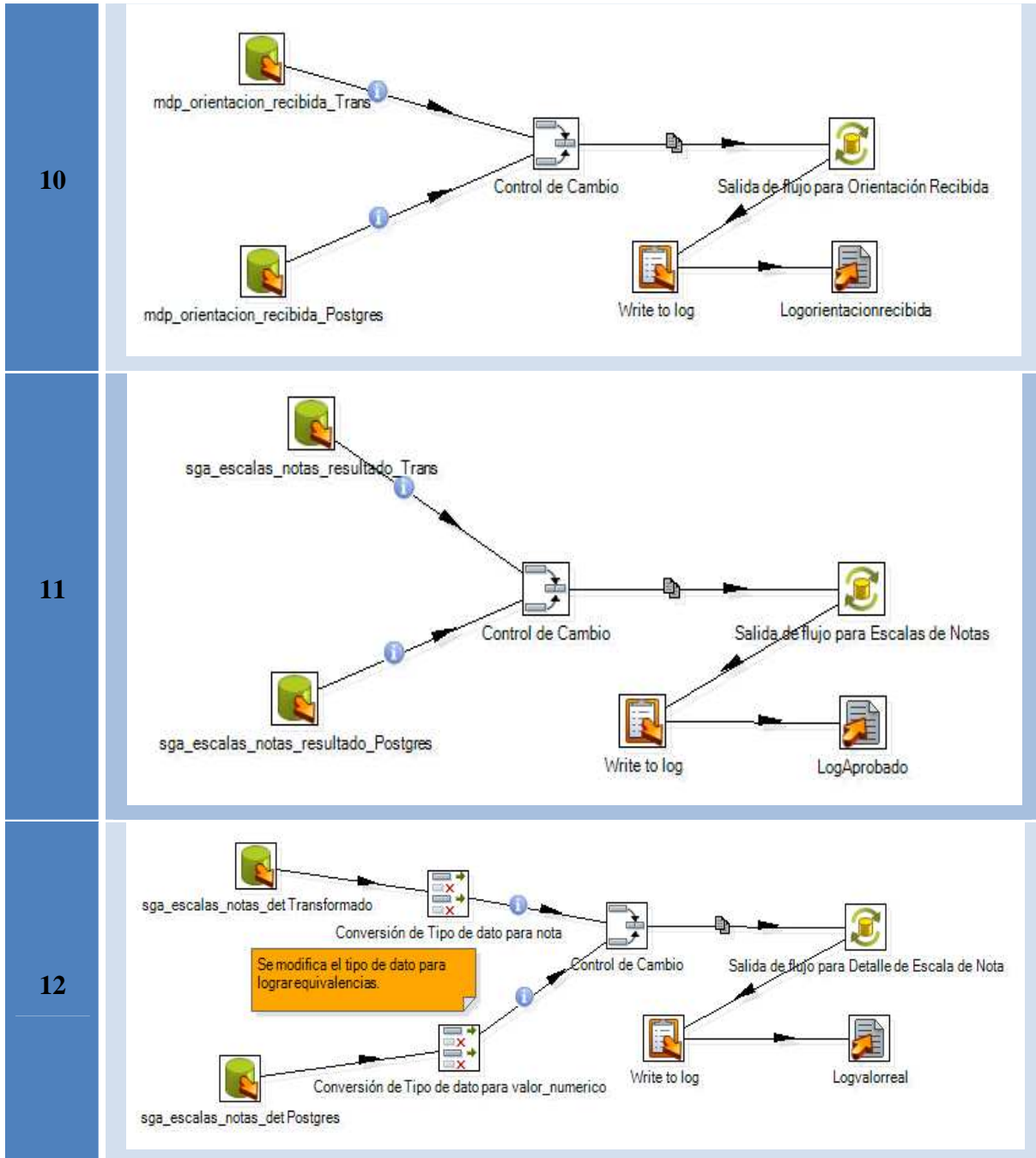
 <p>Salida Fichero de Texto</p>	<p>Este objeto permite una salida de datos a través de un Fichero de Texto.</p> <p>Lo utilizaremos para escribir filas que se registran en el objeto Write to log con el objeto de poder visualizar los resultados en un fichero de texto.</p>
--	--

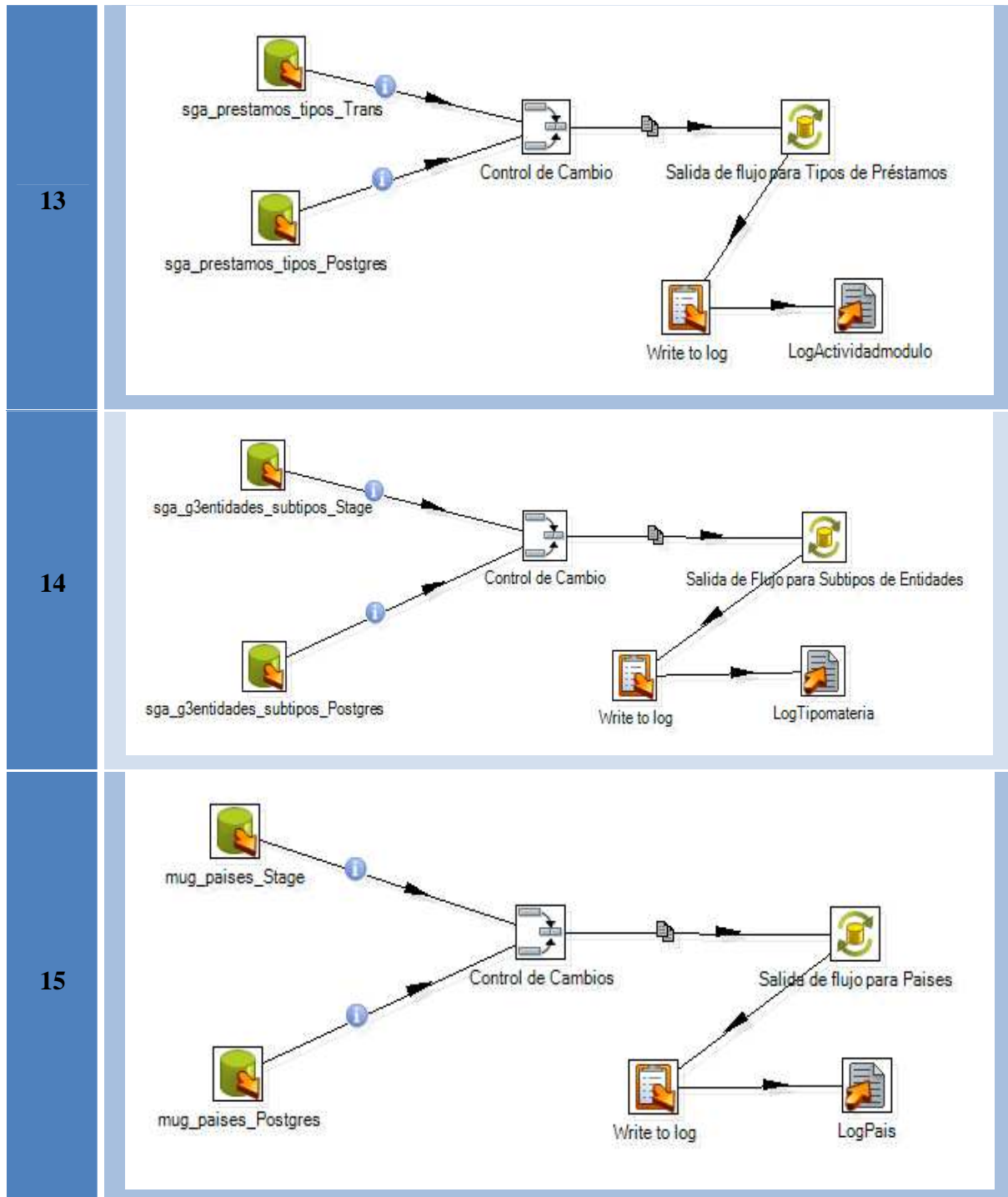
8.1.3.7 Transformaciones para Carga y Control de Cambios

