Guía de Trabajo: Diseño de un Modelo Dimensional para SECOP II

*Procesamiento y Análisis Masivo de Datos Escuela de Ingeniería Julio Garavito Docente: Carlos Isaac Zainea*

*Estudiantes: Miguel Ángel Hernández Vargas y Juan Sebastián Fajardo Acevedo*

**Objetivo:** Aprender a transformar una tabla de datos masiva y compleja en un modelo de datos optimizado (Esquema de Estrella) para el análisis. Esta guía te enseñará a *pensar* como un ingeniero de datos antes de ejecutar el pipeline.

# Parte 1: La Filosofiía - ¿Por Qué No Usar Una Sola Tabla Gigante?

Intentar analizar una tabla con millones de filas y decenas de columnas es como intentar encontrar un libro específico en una biblioteca donde todos los libros están apilados en una sola montaña gigante. Es lento, ineficiente y propenso a errores.

El **modelado dimensional** nos ayuda a organizar esa biblioteca. Creamos un **catálogo de tarjetas de referencia** (las **Tablas de Dimensiones**) que describen los libros (autor, género, año de publicación) y una **lista principal de préstamos** (la **Tabla de Hechos**) que simplemente registra qué libro se prestó y cuándo, usando referencias a las tarjetas del catálogo.

1. **Tablas de Hechos (Facts):** Contienen el **"qué pasó"**. Son el corazón de nuestro análisis. Registran eventos o transacciones. Sus características principales son:
   * Son muy largas (millones o miles de millones de filas).
   * Son "delgadas" (pocas columnas).
   * Contienen **métricas numéricas** (ej: valor del contrato, días adicionados) y claves foráneas (referencias a las dimensiones).
2. **Tablas de Dimensiones (Dimensions):** Contienen el **"quién, qué, dónde, cuándo, cómo y por qué"**. Describen el contexto de los hechos. Sus características son:
   * Son más cortas y anchas (muchas columnas descriptivas).
   * Contienen texto y atributos que no cambian tan a menudo (ej: nombre de la entidad, categoría del proveedor).
   * Tienen una clave primaria única.

# Parte 2: Taller Práctico - Diseñando Nuestro Esquema para SECOP II

## Paso 1: Hacer las Preguntas del Negocio (¡El Paso Más Importante!)

Antes de modelar, pregúntate: **¿Qué queremos responder con estos datos?**

* ¿Qué entidades públicas gastan más dinero?
* ¿En qué departamentos se firman los contratos de mayor valor?
* ¿Cuál es la modalidad de contratación más común para contratos de tecnología?
* ¿Qué proveedores ganan más contratos con el estado?
* ¿Cómo ha evolucionado el valor de los contratos a lo largo del tiempo?

Estas preguntas nos dan pistas sobre qué hechos (métricas) y dimensiones (contexto) son importantes.

## Paso 2: Identificar el "Grano" de la Tabla de Hechos

Pregúntate: ¿Qué representa una única fila en nuestra tabla de hechos? Esta es la decisión más fundamental del diseño. Define el nivel de detalle de tu análisis.

Para el SECOP, la respuesta es simple y clara:

El grano de nuestra tabla de hechos es **un contrato individual**.

## Paso 3: Identificar las Métricas (Los Hechos)

Revisa la lista de columnas del SECOP y busca los **campos puramente numéricos y medibles** asociados a cada contrato. Estos serán los pilares de tu tabla de hechos.

* valor\_del\_contrato
* valor\_pagado
* valor\_pendiente\_de\_pago
* dias\_adicionados
* valor\_de\_pago\_adelantado
* *...y cualquier otra métrica que te interese.*

Tu tabla fact\_contratos empezará a verse así:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id\_contrato (PK)** | **valor\_del\_contr ato** | **valor\_pagado** | **dias\_adicionad os** | **...FKs...** |
| CO1.CON.1234 5 | 150000000 | 75000000 | 30 | ... |

## Paso 4: Identificar el Contexto (Las Dimensiones)

Ahora, busca los campos descriptivos que responden al "quién, qué, dónde, cuándo". Agrupa las columnas relacionadas para crear tus tablas de dimensiones.

