

Trabajo Fin de Grado

Ingeniería de Telecomunicación

Procesamiento y visualización de los datos referentes a la generación y consumo eléctrico en España

Autor: Alfonso Falcón Pérez

Tutor: Juan Antonio Ternero Muñiz

Dpto. Ingeniería Telemática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2022



Trabajo Fin de Grado
Ingeniería de Telecomunicación

**Procesamiento y visualización de los datos
referentes a la generación y consumo eléctrico en
España**

Autor:

Alfonso Falcón Pérez

Tutor:

Juan Antonio Ternero Muñiz

Profesor asociado

Dpto. de Ingeniería Telemática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla
Sevilla, 2022

Trabajo Fin de Grado: Procesamiento y visualización de los datos referentes a la generación y consumo eléctrico en España

Autor: Alfonso Falcón Pérez

Tutor: Juan Antonio Ternero Muñiz

El tribunal nombrado para juzgar el Trabajo arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2022

El Secretario del Tribunal

A mi familia

A mis amigos

A mis maestros

Agradecimientos

Finalizar este proyecto significa ir cerrando mi etapa académica, al menos por ahora. Por ello, y al haber sido tan larga esta etapa, quiero agradecer en primer lugar a mis padres y mi hermano, por apoyarme y no dejarme bajar los brazos cuando no salían las cosas bien.

A mis compañeros de piso que he tenido durante esta etapa, en especial a Paco, que además ha sido compañero de casa de la juventud, de las noches de Champions y de la Isla, y de tardes de estudio en general.

A los pibes del Café 11, por las tardes de desconexión que me hacían falta para charlar de nuestras cosas y poder despejarme de todo lo que me rodeaba.

A los tetrokolos, por las noches de cervezas y risas en la tetería y celebrar mis logros como si fueran suyos.

Y a Rosalía, por aguantar el no verme por estar estudiando y aun así seguir apoyándome en todo momento.

Resumen

El precio de la electricidad en España, que es un servicio básico, depende principalmente de las fuentes energéticas que se usen en la generación, pero también depende de otros factores como la demanda y las interconexiones con los países colindantes. La empresa Red Eléctrica tiene la obligación de publicar los datos relativos al mercado energético, ya que son de interés público, pero son tantos los datos accesibles que puede no estar del todo claro lo que representan.

Para analizar estos registros, existe el Business Intelligence, que permite la explotación, gestión y transformación de datos, de forma que estos sea información procesable a partir de los brutos. Para la consecución de esto, se usan los procesos ETL (Extract, Transform y Load) que se componen de jobs y transformaciones.

El objetivo de este proyecto es el mostrar de forma gráfica algunos de los indicadores más relevantes, respecto al consumo y generación de energía, a partir de las APIs de Red Eléctrica para analizar el precio de la electricidad usando el concepto del Business Intelligence.

Abstract

The price of electricity in Spain, which is a basic service, depends mainly on the energy sources used in generation, but it also depends on other factors such as demand and interconnections with neighboring countries. The company Red Eléctrica has the obligation to publish the data related to the energy market, since they are of public interest, but there is so much data that can be obtained that it may not be entirely clear what is represented.

To analyze these records, there is Business Intelligence, which allows the exploitation, management and transformation of data, so these can be actionable information from raw data. To achieve this, the ETL processes (Extract, Transform and Load) are used, which are made up of jobs and transformations.

The objective of this project is to graphically show some of the most relevant indicators regarding energy consumption and generation, based on the APIs of Red Eléctrica to analyze the price of electricity using the concept of Business Intelligence.

Agradecimientos	vii
Resumen	ix
Abstract	xi
Índice	xiii
Índice de Figuras	xv
Índice de Tablas	xix
1 Introducción	1
1.1 Motivación	1
1.2 Estructura de la memoria	1
2 Tecnologías y entorno de trabajo	2
2.1 APIs	2
2.1.1 REData	2
2.1.2 e-sios	5
2.2 Base de datos	8
2.2.1 Tipo de base de datos	8
2.2.2 Estructura de base de datos	10
2.3 Software utilizado	12
2.4.1 Pentaho Data Integration	12
2.4.2 Power BI	14
2.4.3 SQL Developer	16
3 Transformaciones de datos	18
3.1 Dimensiones	18
3.1.1 DIM_FECHA	18
3.1.2 DIM_HORAS	21
3.2 Indicadores	23
3.2.1 REData	24
3.2.2 e-sios	31
4 Cuadro de mando	39
4.1 Página de inicio	39
4.2 Demanda	41
4.3 Interconexiones internacionales	42
4.4 PVPC	43
4.5 Mapas de REData	46
Referencias	53
Anexo A. Dimensiones	55
Anexo B. Indicadores usados	59
Anexo C. Vistas de base de datos	61



ÍNDICE DE FIGURAS

Figura 2-1. Logotipo de la Red Eléctrica.	2
Figura 2-2. Ejemplo de petición y respuesta a la API de REData.	5
Figura 2-3. Logotipo de e-sios.	5
Figura 2-4. Ejemplo de petición y respuesta a la API de e-sios.	7
Figura 2-5. Logotipo de Oracle Cloud.	8
Figura 2-6. Pantalla de creación de base de datos de Oracle Cloud.	8
Figura 2-7. Pantalla de creación de base de datos de Oracle Cloud (2).	9
Figura 2-8. Pantalla de creación de base de datos de Oracle Cloud (3).	9
Figura 2-9. Pantalla de descarga de wallet de conexión.	10
Figura 2-10. Diagrama relacional de las tablas de base de datos.	12
Figura 2-11. Logotipo de Pentaho.	12
Figura 2-12. Pantalla de creación de base de datos de Oracle Cloud.	14
Figura 2-13. Logotipo de Power BI.	14
Figura 2-14. Pantalla de conexión a base de datos de Power BI.	15
Figura 2-15. Configuración de conexión a base de datos en Power BI Server.	16
Figura 2-16. Logotipo de SQL Developer.	16
Figura 2-17. Pantalla de conexión a base de datos de SQL Developer.	17
Figura 3-1. Transformación de carga de la tabla de DIM_FECHA.	18
Figura 3-2. Paso de Generación de filas.	19
Figura 3-3. Paso de Añadir secuencia.	19
Figura 3-4. Paso de Calculadora.	20
Figura 3-5. Pasos de Data Grid.	20
Figura 3-6. Paso de Salida a tabla (DIM_FECHA).	21
Figura 3-7. Transformación de carga de la tabla DIM_HORAS.	21
Figura 3-8. Pasos de Generar filas y Añadir secuencia.	22
Figura 3-9. Paso de JavaScript.	23
Figura 3-10. Paso de Salida a tabla (DIM_HORAS).	23
Figura 3-11. Job MAIN_JOB.	24
Figura 3-12. Transformación REDATA_EXTRACCION.	24
Figura 3-13. Paso de Entrada de tabla.	24
Figura 3-14. Pasos de Get info sistema y Select values.	25
Figura 3-15. Pasos de Entrada de tabla y Join.	25
Figura 3-16. Paso de Filtrar filas.	26
Figura 3-17. Paso de Switch/case.	26
Figura 3-18. Pasos de Añadir constantes y Concatenar campos.	27
Figura 3-19. Paso de Cliente HTTP.	27

Figura 3-20. Paso de Entrada JSON.	28
Figura 3-21. Paso de Búsqueda en base de datos.	28
Figura 3-22. Paso de JavaScript.	29
Figura 3-23. Paso de Select values.	29
Figura 3-24. Pasos de Búsqueda en base de datos.	29
Figura 3-25. Pasos de Filtrar filas y Get system info.	30
Figura 3-26. Paso de Salida a tabla (TH_INDICADORES).	30
Figura 3-27. Job ESIOS_JOB.	31
Figura 3-27. Transformación ESIOS_PREEXT.	31
Figura 3-28. Pasos de Entrada de tabla y Calculadora.	32
Figura 3-29. Paso de Entrada de tabla.	32
Figura 3-30. Pasos de Añadir constantes y Concatenar campos.	33
Figura 3-31. Detalle de la transformación ESIOS_EXTRACCION.	33
Figura 3-32. Transformación ESIOS_EXTRACCION y paso de Añadir constantes.	34
Figura 3-33. Paso de Cliente HTTP.	34
Figura 3-34. Paso de Entrada JSON.	35
Figura 3-35. Paso de JavaScript.	35
Figura 3-36. Pasos de Filtrar filas y Get system info.	36
Figura 3-37. Paso de Select values.	36
Figura 3-38. Paso de Mapear valores.	37
Figura 3-39. Pasos de Búsqueda en base de datos.	37
Figura 3-40. Pasos de Filtrar filas y Establecer valor de campo a constante.	37
Figura 3-41. Paso de Salida a tabla (TH_INDICADORES).	38
Figura 4-1. Pantalla inicial del cuadro de mando.	39
Figura 4-2. Elemento visual con PVPC actual o más reciente.	39
Figura 4-3. Elementos visuales de porcentaje de generación energética.	40
Figura 4-4. Configuración de la tabla de generación energética.	40
Figura 4-5. Configuración del gráfico de porcentajes de generación energética.	41
Figura 4-6. Pantalla de gráficas de demanda.	41
Figura 4-7. Configuración de las gráficas de demanda.	42
Figura 4-8. Pantalla de gráficas de interconexiones.	42
Figura 4-9. Configuración de la gráfica de interconexión con Andorra.	43
Figura 4-10. Configuración del filtro de la fecha.	43
Figura 4-11. Pantalla de gráficas del PVPC.	44
Figura 4-12. Configuración de la gráfica de barras del PVPC.	44
Figura 4-13. Campos de la tabla de PVPC.	45
Figura 4-14. Configuración de la tabla de PVPC.	45
Figura 4-15. Configuración del cuadro de dato del máximo valor de PVPC entre semana.	46
Figura 4-16. Configuración para mostrar Mapa de formas.	46

Figura 4-17. Pantalla de mapa de REData correspondiente a la demanda energética.	47
Figura 4-18. Configuración de colores del mapa según el valor del indicador.	47
Figura 4-19. Campos de la tabla de demanda energética.	48
Figura 4-20. Agrupación de elementos visuales.	48
Figura 4-21. Pantalla de mapa de REData correspondiente a la generación energética.	49
Figura 4-22. Pantalla de mapa de REData correspondiente a la potencia instalada.	49
Figura 4-23. Configuración del mapa de REData correspondiente al porcentaje de energía generada a partir de fuentes renovables vs no renovables.	50
Figura 4-24. Pantalla de mapa de REData correspondiente al porcentaje de energía generada a partir de fuentes renovables vs no renovables.	50
Figura 4-25. Creación de marcadores.	51
Figura 4-26. Configuración del marcador de Demanda.	51

ÍNDICE DE TABLAS

Tabla 1. Estructura de DIM_CCAA.	10
Tabla 2. Estructura de DIM_HORAS.	10
Tabla 3. Estructura de DIM_FECHA.	11
Tabla 4. Estructura de DIM_INDICADORES.	11
Tabla 5. Estructura de DIM_TIPO_FUENTE.	11
Tabla 6. Estructura de DIM_TRAMOS.	11
Tabla 7. Estructura de TH_INDICADORES.	11
Tabla 8. Contenido de DIM_CCAA.	55
Tabla 9. Contenido de DIM_INDICADORES.	56
Tabla 10. Contenido de DIM_TIPO_FUENTE.	57
Tabla 11. Contenido de DIM_TRAMOS.	57

1 INTRODUCCIÓN

1.1. Motivación

Desde enero de 2021, el precio que pagan los ciudadanos de España por la electricidad ha ido en aumento. Para estabilizar esta situación, desde el Gobierno se planteó una serie de medidas como la de implantar tramos de discriminación horaria. Aun así, no se ha frenado la curva ascendente del PVPC (Precio Voluntario para el Pequeño Consumidor), llegando a picos de 954,01€/MWh (8 de marzo de 2022 a las 19h).

Dado que es un tema candente, se ha optado por hacer este proyecto, en el que se ofrece a la gente de a pie un cuadro de mando o pantalla de visualización en el que poder ver y comparar datos sobre la generación y el consumo de electricidad a partir de las APIs proporcionadas por Red Eléctrica, del grupo Redeia.

Es aquí donde entra en juego el Business Intelligence (BI), que es el conjunto de estrategias y procesos orientado a analizar los datos recopilados de una empresa o administración [1]. Está enfocado a la toma de decisiones dentro de una empresa a partir de este análisis, pero en este caso se trata de facilitar el análisis de datos al público general, ya que estos datos no son de una empresa privada sino publicados por Red Eléctrica.

1.2. Estructura de la memoria

A continuación, se describirá cada una de las partes diferenciadas de esta memoria:

- Introducción: se hace una pequeña introducción al proyecto en el que se expone los motivos de su realización y la estructura de la memoria.
- Tecnologías y entorno de trabajo: se describen las APIs usadas para extraer los datos, el tipo y estructura de la base de datos usada para almacenar los datos tratados para su visualización, y los programas utilizados durante el proyecto.
- Transformaciones de datos: se ven los pasos usados en las transformaciones de datos diferenciando entre las transformaciones que cargan las tablas de dimensiones y la que carga la tabla de hechos a partir de las APIs.
- Cuadro de mando: se enumeran todos los elementos visuales que muestran el análisis de datos en la pantalla de visualización.

2 TECNOLOGÍAS Y ENTORNO DE TRABAJO

En el siguiente apartado se verán los programas y tecnologías que componen el entorno en el que se ha desarrollado este trabajo. Se describirá además las distintas configuraciones dadas para el correcto funcionamiento de dicho entorno.

2.1 APIs

Para obtener los valores de los indicadores de generación y consumo de Red Eléctrica, usaremos dos APIs proporcionadas por esta misma organización. Estas son e-sios y REData.



Figura 2-1. Logotipo de la Red Eléctrica.

2.1.1 REData

REData es una interfaz de programación de aplicación (API) que proporciona un servicio REST para poder obtener diferentes datos del comportamiento del sistema eléctrico español. El uso de esta API es simple, ya que solo permite solicitudes GET para obtener respuestas JSON con los datos deseados [2]. Esta fue integrada en una página web con su mismo nombre, donde se pueden hacer consultas estadísticas guiadas, descargas de informes técnicos y estadísticos y hacer comparativas entre los distintos indicadores [3]. El acceso a esta API es global, por lo tanto, no se necesita ningún tipo de token que añadir a las consultas HTTP.

Los indicadores que se van a usar de esta API son evolución de la demanda energética, estructura de generación de energía por fuente energética, evolución de la generación de energía renovable y no renovable, y la potencia instalada por fuente energética. Todos ellos se obtienen a nivel de comunidad autónoma, lo cual es útil a la hora de representar estos valores gráficamente en un mapa de España.

La petición GET tiene el siguiente formato:

```
GET /{lang}/datos/{category}/{widget}?[query]
```

Siendo {lang} el lenguaje de la respuesta, {category} la categoría del indicador, {widget} el indicador específico de la consulta, y [query] la cadena de parámetros para filtrar la búsqueda de datos. Los parámetros usados son start_date que define la fecha desde la cual se quieren obtener datos, end_date que define la fecha hasta la que se quieren obtener datos, y time_trunc que define la agregación de los datos, según se quiera en hora, día, mes y año.

La respuesta a la consulta sigue el formato JSON, siendo la estructura de la respuesta exitosa:

```
{
  "data": {
    "type": "WIDGET TYPE",
    "id": "WIDGET ID",
```

```

"attributes": {
  "title": "WIDGET NAME",
  "last-update": "DATE",
  "description": "WIDGET DESCRIPTION",
  "meta": {
    "cache-control": {
      "cache": "MISS"
    }
  }
},
"included": [
  {
    "geo_id": GEO_ID,
    "community name": "COMMUNITY NAME",
    "content": {
      "type": "INDICADOR_1 TYPE",
      "id": "INDICADOR_1_ID",
      "groupId": null,
      "attributes": {
        "title": "INDICADOR_1 NAME",
        "description": "INDICADOR_1 DESCRIPTION",
        "type": "INDICADOR 1 TYPE",
        "magnitude": "INDICADOR_1 MAGNITUDE",
        "composite": false,
        "last-update": "DATE",
        "values" : [
          {
            "value": 12345,
            "percentage": "VALUE BETWEEN 0 AND 1",
            "datetime": "DATE"
          }
        ]
      }
    }
  },
  ...
]
}

```

Si la petición hubiera fallado, la respuesta sería con la estructura:

```
{
  "errors": [
    {
      "code": XXX,
      "status": "YYY",
      "title": "ERROR TITLE",
      "detail": "DETAILED ERROR"
    }
  ]
}
```

Para representar un ejemplo de la petición a la API y su respuesta, vamos a usar Postman, un programa que sirve para hacer peticiones HTTP a cualquier API, orientado a hacer pruebas sobre ésta. En este ejemplo, se hace la petición para obtener los datos de la demanda energética mensual para todas las comunidades autónomas desde el 1 de enero de 2022 hasta el día 31 de ese mismo mes. La respuesta tiene formato JSON y, en concreto, la captura muestra el dato del mes de enero para Andalucía.