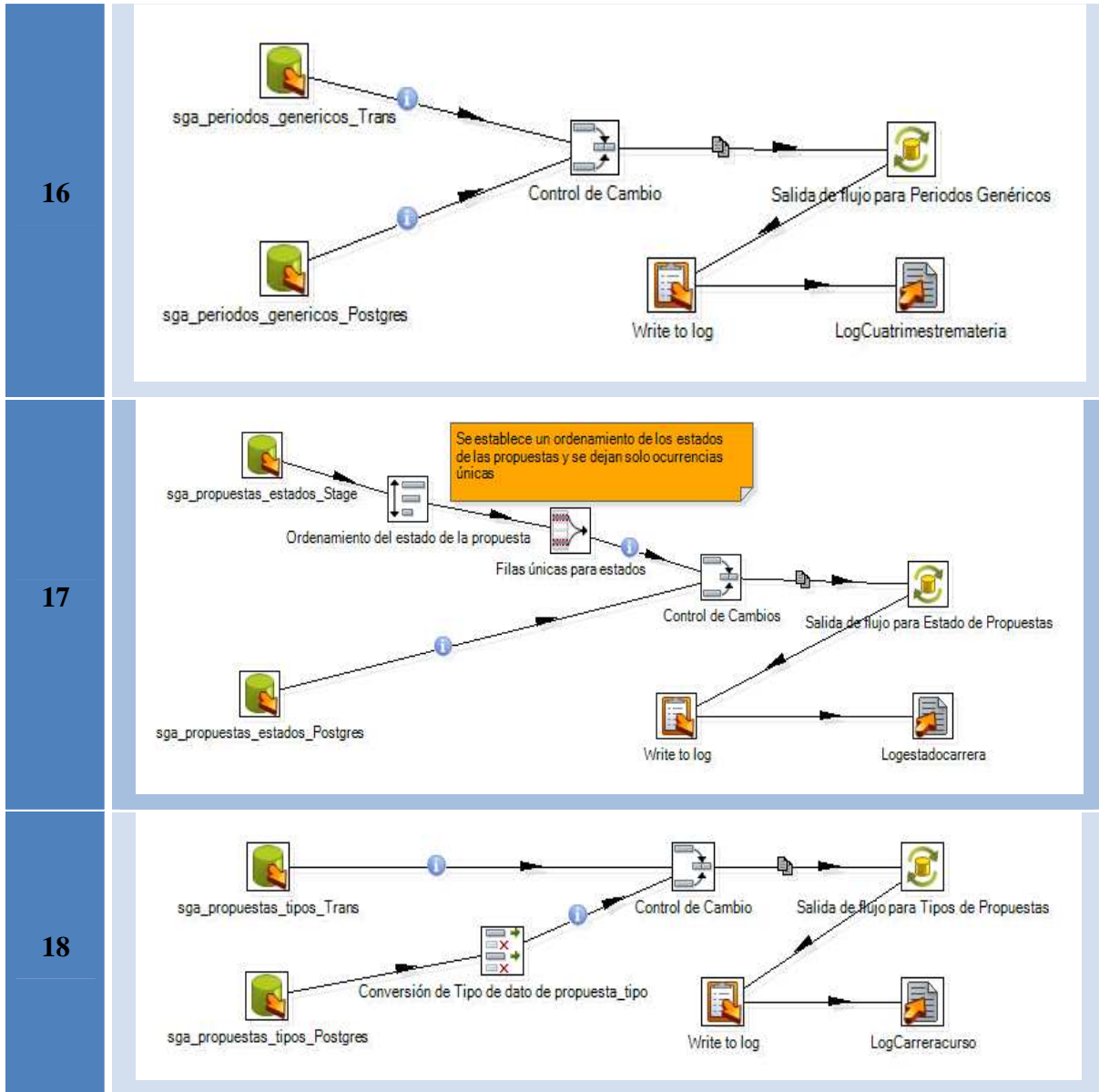


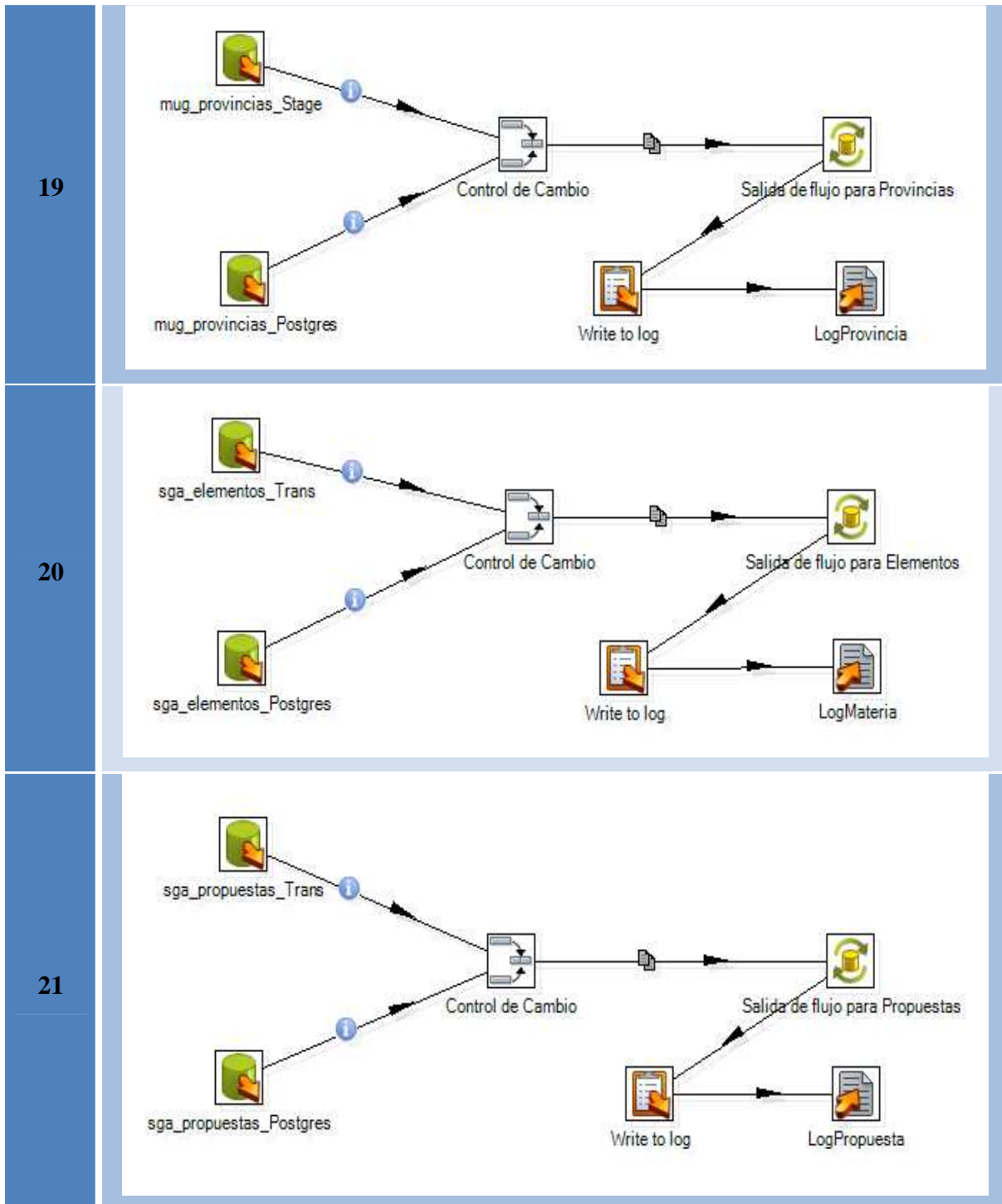


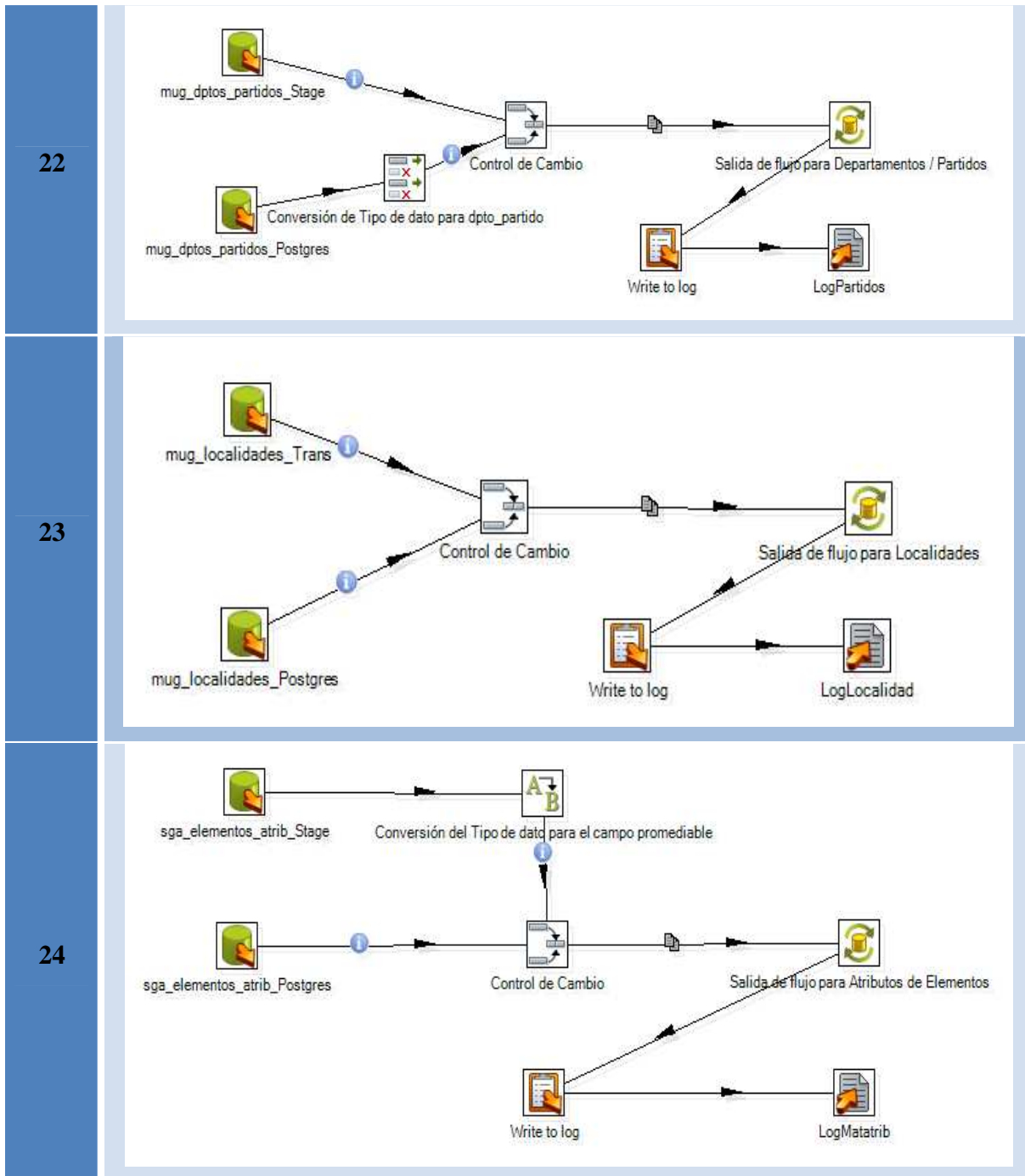


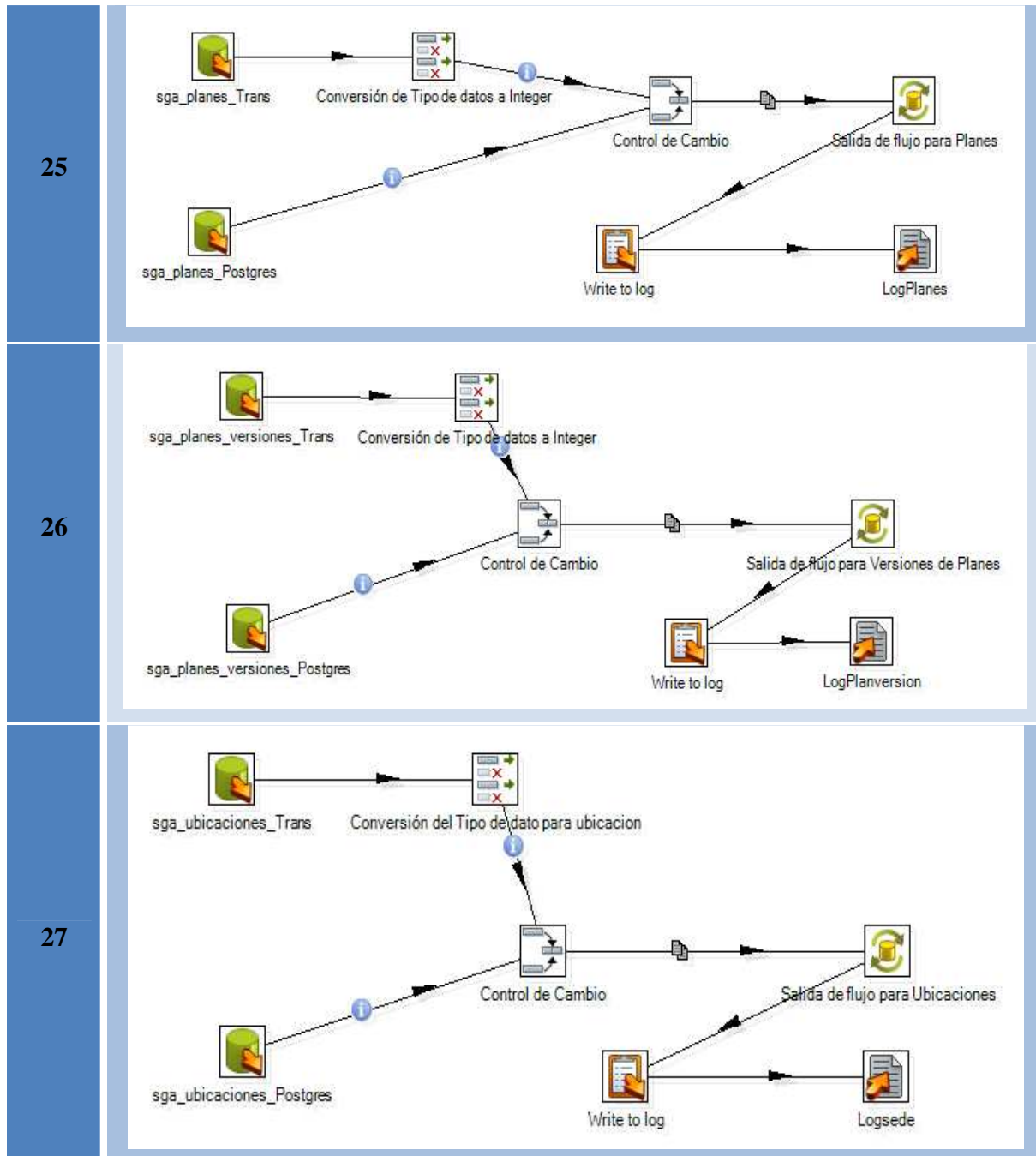


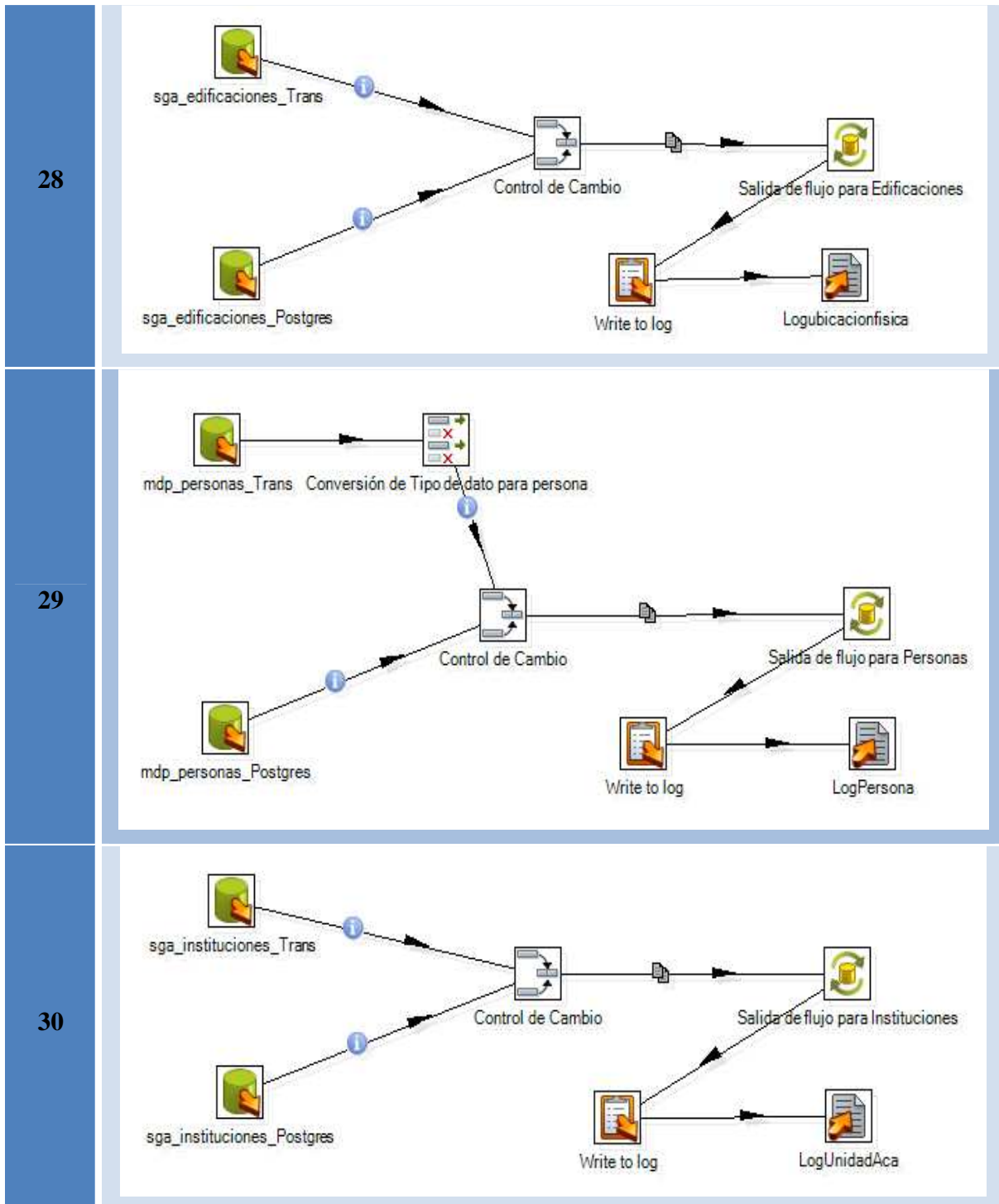


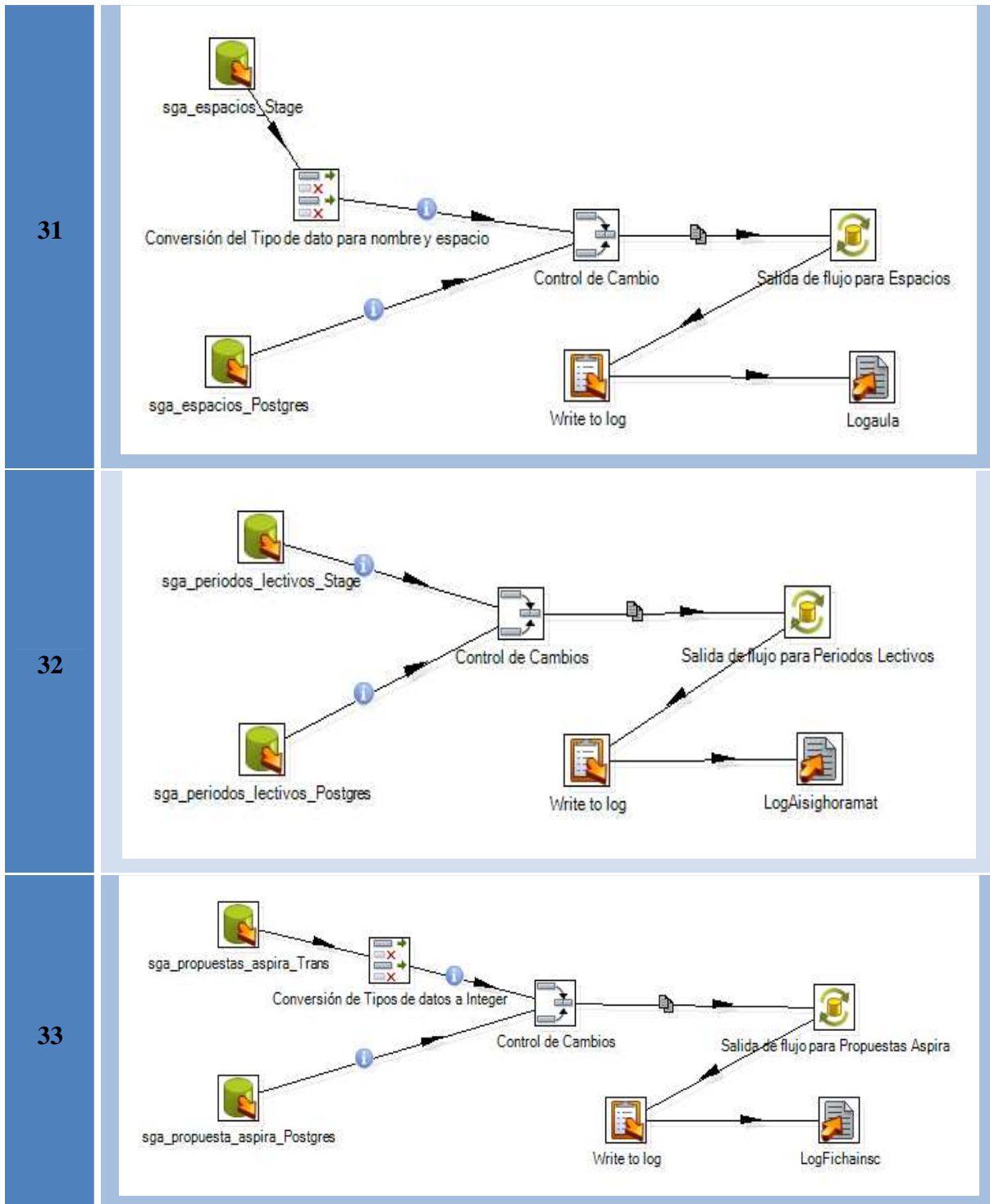


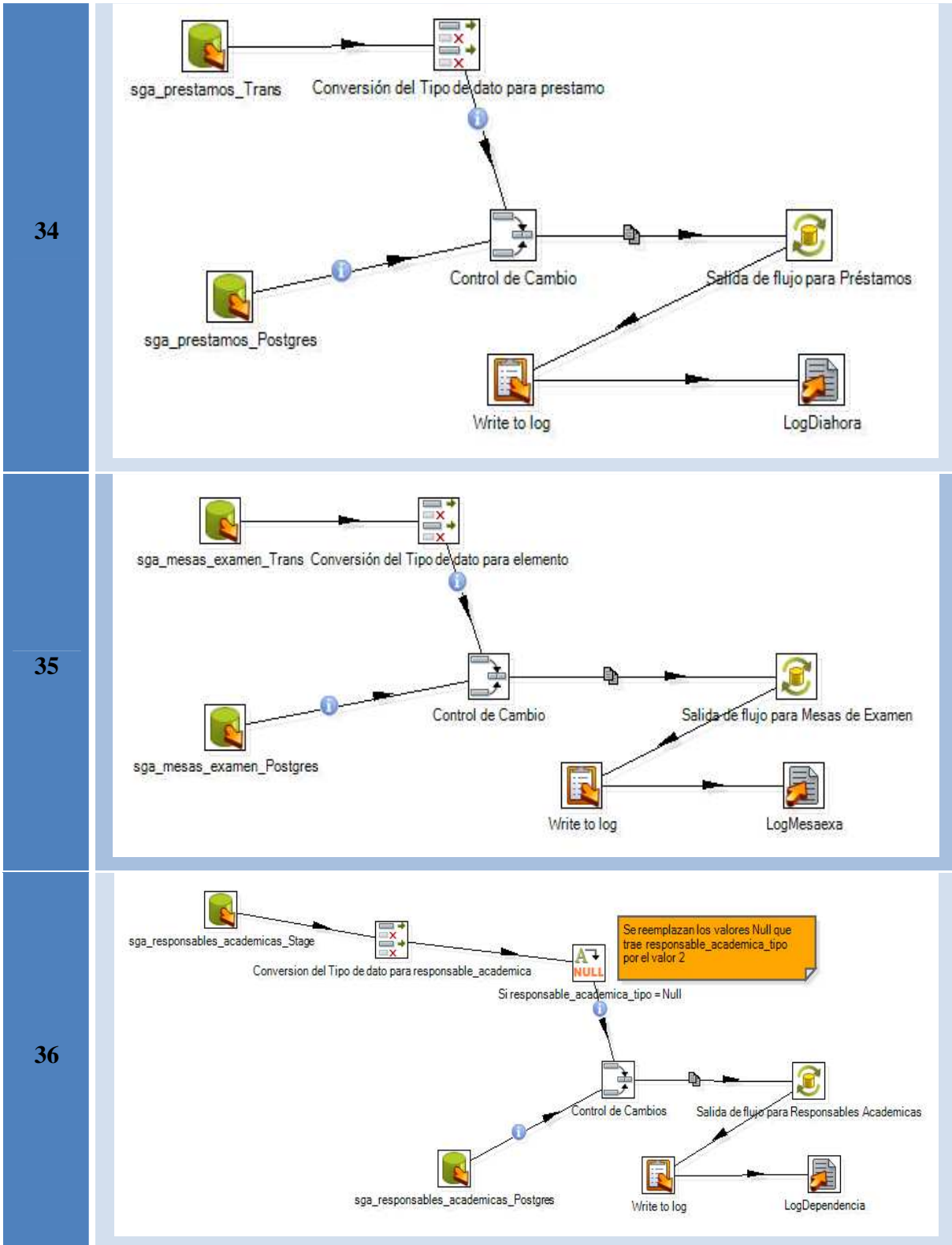