### Dimensión: dim\_entidades (El QUIÉN contrata)

* + id\_entidad (Clave Primaria, generada por nosotros)
  + nit\_entidad
  + nombre\_entidad
  + departamento\_entidad
  + ciudad\_entidad
  + orden (Nacional, Departamental...)
  + sector

### Dimensión: dim\_proveedores (El A QUIÉN se contrata)

* + id\_proveedor (Clave Primaria, generada por nosotros)
  + documento\_proveedor
  + proveedor\_adjudicado
  + es\_pyme

### Dimensión: dim\_geografia (El DÓNDE se ejecuta)

* + id\_geografia (Clave Primaria, generada por nosotros)
  + departamento
  + ciudad
  + *Podrías enriquecerla con códigos DANE, región, etc.*

### Dimensión: dim\_tipo\_contrato (El CÓMO se contrata)

* + id\_tipo\_contrato (Clave Primaria, generada por nosotros)
  + tipo\_de\_contrato
  + modalidad\_de\_contratacion

### Dimensión: dim\_tiempo (El CUÁNDO se firma)

* + id\_tiempo (Clave Primaria, formato YYYYMMDD)
  + fecha\_completa
  + año
  + mes
  + trimestre
  + dia\_de\_la\_semana

1. ¡Tu Turno! Descubriendo Más Dimensiones

La lista de columnas es enorme. Hemos cubierto las dimensiones más obvias, pero hay más información valiosa escondida. Házte las siguientes preguntas para descubrir nuevas dimensiones:

**Dimensiones propuestas.**

1. **Dimensión: dim\_modalidad\_adicional**

Agrupa información sobre formas específicas de contratación o modalidades especiales.

* **id\_modalidad**: Clave primaria única.
* **modalidad\_contrato**: Texto descriptivo de la modalidad (por ejemplo: Licitación Pública, Contratación Directa, Concurso de Méritos).
* **es\_tecnologia**: Flag (Sí/No) si aplica para contratos de tecnología.
* **es\_servicios**: Flag (Sí/No) si aplica para servicios.
* **es\_obra**: Flag (Sí/No) si aplica para obra pública.

1. **Dimensión: dim\_tipo\_pago**

Describe la forma y condiciones de **los pagos del contrato**.

* **id\_tipo\_pago**: Clave primaria única.
* **tipo\_pago**: Texto descriptivo (ejemplo: Pago único, Pago por hitos, Pago adelantado).
* **permite\_anticipos**: Flag (Sí/No) si permite pagos adelantados.
* **plazo\_maximo\_pago**: Número de días máximo permitido para el pago.

**I. dim\_entidad\_supervision**

Representa **la entidad que supervisa o fiscaliza el contrato**.

* **id\_entidad\_supervision**: Clave primaria única.
* **nombre\_entidad\_supervision**: Texto con el nombre de la entidad supervisora.
* **tipo\_entidad\_supervision**: Texto (ejemplo: Ministerio, Gobernación, Alcaldía).
* **departamento**: Departamento donde opera la entidad supervisora.
* **ciudad**: Ciudad donde opera la entidad supervisora.

**J. dim\_categoria\_contrato**

Clasifica los contratos según su **objetivo o tipo de gasto**.

* **id\_categoria\_contrato**: Clave primaria única.
* **categoria**: Texto descriptivo (ejemplo: Salud, Educación, Infraestructura, Tecnología, Servicios).
* **subcategoria**: Texto más específico si aplica (ejemplo: Construcción de vías, Software, Capacitación).
* **es\_prioritario**: Flag (Sí/No) si el contrato es de prioridad nacional o estratégica.
  + **Pregunta 1:** Observa las columnas estado\_contrato, liquidaci\_n, habilita\_pago\_adelantado y el\_contrato\_puede\_ser\_prorrogado. ¿Podrían agruparse en una dim\_estado\_contrato que describa el ciclo de vida y las condiciones administrativas del contrato? **Pista:** Esta dimensión contendría texto y valores booleanos (Si/No).

**Dimensión: dim\_estado\_contrato**

* id\_estado\_contrato: Clave primaria única generada (entero).
* estado\_contrato: Texto descriptivo que indica el estado administrativo del contrato (por ejemplo: Activo, Suspendido, Liquidado, Terminado).
* es\_liquidado: Indicador (Sí/No) que señala si el contrato ya fue liquidado.
* habilita\_pago\_adelantado: Indicador (Sí/No) que señala si el contrato permite realizar pagos adelantados.
* puede\_ser\_prorrogado: Indicador (Sí/No) que señala si el contrato puede ser prorrogado en el tiempo.
  + **Pregunta 2:** Las columnas sobre el dinero como origen\_de\_los\_recursos, destino\_gasto y las columnas de presupuesto\_general\_de\_la\_nacion\_pgn, sistema\_general\_de\_regal\_as, etc., son cruciales. ¿Cómo diseñarías una dim\_recursos\_financieros para categorizar la fuente de los fondos? **Pista:** Podrías tener una columna para el origen principal y varias columnas tipo "bandera" (flag) para los diferentes tipos de recursos.