The screenshot shows a Postman interface for a GET request. The URL is `https://apidatos.ree.es/datos/demanda/evolucion?start_date=2022-01-01T00:00&end_date=2022-01-31T23:59&time_trunc=month&all_ccaa=allCcaa&tecno_select=undefined`. The parameters section is expanded, showing four checked parameters: `end_date` (2022-01-31T23:59), `time_trunc` (month), `all_ccaa` (allCcaa), and `tecno_select` (undefined). The response status is 200 OK. The response body is displayed in JSON format, showing a single data object with the following structure:

```
1 {
2   "data": {
3     "type": "Evolución de la demanda",
4     "id": "dem1",
5     "attributes": {
6       "title": "Evolución de la demanda",
7       "last-update": null,
8       "description": null
9     },
10    "meta": {
11      "cache-control": {
12        "cache": "MISS"
13      }
14    }
15  },
16  ,
```

```

16   "included": [
17     {
18       "geo_id": 4,
19       "community_name": "Andalucía",
20       "content": {
21         "type": "Demanda",
22         "id": "10339",
23         "groupId": null,
24         "attributes": {
25           "title": "Demanda",
26           "description": null,
27           "color": "#ffea00",
28           "type": null,
29           "magnitude": null,
30           "composite": false,
31           "last-update": "2022-05-26T14:38:08.000+02:00",
32           "values": [
33             {
34               "value": 3588063.789,
35               "percentage": 1,
36               "datetime": "2022-01-01T00:00:00.000+01:00"
37             }
38           ]
39         }
40       }
41     }
42   ]

```

Figura 2-2. Ejemplo de petición y respuesta a la API de REData.

2.1.2 e·sios

La API de e·sios es un sistema de información denominado Sistema de Información del Operador del Sistema. Publica datos desde el 31 de marzo de 2014, estando a disposición de cualquier persona ya que, según la legislación, Red Eléctrica tiene la obligación de hacer públicos los resultados de los mercados o procesos de operación del sistema [4]. En general, los datos de e·sios son más técnicos que los de REData. Para el acceso a las consultas a la API, es necesario obtener un token único y personal el cual se pide mandando un correo a consultasios@ree.es. Este token debe introducirse en la cabecera Authorization de la petición.



Figura 2-3. Logotipo de e·sios.

El principal indicador que nos ofrece es el del Precio Voluntario para el Pequeño Consumidor (PVPC), el cual se publica a las 20:00 para cada hora del día siguiente. Además de este, se ha usado los indicadores de generación en tiempo real por fuente energética, interconexiones internacionales y los de demanda prevista, programada y real.

La petición GET tiene el siguiente formato:

```
GET /indicator /{id}?[query]
```

Siendo {id} el identificador del indicador y [query] la cadena de parámetros para filtrar la búsqueda de datos. Los parámetros usados son start_date que define la fecha desde la cual se quieren obtener datos, time_agg que define como agregar los valores, en nuestro caso media, y time_trunc que define la agregación de los datos, según se quiera en hora, día, mes y año.

La respuesta a la consulta sigue el formato JSON, siendo la estructura de la respuesta exitosa:

```

{
  "indicator": {
    "name": "INDICATOR_NAME",
    "short_name": " INDICATOR_NAME",

```

```

    "id": ID INDICATOR,
    "composited": false,
    "step_type": "linear",
    "disaggregated": false,
    "magnitud": [
      {
        "name": "MAGNITUDE_NAME",
        "id": ID_MAGNITUDE
      }
    ],
    "tiempo": [
      {
        "name": "TIME_RANGE_NAME",
        "id": ID_TIME_RANGE
      }
    ],
    "geos": [
      {
        "geo_id" : GEO_ID,
        "geo_name": "GEO_NAME"
      }
    ],
    "values_updated_at": "DATE",
    "values": [
      {
        "value": VALUE,
        "datetime": "DATE",
        "datetime_utc": "DATE_UTC",
        "tz_time": "TZ TIME",
        "geo_id": GEO ID,
        "geo_name": "GEO_NAME"
      },
      ...
    ]
  }
}

```

Si la petición hubiera fallado, la respuesta sería con la estructura:

```

{
  "status": "ERROR_ID",
  "error": "ERROR_MESSAGE",
}

```

Si se realizara la petición sin añadir el token personal a la cabecera, el mensaje que aparece es el siguiente:
 HTTP Token: Access denied.

Al igual que para REData, se va a usar Postman para mostrar un ejemplo de la petición y respuesta de la API de e-sios. En este caso, se va a obtener el dato del PVPC horario del 1 de enero de 2022 desde las 00h hasta las 10h agrupado por sistemas eléctricos (península, Baleares, Canarias, Ceuta y Melilla). Importante para realizar la petición correctamente es añadir el token personal en la cabecera Authorization. En concreto, en la captura se observan los datos del PVPC para las 00h de la península, de Canarias y de Baleares.

The screenshot shows a Postman interface for a GET request to the URL: `https://api.esios.ree.es/indicators/1001?start_date=2022-01-01T00%3A00&end_date=2022-01-01T10%3A00&time_agg=average&time_trunc=month&geo_agg=sum&geo_trunc=electric_system`. The Headers tab is active, showing a token in the Authorization header. The Body tab shows the JSON response:

```

1  {
2    "indicator": {
3      "name": "Término de facturación de energía activa del PVPC 2.0TD",
4      "short_name": "PVPC T. 2.0TD",
5      "id": 1001,
6      "composited": false,
7      "step_type": "linear",
8      "disaggregated": true,
9      "magnitud": [
10     {
11       "name": "Precio",
12       "id": 23
13     }
14   ],
15   "tiempo": [
16     {
17       "name": "Hora",
18       "id": 4
19     }
20   ]
21 }

```

The response also includes a `"values_updated_at"` field and a `"values"` array with three entries for different regions:

```

43  "values_updated_at": "2021-12-31T20:16:32.000+01:00",
44  "values": [
45    {
46      "value": 204.51,
47      "datetime": "2022-01-01T00:00:00.000+01:00",
48      "datetime_utc": "2021-12-31T23:00:00Z",
49      "tz_time": "2021-12-31T23:00:00.000Z",
50      "geo_id": 8741,
51      "geo_name": "Peninsula"
52    },
53    {
54      "value": 204.51,
55      "datetime": "2022-01-01T00:00:00.000+01:00",
56      "datetime_utc": "2021-12-31T23:00:00Z",
57      "tz_time": "2021-12-31T23:00:00.000Z",
58      "geo_id": 8742,
59      "geo_name": "Canarias"
60    },
61    {
62      "value": 204.51,
63      "datetime": "2022-01-01T00:00:00.000+01:00",
64      "datetime_utc": "2021-12-31T23:00:00Z",
65      "tz_time": "2021-12-31T23:00:00.000Z",
66      "geo_id": 8743,
67      "geo_name": "Baleares"
68    }
69  ]

```

Figura 2-4. Ejemplo de petición y respuesta a la API de e-sios.

2.2 Base de datos

A continuación, se detalla los aspectos de la base de datos utilizada para almacenar los datos obtenidos de las APIs anteriormente vistas.

2.2.1 Tipo de base de datos

Para este trabajo se ha usado una base de datos autónoma en la nube de Oracle que provee al registrar una cuenta gratuita. Que sea autónoma significa que las tareas de mantenimiento se realizan automáticamente mientras la base de datos esté en funcionamiento.



Figura 2-5. Logotipo de Oracle Cloud.

Hay distintos tipos de base de datos que Oracle da a elegir [5]:

-Data warehouse: está optimizado para el procesamiento analítico de datos, es decir, tiene un alto rendimiento a la hora de realizar consultas con una gran cantidad de datos.

-Transaction processing: está enfocado a las aplicaciones OLTP, que se caracteriza por sus numerosas y simples transacciones.

-JSON: es una base de datos de documentos en la nube orientada a las aplicaciones centradas en JSON.

-APEX: enfocado en el servicio con el mismo nombre, que es una plataforma para el desarrollo de aplicaciones con poco código de programación y basadas en datos.

La creación de la base de datos se realiza siguiendo los siguientes pasos:

The screenshot shows the 'Create Autonomous Database' form in Oracle Cloud. It has a title 'Create Autonomous Database' and a subtitle 'Provide basic information for the Autonomous Database'. There are three input fields: 'Compartment' with a dropdown menu showing 'jfernero001 (root)', 'Display name' with the value 'DB-20220518104905', and 'Database name' with the value 'DB20220518104905'. Below these fields is a section titled 'Choose a workload type' with four options: 'Data Warehouse' (selected with a checkmark), 'Transaction Processing', 'JSON', and 'APEX'. At the bottom, there are two buttons: 'Create Autonomous Database' and 'Cancel'.

Figura 2-6. Pantalla de creación de base de datos de Oracle Cloud.

Se le pone un nombre y se selecciona el tipo de base de datos de los cuatro vistos anteriormente, siendo nuestro caso un Data Warehouse.

Configure the database

Always Free ⓘ
 Show only Always Free configuration options

ⓘ If your Always Free Autonomous Database has no activity for 7 consecutive days, the database will be automatically stopped. Your data will be preserved, and you can restart the database to continue using it. If the database remains stopped for 3 months, it will be reclaimed. [Learn more.](#)

Choose database version
 19c

OCPU count *Read-Only*
 1
Always Free Autonomous databases can utilize up to 1 core. The CPU core count cannot be adjusted.

OCPU auto scaling
Allows system to use up to three times the number of cores specified by the OCPU count as the workload increases. [Learn more.](#)

Storage (TB) *Read-Only*
 0.02
Always Free Autonomous databases can utilize up to 0.02 TB (20 GB) of storage. The storage size cannot be adjusted.

Storage auto scaling
Allows system to expand up to three times the reserved storage.

Figura 2-7. Pantalla de creación de base de datos de Oracle Cloud (2).

Se elige la configuración, en nuestro caso a ser una cuenta gratuita únicamente podemos elegir entre distintas versiones de Oracle, de la cual seleccionamos la más reciente.

Create administrator credentials ⓘ

Username *Read-Only*
 ADMIN
ADMIN username cannot be edited.

Password

Confirm password

Choose network access

Access Type

Secure access from everywhere Allow users with database credentials to access the database from the internet. ✓	Secure access from allowed IPs and VCNs only Restrict access to specified IP addresses and VCNs.	Private endpoint access only Restrict access to a private endpoint within an OCI VCN.
---	--	---

The virtual cloud network option is not available for OCI Free Tier accounts.

Require mutual TLS (mTLS) authentication ⓘ
If you select this option, mTLS will be required to authenticate connections to your Autonomous Database.

Figura 2-8. Pantalla de creación de base de datos de Oracle Cloud (3).

Se indica la contraseña para el usuario ADMIN, que es el usuario administrador de base de datos que usaremos. Por último, se selecciona una restricción para la conexión por red, eligiendo en nuestro caso el acceso universal.

Como se comentó anteriormente, la elegida es Data warehouse, ya que vamos a tener una gran cantidad de datos. Las características de esta es que tiene 20GB de almacenamiento, tiene la versión 19c de Oracle, y para su conexión se hace uso de credenciales de cliente, a partir de ahora Wallet. Este wallet tiene distintos tipos de conexión, high, medium o low, según la prioridad que se le quiera dar para asignarles recursos a dicha conexión.

Para la descarga de este wallet, se debe ir a la página donde está alojada la base de datos dentro de <https://cloud.oracle.com>. Una vez en la pantalla inicial de la base de datos, se hace click en 'DB Connection' y acto seguido descargar el Instance Wallet. El Instance Wallet solo permite la conexión a una base de datos específica, mientras que el Regional Wallet está dirigida a un usuario para poder acceder a todas las instancias de servicios habitados en la cuenta de la Oracle Cloud [6].

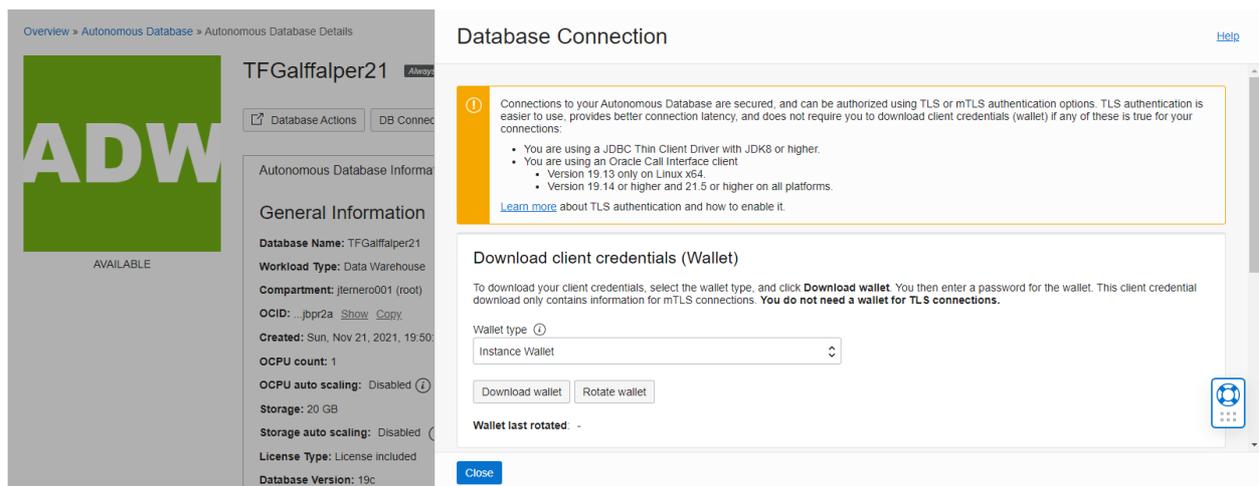


Figura 2-9. Pantalla de descarga de wallet de conexión.

Tras esto se pedirá una contraseña asociada al wallet y, tras ello, se descargará un archivo comprimido donde vienen varios ficheros los cuales usaremos más adelante para realizar la conexión de los distintos programas con la base de datos.

2.2.2 Estructura de base de datos

El diseño de la base de datos sigue el modelo dimensional, donde tenemos los datos comunes en tablas llamadas dimensiones y los valores que se registran a lo largo del tiempo guardados en una tabla de hechos. Las dimensiones son tablas simples y de pocos registros en las que se ofrece información característica de la tabla de hechos, a la cual se une mediante un campo clave. La tabla de hechos es la tabla principal del sistema, la cual contiene las claves para unir con las dimensiones y además las medidas o valores que se quieren analizar. Al tener muchos registros, todos o la gran mayoría de sus campos son numéricos, ya que ocupan menos almacenamiento que cadenas de caracteres.

En nuestro caso, las dimensiones dadas son las de comunidades autónomas (DIM_CCAA), la de fecha (DIM_FECHA), la de hora (DIM_HORAS), la de indicadores (DIM_INDICADORES), la fuente energética (DIM_TIPO_FUENTE) y la de tramos horarios de facturación (DIM_TRAMOS). Mientras, la tabla de hechos de nuestro proyecto (TH_INDICADORES) guarda los datos horarios, diarios o mensuales de los distintos indicadores obtenidos de las APIs vistas anteriormente.

A continuación, se muestra la estructura de cada una de las tablas:

DIM_CCAA	
NOMBRE COLUMNA	TIPO DATO
ID_CCAA (PK)	INTEGER
DS_CCAA	VARCHAR2 (50 CHAR)

Tabla 1. Estructura de DIM_CCAA.

DIM_HORAS	
NOMBRE COLUMNA	TIPO DATO
ID_HORA (PK)	INTEGER
DS_HORA	VARCHAR2 (10 CHAR)
ID_TRAMO	INTEGER
N_HORA	VARCHAR2 (5 CHAR)

Tabla 2. Estructura de DIM_HORAS.

DIM_FECHA	
NOMBRE COLUMNA	TIPO DATO
ID_FECHA (PK)	INTEGER
D_FECHA	DATE
N_DIAS	INTEGER
N_MES	INTEGER
N_ANYO	INTEGER
N_SEMANA	INTEGER
DS_DIA_SEMANA	VARCHAR2 (10 CHAR)
DS_MES	VARCHAR2 (15 CHAR)
N_DIA_SEMANA	INTEGER

Tabla 3. Estructura de DIM_FECHA.

DIM_INDICADORES	
NOMBRE COLUMNA	TIPO DATO
ID_INDICADOR (PK)	INTEGER
DS_INDICADOR_CORTA	VARCHAR2 (100 CHAR)
DS_INDICADOR	VARCHAR2 (500 CHAR)
DS_ORIGEN	VARCHAR2 (20 CHAR)

Tabla 4. Estructura de DIM_INDICADORES.