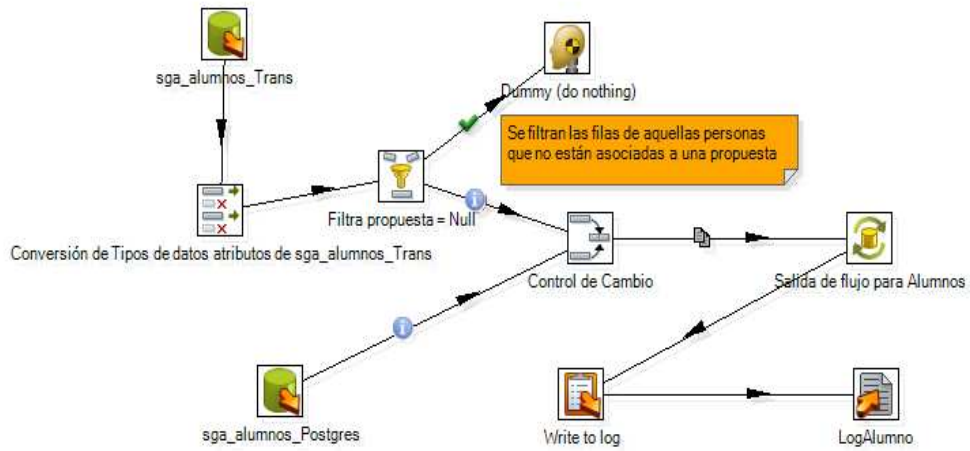




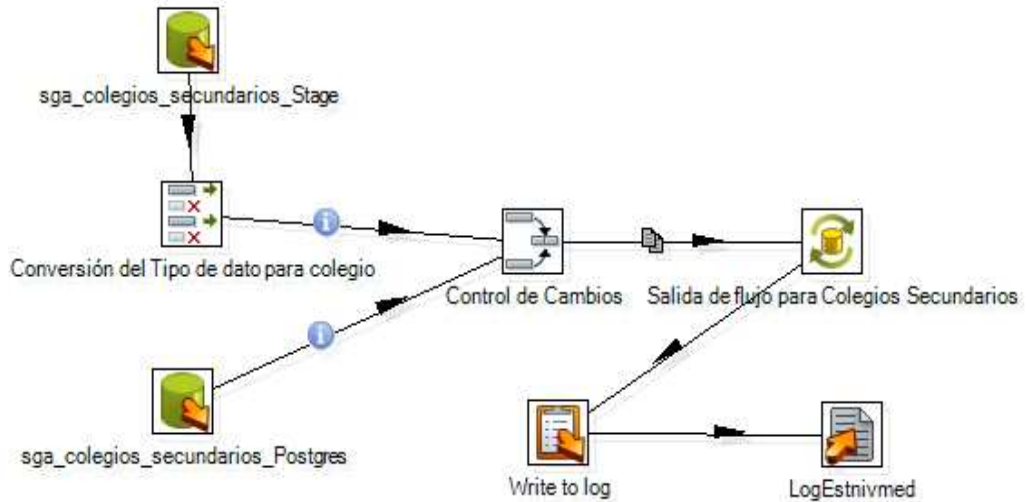




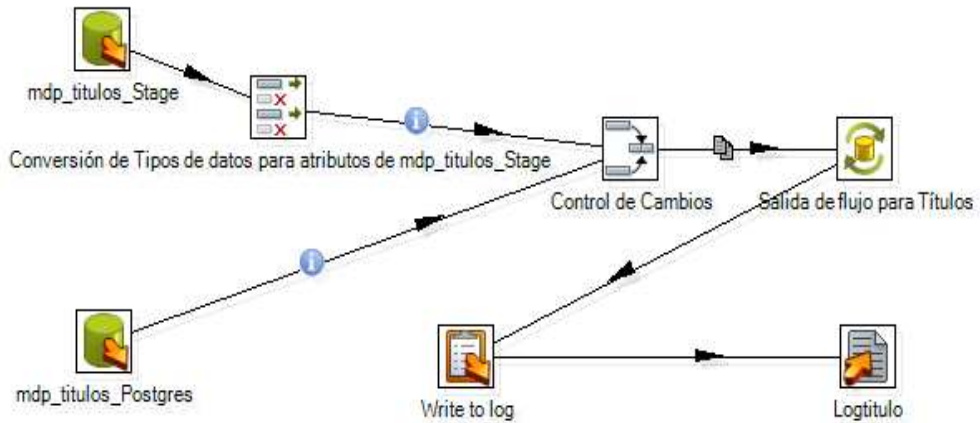
37

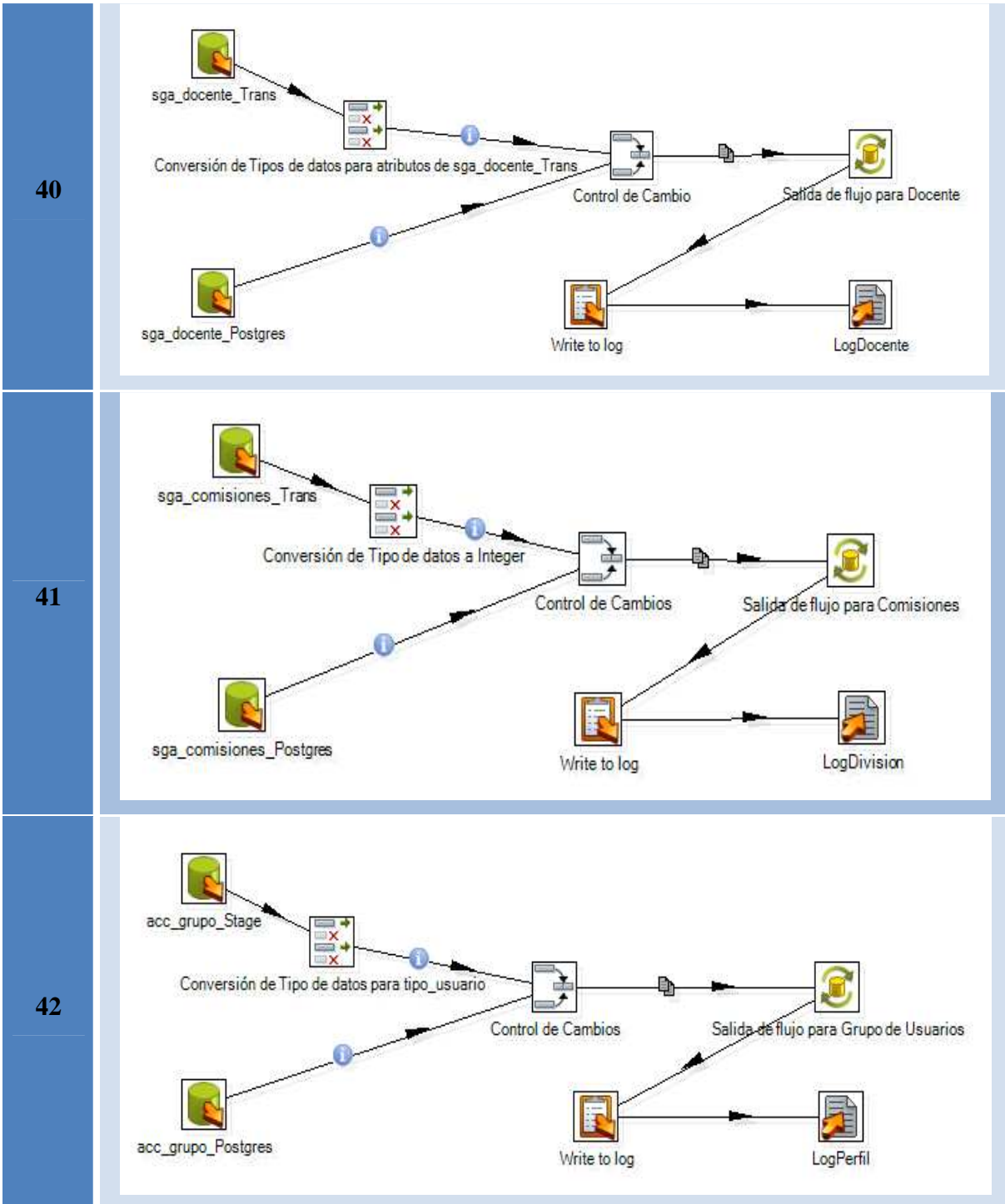


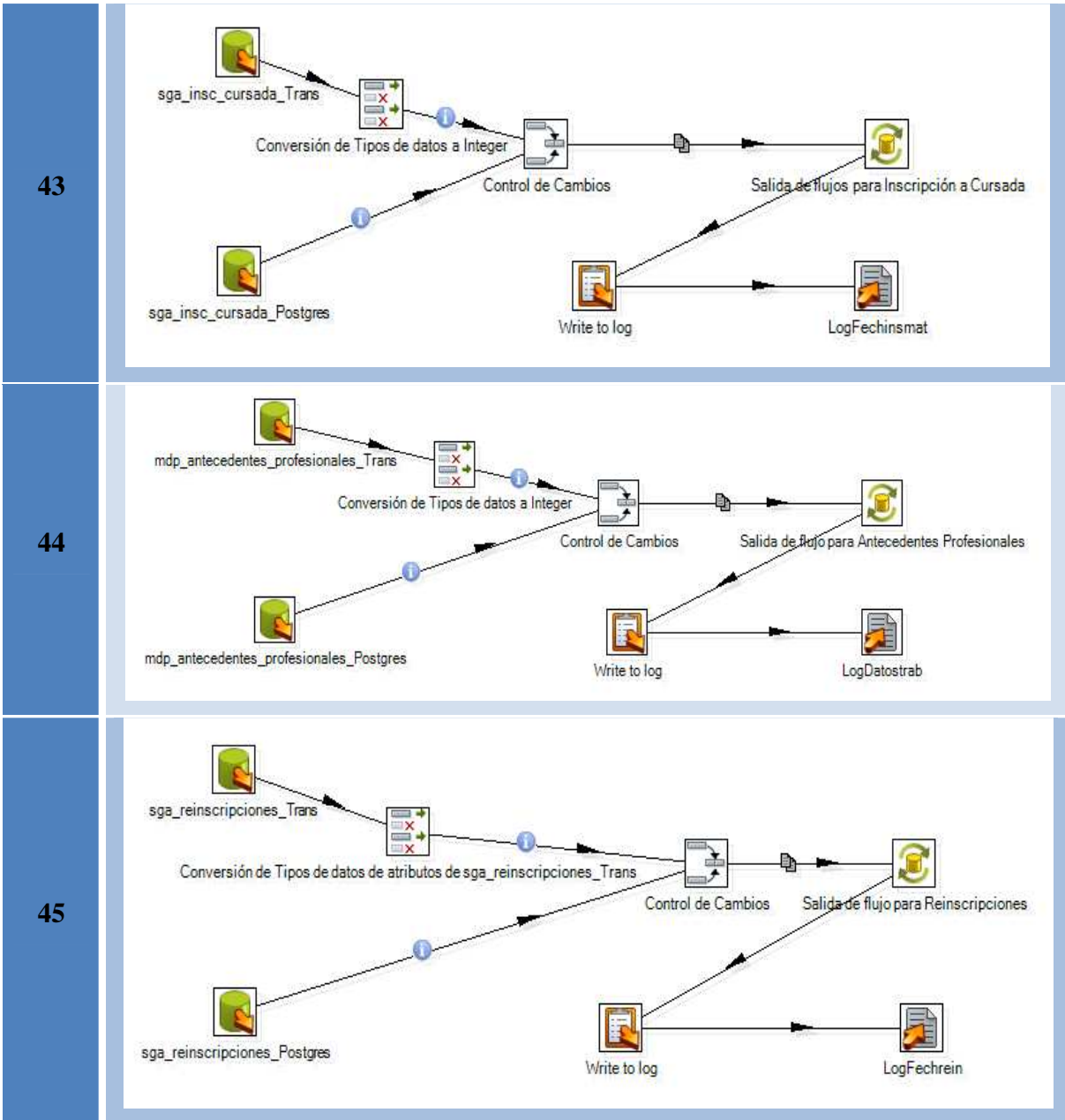
38

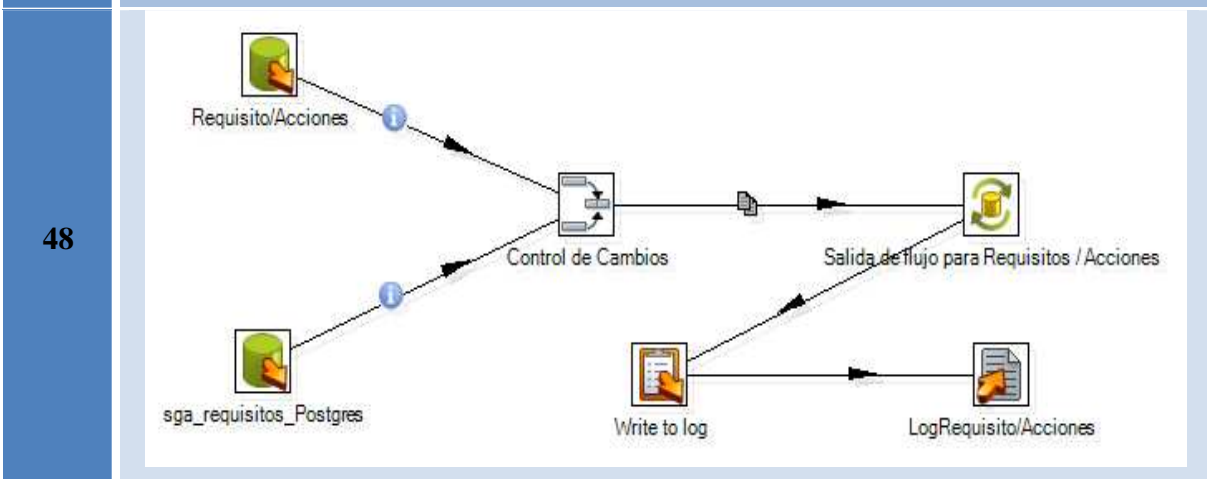
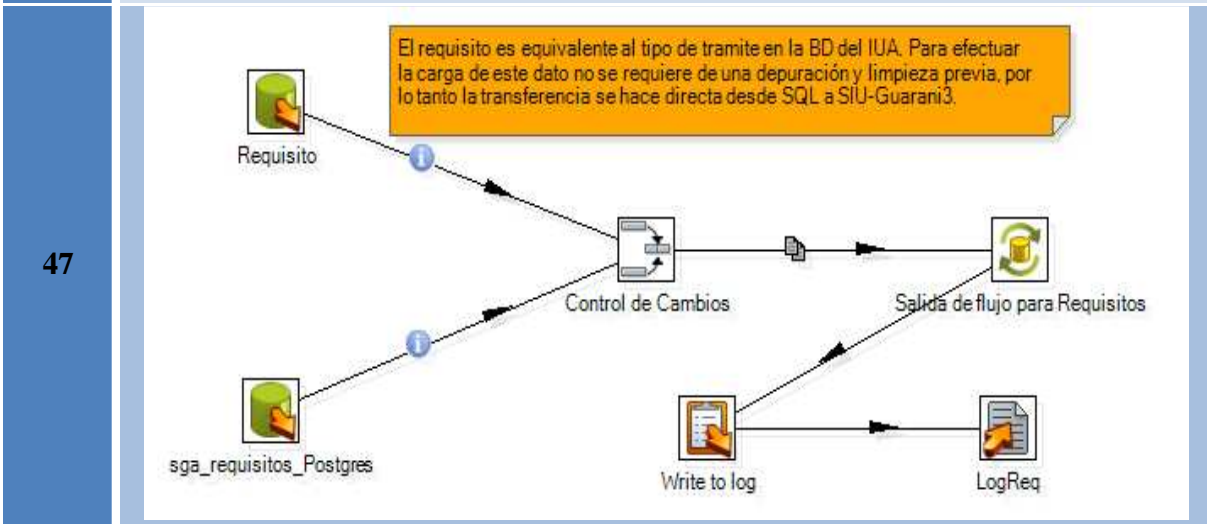
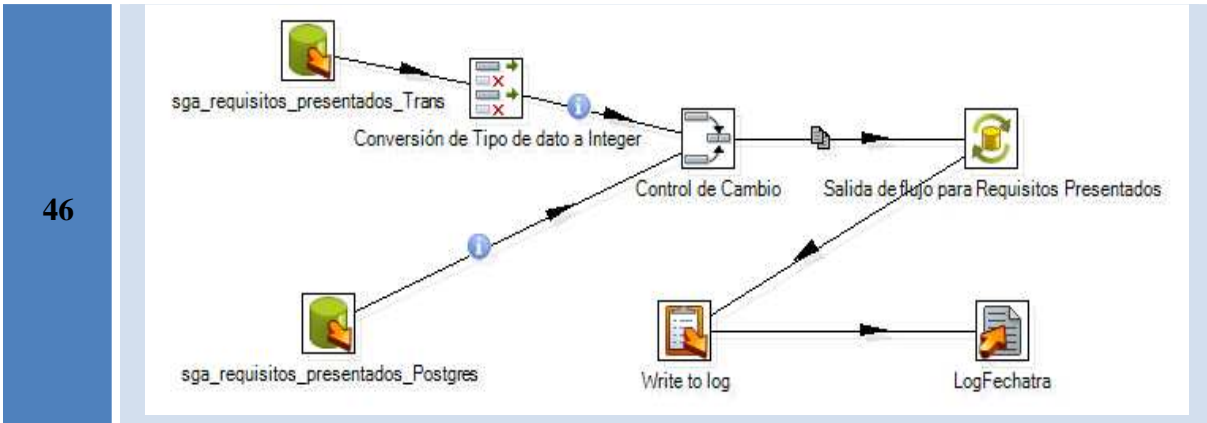