**Dimensión: dim\_recursos\_financieros**

* id\_recurso: Clave primaria única (entero).
* origen\_principal: Texto que indica de dónde provienen los recursos (ejemplo: Presupuesto General de la Nación, Sistema General de Regalías, Recursos propios, etc.).
* destino\_gasto: Texto que indica a qué se orienta el gasto.
* flag\_pgn: Indicador (Sí/No) si provienen del Presupuesto General de la Nación.
* flag\_regalias: Indicador (Sí/No) si provienen de Regalías.
* flag\_sgp: Indicador (Sí/No) si provienen del Sistema General de Participaciones.
* flag\_otros: Indicador (Sí/No) para otros tipos de fuentes.
  + **Pregunta 3:** Columnas como espostconflicto, obligaci\_n\_ambiental u obligaciones\_postconsumo son "banderas" (flags) importantes que califican al contrato.

¿Valdría la pena crear una dim\_marcadores\_especiales para agrupar estos indicadores?

Sí, creemos que es valioso hacer esa dimensión. Ya que la idea de la dimensión dim\_responsables es unificar en una sola tabla todas las personas relacionadas con un contrato, como el ordenador del gasto, el supervisor o el ordenador de pago, en lugar de crear tres dimensiones separadas. Esto simplifica el modelo, reduce redundancia y permite asignar a cada persona un rol específico dentro del contrato, facilitando consultas y mantenimientos futuros sin perder información sobre quién hace qué.

**Dimensión: dim\_marcadores\_especiales**

Agrupa banderas (flags) que cualifican al contrato con condiciones particulares.

* id\_marcador: Clave primaria única (entero).
* es\_postconflicto: Indicador (Sí/No) si el contrato se relaciona con el postconflicto.
* obligacion\_ambiental: Indicador (Sí/No) si tiene obligaciones ambientales.
* obligacion\_postconsumo: Indicador (Sí/No) si incluye obligaciones postconsumo.
  + **Pregunta 4 (Desafío):** Analiza las columnas nombre\_ordenador\_del\_gasto, nombre\_supervisor y nombre\_ordenador\_de\_pago. Todas se refieren a personas. ¿Sería eficiente tener una dimensión para cada uno, o podrías crear una única dim\_responsables que contenga a todas las personas y una columna adicional que describa su rol en el contrato?

Creemos que sería más eficiente crear una sola dimensión llamada dim\_responsables que contenga a todas las personas involucradas en los contratos. En lugar de tener tres dimensiones separadas (ordenador del gasto, supervisor, ordenador de pago), concentras la información en una sola tabla y agregas una columna rol\_responsable que indique el rol de cada persona en el contrato. Ya que esto reduce la duplicación de datos, facilita las consultas y mantiene un modelo más limpio y manejable, sin perder detalle sobre quién cumple qué función.

**Dimensión: dim\_responsables**

Unifica todas las personas relacionadas con un contrato (ordenador, supervisor, pagador).

* id\_responsable: Clave primaria única (entero).
* nombre\_responsable: Texto con el nombre completo de la persona.
* tipo\_documento: Identificación del responsable (CC, CE, NIT, etc.).
* rol\_responsable: Texto que describe el rol que cumple en el contrato (ejemplo: Ordenador del gasto, Supervisor, Ordenador de pago).

## Paso 5: Ensamblar el Esquema de Estrella

Ahora, tu tabla de hechos (fact\_contratos) se completa con las claves foráneas (Foreign Keys

- FK) que apuntan a las dimensiones. Será una tabla muy optimizada:

Tabla: fact\_contratos

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **id\_contrato (PK)** | **id\_entidad (FK)** | **id\_proveed or (FK)** | **id\_geografi a (FK)** | **id\_tipo\_con trato (FK)** | **id\_tiempo (FK)** | **valor\_del\_co ntrato** | **dias\_adic ionados** |
| CO1.CON. 12345 | 101 | 5001 | 32 | 4 | 20230515 | 150000000 | 30 |
| CO1.CON. 12346 | 205 | 5002 | 16 | 2 | 20230516 | 8000000 | 0 |