DIM_TIPO_FUENTE	
NOMBRE COLUMNA	TIPO DATO
ID_TIPO (PK)	INTEGER
DS_TIPO	VARCHAR2 (50 CHAR)

Tabla 5. Estructura de DIM_TIPO_FUENTE.

DIM_TRAMOS	
NOMBRE COLUMNA	TIPO DATO
ID_TRAMO (PK)	INTEGER
DS_TRAMO	VARCHAR2 (15 CHAR)

Tabla 6. Estructura de DIM_TRAMOS.

TH_INDICADORES	
NOMBRE COLUMNA	TIPO DATO
ID_INDICADOR (PK)	INTEGER
N_VALOR	NUMBER (15,3)
ID_LOCALIZACION (FK)	INTEGER
ID_FECHA (FK)	INTEGER
ID_HORA (FK)	INTEGER
ID_TIPO (FK)	INTEGER
F_CREACION	DATE
ID_TRAMO (FK)	INTEGER

Tabla 7. Estructura de TH_INDICADORES.

Las tablas se relacionan con la siguiente estructura:

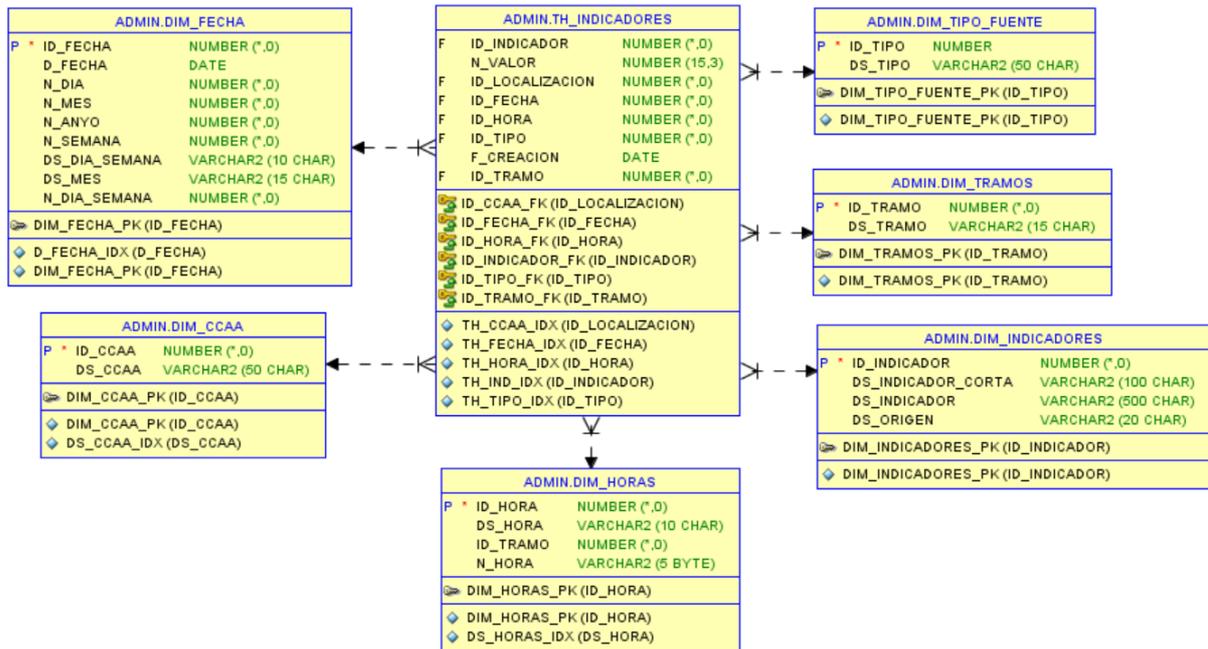


Figura 2-10. Diagrama relacional de las tablas de base de datos.

Además, para facilitar la visualización y manejo de los datos por indicador, se han creado vistas para mejorar el modelado de la visualización de datos.

Para la optimización de las búsquedas y de uniones entre tablas, se utilizan los índices. Los índices son una estructura de datos con punteros a los registros de una tabla de forma ordenada, lo que proporciona un acceso directo y rápido a las filas.

2.3 Software utilizado

2.3.1 Pentaho Data Integration

Pentaho Data Integration (PDI) es un conjunto de componentes de Pentaho Suite, de la empresa Hitachi, para crear procesos de extracción, transformación y carga de datos (ETL por sus siglas en inglés) [7]. Este es uno de los softwares más usados en el mundo del Business Intelligence (BI) por ser muy intuitivo, disponible para Linux y Windows, y además open source.



Figura 2-11. Logotipo de Pentaho.

Spoon es la interfaz gráfica de PDI, donde se crean los procesos ETL. Estos procesos constan de transformaciones y jobs, que son conjuntos de transformaciones. La extensión de los ficheros de transformaciones es '.ktr' y la de los jobs '.kjb', ambos con lenguaje XML. Es el principal programa que se va a usar en este trabajo. Otros componentes de PDI son Carte, que es un servidor web simple para ejecutar transformaciones y jobs de forma remota [8], Pan, que es la herramienta de línea de comandos para ejecutar

transformaciones, y Kitchen, que es la herramienta de línea de comandos para ejecutar jobs.

En este caso específico vamos a usar la versión Community 8.2 al ser una de las más estables actualmente.

2.3.1.1 Configuración de Pentaho Data Integration

Esta configuración está orientada para Windows. Para el correcto funcionamiento de PDI, se depende del JDK 1.8 de Java, por lo que además habría que añadir a la carpeta /data-integration/lib el controlador de base de datos de Oracle 'ojdbc8.jar', 'oraclepki.jar', 'osdt_cert.jar' y 'osdt_core.jar'. Además, se ha de modificar varios archivos. El primero de ellos será custom.properties, ubicado en la carpeta system/karaf/etc/ dentro de data-integration, al cual se le añadirá al final del fichero la siguiente línea:

```
org.apache.karaf.security.providers=oracle.security.pki.OraclePKIProvider
```

A continuación, se editará el fichero Spoon.bat de la carpeta /data-integration. Antes del bloque REM*****Run...*****, habrá que añadir las siguientes líneas:

```
@REM Java options to support Oracle Wallet secure connections
@REM Assumes that the cwallet.sso file is in data-integration\Wallet\ folder
set OPT="-Djavax.net.ssl.keyStore=%KETTLE_DIR%\Wallet\cwallet.sso" %OPT%
set OPT="-Djavax.net.ssl.keyStoreType=SSO" %OPT%
set OPT="-Doracle.net.tns_admin=%KETTLE_DIR%\Wallet" %OPT%
```

Estas líneas son para indicar donde están los archivos necesarios para conectarse a la base de datos. Hay que crear la carpeta Wallet dentro de la carpeta /data-integration. Dentro de ese directorio hay que copiar los ficheros 'cwallet.sso', 'sqlnet.ora', 'tnsnames.ora' y 'trustore.jks' extraídos del wallet comprimido descargado anteriormente en el apartado '2.2.1 Tipo de base de datos' [9].

Una vez seguidos estos pasos, ya se puede realizar la conexión a la base de datos. Para ello, se debe ejecutar Spoon y crear una transformación o job en el que se vaya a trabajar. Se hace click derecho en la pestaña de 'Database connection' y en 'New', para crear una nueva conexión, tras lo que aparecerá una pantalla. Se debe seleccionar 'Generic database' dentro de 'Connection type', en el apartado 'Settings' seleccionar 'Oracle' dentro de 'Dialect', insertar 'jdbc:oracle:thin:@' seguido del nombre de la conexión del wallet (el cual se puede ver en /data-integration/Wallet/tnsnames.ora), en este caso 'tfgalffalper21_high', en el campo 'Custom connection URL, y 'oracle.jdbc.driver.OracleDriver' en el campo 'Custom driver class name'. Para finalizar, se debe meter las credenciales del usuario de base de datos, en este caso ADMIN, darle un nombre a la conexión y probar si es correcta.

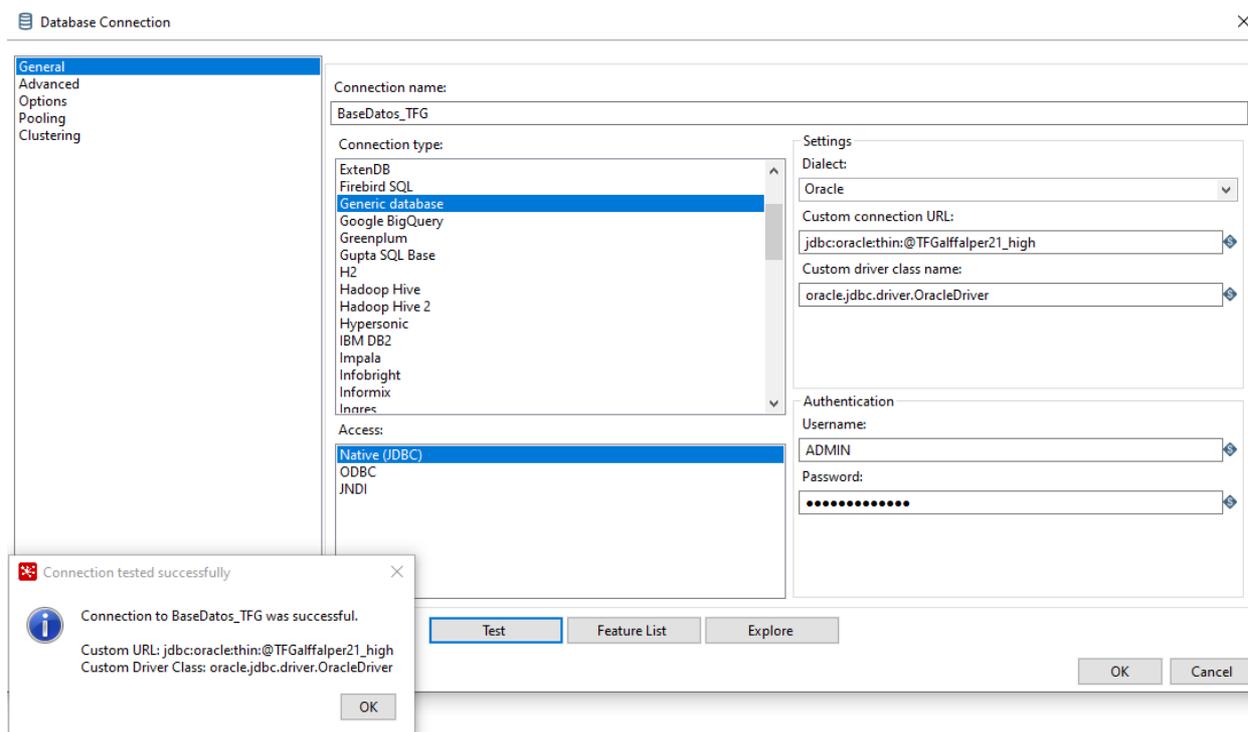


Figura 2-12. Pantalla de creación de base de datos de Oracle Cloud.

2.3.2 Power BI

Power BI es un conjunto de herramientas para presentar de forma gráfica los datos provenientes de diversos orígenes. Se compone de Power BI Service, que es el servicio de visualización en la nube, y el programa Power BI Desktop para crear los paneles con la información [10].



Figura 2-13. Logotipo de Power BI.

Los proyectos que se crean en Power BI Desktop se guardan con el formato '.pbix', y se publican mediante el botón 'Publicar' de la aplicación directamente en la nube a la cuenta asociada al programa.

Una alternativa a este programa es Schema Workbench, algo menos intuitivo, pero más orientado a cubos Mondrian, algo que se buscaba en primer momento. Era la primera opción ya que pertenece a la suite de Pentaho, pero no tiene la opción de conectarse a las bases de datos de Oracle Cloud, la cual es la usada en este proyecto.

Power BI Desktop es gratuito, pero para publicar en la nube debes tener una cuenta Microsoft Office 365. La versión utilizada es la más reciente a la hora de realizar este trabajo, la 2.104.

2.3.2.1 Configuración de Power BI

Para hacer posible la conexión de Power BI Desktop, es necesario realizar varias acciones previas. Es necesario instalar el proveedor de acceso de datos de Oracle ODP.NET. Se debe descargar el fichero ODAC XCopy [11], descomprimirlo y ejecutar en la ventana de comandos de Windows en modo administrador las

siguientes líneas:

```
install.bat odp.net4 <directorio de instalación> odp64
OraProvCfg /action:gac /providerpath:"Oracle.DataAccess.dll"
OraProvCfg /action:config /product:odp /frameworkversion:v4.0.30319
/providerpath:"Oracle.DataAccess.dll"
```

También hay que instalar Instant Client, que permite el despliegue de aplicaciones que se conectan a bases de datos de Oracle [12]. Finalmente, añadir a las variables de entorno del sistema, dentro del Path, la ruta donde se ha instalado ODP, en este caso C:\odp64, además de añadir una nueva variable de sistema llamada TNS_ADMIN con la ruta network/admin de la carpeta donde se haya instalado Instant Client. Además, sustituir el tnsnames.ora de esta carpeta por el del wallet.

Una vez, realizada esta configuración, al abrir Power BI aparecerá una ventana y se seleccionará ‘Obtener datos’. Tras ello, se selecciona ‘Oracle database’ y como servidor indicamos el nombre de la conexión que está en tnsname.ora, en este caso ‘tfgalffalper21_high’, seleccionando posteriormente la opción de DirectQuery. Esto último es para que no almacene datos en el informe, sino que cada vez que se visualice, haga las consultas a la base de datos y obtenga los datos actualizados. Para finalizar, se selecciona las tablas o vistas de las cuales se quieren visualizar datos [13].



Figura 2-14. Pantalla de conexión a base de datos de Power BI.

Una vez terminado el cuadro de mando, se publica en la cuenta de la nube de Power BI Server. Para que desde el servidor pueda obtener los datos de las tablas se debe configurar un gateway en la máquina donde se tenga Power BI Desktop. En la página web de Power BI, una vez iniciada sesión, se clic en Configuración>Descargar>Puerta de enlace de datos, lo que abrirá otra ventana con la página de la descarga. Se inicia la instalación introduciendo el correo electrónico de la cuenta de Microsoft y se clic en ‘Register a new gateway on this computer’. Tras ello se le da un nombre y se pone una contraseña de recuperación.

A continuación, volver a la página de Power BI, hay que ir a Configuración>Configuración>Administrar puertas de enlace, donde aparecerá la que se ha instalado en el equipo. Una vez ahí, se selecciona Nuevo origen de datos y aparecerá una ventana de configuración en la que debemos seleccionar el gateway instalado, poner un nombre al origen de datos, seleccionar el tipo de origen Oracle, indicar el nombre de la conexión, que en este caso será ‘tfgalffalper21_high’, e introducir las credenciales de base de datos [14].

Nuevo origen de datos ×

Nombre de clúster de puerta de enlace *

Nombre del origen de datos *

Tipo de origen de datos *

Servidor *

Autenticación ⓘ

Método de autenticación *

Nombre de usuario *

Contraseña *

Omitir conexión de prueba

Figura 2-15. Configuración de conexión a base de datos en Power BI Server.

2.3.3 SQL Developer

Oracle SQL Developer es una herramienta que proporciona editores para ejecutar consultas SQL y modificaciones sobre una base de datos. Este programa solo dispone de conexiones a bases de datos de tipo Oracle, incluyendo los Oracle Cloud [15].



Figura 2-16. Logotipo de SQL Developer.

La versión utilizada es la más actualizada en el momento de realizar este trabajo, la 21.4.3.

2.3.3.1 Configuración de SQL Developer

Para realizar la conexión a la base de datos en la nube de Oracle hay que seguir las instrucciones aquí descritas. Una vez abierto SQL Developer, se le dará a la cruz verde para crear una nueva conexión, tras lo que aparecerá una ventana de configuración. Lo primero será darle un nombre a la nueva conexión y seleccionar que tipo de base de datos es a la que nos queremos conectar, siendo por defecto Oracle. Seguido a esto, se introducirá las credenciales del usuario de base de datos, en este caso ADMIN. En ‘Tipo de conexión’ se debe seleccionar ‘Cartera de Cloud’, especificando en ‘Archivo de configuración’ la ruta donde está el wallet comprimido que se descargó en el apartado 2.2.1. Tras ello, detectará las conexiones disponibles en el wallet, seleccionando en este caso ‘tfgalfalpe21_high’. Una vez finalizada la configuración, se probará la conexión para comprobar que todo es correcto.