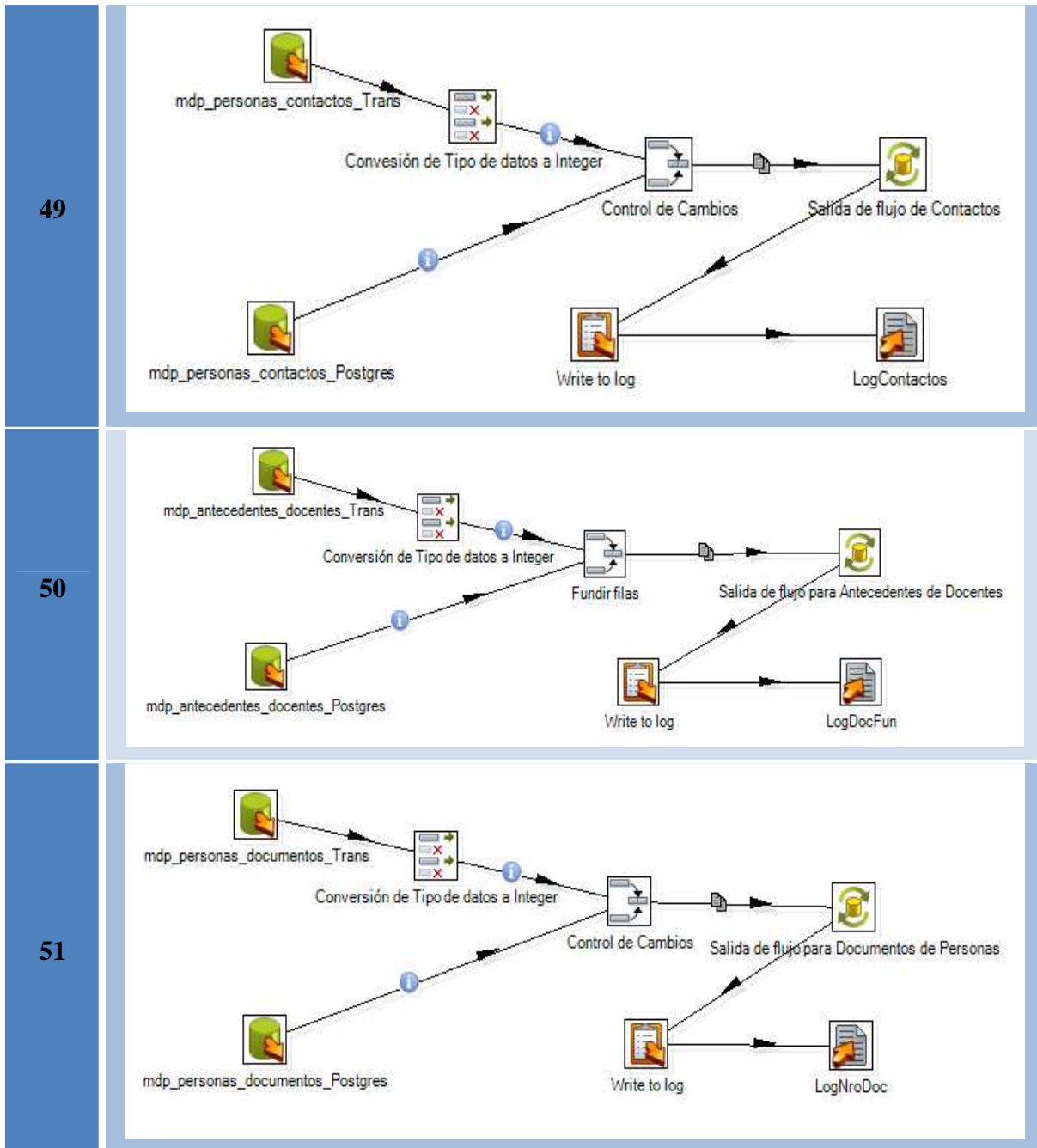
39

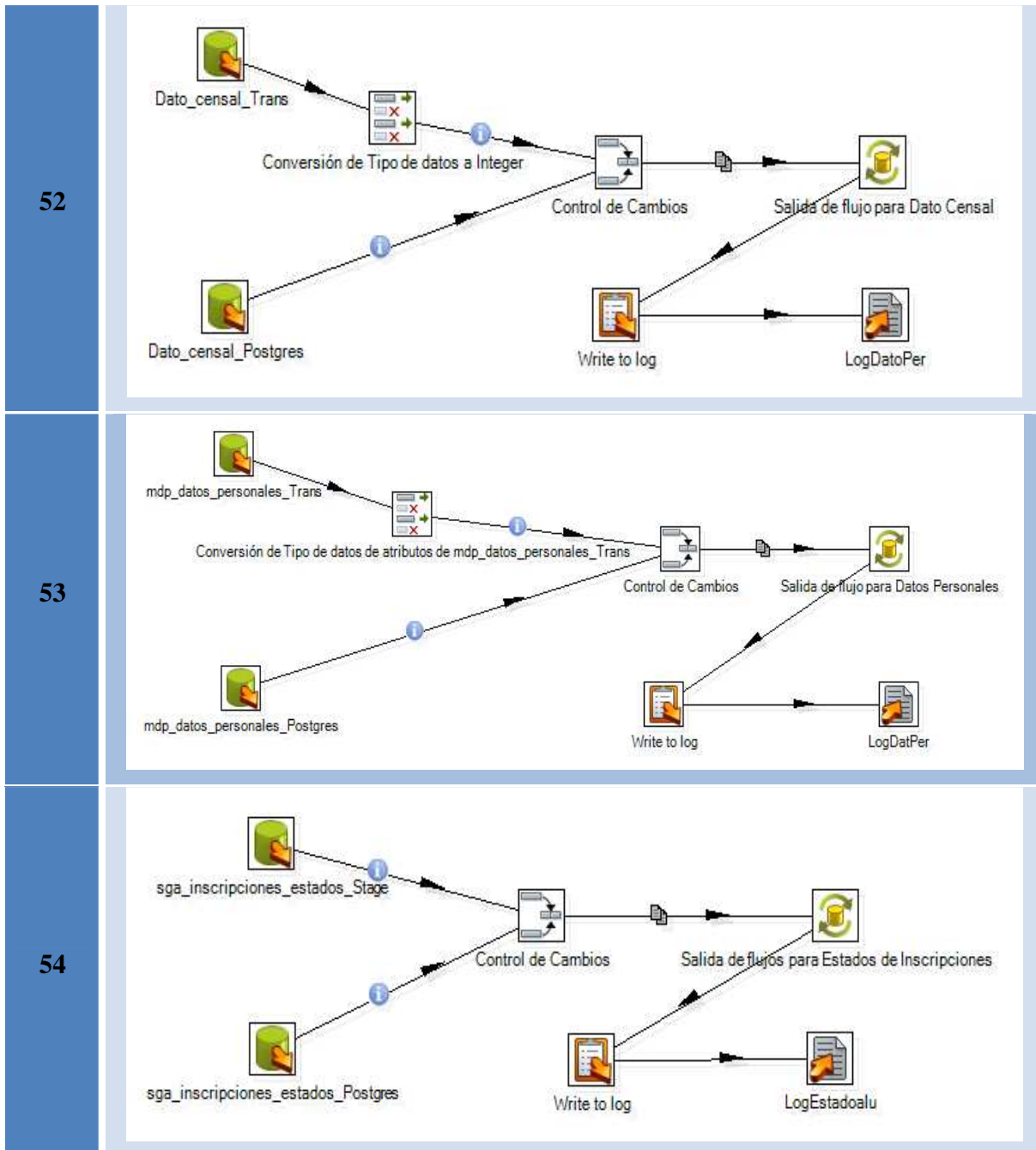


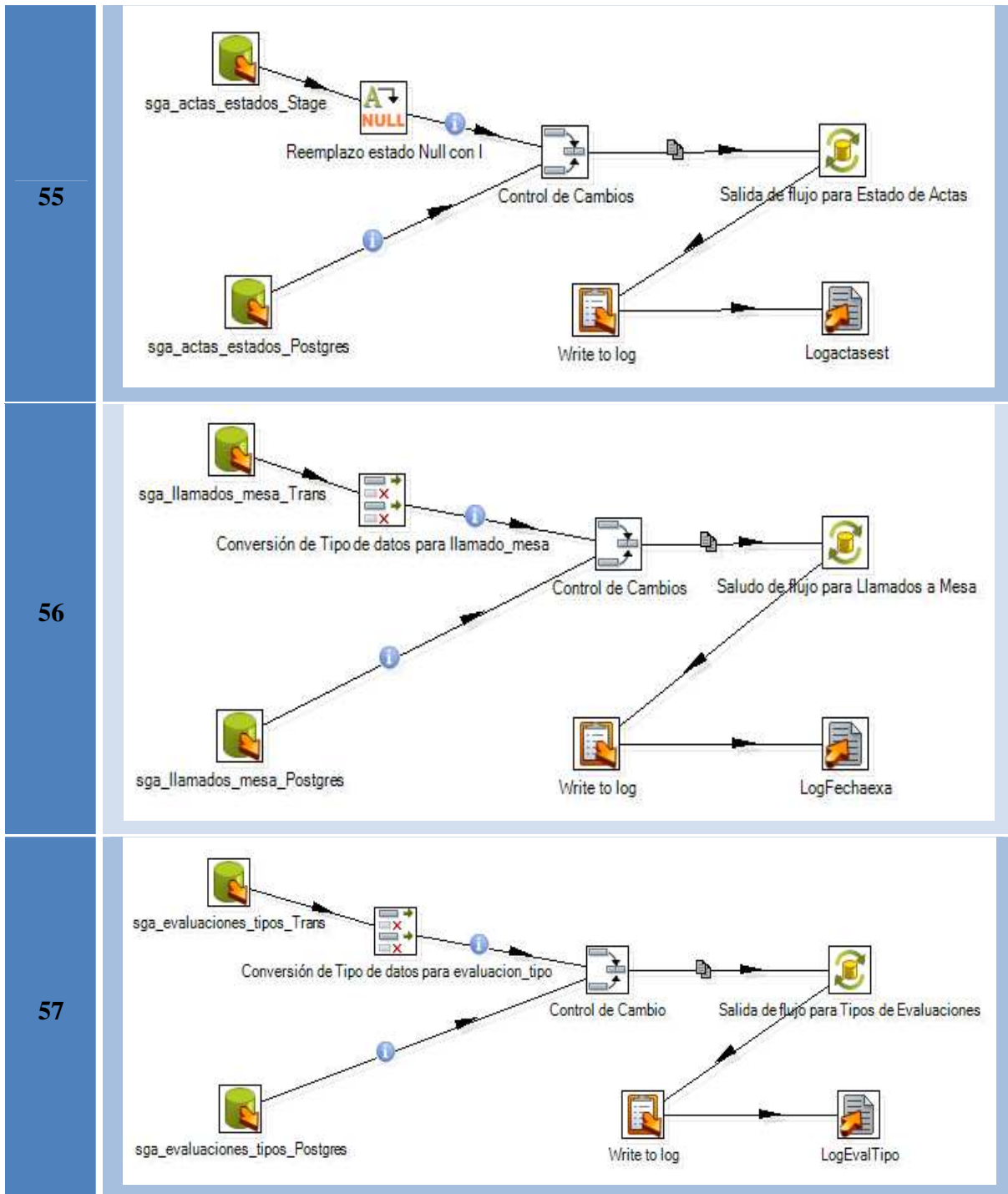


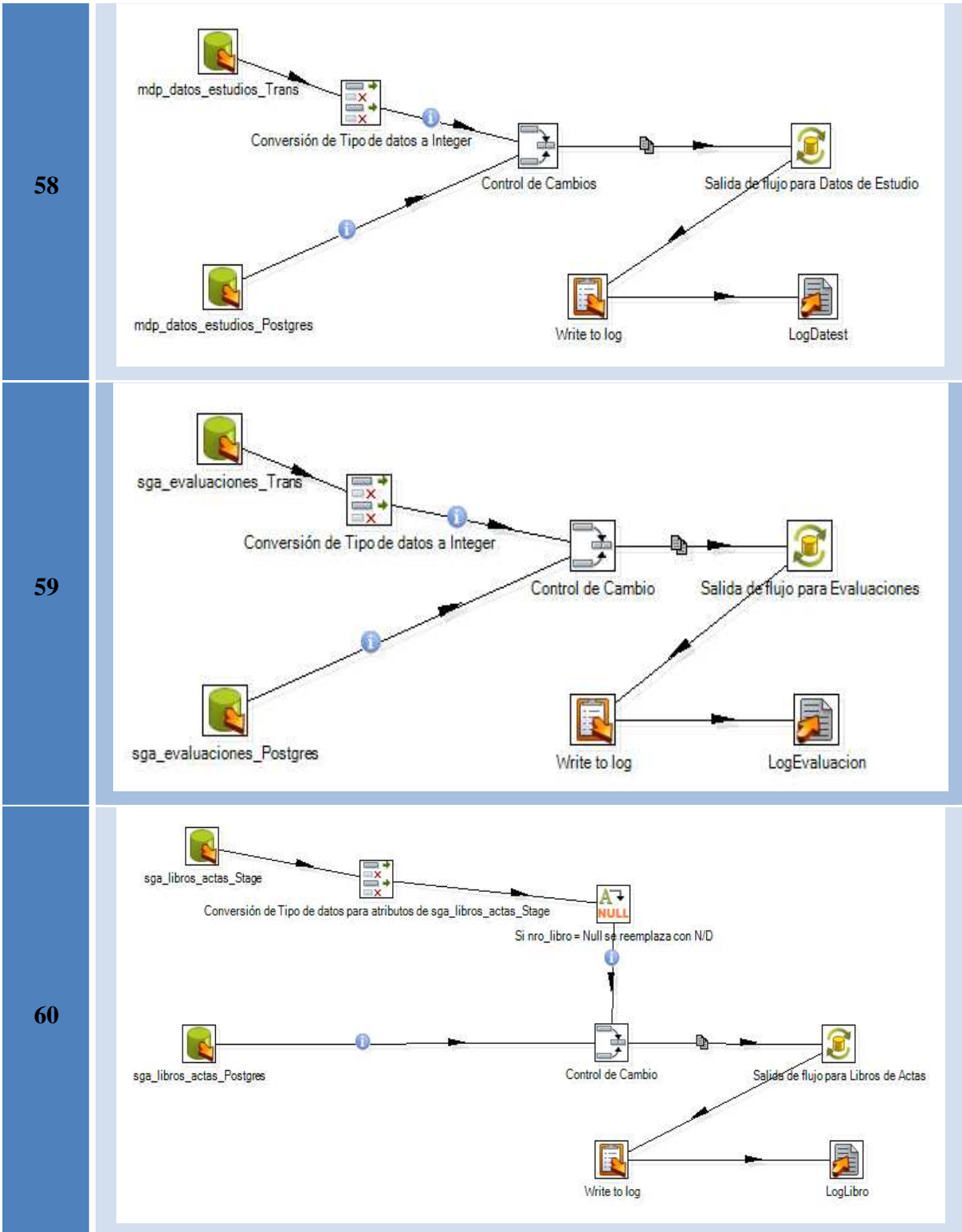




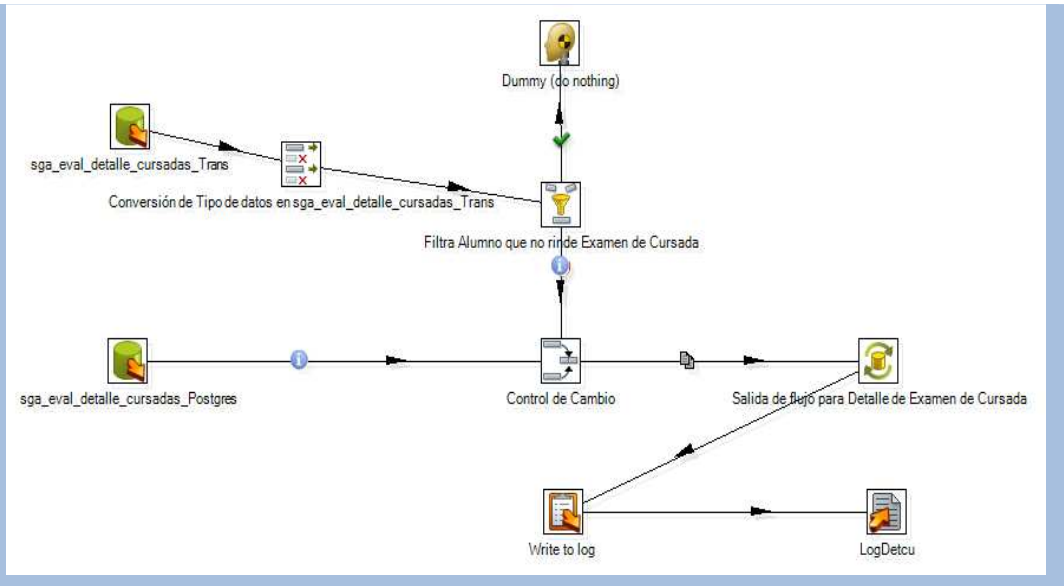




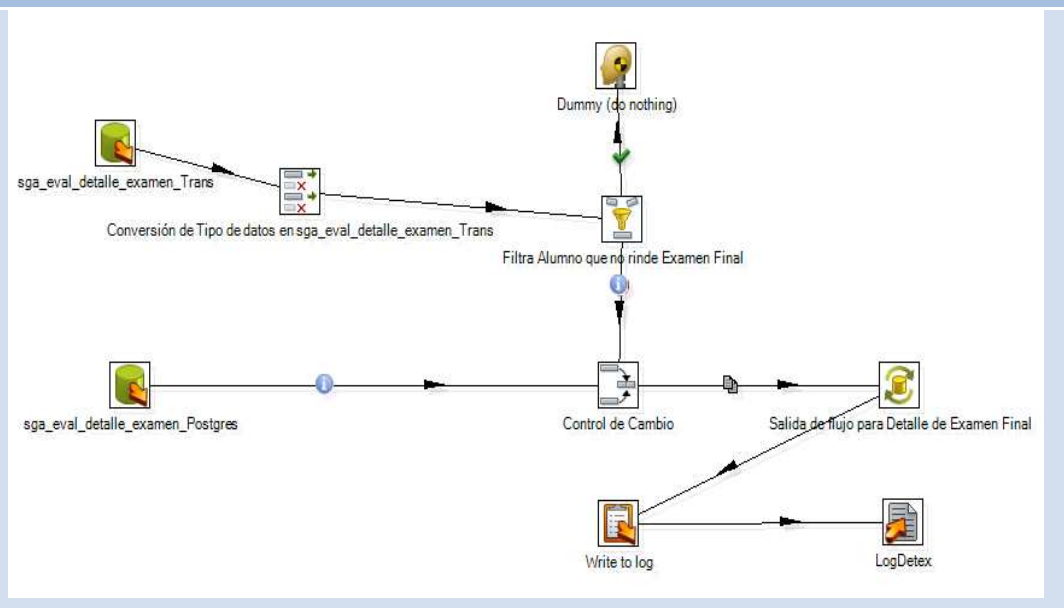


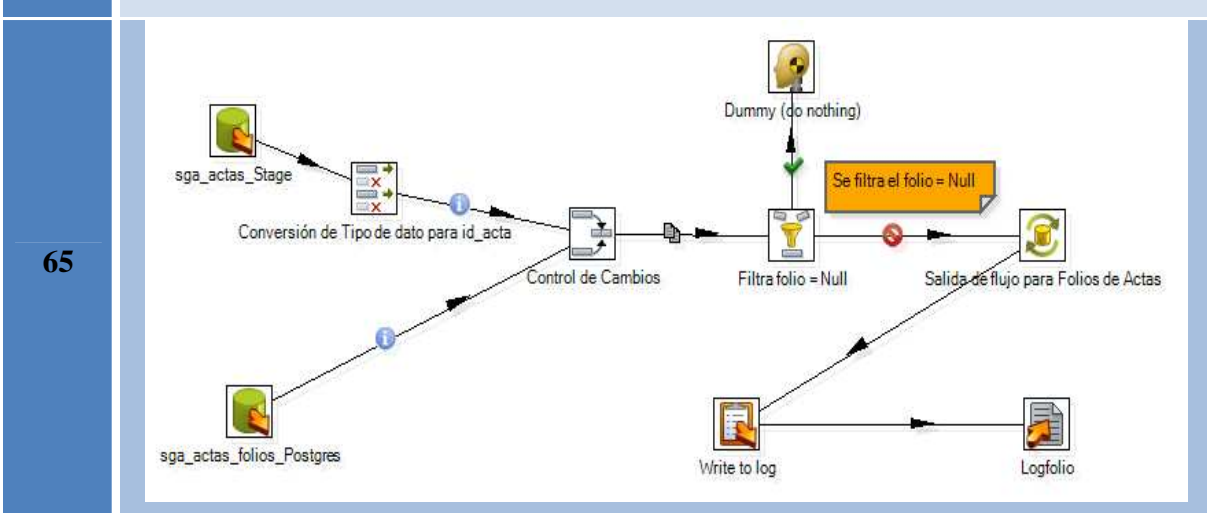
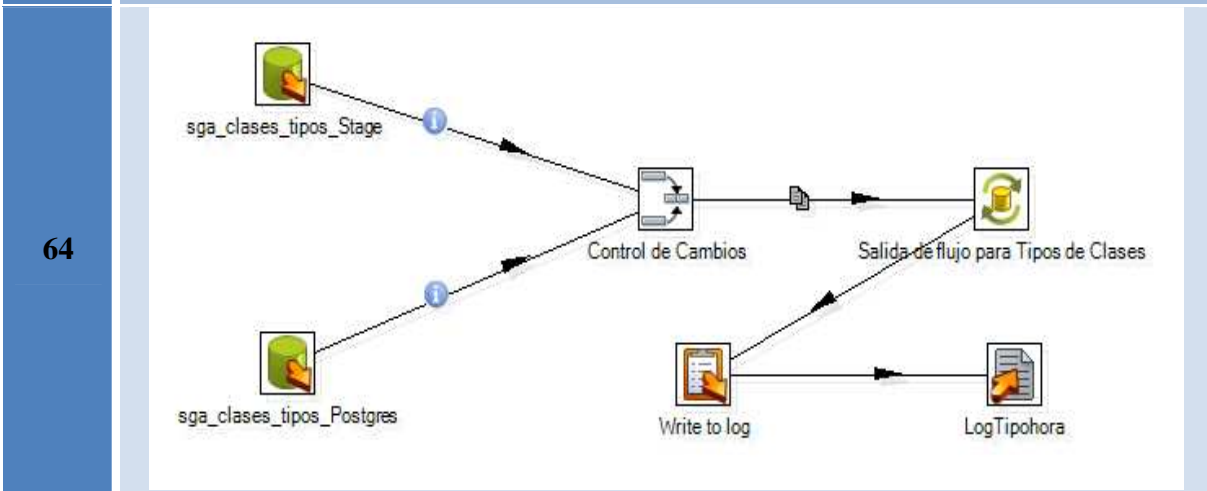
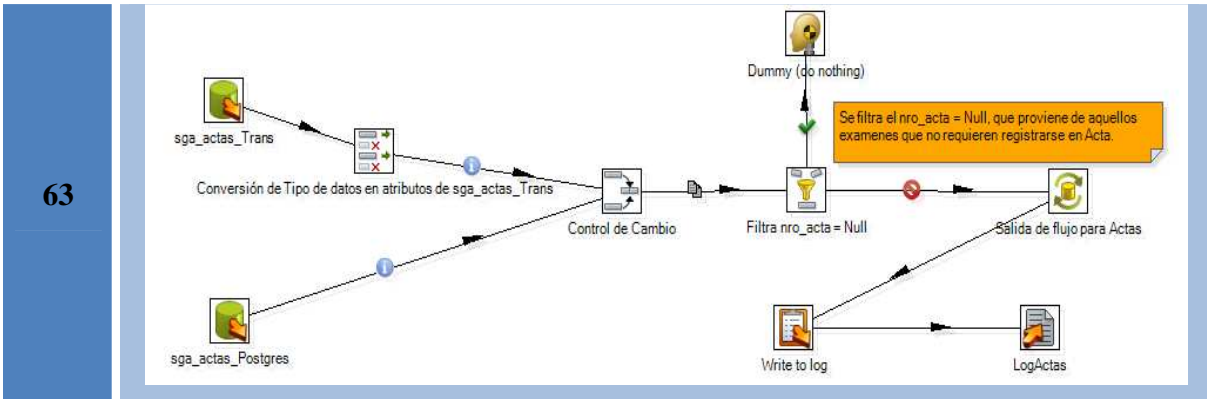


61



62





8.1.3.8 Ejemplos de Pasos ejecutados durante la Carga de datos y Control de Cambios

- *Ejemplo Edición del Paso **Entrada de Tabla** “sga_años_academicos_Trans”.*

En la **imagen 8.1.3.8.a** se observa la consulta a la tabla sga_años_academicos_Trans, de los datos transformados en la base de datos intermedia (Stage), previo a su migración.

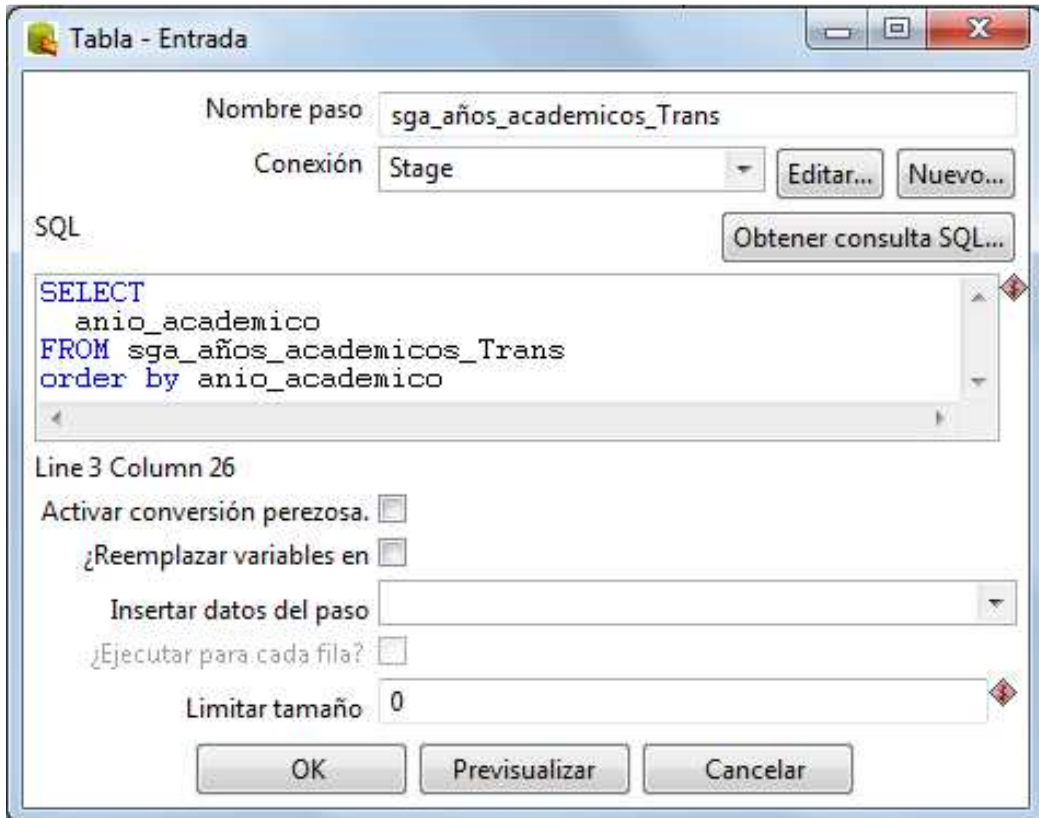


Imagen 8.1.3.8.a

- *Ejemplo Edición del Paso **Mapeo de Valores**.*

En la **imagen 8.1.3.8.b** se observa el mapeo para el nombre de la modalidad cursada, considerando valor origen y destino.

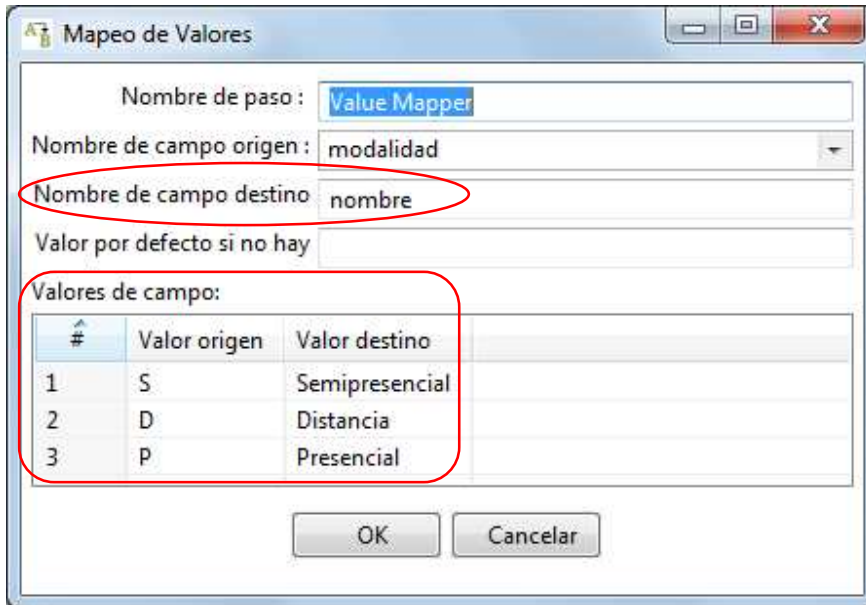


Imagen 8.1.3.8.b

- *Ejemplo Edición del Paso Operaciones con Cadenas.*

En la **imagen 8.1.3.8.c** se observa un recorte de espacios (trim type right) para el campo nombre que corresponde a la tabla sga_acciones_Stage.

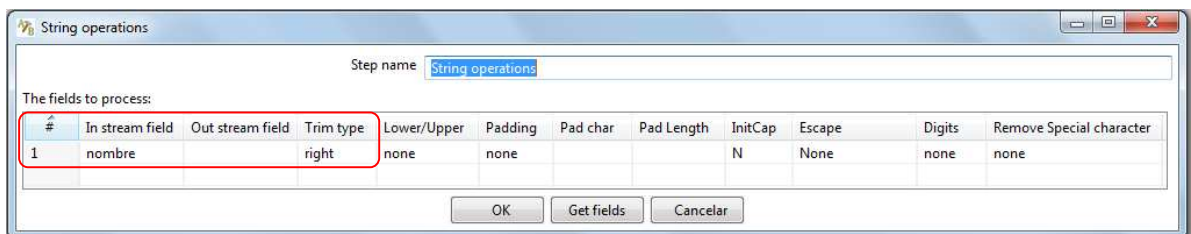


Imagen 8.1.3.8.c

- *Ejemplo Edición del Paso Selecciona / Renombra Valores.*

En la **imagen 8.1.3.8.d** se observa la modificación a Integer del tipo de dato de los campos que provienen de la tabla sga_planes_Trans.

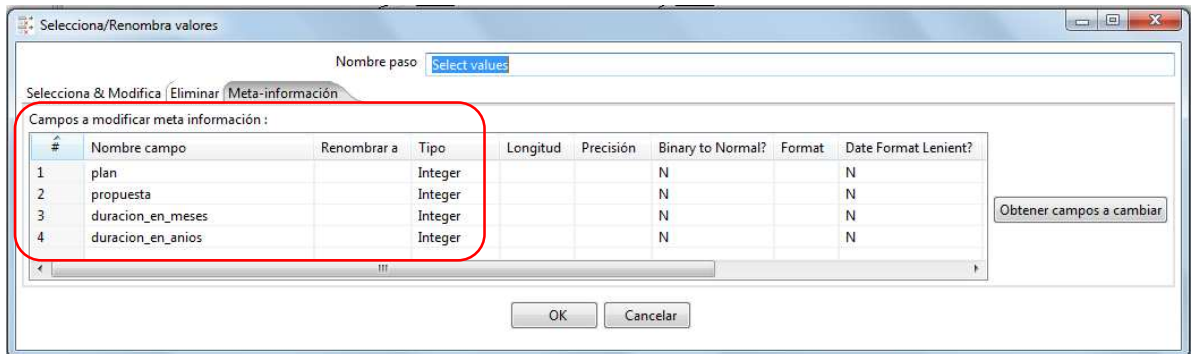


Imagen 8.1.3.8.d

- *Ejemplo Edición del Paso Ordenar Filas.*

En la **imagen 8.1.3.8.e** se observa el ordenamiento de filas de manera ascendente para el campo tipo_usuario de la tabla acc_tipos_usuarios_Stage.