# Parte 3: Optimización y Estrategia de Carga

## ¿Por Qué Este Diseño es Mucho Mejor?

1. **Consultas Más Rápidas:** Las uniones (joins) entre tablas usando claves numéricas (ej: 101) son increíblemente más rápidas que comparar cadenas de texto largas (ej: "INSTITUTO NACIONAL DE VÍAS"). El motor de la base de datos compara números, lo cual es computacionalmente mucho más simple.
2. **Menor Almacenamiento:** Evitas repetir millones de veces textos largos. Solo almacenas un número pequeño en la tabla de hechos. Esto reduce drásticamente el tamaño total de los datos.
3. **Consistencia de Datos (Fuente Única de Verdad):** Si el nombre de una entidad cambia, solo lo corriges en un lugar: dim\_entidades. Todos los análisis que usen esa tabla se actualizarán automáticamente.
4. **Mantenimiento Sencillo:** Las dimensiones cambian poco. Puedes actualizarlas una vez al día. La tabla de hechos es la que crece constantemente.

## La Regla de Oro: Carga la Tabla de Hechos por Lotes

Incluso con este modelo optimizado, la tabla de hechos será masiva. **NUNCA** intentes cargar todos los contratos de la historia en un solo proceso.

* + **¿Por qué?** Agotarás la memoria de Spark, sobrecargarás la API del SECOP y si el proceso falla a la mitad, perderás todo el trabajo.
  + **¿Cómo?** Procesa los datos por lotes temporales. La estrategia más lógica es **procesar los contratos año por año, o incluso mes por mes**.

### Implementación (Pseudo-código):

AÑOS\_A\_PROCESAR = [2022, 2023, 2024]

for año in AÑOS\_A\_PROCESAR:

# 1. Descargar los contratos del 'año' df\_crudo\_año = descargar\_datos\_de\_secop(año)

# 2. Transformar (hacer joins con dimensiones para obtener FKs) df\_hechos\_año = transformar\_a\_tabla\_de\_hechos(df\_crudo\_año)

# 3. Añadir el lote a la tabla Delta en modo 'append' df\_hechos\_año.write.format("delta").mode("append").save("s3a://gold/facts/contratos")

# Parte 4: Pro-Tips para Construir tus Queries SoQL

Para extraer los datos de la API eficientemente, usa el poder de SoQL para que el servidor trabaje por ti.

Tip 1: Pide solo las columnas que necesitas (select)

No traigas las 80+ columnas si solo vas a usar 10 para una dimensión. Es mejor y más seguro usar una librería como sodapy en Python.

# Ejemplo con sodapy para traer solo lo necesario para dim\_entidades import pandas as pd

from sodapy import Socrata

# Conexión al cliente (no se necesita token para datos públicos) client = Socrata("[www.datos.gov.co](http://www.datos.gov.co/)", None)

# El parámetro 'select' corresponde al $select de SoQL

# El parámetro 'limit' es una buena práctica para no sobrecargar la API results = client.get(

"jbjy-vk9h",

select="nit\_entidad, nombre\_entidad, departamento, ciudad, orden", limit=1000

)

results\_df = pd.DataFrame.from\_records(results) print(results\_df.head())

Tip 2: Filtra en el origen, no en casa (where)

No descargues todos los contratos para luego filtrarlos en Spark. Pídele a la API que te dé solo los de 2023.

# El parámetro 'where' en sodapy usa la sintaxis de SoQL

query\_2023 = "fecha\_de\_firma >= '2023-01-01T00:00:00' AND fecha\_de\_firma < '2024-01-01T00:00:00'"

results = client.get("jbjy-vk9h", where=query\_2023, limit=5000)

Tip 3: ¡Usa la agregación de la API para crear dimensiones! (group)

Este es el truco más poderoso. Para crear dim\_entidades, no necesitas descargar todos los contratos. Pídele a la API que te devuelva los valores únicos.

# El parámetro 'group' en sodapy corresponde al $group de SoQL

# Esto es mucho más rápido y eficiente que hacer un .distinct() en Spark results = client.get(

"jbjy-vk9h",

select="nit\_entidad, nombre\_entidad, departamento, ciudad, orden", group="nit\_entidad, nombre\_entidad, departamento, ciudad, orden"

)

Tip 4: Usa siempre la paginación (limit y offset)

Para cualquier consulta que devuelva muchos datos, nunca pidas todo de una vez. Pide los datos en "páginas" dentro de un bucle.

# Trae los primeros 50,000 resultados

results\_pagina\_1 = client.get("jbjy-vk9h", limit=50000, offset=0, where=...)

# En la siguiente iteración, trae los siguientes 50,000

results\_pagina\_2 = client.get("jbjy-vk9h", limit=50000, offset=50000, where=...)

Taller

---- Carga datos

---- dim\_tiempo

---- dim\_responsables