The image shows a screenshot of the 'Conectar' (Connect) dialog box in SQL Developer. The dialog is titled 'Conectar' and has a 'Name' field containing 'BaseDatos_TFG'. Below this, the 'Tipo de Base de Datos' (Database Type) is set to 'Oracle'. The 'Información de usuario' (User Information) section is active, showing 'Usuario de Proxy' (Proxy User) as the selected user type. The 'Tipo de autenticación' (Authentication Type) is 'Por defecto' (Default). The 'Usuario' (User) field contains 'ADMIN', and the 'Rol' (Role) field contains 'valor por defecto'. The 'Contraseña' (Password) field is masked with dots, and the 'Guardar Contraseña' (Save Password) checkbox is checked. The 'Tipo de Conexión' (Connection Type) is set to 'Cartera de Cloud' (Cloud Wallet). The 'Detalles' (Details) section is active, showing 'Archivo de Configuración' (Configuration File) as 'alcomp\Documents\TFG\Wallet_TFGalffalper21.zip' and 'Servicio' (Service) as 'tfgalffalper21_high'. There are buttons for 'Examinar...' (Browse...), 'Configurar QSS clásico' (Configure Classic QSS), 'Guardar' (Save), 'Borrar' (Delete), 'Probar' (Test), 'Conectar' (Connect), and 'Cancelar' (Cancel).

Figura 2-17. Pantalla de conexión a base de datos de SQL Developer.

3 TRANSFORMACIONES DE DATOS

Para obtener los datos de las APIs y transformarlos a las estructuras de tablas que se desea, se crearán una serie de transformaciones y jobs para cargar registros datos en la base de datos.

3.1 Dimensiones

Para rellenar las tablas de las dimensiones DIM_CCAA, DIM_INDICADORES, DIM_TIPO_FUENTE y DIM_TRAMOS se ha usado sentencias INSERT INTO de SQL con los datos deseados, ya que son tablas estáticas y simples de rellenar. Sin embargo, para las tablas DIM_FECHA y DIM_HORAS se han creado una transformación para cargar los datos en cada una de ellas.

3.1.1 DIM_FECHA

La transformación está compuesta por los pasos que se ven a continuación.

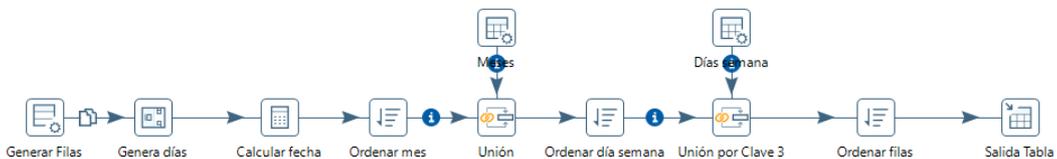


Figura 3-1. Transformación de carga de la tabla de DIM_FECHA.

En el paso de Generar Filas se generan tantas filas como quiera, en este caso 1000, y dos columnas constantes: una con la primera fecha que va a tener la dimensión, en formato YYYYMMDD, que va a ser 20200401, y otra columna con el valor constante -1. Este último valor se usará más adelante para hacer cálculos.

#	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Value	Set empty string?
1	FECHA_BASE	Date	yyyyMMdd						20200401	N
2	UNO	Integer							-1	N

Figura 3-2. Paso de Generación de filas.

El siguiente paso es el de Genera días, que lo que hace es añadir a las dos columnas anteriores otra columna con una secuencia ascendente de números desde el 0 hasta el número de filas que provienen del flujo de datos. Se puede elegir el valor del incremento, para este caso será de uno en uno.

The screenshot shows a dialog box titled "Add sequence". It is divided into two main sections. The first section, "Use a database to generate the sequence", includes a checkbox "Use DB to get sequence?" which is unchecked. Below this are fields for "Connection" (set to "LUZ"), "Schema name", and "Sequence name" (set to "SEQ_"). The second section, "Use a transformation counter to generate the sequence", includes a checked checkbox "Use counter to calculate sequence?". Below this are fields for "Counter name (optional)", "Start at value" (set to "0"), "Increment by" (set to "1"), and "Maximum value" (set to "999999999"). At the bottom of the dialog are "OK" and "Cancel" buttons. A "Help" button is located in the bottom-left corner.

Figura 3-3. Paso de Añadir secuencia.

A continuación, se usa el paso Calculadora al que hemos llamado Calcular fecha como descripción. Dentro de este componente hay variedad de operaciones, la mayoría relacionadas con el tipo de dato Date. Los dos primeros cálculos son la suma de la fecha base y la secuencia de números, lo que hace que se cree una secuencia de fechas. La diferencia de ambas es el tipo de dato y el formato que tienen, de la primera obtenemos el identificador (ID_FECHA) que tendrá el registro en DIM_FECHA con el formato YYYYMMDD, y de la segunda obtenemos la FECHA de tipo Date y con formato YYYY-MM-DD. A partir de esta fecha se calcula otra auxiliar FECHA_AUX que será el día anterior a FECHA, sumando el valor constante -1 visto antes, y tendrá su mismo formato. A partir de FECHA_AUX calculamos el número de día de la semana, ya que para el programa la semana comienza en domingo, por lo que hay que usar el día anterior al día real para tener qué día de la semana es de forma correcta. Para el resto de los campos, que son el año, el mes, el día del mes y la semana del año, se usa FECHA.

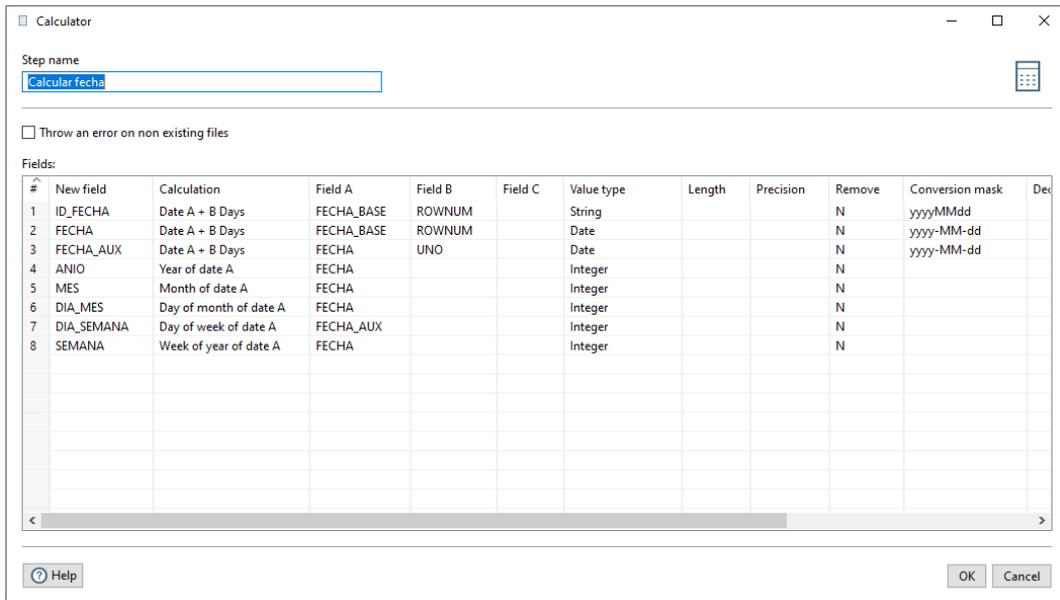


Figura 3-4. Paso de Calculadora.

Lo siguiente es dar los nombres a los días y a los meses. Para ello se han creado dos Data Grid, que generan dos tablas estáticas en la transformación. Para unir las al flujo de datos, se ha usado un Merge Join, donde se seleccionan las claves de unión de cada flujo y el tipo de unión. Las claves deben estar ordenadas antes de la unión, por lo que se colocan pasos Sort rows.

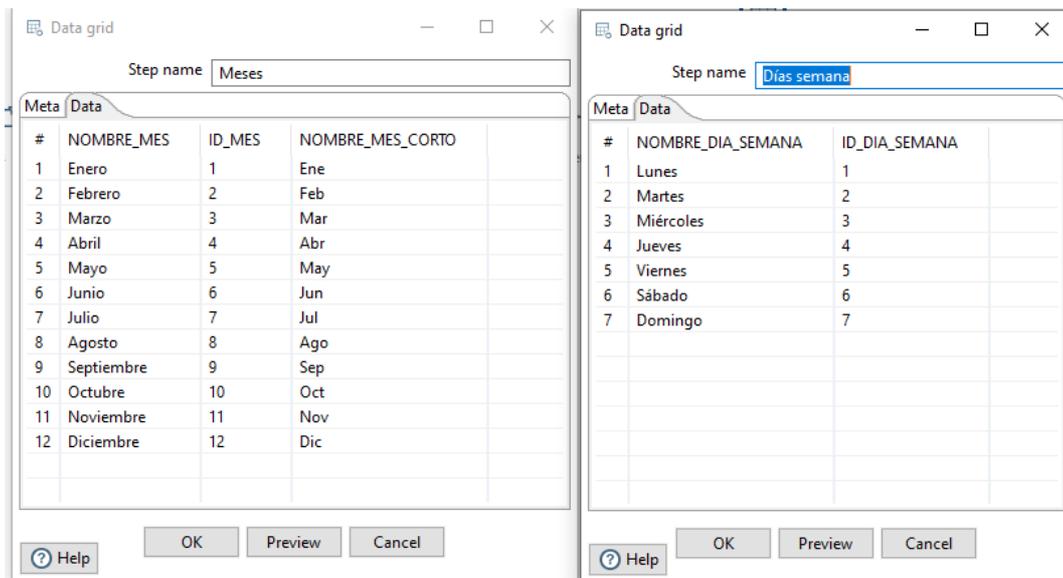


Figura 3-5. Pasos de Data Grid.

Por último, se ordena por la fecha para volcar los datos a la tabla DIM_FECHA. Se utiliza el paso llamado Table output, donde se indica la conexión a la base de datos deseada, la tabla y los campos que se introducen en cada columna de ésta.

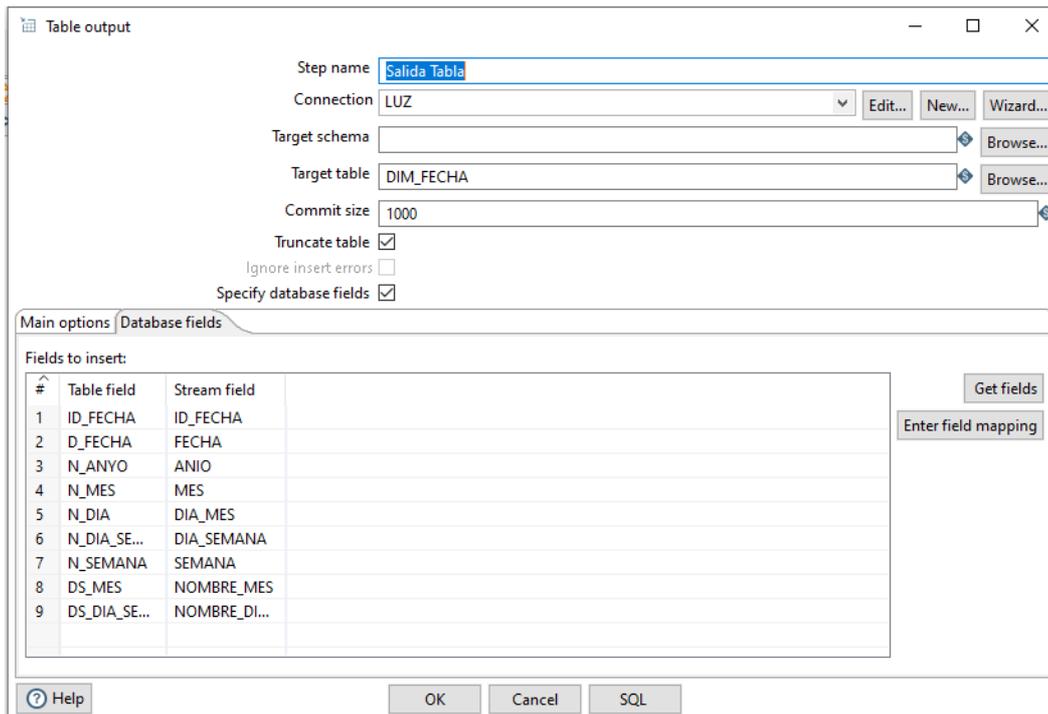


Figura 3-6. Paso de Salida a tabla (DIM_FECHA).

3.1.2 DIM_HORAS

La transformación para cargar la dimensión DIM_HORAS es más simple que la anterior.



Figura 3-7. Transformación de carga de la tabla DIM_HORAS.

Se compone de un paso de Generar Filas en la que se generan 24 filas con el número 0. Tras éste viene un Genera secuencia, en el que se incrementa una unidad cada fila, lo que hace las 24 horas.

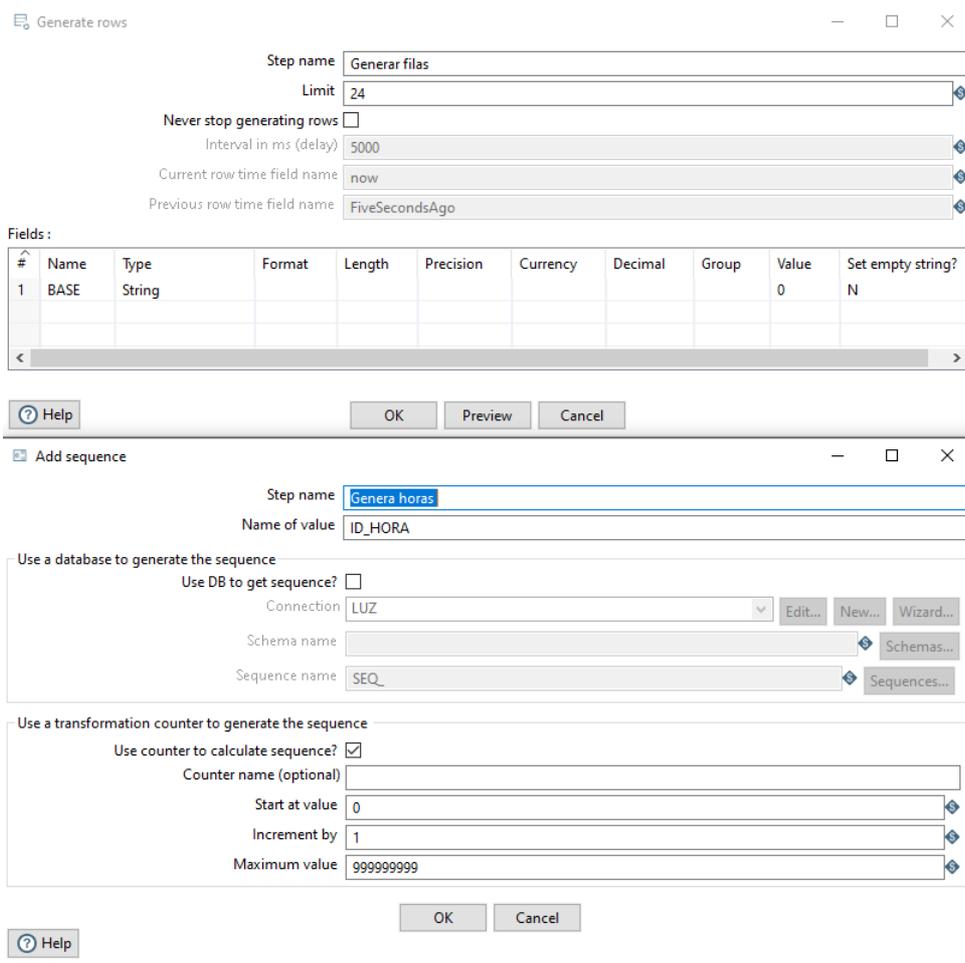


Figura 3-8. Pasos de Generar filas y Añadir secuencia.

El paso más complejo de esta transformación es el Modified JavaScript value, que es un componente donde se introduce código JavaScript para transformar los valores de las columnas a utilizar. En este caso, usamos el código para asignar el ID_TRAMO del tramo horario de facturación a cada una de las horas, los cuales son el tramo valle (ID_TRAMO=1) de 00h a 08h, el tramo llano (ID_TRAMO=2) de 08h a 10h, de 14h a 18h y de 22h a 24h, y el tramo punta (ID_TRAMO=3) de 10h a 14h y de 18h a 22h [16]. Además, se pasa el número de hora a cadena de caracteres para que tenga el formato de dos dígitos (N_HORA) y el formato digital con minutos (N_HORA_MIN).