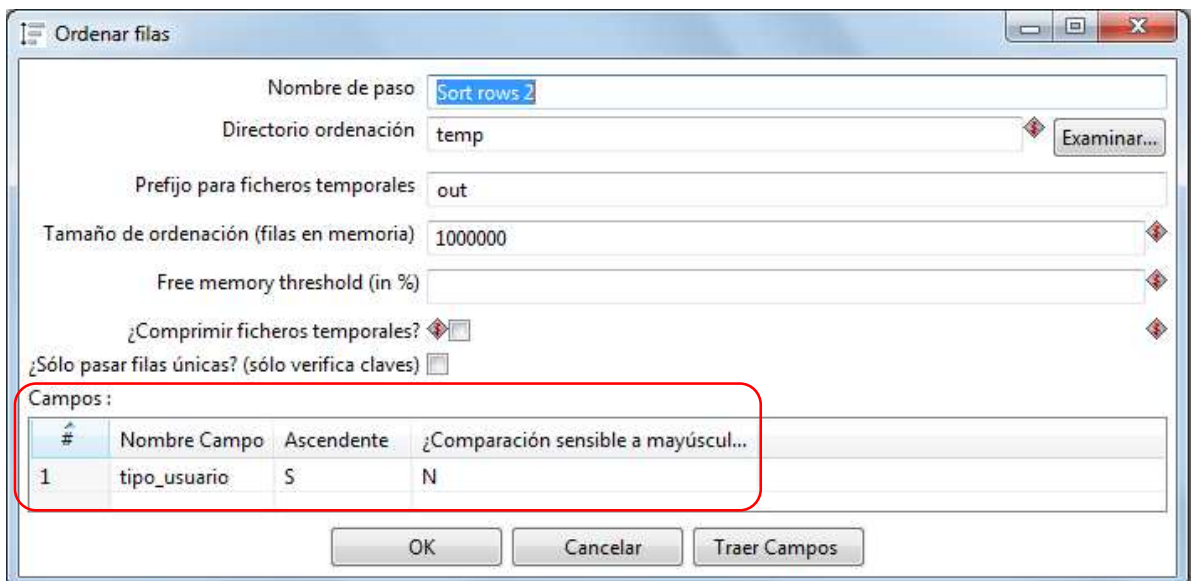


Imagen 8.1.3.8.e

- *Ejemplo Edición del Paso Filas Únicas.*

En la imagen 8.1.3.8.f se observa la aplicación de filas únicas para el campo estado de la tabla sga_propuestas_estado_Stage.

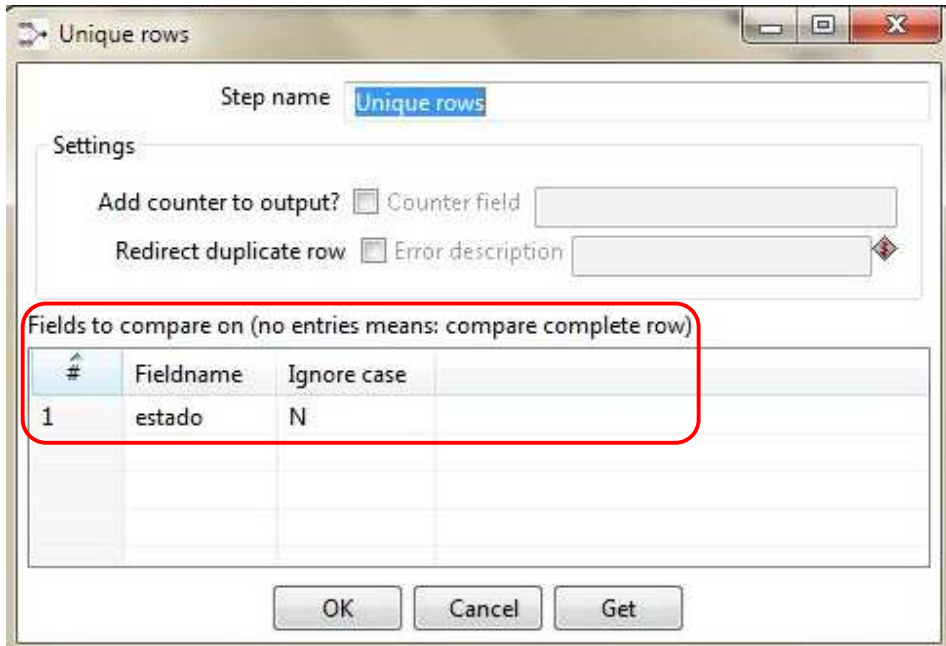


Imagen 8.1.3.8.f

- *Ejemplo Edición del Paso Si el Valor del Campo es Nulo*

En la **imagen 8.1.3.8.g** se observa el reemplazo con un valor 2 de aquellos valores nulos que trae el campo responsable_academica_tipo de la tabla sga_responsables_academicas_Stage.

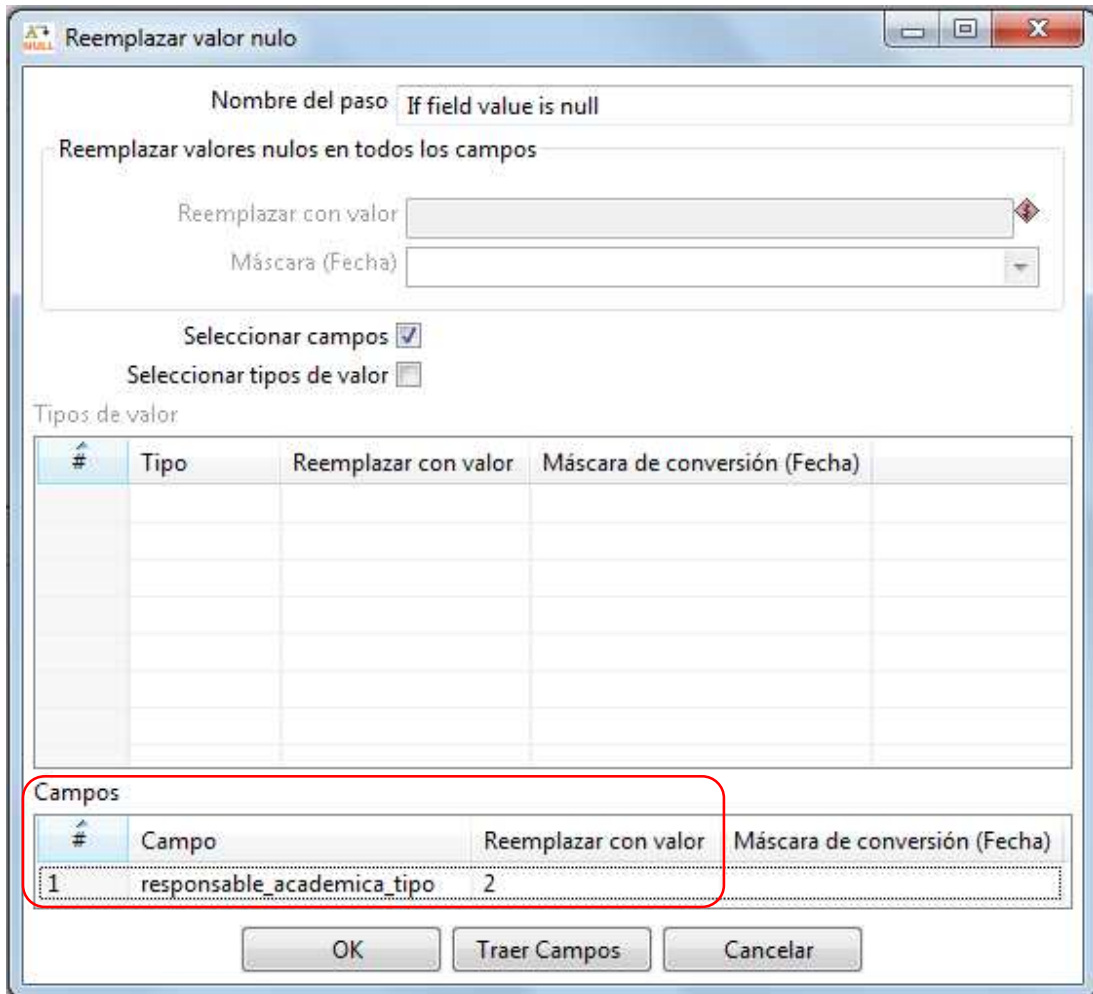


Imagen 8.1.3.8.g

- *Ejemplo Edición del Paso **Filtrar Filas***

En la **imagen 8.1.3.8.h** se puede observar el filtrado de números de actas que tienen valor null. Este paso al detectar nulos no hace nada, es decir envía los datos al paso Dummy. Los valores que se identifican como no nulos siguen en la corriente de secuencia.

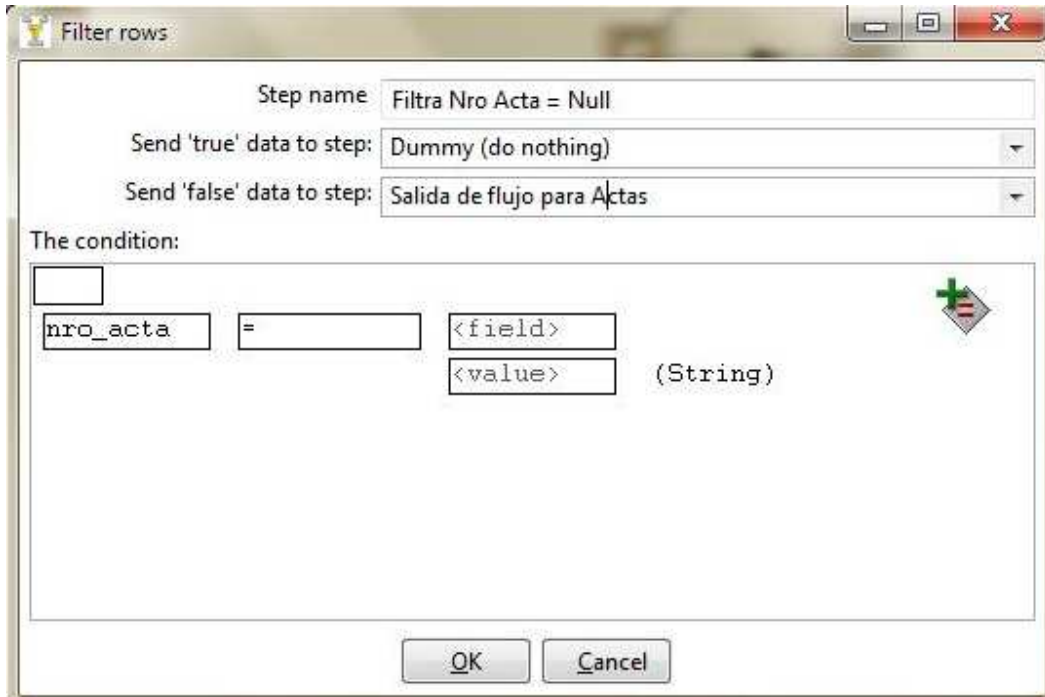


Imagen 8.1.3.8.h

- *Ejemplo Edición del Paso Transformación Simulada (No hace nada)*

En la **imagen 8.1.3.8.i** se observa que el paso no hace nada, solo recibe información que se presenta como flujo de datos falsos.

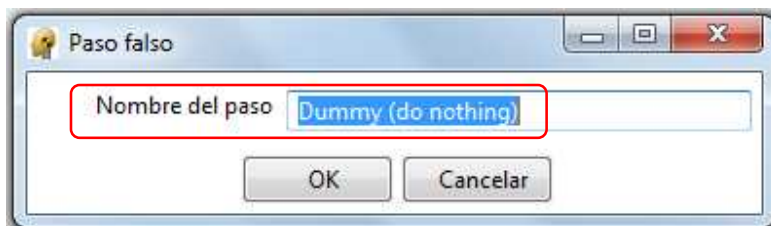


Imagen 8.1.3.8.i

- *Ejemplo Edición del Paso Fundir Filas*

En la **imagen 8.1.3.8.j** se observa la comparación de la información de ambas tablas (mdp_orientacion_recibida_Trans y mdp_orientacion_recibida_Postgres) gestionando el control de cambios.

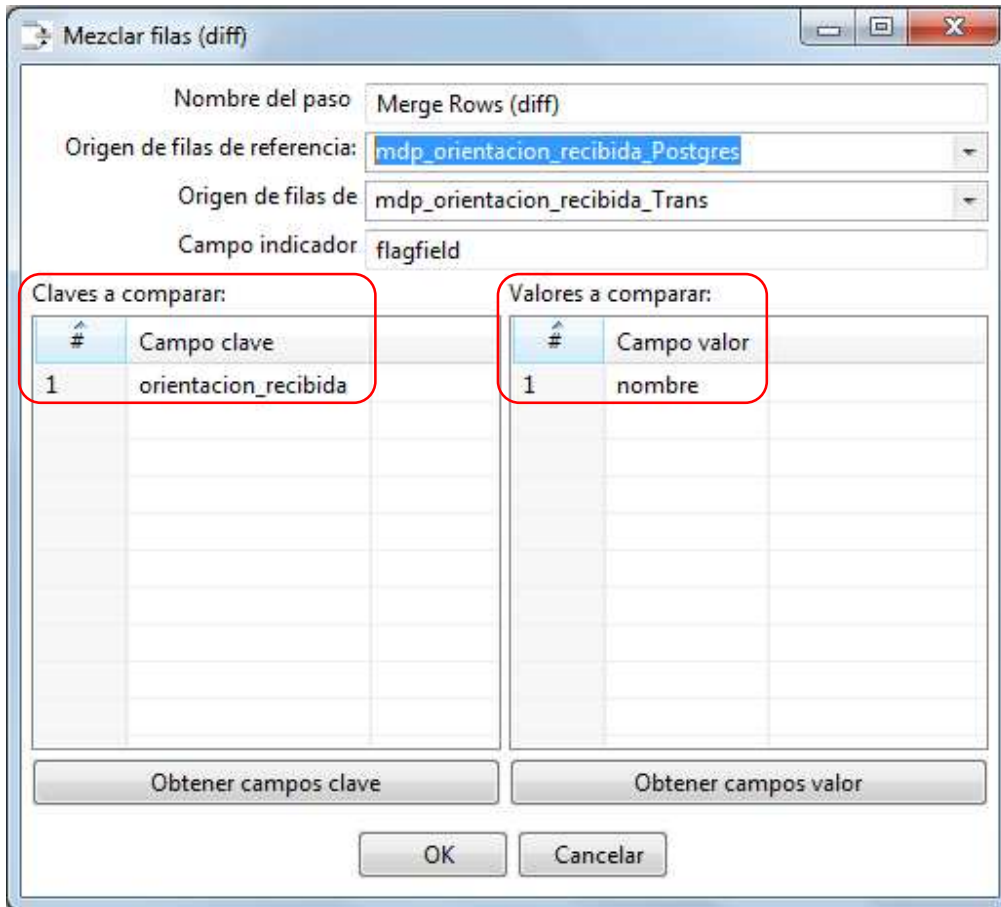


Imagen 8.1.3.8.j

- *Ejemplo Edición del Paso Synchronize after merge*

En la **imagen 8.1.3.8.k** se observan los campos que pueden actualizarse en la salida del flujo de información.

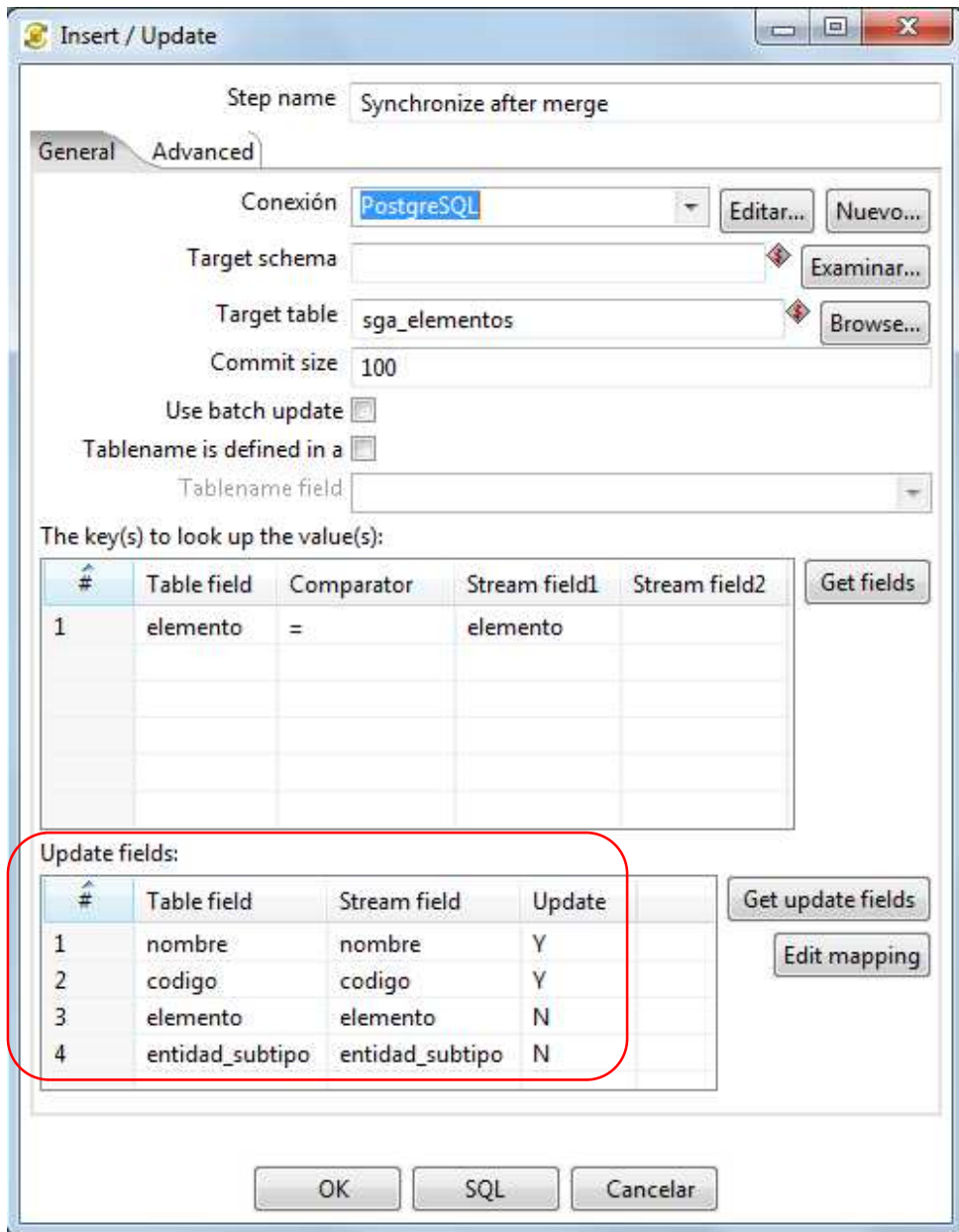


Imagen 8.1.3.8.k

- *Ejemplo Edición del Paso Write data to log*

En la **imagen 8.1.3.8.l** se visualizan los datos de carga de la tabla sga_requisitos_presentados_Trans que se escribirán al log. Asimismo se observa el campo flagfield que se utiliza como bandera y que permite indicar si se ha modificado o no un registro.

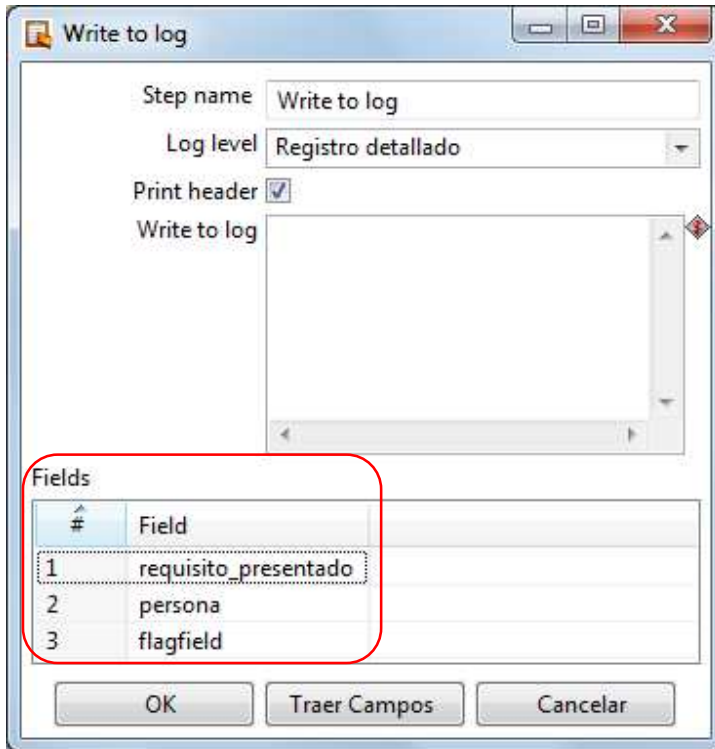


Imagen 8.1.3.8.l

- *Ejemplo Edición del Paso Salida Fichero de Texto*

En la **imagen 8.1.3.8.m** se observa la extensión determinada (.txt) en el que se visualizarán los datos de salida de la tabla sga_comisiones_Trans.