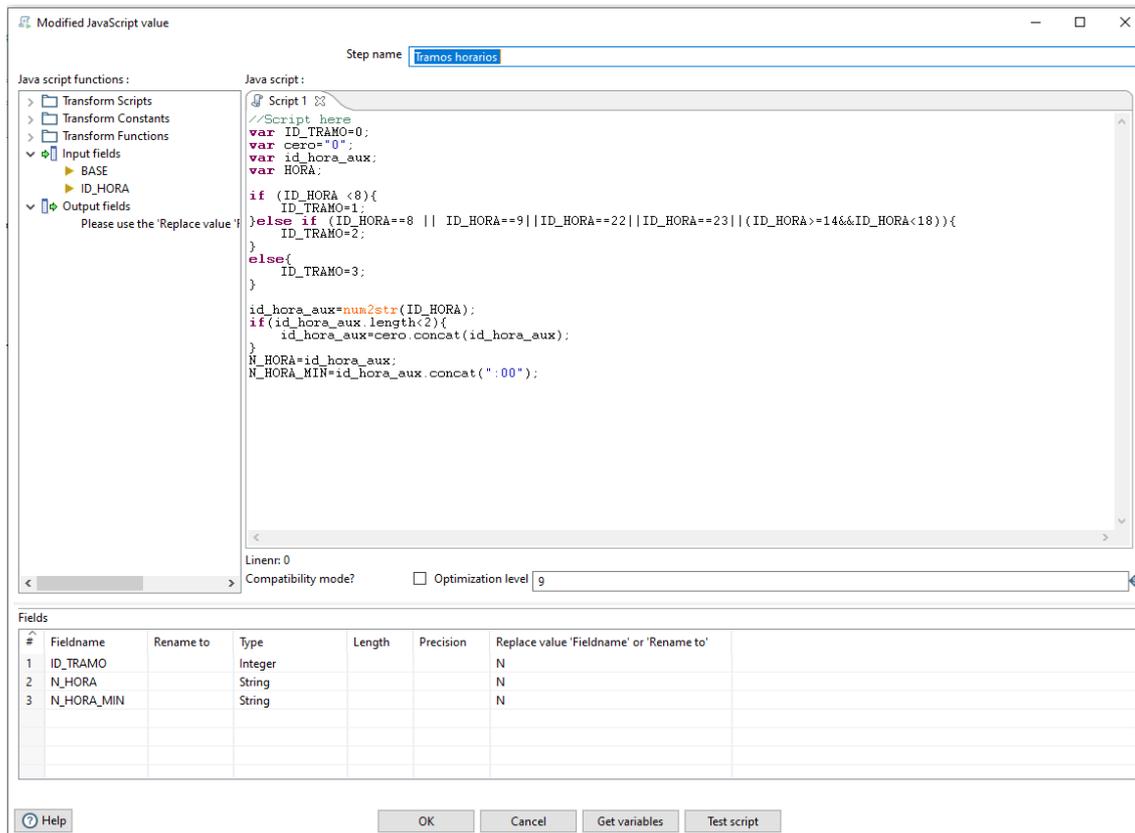


Figura 3-9. Paso de JavaScript.

Para finalizar, se usa el Table output para pasar los datos transformados a la dimensión DIM_HORAS.

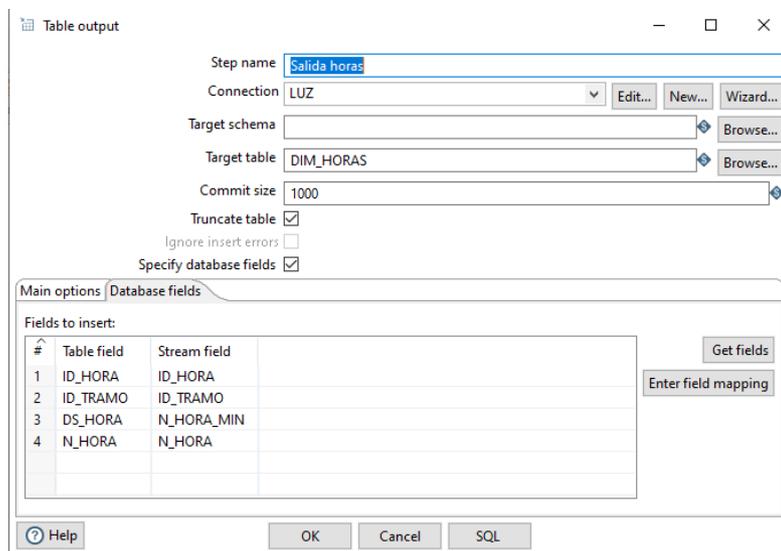


Figura 3-10. Paso de Salida a tabla (DIM_HORAS).

3.2 Indicadores

A continuación, se comentan los jobs y transformaciones creadas para obtener y cargar los datos procedentes de las APIs. Principalmente tenemos un job principal compuesto por la transformación referente a REData y el job de e-sios. Lanzando este proceso se ejecutan ambos en serie, aunque se pueden ejecutar por separado ya que son independientes.



Figura 3-11. Job MAIN_JOB.

3.2.1 REData

La transformación de REData está compuesta por los siguientes pasos.

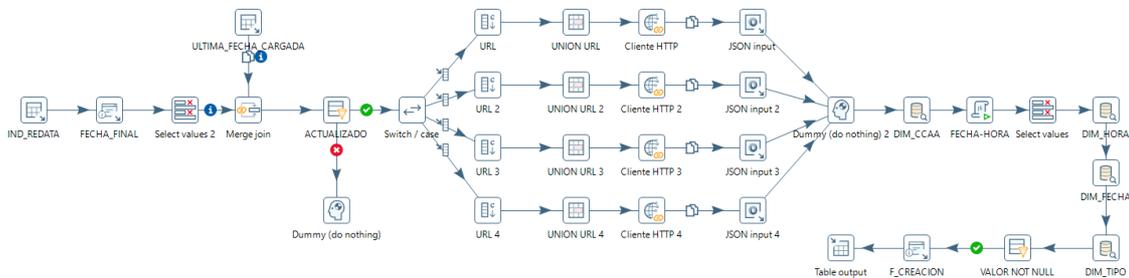


Figura 3-12. Transformación REDATA_EXTRACCION.

El inicio de esta transformación viene dado por un Table input con una consulta que devuelve los identificadores de los indicadores cuyo DS_ORIGEN tenga el valor 'REData' de la tabla DIM_INDICADORES.

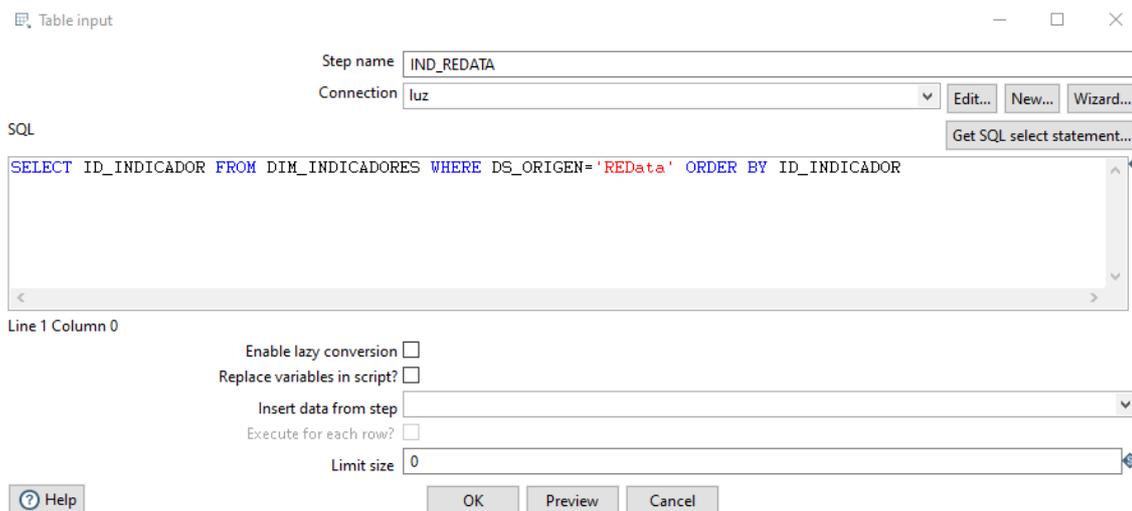


Figura 3-13. Paso de Entrada de tabla.

A esa columna de identificadores, se añade la fecha del último día del mes de la fecha de ejecución, cambiando el formato en el siguiente 'Select values' a 'yyyy-MM-dd'. Esta será la fecha que irá en el parámetro de end_date de la petición a la API.

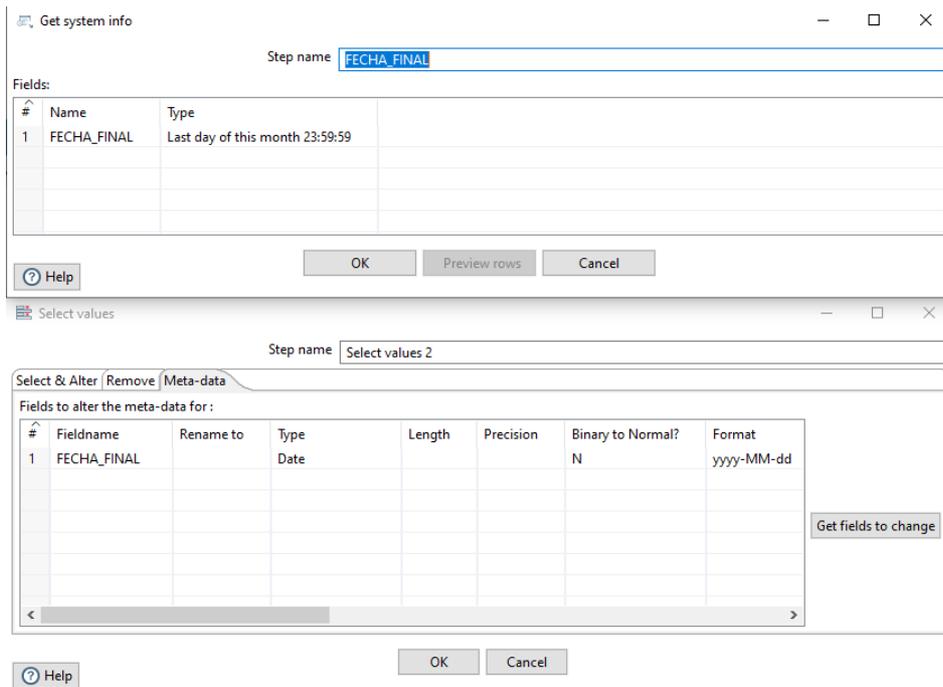


Figura 3-14. Pasos de Get info sistema y Select values.

En la parte superior, se encuentra otro 'Table input', cuya consulta extrae la última fecha cargada de cada indicador de REData con el formato 'yyyy-MM-dd'. Si no hay una fecha máxima quiere decir que no hay datos relacionados con ese indicador en la tabla, por lo que la FECHA_INICIO tomará como valor el 2022-06-01 para que los datos de la petición empiecen por esa fecha. Una vez extraídos los datos, se añaden al flujo de datos con un 'Merge join' uniéndose mediante los ID_INDCADOR.

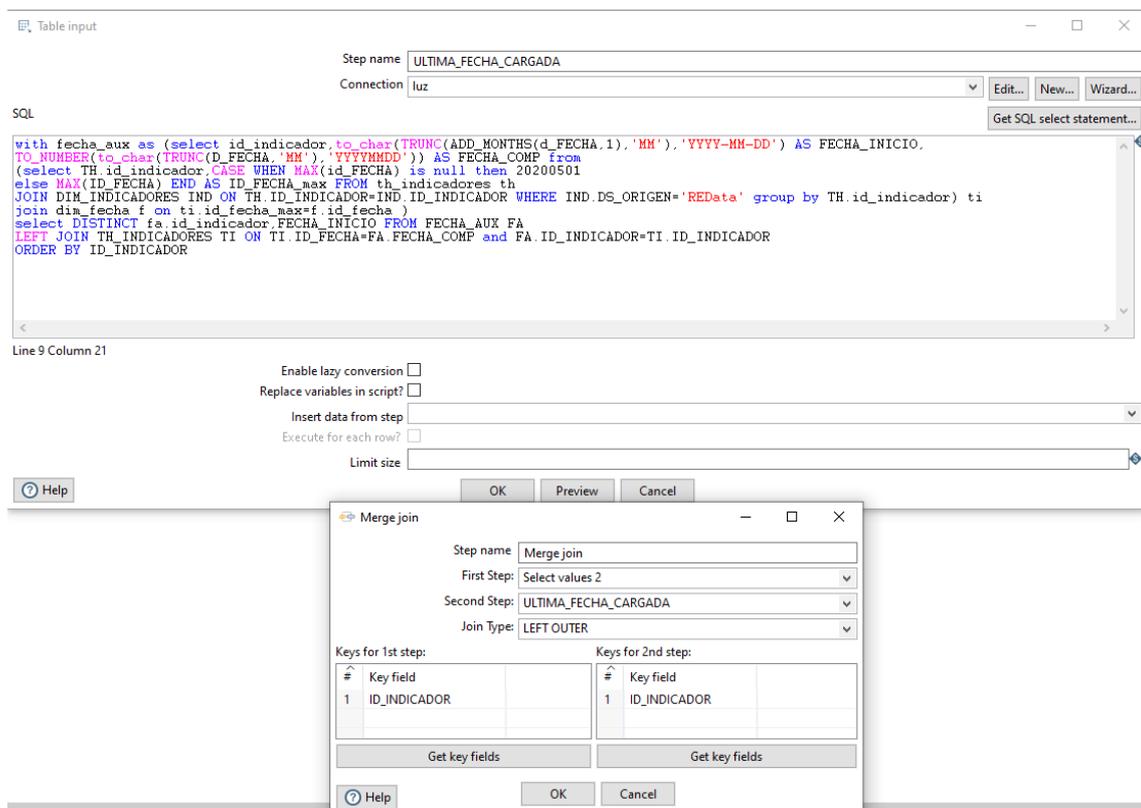


Figura 3-15. Pasos de Entrada de tabla y Join.

Es importante comprobar que FECHA_INICIO es menor que FECHA_FINAL, que ocurre cuando se han cargado los datos del mes actual.



Figura 3-16. Paso de Filtrar filas.

A partir de aquí se ramifica el flujo de datos en cuatro caminos similares, uno para cada petición a los indicadores de REData. Los registros se distribuyen entre las ramificaciones según el ID_INDICADOR que tenga, filtrado por el paso 'Switch/case'.

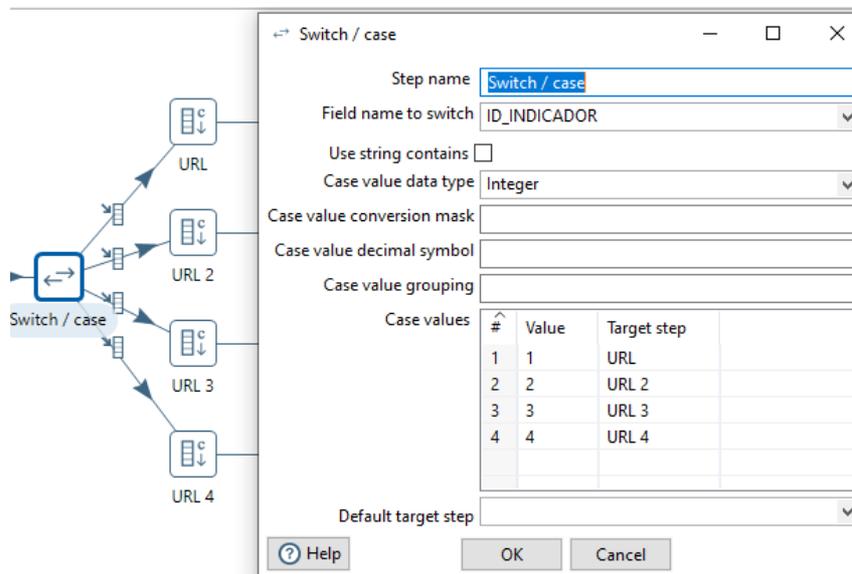


Figura 3-17. Paso de Switch/case.

El principio de cada camino es un paso en el que se añaden distintas constantes, los cuales son partes de la URL para la petición de cada indicador, y se concatenan en el siguiente paso añadiendo los campos FECHA_INICIO y FECHA_FINAL, formando la URL final.

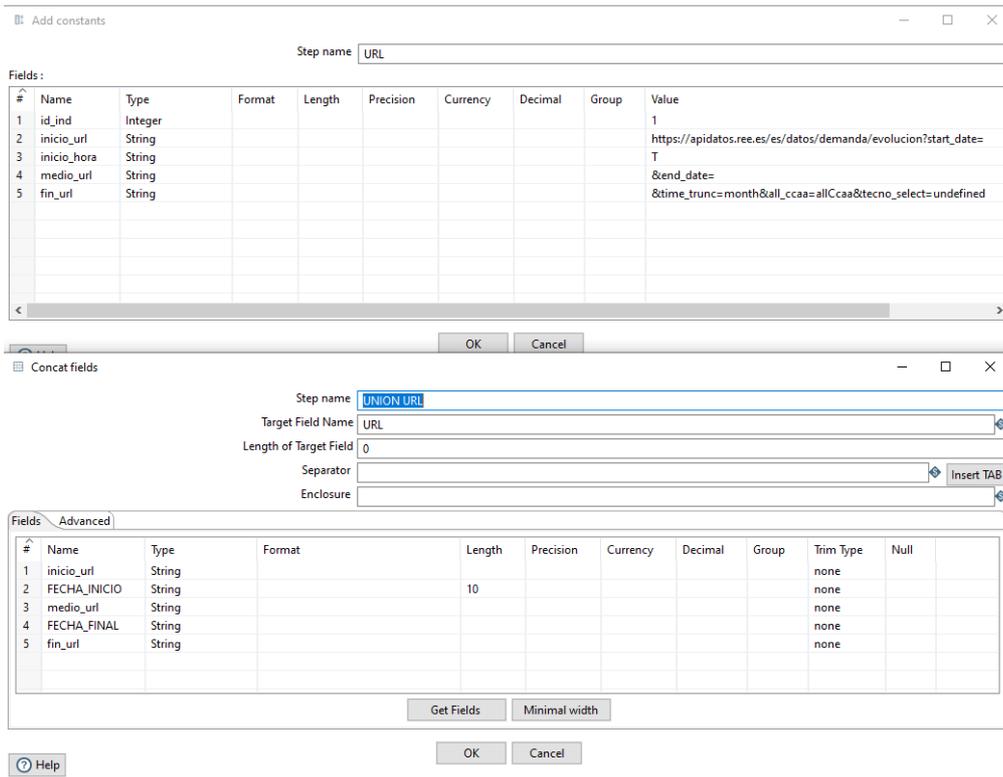


Figura 3-18. Pasos de Añadir constantes y Concatenar campos.

El paso siguiente se llama HTTP Client, que es el que hace la petición a la API para extraer los datos de REData. Como ya está formada la URL con los parámetros, se selecciona la casilla de 'Accept URL from field?' para seleccionar el campo del flujo de datos. Se indica en 'Result field name' el nombre del campo que contiene la respuesta JSON de la API.

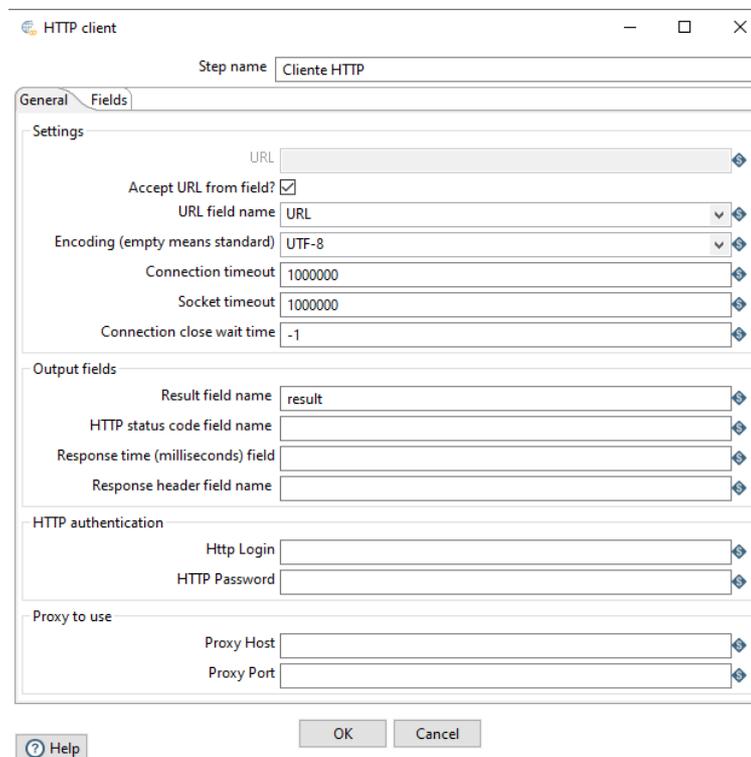


Figura 3-19. Paso de Cliente HTTP.

Con la respuesta JSON, se procede a seleccionar únicamente los campos deseados, en este caso serán el nombre de la comunidad autónoma (community_name), la fecha a la que se refiere el dato (datetime, con

formato YYYY-MM-DDTHH:MM:SS, por ejemplo 2021-02-21T00:00:00), el tipo de fuente energética (tipo, puede traer el valor o no, depende del indicador) y el valor del propio indicador (value). Estas ramas acaban con los mismos campos en el flujo de datos, por lo cual unimos estos flujos en uno para tratarlos de la misma forma.

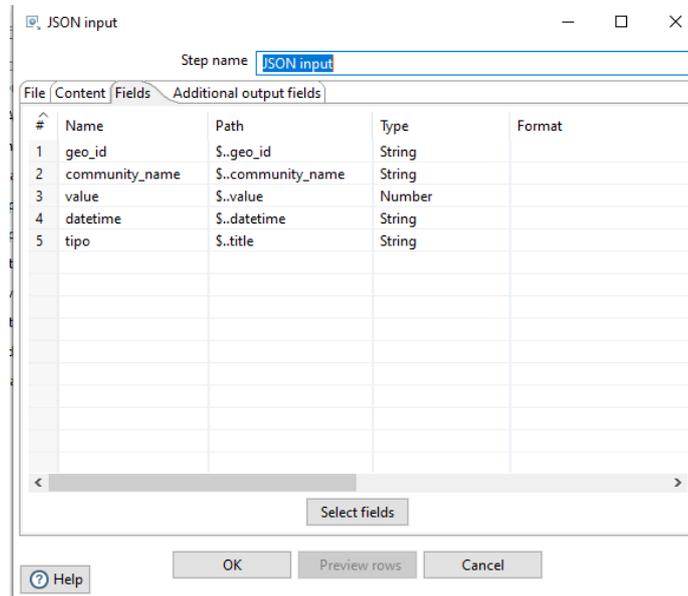


Figura 3-20. Paso de Entrada JSON.

A continuación de esto, hay un paso Database Lookup, el cual busca en la tabla DIM_CCAA el ID_CCAA cuyo DS_CCAA coincida con community_name de la respuesta de la API.

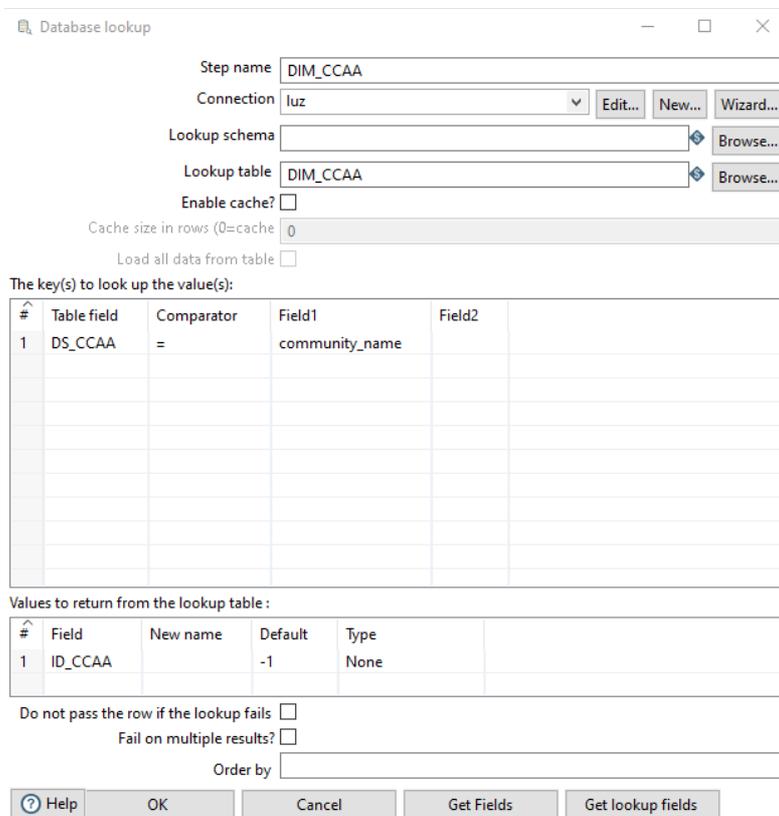


Figura 3-21. Paso de Búsqueda en base de datos.

El siguiente paso es un Modified JavaScript value, donde se usa código JavaScript para extraer del campo datetime la fecha y la hora y transformarlos al formato para hacer las búsquedas en las dimensiones.

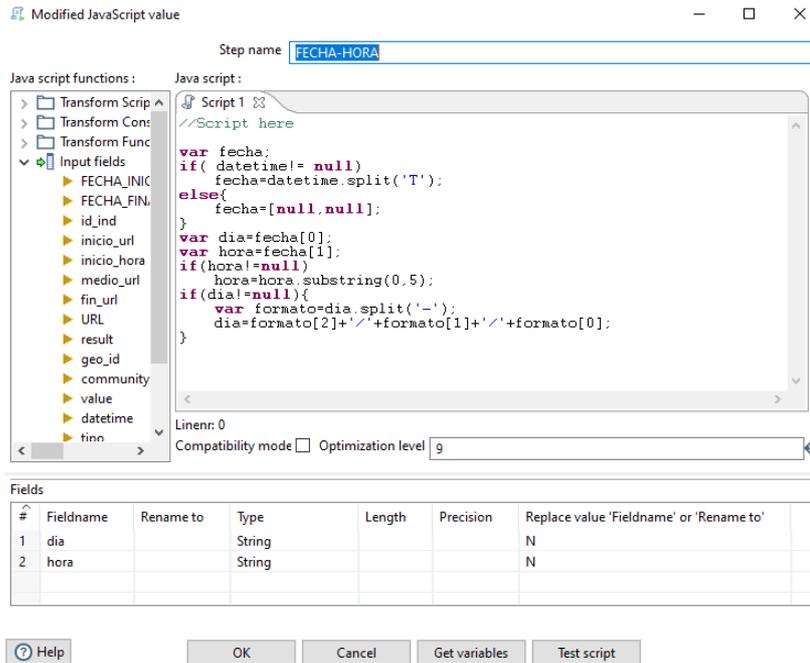


Figura 3-22. Paso de JavaScript.

Seguido de ello, se usa el paso Select values, en el cual se puede seleccionar los campos que siguen en el flujo, los campos que se eliminan y cambios de tipo o formato de estos. En este caso se usará con último fin, para convertir la fecha de String a Date y el value a Number con 3 decimales.

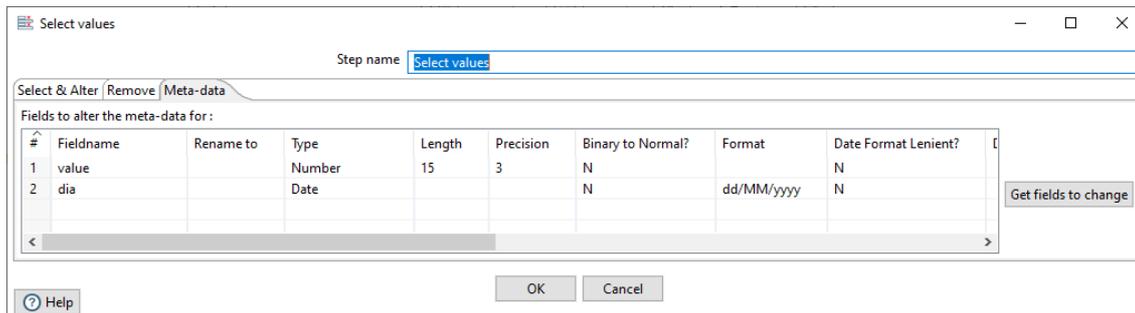


Figura 3-23. Paso de Select values.

Los siguientes pasos son Database Lookup al igual que el de búsqueda en DIM_CCAA, pero para buscar los identificadores de las tablas DIM_HORA, DIM_FECHA y DIM_TIPO donde coincidan sus valores con los de la hora y fecha modificadas en el paso anterior y el tipo proveniente de la respuesta de la API.

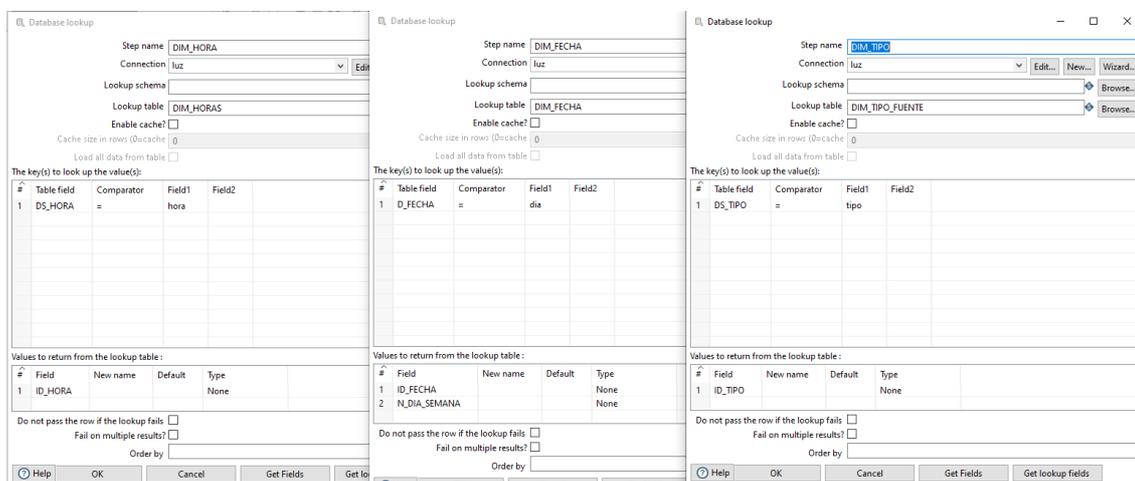


Figura 3-24. Pasos de Búsqueda en base de datos.

Antes de introducir los valores en la tabla TH_INDICADORES, filtramos los registros que tienen su valor nulo, los cuales provienen de las cabeceras del JSON que no interesa cargar en la tabla de hechos, y que el ID_CCAA sea distinto a 0, ya que este es el asociado a 'Todos' y solo interesa los valores de las comunidades autónomas por separado. Para tener un control, se añade también la fecha en la que se ejecuta la transformación, la cual se introduce en el campo F_CREACION de la tabla.

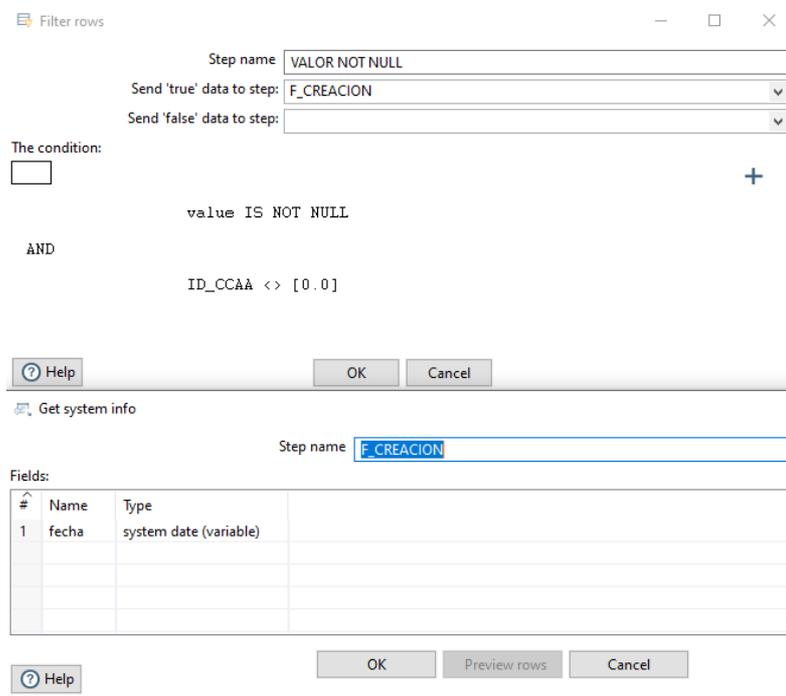


Figura 3-25. Pasos de Filtrar filas y Get system info.

Por último, se selecciona los campos del flujo de datos que cargar en la tabla TH_INDICADORES dentro del Table Output. La razón por la que se usaron antes los pasos Database Lookup es porque se busca que la tabla de hechos sea lo más optima posible, aligerando su almacenamiento, por lo que se vuelca en esta los indicadores de las dimensiones en vez de cargar los valores literales, ya que los campos de tipo numérico son más óptimos.