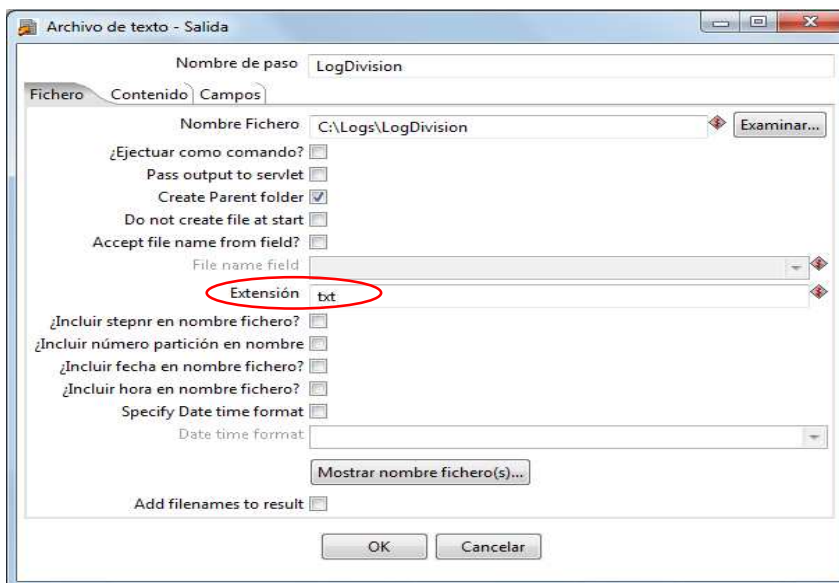


Imagen 8.1.3.8.m

En la **imagen 8.1.3.8.n** se observan los campos de salida que se mostrarán en el fichero de texto.

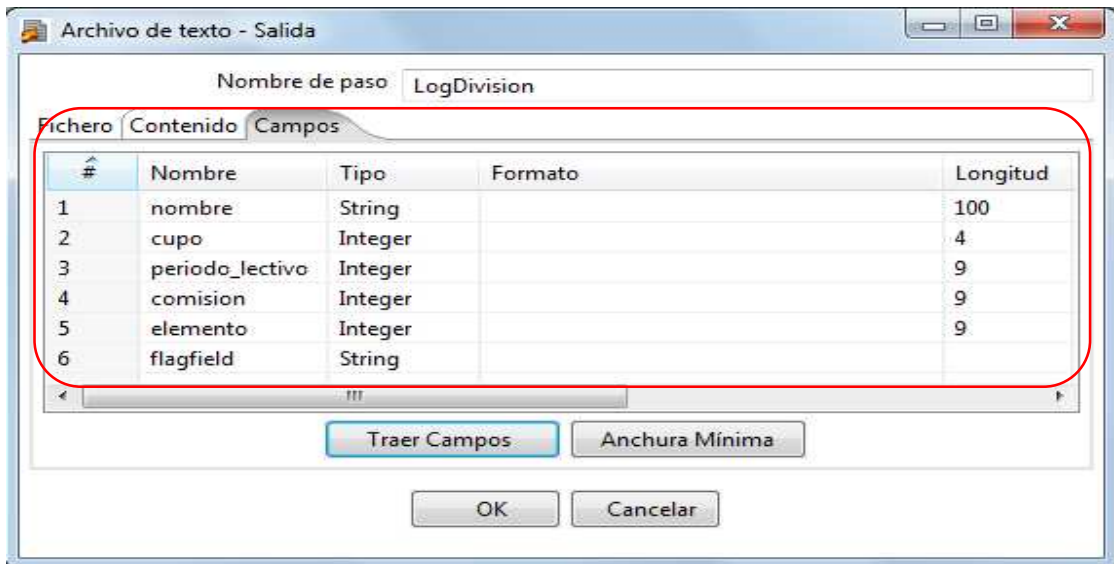


Imagen 8.1.3.8.n

Cabe aclarar que todos los ejemplos ilustrados corresponden a diferentes transformaciones de la etapa de Carga y Control de cambios.

8.1.3.6 Trabajo para Carga y Control de Cambios

A continuación se ilustra en la **Fig. 8.1.3.6** el Trabajo planificado en Kettle que ejecuta automáticamente y en orden secuencial cada una de las transformaciones para la Carga y Control de cambios:

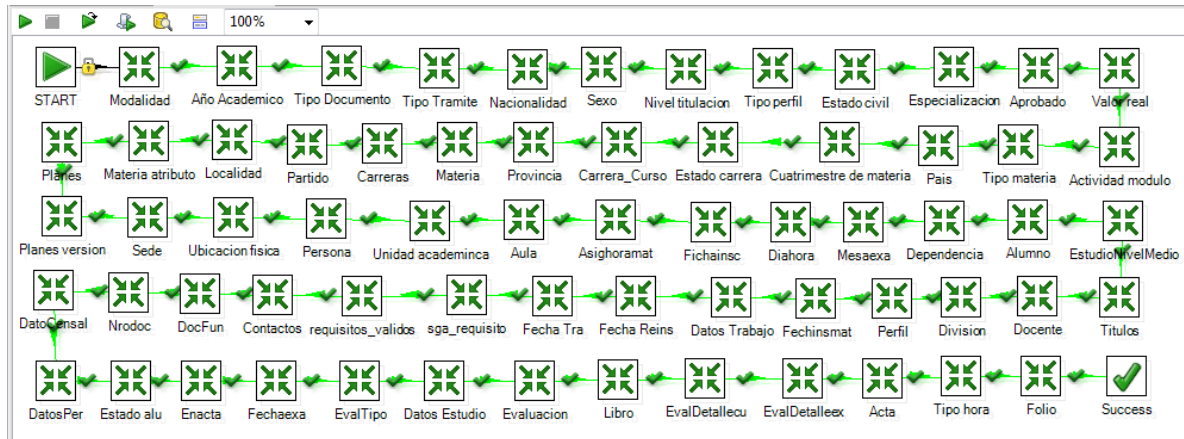


Fig. 8.3.1.6

El resultado de la ejecución del trabajo previamente ilustrado quedará demostrado en la etapa de Pruebas, en el caso de test que aplique al mismo.

8.1.3.7 Trabajo para secuencia ETL

En la siguiente ilustración (**Fig. 8.1.3.7**) se expone el trabajo global que contiene todos los trabajos anteriormente mencionados, invocando a los mismos y ejecutándolos en el orden ETL que se encuentran establecidos:



Fig. 8.1.3.7

El resultado de la ejecución del trabajo previamente ilustrado quedará demostrado en la etapa de Pruebas, en el caso de test que aplique al mismo.

Etapa 5

PRUEBAS Y PUESTA EN MARCHA

En esta etapa se realizarán las pruebas de eficacia operativa que determinarán si los resultados esperados son acordes con los resultados obtenidos de la migración a SIU-Guarani3. Además, se describirán las principales acciones a realizar para que el modelo o solución propuesta pueda aplicarse a la realidad.

9.1 PRUEBAS

Casos de Testing

Para llevar a cabo esta tarea se proponen los siguientes casos de test:

9.1.1 Caso de Test 1 – Control de Cambios en tabla: “*mdp_personas*”

Ejecutamos el trabajo de migración sin hacer modificaciones previas a información contenida en la BD del IUA y analizamos una muestra de 20 (veinte) registros de la tabla *mdp_personas* a partir del log de salida que se observa a continuación:

persona	apellido	nombres	sexo	fecha_nacimiento	localidad_nacimiento	nacionalidad	flagfield
1	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000	2942	2	identical
2	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000	2137	5	identical
3	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
4	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
5	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
6	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000	2942	27	identical
7	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000	2942	23	identical
8	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000	2942	23	identical
9	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
10	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
11	APELLIDO 170	NOMBRE 170	N	1980/01/01 00:00:00.000	2942	5	identical
12	APELLIDO 171	NOMBRE 171	N	1980/01/01 00:00:00.000		5	identical
13	APELLIDO 172	NOMBRE 172	N	1980/01/01 00:00:00.000		5	identical
14	APELLIDO 173	NOMBRE 173	N	1980/01/01 00:00:00.000		5	identical
15	APELLIDO 175	NOMBRE 175	M	1980/01/01 00:00:00.000	2942	5	identical
16	APELLIDO 177	NOMBRE 177	N	1980/01/01 00:00:00.000	2942	5	identical
17	APELLIDO 178	NOMBRE 178	M	1980/01/01 00:00:00.000	2942	5	identical
18	APELLIDO 179	NOMBRE 179	M	1980/01/01 00:00:00.000		5	identical
19	APELLIDO 180	NOMBRE 180	M	1980/01/01 00:00:00.000		5	identical
20	APELLIDO 181	NOMBRE 181	F	1980/01/01 00:00:00.000	2942	5	identical

El campo *flagfield* = *identical* significa que los datos ya habían sido transferidos a SIU-Guarani3 y que no se han modificado luego de una nueva ejecución de trabajo de migración.

Cabe aclarar que el personal de Base de Datos del IUA nos entregó la información con algunos campos genéricos por cuestiones de confidencialidad y privacidad de la información. Por lo tanto el apellido del alumno se registra como “APELLIDO ALUMNO” y los nombres como “NOMBREA NOMBREB”; mientras que para docente el apellido se registra como “APELLIDO” seguido de un número, y el nombre se registra como “NOMBRE” también seguido por un número. Por ej.: APELLIDO 170, NOMBRE 170. Asimismo, existen campos vacíos o cargados por defecto ya que así se originaron inicialmente en la Base de Datos del IUA.

Con el objetivo de verificar el funcionamiento del Control de Cambios implementado en las transformaciones de carga realizamos la modificación en SQL de los registros (apellido y nombres) que corresponden a la persona 1 y 11 de la tabla *mdp_personas* en SIU-Guarani3. Ejecutamos nuevamente el trabajo y consultamos el resultado que se observa a continuación:

persona	apellido	nombres	sexo	fecha_nacimiento	localidad_nacimiento	nacionalidad	flagfield
1	Rosso	Mauricio	M	1980/01/01 00:00:00.000	2942	2	changed
2	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000	2137	5	identical
3	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
4	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
5	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
6	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000	2942	27	identical
7	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000	2942	23	identical
8	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000	2942	23	identical
9	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
10	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
11	Rodriguez	Marcos	N	1980/01/01 00:00:00.000	2942	5	changed
12	APELLIDO 171	NOMBRE 171	N	1980/01/01 00:00:00.000		5	identical
13	APELLIDO 172	NOMBRE 172	N	1980/01/01 00:00:00.000		5	identical
14	APELLIDO 173	NOMBRE 173	N	1980/01/01 00:00:00.000		5	identical
15	APELLIDO 175	NOMBRE 175	M	1980/01/01 00:00:00.000	2942	5	identical
16	APELLIDO 177	NOMBRE 177	N	1980/01/01 00:00:00.000	2942	5	identical
17	APELLIDO 178	NOMBRE 178	M	1980/01/01 00:00:00.000	2942	5	identical
18	APELLIDO 179	NOMBRE 179	M	1980/01/01 00:00:00.000		5	identical
19	APELLIDO 180	NOMBRE 180	M	1980/01/01 00:00:00.000		5	identical
20	APELLIDO 181	NOMBRE 181	F	1980/01/01 00:00:00.000	2942	5	identical

Inspeccionamos que en el campo de bandera o marca (flagfield) se modificó a “*changed*” para los registros actualizados, siendo acorde a los efectos de funcionar efectivamente.

Luego realizamos una tercera corrida del trabajo sin modificaciones previas, el cuál arrojó el siguiente resultado:

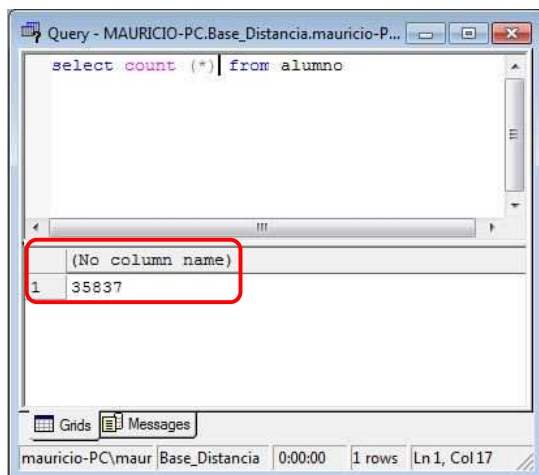
persona	apellido	nombres	sexo	fecha_nacimiento	localidad_nacimiento	nacionalidad	flagfield
1	Rosso	Mauricio	M	1980/01/01 00:00:00.000	2942	2	identical
2	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000	2137	5	identical
3	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
4	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
5	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
6	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000	2942	27	identical
7	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000	2942	23	identical
8	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000	2942	23	identical
9	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
10	APELLIDO ALUMNO	NOMBREA NOMBREB	M	1980/01/01 00:00:00.000		23	identical
11	Rodriguez	Marcos	N	1980/01/01 00:00:00.000	2942	5	identical
12	APELLIDO 171	NOMBRE 171	N	1980/01/01 00:00:00.000		5	identical
13	APELLIDO 172	NOMBRE 172	N	1980/01/01 00:00:00.000		5	identical
14	APELLIDO 173	NOMBRE 173	N	1980/01/01 00:00:00.000		5	identical
15	APELLIDO 175	NOMBRE 175	M	1980/01/01 00:00:00.000	2942	5	identical
16	APELLIDO 177	NOMBRE 177	N	1980/01/01 00:00:00.000	2942	5	identical
17	APELLIDO 178	NOMBRE 178	M	1980/01/01 00:00:00.000	2942	5	identical
18	APELLIDO 179	NOMBRE 179	M	1980/01/01 00:00:00.000		5	identical
19	APELLIDO 180	NOMBRE 180	M	1980/01/01 00:00:00.000		5	identical
20	APELLIDO 181	NOMBRE 181	F	1980/01/01 00:00:00.000	2942	5	identical

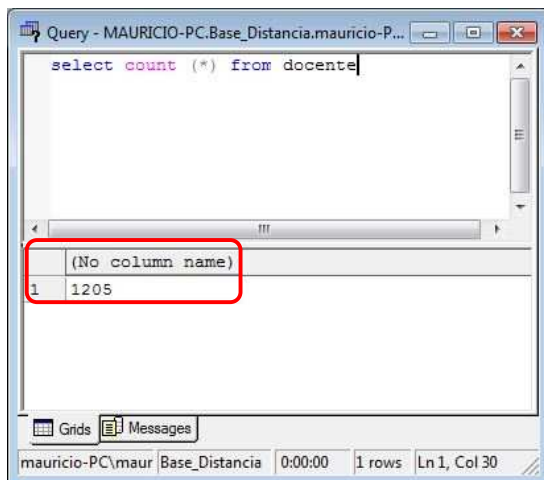
Verificamos que para todos los registros el campo flagfield retoma a su estado “*identical*”, debido a que en la última corrida no hubo modificaciones, siendo acorde a los efectos de funcionar efectivamente.

9.1.2 Caso de Test 2 – Control de Cantidad de Alumnos y Docentes Migrados

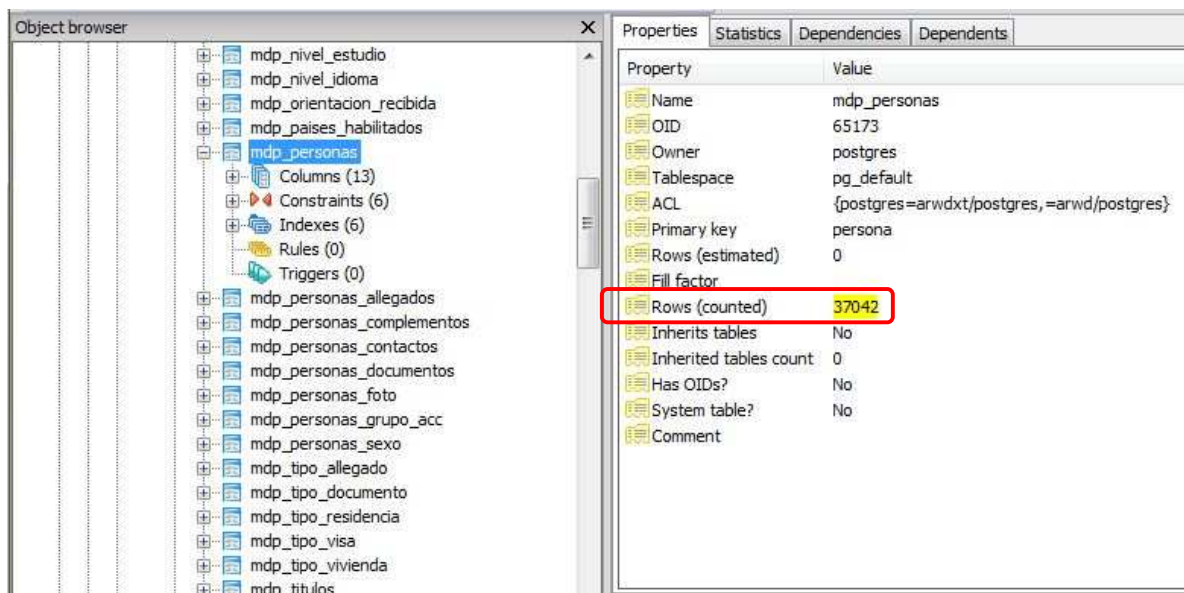
Efectuamos una consulta en SQL Server de la totalidad de Alumnos y Docentes ingresados en la Base de Datos del IUA:

Alumnos en IUA:



Docentes en IUA:

Verificamos que la cantidad de alumnos cargados corresponde a 35.837, mientras que la cantidad de docentes es de 1.205, sumados ambos hacen un total de 37.042, número que debería quedar plasmado para la totalidad de las filas contadas de la tabla *mdp_personas* de SIU-Guarani3. Asimismo, ingresamos a PostgreSQL e inspeccionamos el valor de Rows (counted) registrado en la tabla *mdp_personas* y efectivamente coincide con el total de alumnos y docentes que fueron transferidos desde la BD del IUA:



De esta manera, se concluye que los resultados obtenidos son los esperados, siendo efectiva la validación para el control de la cantidad de alumnos y docentes migrados.

9.1.3 Caso de Test 3 – Control de Cambios en tabla: “*sga_prupuestas*”

Ejecutamos una primera corrida del trabajo de migración para verificar la Carga inicial de datos en la tabla *sga_prupuestas* y analizamos una muestra de 20 (veinte) registros partir del log de salida que se observa a continuación:

propuesta	nombre	estado	codigo	propuesta_tipo	flagfield
1	ACTIVIDAD DE EXTENSIÓN	0	0	3	new
2	ACTIVIDAD DE EXTENSIÓN	0	A5	3	new
3	CURSO PSGR INAEROESPAC	1	AER	6	new
4	ALEMAN	1	ALE	6	new
5	PPG Auditoría en Salud	1	AUD	6	new
6	PPG Auditoría en Salud	1	AUD	5	new
7	ACTIVIDAD DE EXTENSIÓN	1	C01	6	new
8	ACTIVIDAD DE EXTENSIÓN	1	C02	6	new
9	ACTIVIDAD DE EXTENSIÓN	1	C03	6	new
10	DISEÑO MECANICO 4	1	C04	6	new
11	DISEÑO MECANICO 5	1	C05	6	new
12	DISEÑO MECANICO 6	1	C06	6	new
13	DISEÑO MECANICO 7	1	C07	6	new
14	CURSILLO PREPARATORIO A DISTANCIA	0	CAD	3	new
15	CURSO PREPARATORIO A DISTANCIA	1	CAD	5	new
16	CURSOS DE ACTUALIZACIÓN	1	CAT	6	new
17	ADMINISTRACION DE LA CALIDAD	1	CC	3	new
18	CURSO INTRODUCTORIO (CIMEI)	0	CI	3	new
19	CURSO DE ADMISION	1	CI	6	new
20	CURSO A DISTANCIA DE COMPRESION LECTORA	1	CLD	6	new

El campo *flagfield* = *new* significa que los datos han impactado por primera vez en la base de datos de SIU-Guarani3.