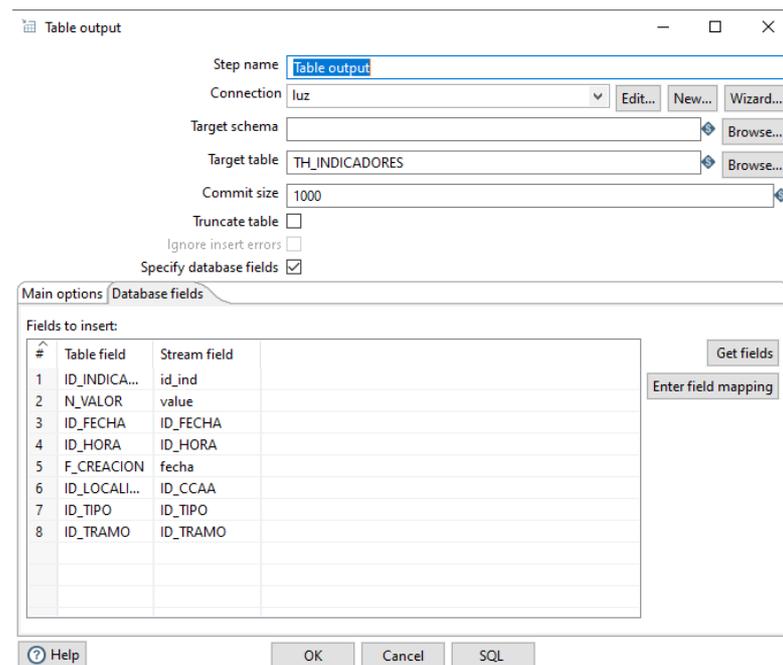


Figura 3-26. Paso de Salida a tabla (TH_INDICADORES).

3.2.2 e-sios

Los indicadores usados de la API de e-sios son más numerosos que los de REData, por lo que, en vez de ramificar la transformación por cada indicador, se ha optado por crear un bucle en el que se lance la transformación tantas veces como indicadores haya pasándole este como parámetro. Por ello, el job de la API de e-sios es más complejo que el de REData.



Figura 3-27. Job ESIOS_JOB.

Este job se compone de una transformación llamada INDICADORES_ESIOS y de otra transformación llamada EXTRACCION_ESIOS. En la primera transformación se observan los siguientes pasos.

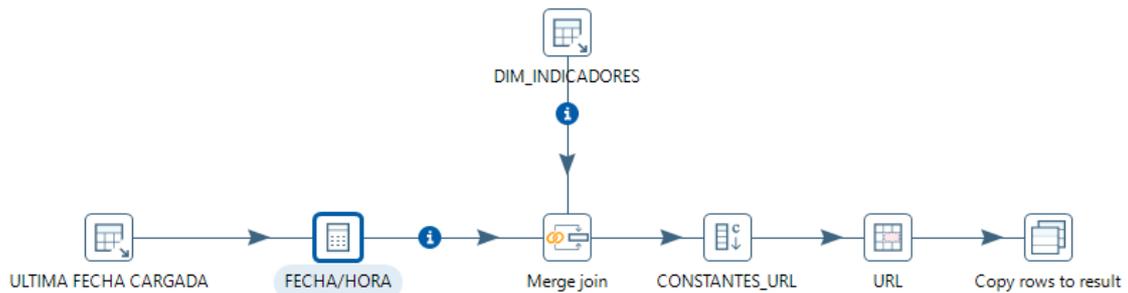


Figura 3-27. Transformación ESIOS_PREEXT.

En el primer paso, se obtiene el identificador de la última fecha cargada de cada indicador de e-sios al igual que al comienzo de la transformación de REData, pero además añadimos la fecha D_FECHA y la siguiente hora a la última cargada. Esta hora se sumará a la fecha D_FECHA con el paso 'Calculator' para obtener la fecha y hora desde la cual hay que extraer datos de la API.

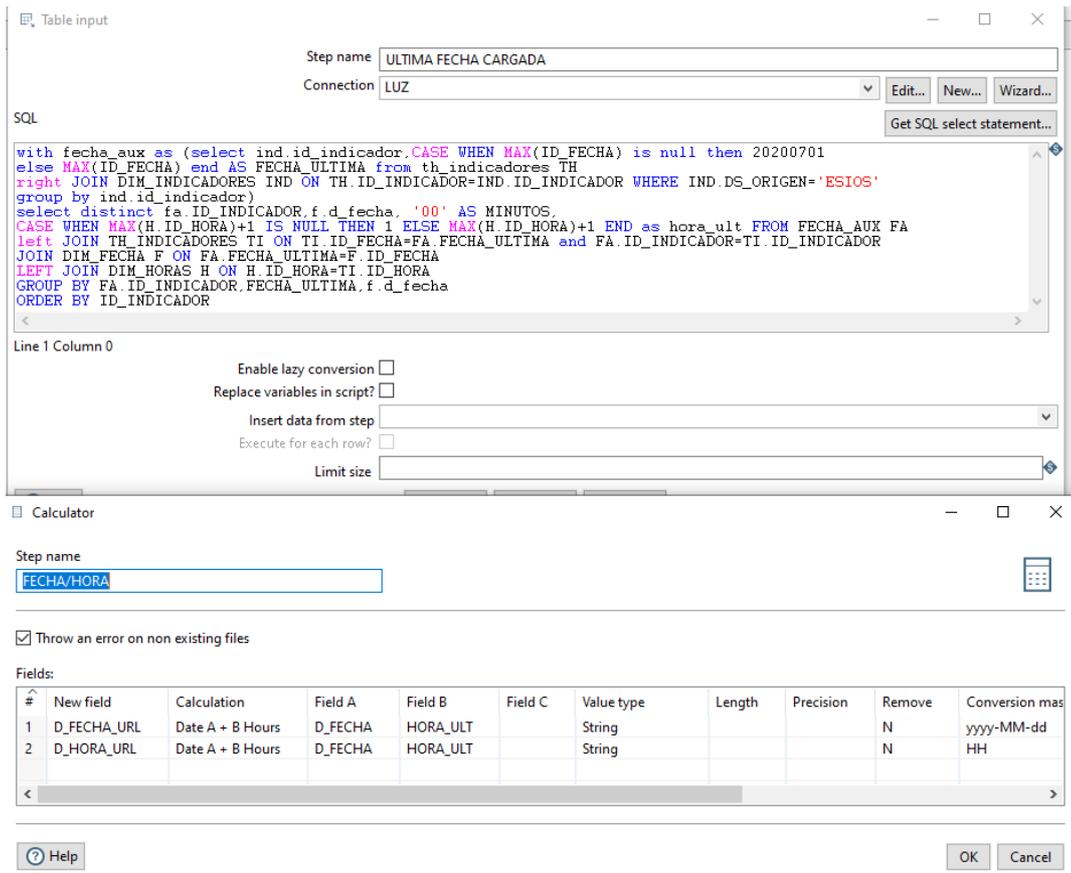


Figura 3-28. Pasos de Entrada de tabla y Calculadora.

Se obtienen los identificadores de los indicadores provenientes de DIM_INDICADORES con DS_ORIGEN='ESIOS' y se realiza una unión mediante ID_INDICADOR con el flujo de datos anterior.

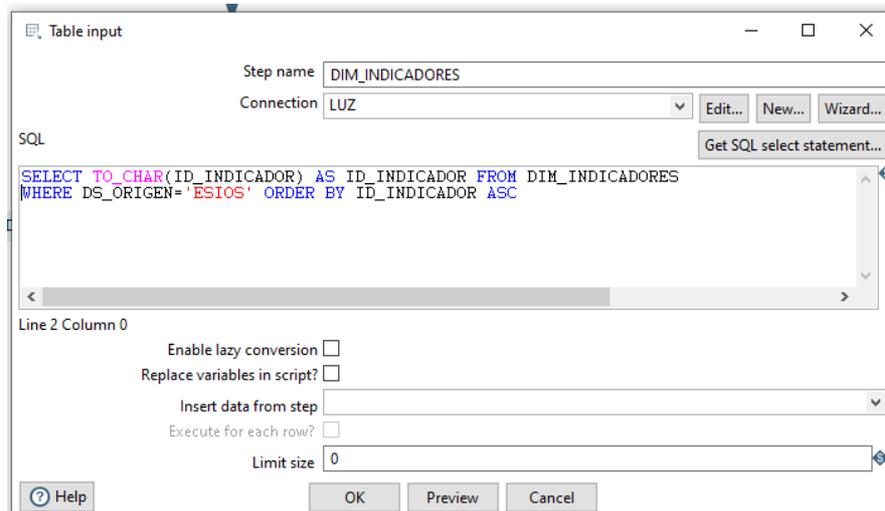


Figura 3-29. Paso de Entrada de tabla.

Además, se añaden varias columnas constantes con los componentes de la URL de la petición, los cuales se concatenan en el Concat fields con la fecha antes obtenida y con los indicadores para así tener un enlace de petición por cada uno de ellos.

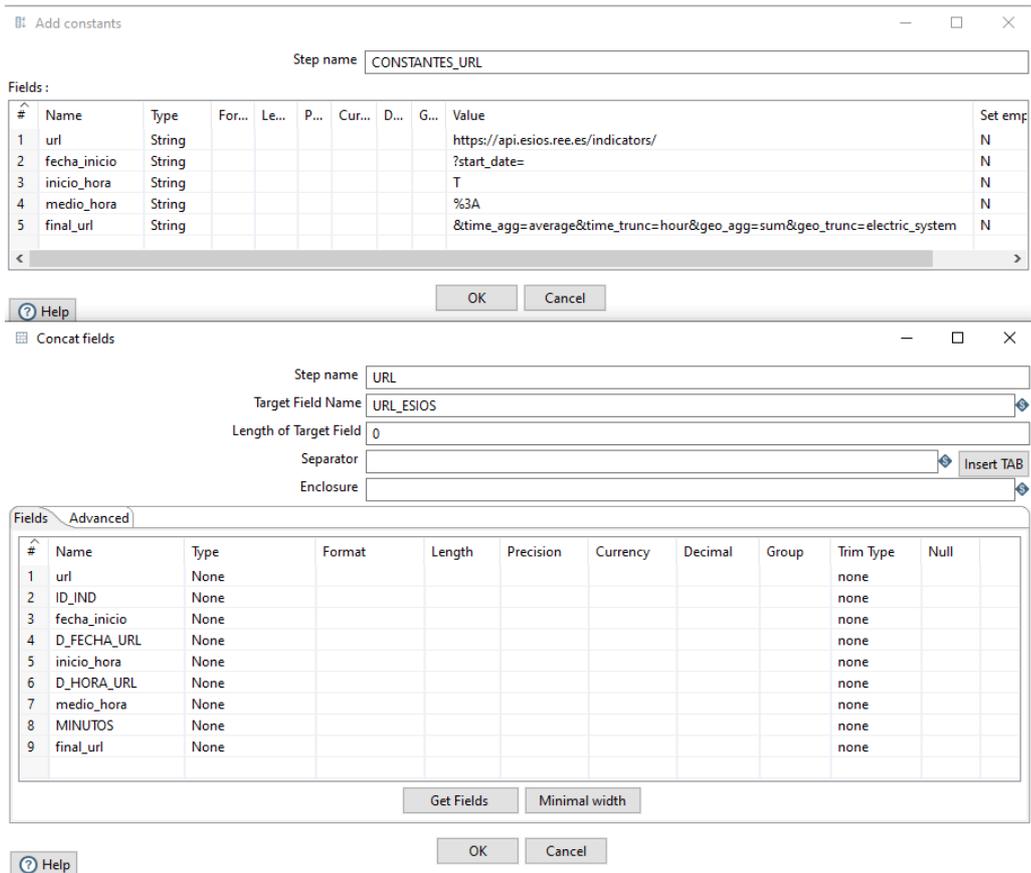


Figura 3-30. Pasos de Añadir constantes y Concatenar campos.

Por último, se usa el paso 'Copy rows to result', que hace que las columnas persistan más allá de esta transformación y se puedan usar más adelante dentro del job al que pertenece.

Finalizada esta transformación, se inicia la siguiente, la cual es la encargada de la extracción de datos de la API. Para conseguir que se ejecute para cada indicador, se debe seleccionar la opción 'Execute every input row'.

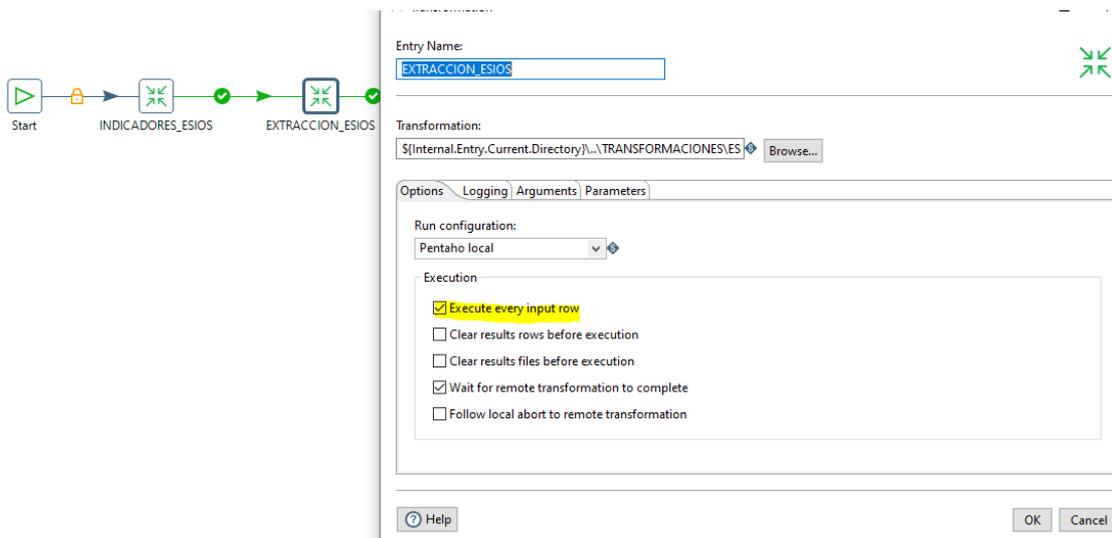


Figura 3-31. Detalle de la transformación ESIOS_EXTRACCION.

Esta transformación se inicia con el paso Get rows from result, de donde obtenemos los campos URL_ESIOS e ID_INDICADOR de la transformación anterior. Tras ello, se añade como constante el campo Authorization de la petición a la API, donde va incluido el token personal de acceso (Token token="xxxx").

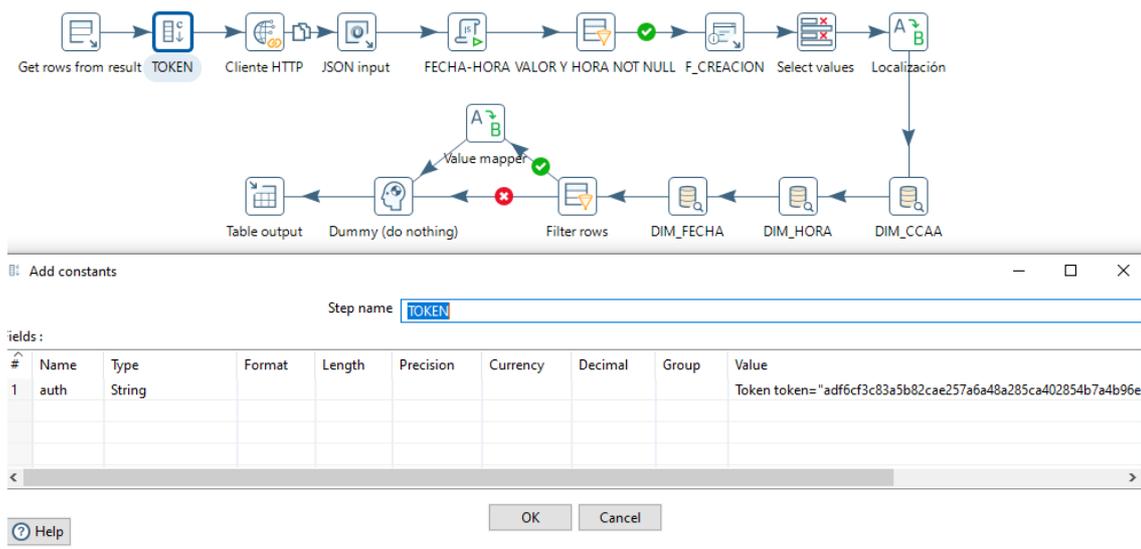


Figura 3-32. Transformación ESIOS_EXTRACCION y paso de Añadir constantes.

El siguiente paso es el de HTTP Client para realizar la petición GET a la API e·sios. Al igual que en la transformación de REData, se usa la URL_ESIOS que viene del flujo de datos. La única diferencia es que en este caso se debe añadir la cabecera Authorization, seleccionándola en la pestaña Fields en la parte de HTTP Headers.