Luego ejecutamos una segunda corrida del trabajo sin hacer modificaciones previas a información contenida en la BD del IUA y analizamos la misma muestra de 20 (veinte) registros de la tabla *sga_prupuestas*:

propuesta	nombre	estado	codigo	propuesta_t	flagfield
1	ACTIVIDAD DE EXTENSIÓN	0	0	3	identical
2	ACTIVIDAD DE EXTENSIÓN	0	A5	3	identical
3	CURSO PSGR INAEROESPAC	1	AER	6	identical
4	ALEMAN	1	ALE	6	identical
5	PPG Auditoría en Salud	1	AUD	6	identical
6	PPG Auditoría en Salud	1	AUD	5	identical
7	ACTIVIDAD DE EXTENSIÓN	1	C01	6	identical
8	ACTIVIDAD DE EXTENSIÓN	1	C02	6	identical
9	ACTIVIDAD DE EXTENSIÓN	1	C03	6	identical
10	DISEÑO MECANICO 4	1	C04	6	identical
11	DISEÑO MECANICO 5	1	C05	6	identical
12	DISEÑO MECANICO 6	1	C06	6	identical
13	DISEÑO MECANICO 7	1	C07	6	identical
14	CURSILLO PREPARATORIO A DISTANCIA	0	CAD	3	identical
15	CURSO PREPARATORIO A DISTANCIA	1	CAD	5	identical
16	CURSOS DE ACTUALIZACIÓN	1	CAT	6	identical
17	ADMINISTRACION DE LA CALIDAD	1	CC	3	identical
18	CURSO INTRODUCTORIO (CIMEI)	0	CI	3	identical
19	CURSO DE ADMISION	1	CI	6	identical
20	CURSO A DISTANCIA DE COMPRENSION LECTORA	1	CLD	6	identical

El campo *flagfield* = *identical* significa que los datos ya habían sido transferidos a SIU-Guarani3 y que no se han modificado luego de una nueva ejecución de trabajo de migración.

Con el objetivo de verificar el funcionamiento del Control de Cambios implementado en las transformaciones de carga realizamos la modificación en SQL del campo “nombre” que corresponde a la propuesta 7 y además efectuamos el ingreso de un nuevo registro; ambas operaciones deberán actualizarse en la tabla *sga_propuestas* de SIU-Guarani3. Ejecutamos nuevamente el trabajo y consultamos el resultado que se observa a continuación:

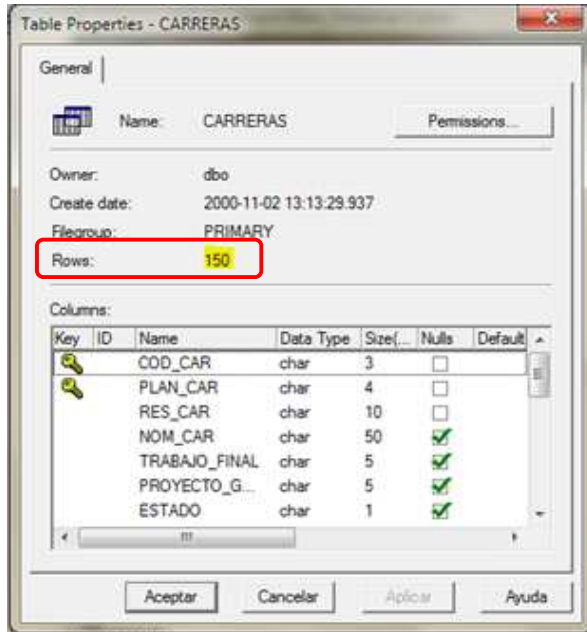
propuesta	nombre	estado	codigo	propuesta_tipo	flagfield
1	ACTIVIDAD DE EXTENSIÓN	0	0	3	identical
2	ACTIVIDAD DE EXTENSIÓN	0	A5	3	identical
3	CURSO PSGR INAEROESPAC	1	AER	6	identical
4	ALEMAN	1	ALE	6	identical
5	PPG Auditoría en Salud	1	AUD	6	identical
6	PPG Auditoría en Salud	1	AUD	5	identical
7	ACTIVIDAD DE EXTENSIÓN Modificado	1	C01	6	changed
8	ACTIVIDAD DE EXTENSIÓN	1	C02	6	identical
9	ACTIVIDAD DE EXTENSIÓN	1	C03	6	identical
10	DISEÑO MECANICO 4	1	C04	6	identical
11	DISEÑO MECANICO 5	1	C05	6	identical
12	DISEÑO MECANICO 6	1	C06	6	identical
13	DISEÑO MECANICO 7	1	C07	6	identical
14	CURSILLO PREPARATORIO A DISTANCIA	0	CAD	3	identical
15	CURSO PREPARATORIO A DISTANCIA	1	CAD	5	identical
16	CURSOS DE ACTUALIZACIÓN	1	CAT	6	identical
17	ADMINISTRACION DE LA CALIDAD	1	CC	3	identical
18	CURSO INTRODUCTORIO (CIMEI)	0	CI	3	identical
19	CURSO DE ADMISION	1	CI	6	identical
20	CURSO A DISTANCIA DE COMPRENSION LECTOR	1	CLD	6	identical
151	TECNICO SUPERIOR EN ADMINISTRACION DE EN	0	TS	3	new

Inspeccionamos que en el campo de bandera o marca (flagfield) se modificó a “*changed*” en el registro que corresponde al dato cambiado y se determinó en “*new*” sobre el registro último creado, siendo acorde a los efectos de funcionar efectivamente.

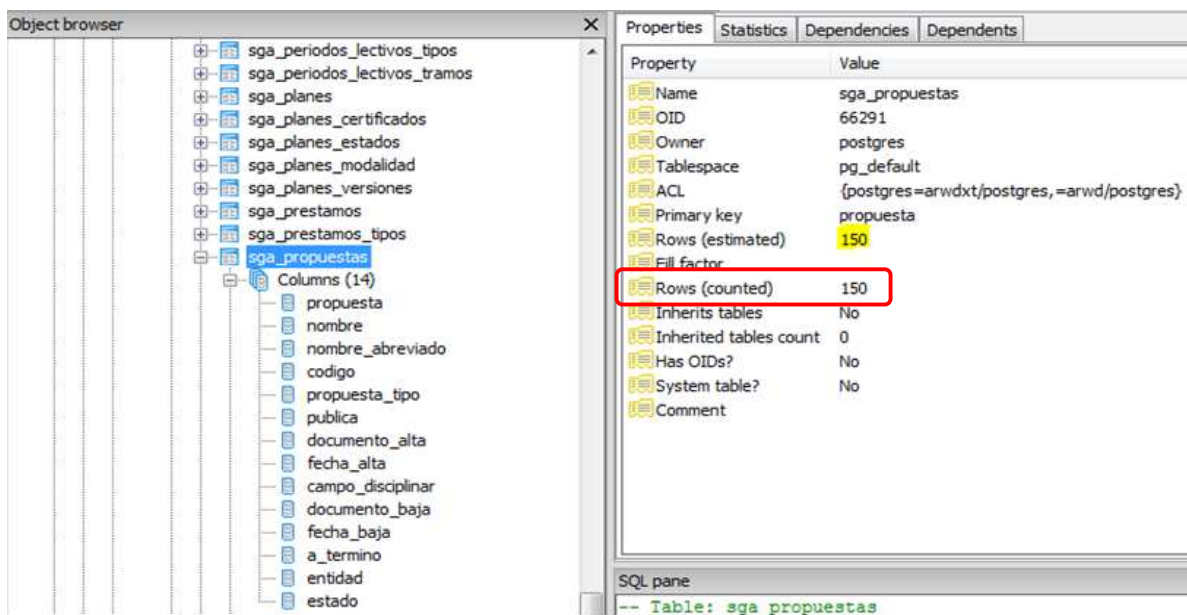
9.1.4 Caso de Test 4 – Control de Cantidad de Carreras Migradas

Efectuamos una consulta en SQL Server de la totalidad de Carreras cargadas en la Base de Datos del IUA:

Carreras en IUA



Verificamos que la cantidad de Carreras cargadas corresponde a un total de 150. Asimismo, ingresamos a PostgreSQL e inspeccionamos el valor de Rows (counted) registrado en la tabla *sga_propuestas* y efectivamente coincide con el total de Carreras que fueron transferidos desde la BD del IUA:



De esta manera, se concluye que los resultados obtenidos son los esperados, siendo efectiva la validación para el control de la cantidad de Carreras migradas.

9.1.5 Caso de Test 5 – Control de Combinación de Registros de Diferentes Tablas

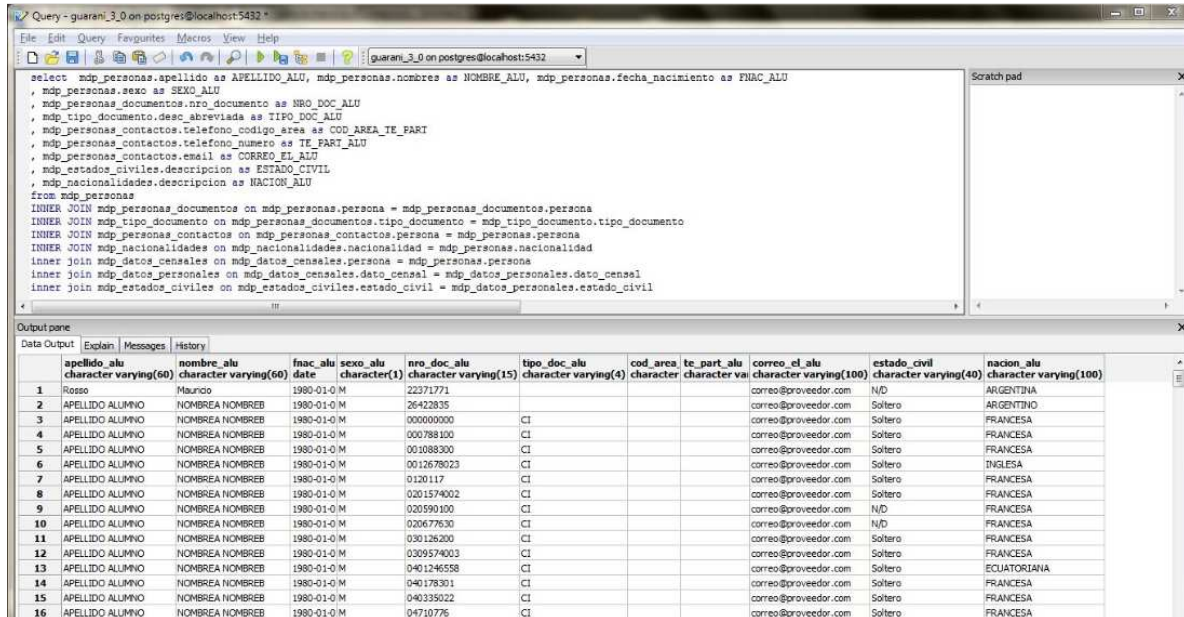
Efectuamos una consulta de la tabla *Alumno* en la Base de Datos del IUA, donde se analizará la integridad y existencia de los primeros 16 registros transferidos a SIU-Guarani3. A continuación observamos el Select ejecutado en SQL:

The screenshot shows a window titled "SQL Query Analyzer - [Query - MAURICIO-PC_Base_Distancia.mauricio-PC\mauricio - Untitled1]*". The query entered is: `select APELLIDO_ALU, NOMBRE_ALU, FNAC_ALU, SEXO_ALU, NRO_DOC_ALU, TIPO_DOC_ALU, COD_AREA_TE_PART, TE_PART_ALU, CORREO_EL_ALU, ESTADO_CIVIL, NACION_ALU from alumno;`

	APELLIDO_ALU	NOMBRE_ALU	FNAC_ALU	SEXO_ALU	NRO_DOC_ALU	TIPO_DOC_ALU	COD_AREA_TE...	TE_PAR...	CORREO_EL_ALU	ESTADO_CIVIL	NACION_ALU
1	Rosso	Mauricio	1980-01-01 0...	M	22371771		NULL	NULL	correo@proveedor.co...	NULL	ARGENTINA
2	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	26422835		NULL	NULL	correo@proveedor.co...	Soltero	ARGENTINO
3	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	000000000	CI	NULL	NULL	correo@proveedor.co...	Soltero	FRANCESA
4	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	000788100	CI	NULL	NULL	correo@proveedor.co...	Soltero	FRANCESA
5	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	001088300	CI	NULL	NULL	correo@proveedor.co...	Soltero	FRANCESA
6	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	0012678023	CI	NULL	NULL	correo@proveedor.co...	Soltero	INGLESA
7	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	0120117	CI	NULL	NULL	correo@proveedor.co...	Soltero	FRANCESA
8	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	0201574002	CI	NULL	NULL	correo@proveedor.co...	Soltero	FRANCESA
9	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	020590100	CI	NULL	NULL	correo@proveedor.co...	NULL	FRANCESA
10	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	020677630	CI	NULL	NULL	correo@proveedor.co...	NULL	FRANCESA
11	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	030126200	CI	NULL	NULL	correo@proveedor.co...	Soltero	FRANCESA
12	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	0309574003	CI	NULL	NULL	correo@proveedor.co...	Soltero	FRANCESA
13	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	0401246558	CI	NULL	NULL	correo@proveedor.co...	Soltero	ECUATORIANA...
14	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	040178301	CI	NULL	NULL	correo@proveedor.co...	Soltero	FRANCESA
15	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	040335022	CI	NULL	NULL	correo@proveedor.co...	Soltero	FRANCESA
16	APELLIDO ALUMN...	NOMBREA NOMBR...	1980-01-01 0...	M	04710776	CI	NULL	NULL	correo@proveedor.co...	Soltero	FRANCESA

Cabe aclarar que por una cuestión de normalización y estructura de SIU-Guarani3 los datos se migran a diferentes tablas en PostgreSQL y no están contenidos en una sola (*tabla Alumno*) como lo es en la base de datos del IUA.

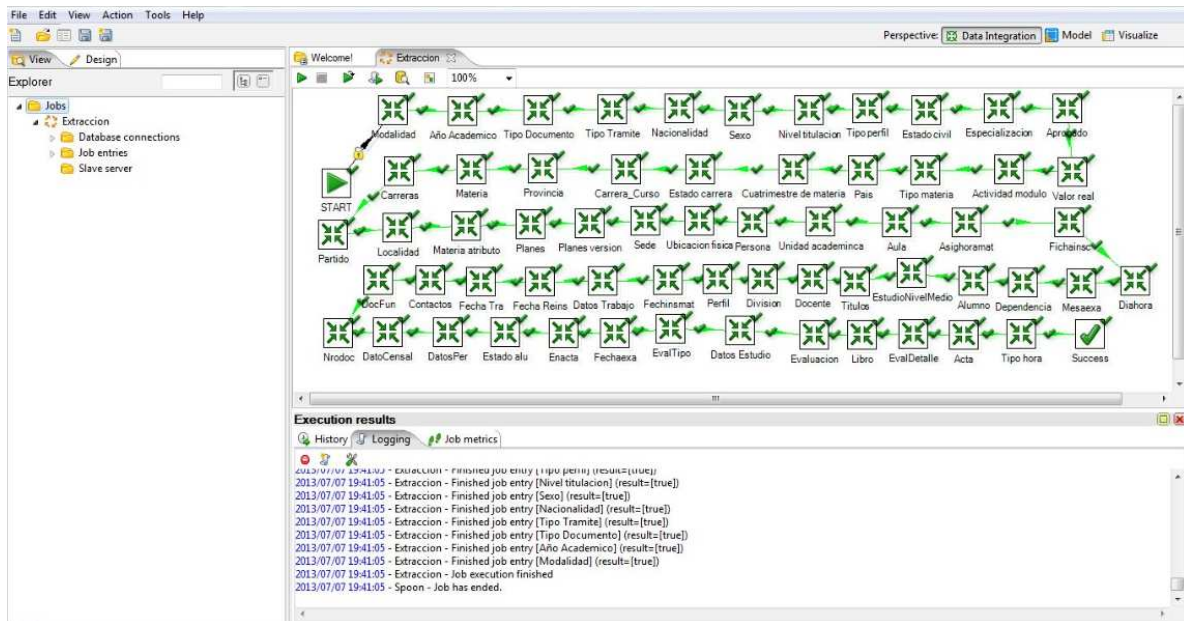
Para verificar que los registros de la tabla señalada se hayan transferido e integrado correctamente en SIU-Guarani3 realizamos la siguiente consulta que permite calcular el producto cruzado de los registros analizados en las diferentes tablas involucradas de PostgreSQL:



Verificamos que los 16 registros inspeccionados se integraron adecuadamente, garantizando de esta manera integridad y exactitud en la información migrada.

9.1.6 Caso de Test 6 – Control de Ejecución Trabajo de Extracción de Datos

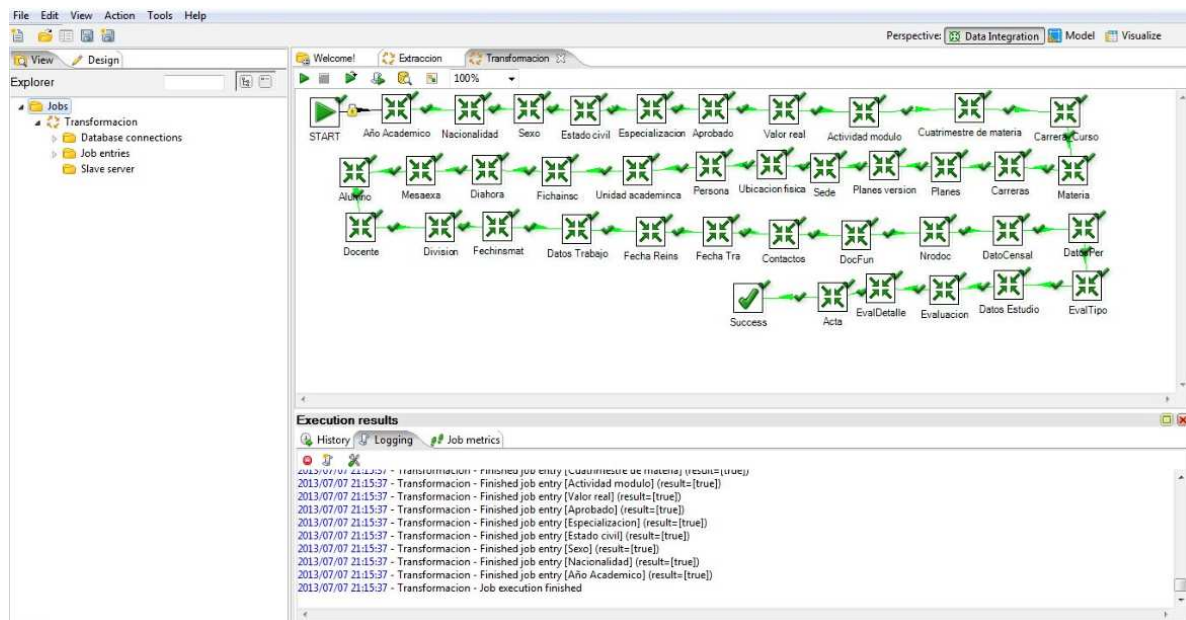
Ejecutamos el trabajo que contiene ordenadamente todas las transformaciones de Extracción de Datos:



Una vez ejecutado, inspeccionamos que la finalización del trabajo ha resultado ser exitosa, situándonos en un contexto en el que todos los datos extraídos de la base de datos del IUA ya residen en una base de datos Stage, y están dadas las condiciones para iniciar con la tarea de limpieza y depuración de los datos.

9.1.7 Caso de Test 7 – Control de Ejecución Trabajo Depuración y Limpieza de Datos

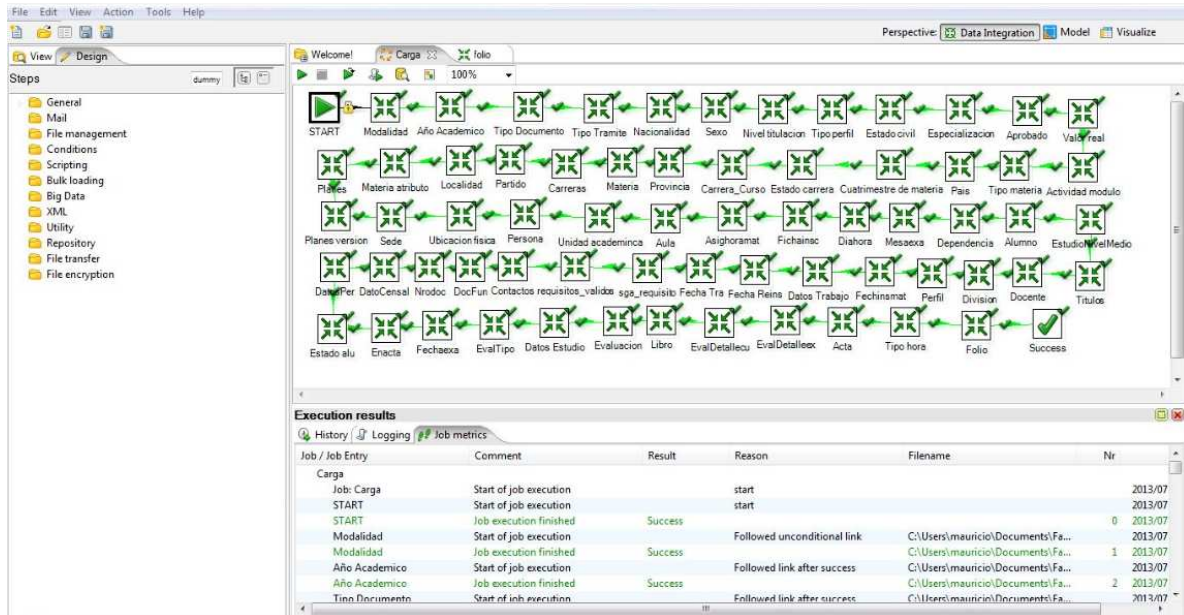
Ejecutamos el trabajo que contiene ordenadamente todas las transformaciones de Depuración y Limpieza de Datos:



Una vez ejecutado, inspeccionamos que la finalización del trabajo ha resultado ser exitosa, situándonos en un contexto en el que todos los datos ya se encuentran depurados en Stage, y están dadas las condiciones para iniciar con la transferencia (carga y control de cambios) de la información a la base de datos de SIU-Guarani3.

9.1.8 Caso de Test 8 – Control de Ejecución Trabajo Carga y Control de Cambios

Ejecutamos el trabajo que contiene ordenadamente todas las transformaciones de Carga y Control de Cambios:



Una vez ejecutado, inspeccionamos que la finalización del trabajo ha resultado ser exitosa, situándonos en un contexto en el que todos los datos ya se encuentran transferidos a SIU-Guarani3.

9.1.9 Caso de Test 9 – Control de Ejecución Trabajo Global

Ejecutamos el trabajo global que invoca a los 3 (tres) trabajos que se validaron anteriormente:

The screenshot displays an ETL (Extract, Transform, Load) job execution interface. The top part shows a workflow diagram with five stages: START, Extraccion, Transformacion, Carga, and Success. The bottom part shows the 'Execution results' table, which provides a detailed log of the job's progress.

Job / Job Entry	Comment	Result	Reason	Filename	Nr	Log date
ETL						
Job: ETL	Start of job execution		start			2013/07/29 23:02:47
START	Start of job execution		start			2013/07/29 23:02:47
START	Job execution finished	Success			0	2013/07/29 23:02:47
Extraccion	Start of job execution		Followed unconditional link	C:\Users\ma...		2013/07/29 23:02:47
Extraccion	Job execution finished	Success		C:\Users\ma...	1	2013/07/29 23:26:40
Transformacion	Start of job execution		Followed link after success	C:\Users\ma...		2013/07/29 23:26:40
Job: Transformacion						
Transformacion	Job execution finished	Success		C:\Users\ma...	2	2013/07/29 23:35:11
Carga	Start of job execution		Followed link after success	C:\Users\ma...		2013/07/29 23:35:11
Job: Carga						
Carga	Job execution finished	Success		C:\Users\ma...	3	2013/07/29 23:47:42
Success	Start of job execution		Followed link after success			2013/07/29 23:47:42
Success	Job execution finished	Success			3	2013/07/29 23:47:42
Job: ETL	Job execution finished	Success	finished		3	2013/07/29 23:47:42

Verificamos que la ejecución del trabajo global finalizó satisfactoriamente, lo que nos determina que cualquier transferencia requerida a SIU-Guarani3 sea realizada a partir de dicho trabajo.

De esta manera se concluye con la etapa de pruebas, corroborando que el resultado de cada uno de los casos de test planificados ha sido el esperado, siendo el producto bajo pruebas el adecuado para el propósito del proyecto, cumpliendo con los criterios de aceptación definidos por los interesados de dicho proyecto.

9.2 PUESTA EN MARCHA

9.2.1 Infraestructura Computacional

La plataforma tecnológica existente hoy en día es la siguiente:

- Arquitectura: 64 bits.
- Procesador: Intel Xeon E5420 – 2.5 GHz
- Pentaho instalado sobre Windows 2003 Server.

9.2.2 Configuración de la Base de Datos

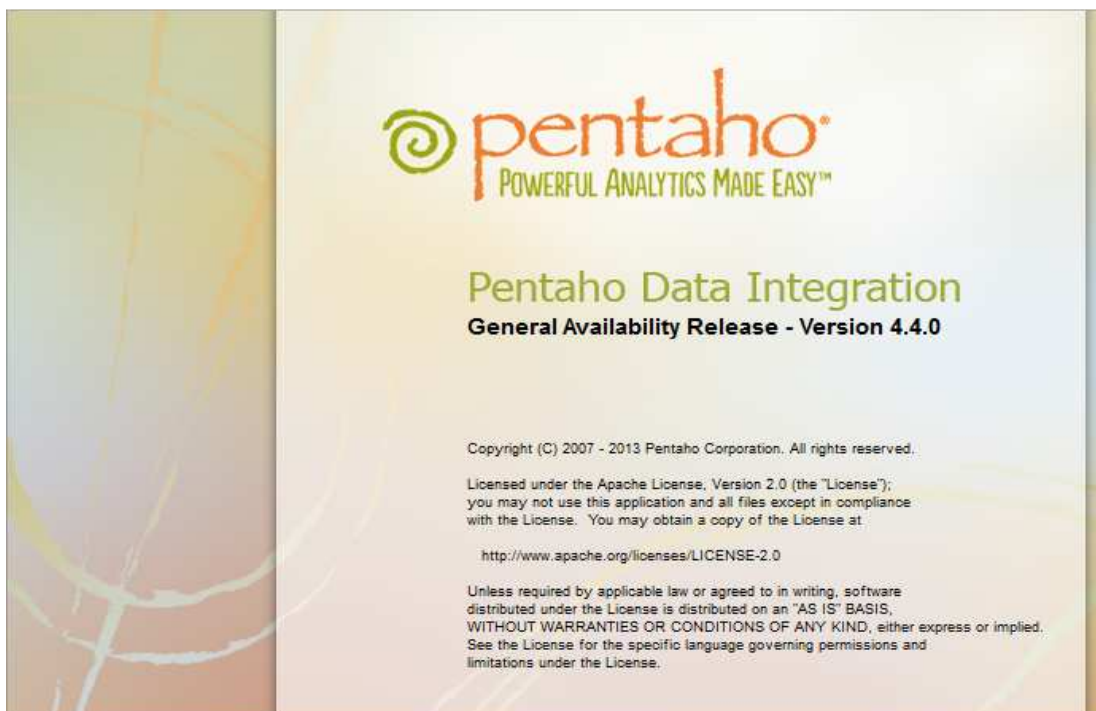
- Para la configuración del motor de base de datos PostgreSQL se deben realizar los siguientes pasos:
- Descargar el paquete postgresQL 8.3 de página de descargas de postgresQL: <http://www.postgresql.org/download/>
- Instalar el paquete postgresQL 8.3 que incluye la interfaz gráfica pgAdmin III.
- Configurar la conexión local que se va a utilizar para la creación de las bases de datos. Para ello se ingresan los siguientes datos:
 - Nombre: PostgreSQL 8.3
 - Servidor: localhost
 - Puerto: 5432
 - BD de Mantenimiento: postgres
 - Usuario: postgres
 - Contraseña: *****
 - Servicio: postgresql-8.3

9.2.3 Configuración de Pentaho

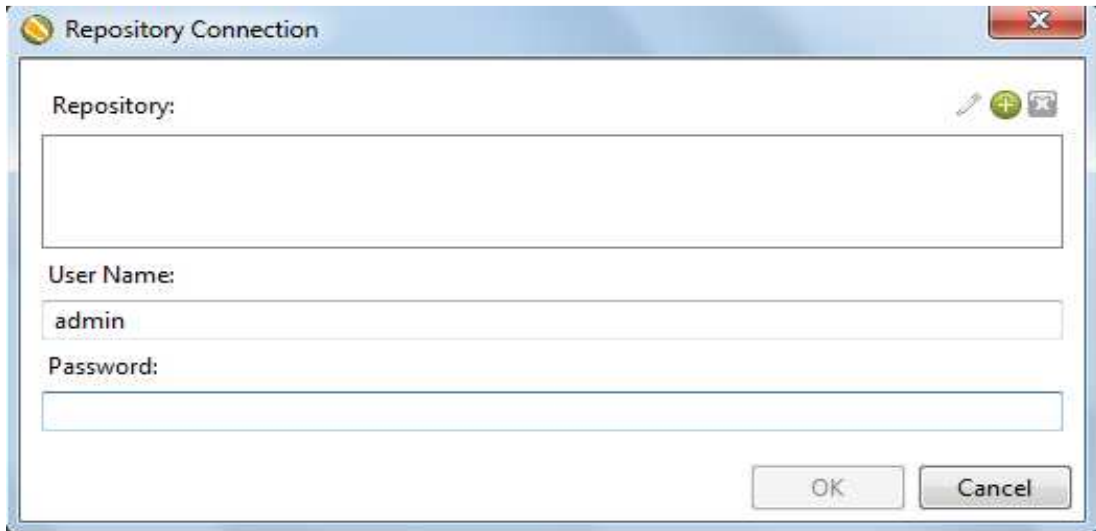
Para la configuración de Pentaho se debe descargar la suite Pentaho de la página:

<http://sourceforge.net/projects/pentaho/files/Data%20Integration/4.4.0-stable/>, luego se tienen que configurar las herramientas que se utilizarán para el proceso ETL, que son las siguientes:

- **Plataforma:** para la instalación de la plataforma se sigue el Wizard que proporciona el instalador de Pentaho. En ella se registran las contraseñas que se utilizarán para el usuario administrador y para realizar publicaciones. Además se incluye la creación de un usuario cliente estándar con una contraseña por defecto (Joe/password).
- **Data Integration (Kettle):** Esta herramienta servirá para la construcción de los procesos ETL, específicamente se usará la aplicación Spoon, una interfaz gráfica que permite arrastrar los componentes de entrada, transformación y salida de los procesos. Para su configuración se debe iniciar Pentaho Data Integration desde la ruta: C:\Archivos de programa\pentaho\design-tools\data-integration\spoon.bat y se mostrará la siguiente pantalla de inicio:



Luego de unos segundos se oculta automáticamente la pantalla de inicio y se abre la siguiente ventana para el logon de acceso a Spoon:



Las conexiones a las Bases de Datos que serán utilizadas en nuestro proyecto de migración se establecen de la siguiente manera:

- Base_Distancia: base de datos del Sistema de Gestión Académica del Instituto Universitario Aeronáutico.

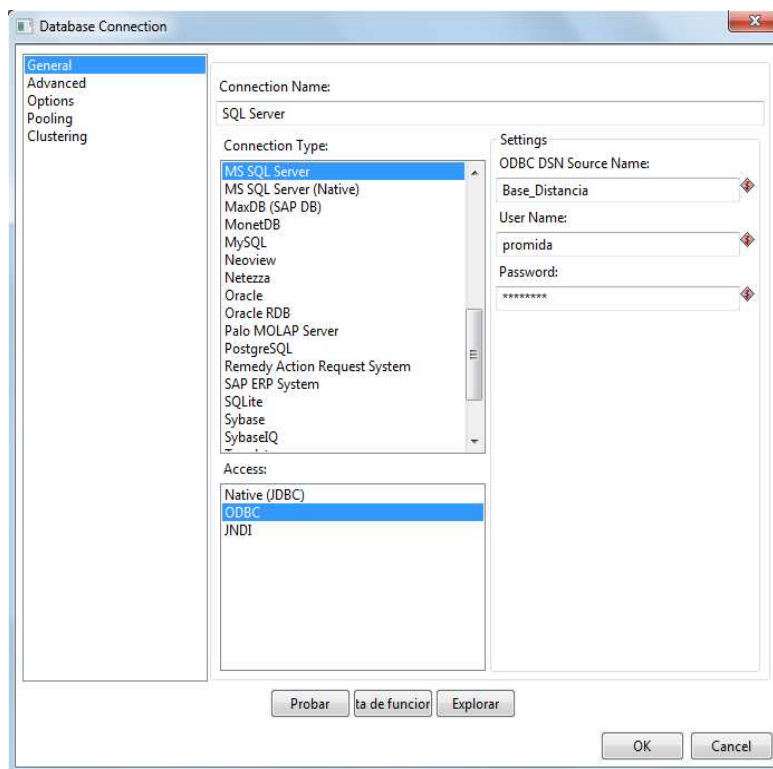


Ilustración A – Conexión a Base_Distancia

- Stage: base de datos intermedia donde se realiza la transformación de datos (depuración y limpieza).

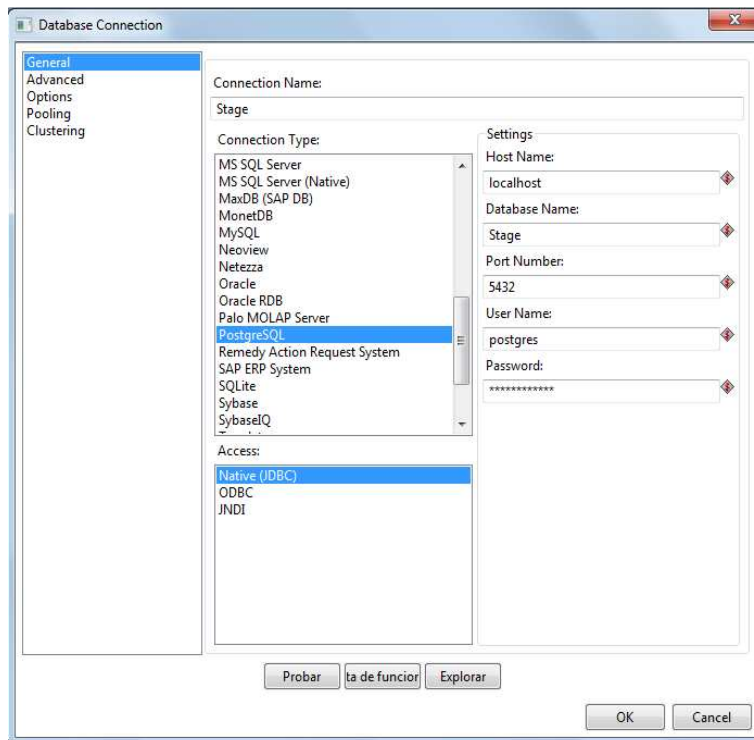


Ilustración B – Conexión a Stage

- SIU-Guarani3: base de datos donde se transfieren o migran los datos transformados.

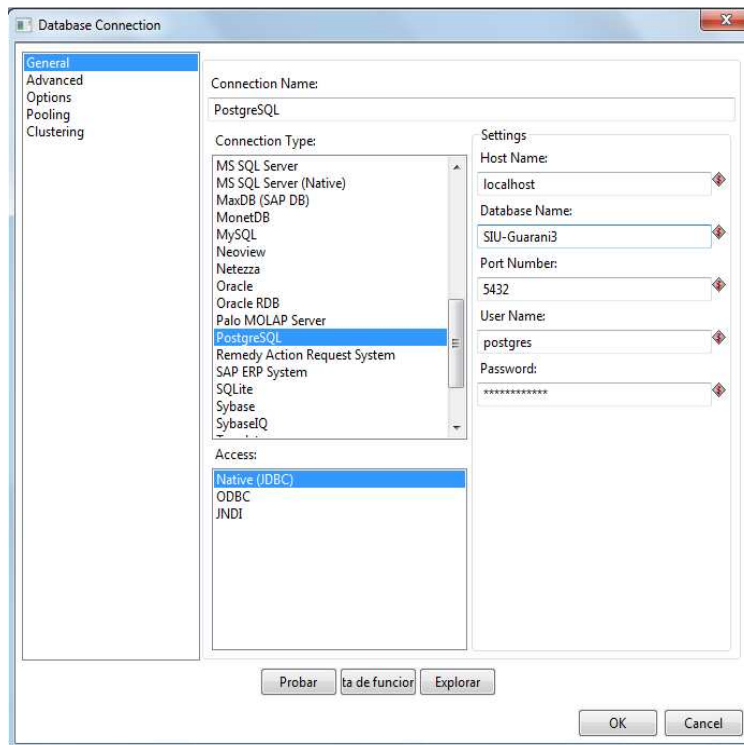


Ilustración C – Conexión a SIU-Guarani3

Etapa 6

CONCLUSIONES

Finalmente se ha llegado a la conclusión del presente proyecto, recorriéndose un largo camino hacia un final exitoso, no sólo por los resultados obtenidos, sino también por la experiencia en sí misma.

Recordando que el objetivo general de este trabajo es realizar un proceso de migración que permita garantizar la transferencia adecuada de los datos desde la Base de Datos del Sistema Integral de Gestión Académica del I.U.A a la Base de Datos del Sistema SIU-Guaraní3, se efectuó una recopilación de los aspectos esenciales que deben ser considerados a la hora de construir un proceso ETL para la migración de los datos, poniendo en práctica tales aspectos con la incorporación de una herramienta Open Source que permitió facilitar el diseño y construcción para la integración de los datos. Con este complemento, se pudo especificar y estandarizar toda una secuencia de pasos lógicos para trasladar la información entre los diferentes motores de bases de datos inmersos en el proyecto.

Considerando los objetivos específicos y su cumplimiento en el trabajo realizado se deduce lo siguiente:

- “Garantizar la consistencia, integridad y seguridad necesaria de los datos que serán transferidos al sistema destino”. Luego de inspeccionar los resultados de la ejecución de transformaciones, trabajos, y pruebas que se dejaron plasmados en el presente documento se verificó que se ha cumplido con los tres conceptos señalados.
- “Seleccionar la/s herramienta/s más adecuada/s para llevar a cabo la migración y conversión de los datos desde la base de datos de origen a la de destino”. A través de una comparación multicriterio por scoring entre alternativas de herramientas ETL Open Source, Kettle de la suite Pentaho obtiene la ponderación más alta, siendo la mejor alternativa; dicha plataforma tiene una mayor robustez y versatilidad, incluyendo todos los componentes requeridos para dar solución a lo que exige el proyecto.
- “Seleccionar la/s metodología/s más adecuada/s para llevar a cabo la migración y conversión de los datos desde la base de datos de origen a la de destino”. Se detallaron

los principales aspectos a considerar en cada una de las etapas del diseño de la metodología ETL en la cual se incluye la definición de algunas técnicas para optimizar el proceso. Se incluyen las recomendaciones fundamentales que se deben tener en cuenta a la hora de diseñar y desarrollar la etapa de integración de datos de un proyecto de migración. Sin embargo, es importante tener en cuenta que no existe un enfoque estándar para abordar este tipo de proyectos, ya que cada contexto organizacional es distinto. La idea es tener las bases para poder escoger los aspectos y recomendaciones que mejor se adapten al contexto organizacional en que se está trabajando.

- “Disponer de datos históricos del sistema de información actual en el nuevo sistema”. Luego de la ejecución del trabajo final, que engloba la corrida secuencial de los trabajos de Extracción, Transformación, y Carga, verificamos que la información histórica relacionada con la Persona, la Carrera, y el Cursado incluida en el proceso de migración existe en la base de datos de SIU-Guaraní3.
- “Contar con una base de datos intermedia (Staging) como parte del proceso de migración”. Se ha creado una base de datos Staging que permite cambios de ella hacia atrás, es decir en dirección de la fuente, otorgando la posibilidad de crear una migración desde otro sistema a SIU-Guaraní3. Esta situación se traduce a que el proceso podrá ser reutilizado para futuros proyectos cuyo objetivo sea migrar sus datos a SIU-Guaraní3.

Como resultado del trabajo desarrollado, se ha presentado una solución al problema de la migración de los datos para adecuarse a los nuevos requisitos que surgen con el paso del tiempo. Se ha mostrado la arquitectura de una herramienta Open Source que dirige a conexiones exitosas, con una interfaz amigable, intuitiva, y precisa, claves en el éxito del funcionamiento del proceso y en consecuencia al traslado efectivo de la información.

El proyecto “ProMiDa” ha resultado adecuado a los propósitos planteados inicialmente en el TFG, a un bajo costo, flexible y adaptable, pudiendo construir un primer prototipo para la migración de datos a SIU-Guaraní3.

10. REFERENCIAS

- *Pentaho Kettle Solutions. Building Open Source ETL Solutions with Pentaho Data Integration: Matt Casters, Roland Bauman, Jos Van Dongen*
- *Groff, J. Sql Manual de Referencia. 2.ed. Madrid-España: MC Graw Hill, 2003*
- *Metodología ETL acorde a Buenas Prácticas y Recomendaciones de la disciplina. Memoria (Ingeniero Civil en Informática y Telecomunicaciones) -- Universidad Diego Portales. Autor: Silva Torres, Carolina. Santiago, Chile, 2012*
- *Factores críticos de éxito en el proceso de migración de bases de datos relacionales. Obtenido de: <http://www.paginaspersonales.unam.mx>*
- http://es.wikipedia.org/wiki/Migraci%C3%B3n_de_datos
- <http://community.pentaho.com>
- <http://wiki.pentaho.com/display/EAI/Latest+Pentaho+Data+Integration+%28aka+Kettle%29+Documentation>
- <http://churriwifi.wordpress.com/2010/05/01/16-procesos-etl-escenarios-para-el-diseno-de-los-procesos>
- <http://www.redopenbi.com/group/pentahodataintegration>
- <http://es.scribd.com/doc/97615507/Lab-ETL-Pentaho>
- <http://forums.pentaho.com/forum.php>
- <http://repositorio.siu.edu.ar/trac/Portal-G3>