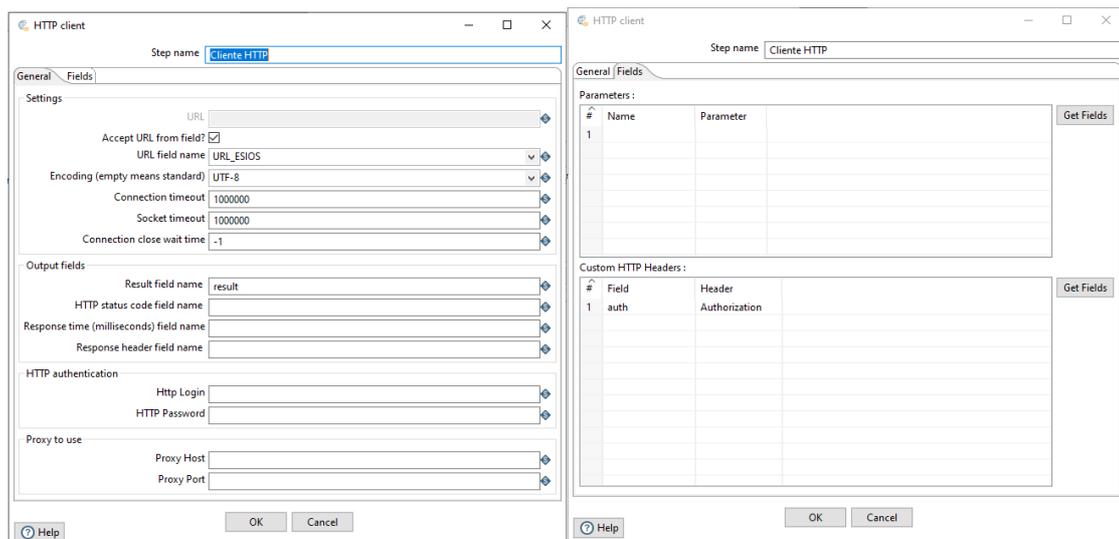


Figura 3-33. Paso de Cliente HTTP.

Una vez obtenido la respuesta de la API, se escoge los campos deseados del campo JSON. Para este caso se seleccionan el nombre de la localización (geo_name), el valor del indicador (value) y la fecha del registro (datetime, con formato YYYY-MM-DDTHH:MM:SS).

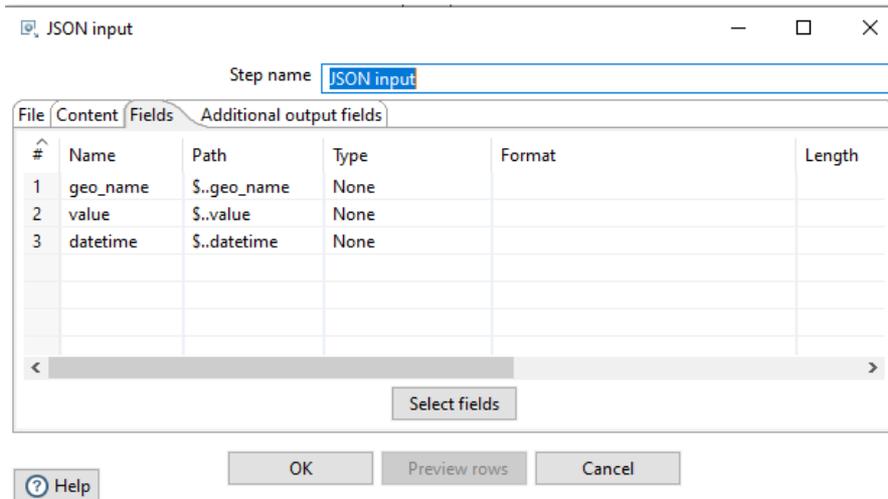


Figura 3-34. Paso de Entrada JSON.

El próximo paso es un Modified JavaScript value, donde se usa código JavaScript para extraer del campo datetime la fecha y la hora y transformarlos al formato para hacer las búsquedas en las dimensiones.

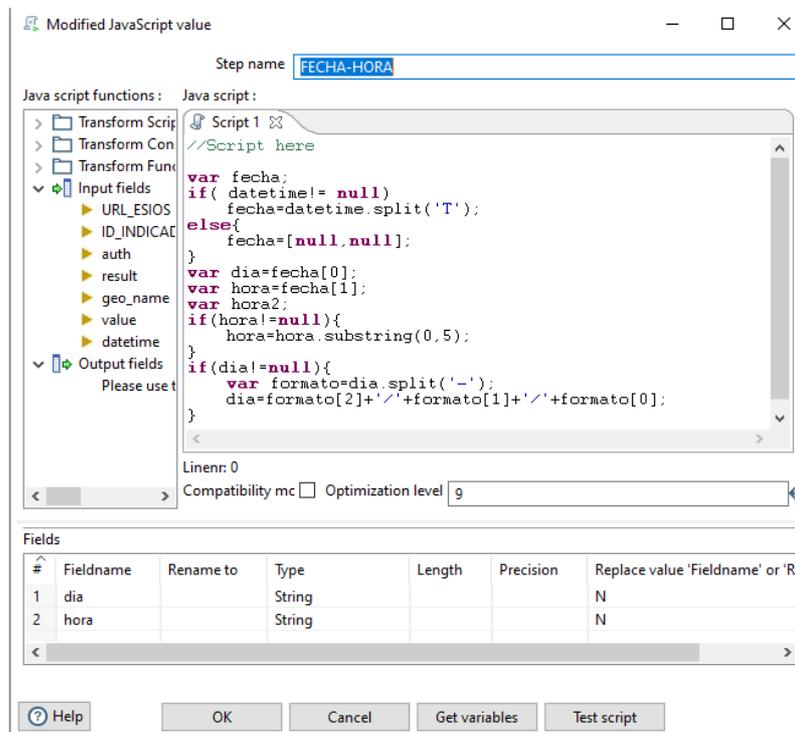


Figura 3-35. Paso de JavaScript.

A continuación, se filtra por los registros que no tienen ni la hora ni el valor nulos, ya que estos pertenecen a la cabecera de la respuesta. Además, se añade la fecha actual como F_CREACION para tener un control sobre cuándo se ha insertado el dato en la tabla de hechos.

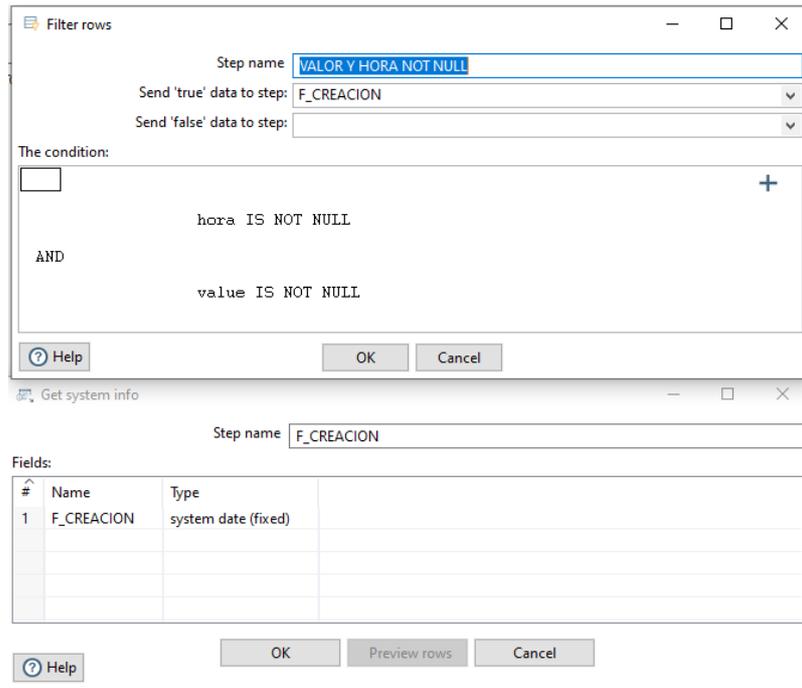


Figura 3-36. Pasos de Filtrar filas y Get system info.

En el siguiente Select value, se cambia de tipo y formato varios campos. El value se pasa a Number con 3 decimales, la fecha del registro al tipo Date manteniendo el formato DD/MM/YYYY, y se cambia el formato a F_CREACION para añadirle la hora.

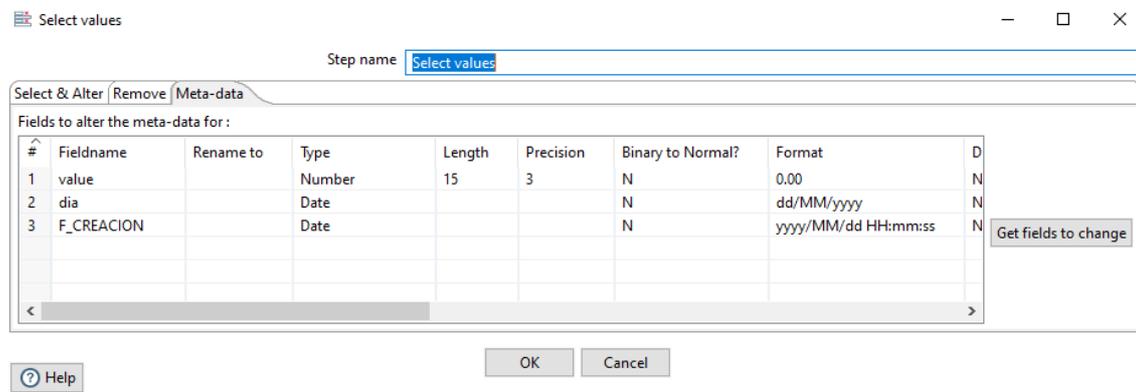


Figura 3-37. Paso de Select values.

Antes de hacer las búsquedas de las dimensiones, se cambia de valor varios registros de geo_name, ya que los valores literales que están en DIM_CCAA son los usados en REData, por lo que se mapean para realizar correctamente la búsqueda en la dimensión.

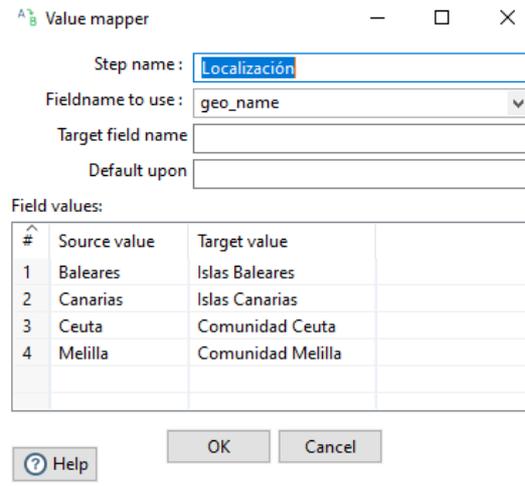


Figura 3-38. Paso de Mapear valores.

Las búsquedas de identificadores son en las dimensiones DIM_CCAA, donde se busca que el ID_CCAA tenga asociado el DS_CCAA con el valor de geo_name, DIM_HORA, donde se busca que el ID_HORA y el ID_TRAMO tengan asociado el DS_HORA con el valor de hora (el campo proveniente del flujo de la transformación), y DIM_FECHA, donde se busca que el ID_FECHA y el N_DIA_SEMANA tengan asociado el DS_FECHA con el valor de día (el campo proveniente del flujo de la transformación).

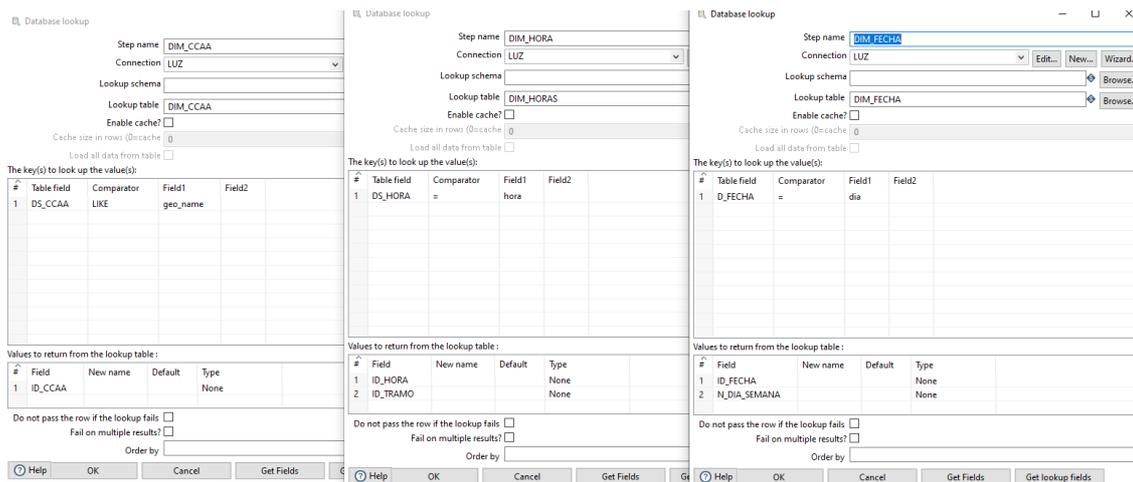


Figura 3-39. Pasos de Búsqueda en base de datos.

A partir de aquí, se bifurca el camino, donde los registros que pertenecen a los días sábado o domingo (N_DIA_SEMANA es 6 o 7) cambian su ID_TRAMO al asociado a la hora valle.

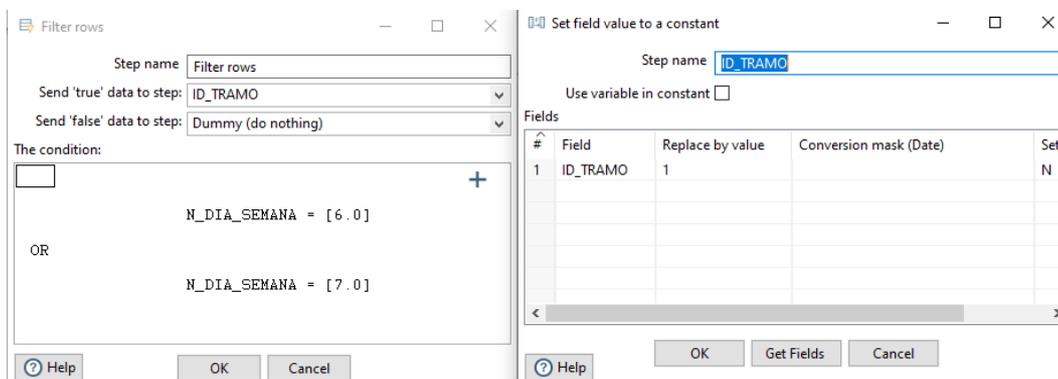


Figura 3-40. Pasos de Filtrar filas y Establecer valor de campo a constante.

Por último, se seleccionan los campos del flujo de datos a introducir en las columnas de la tabla de hechos TH_INDICADORES. Una vez finalizada esta transformación, se vuelve a ejecutar con otro indicador distinto

hasta que se acabe la lista proveniente de la transformación INDICADORES_ESIOS.

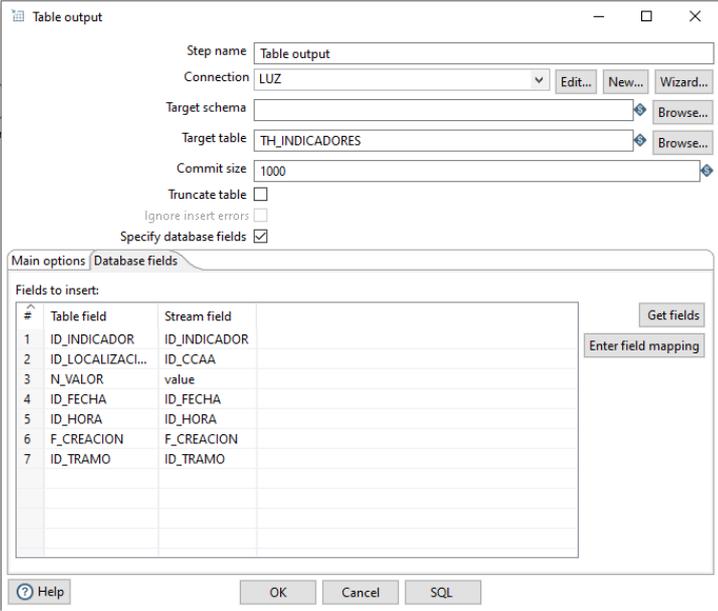


Figura 3-41. Paso de Salida a tabla (TH_INDICADORES).

4 CUADRO DE MANDO

El cuadro de mando es donde se va a poder ver los datos transformados de forma gráfica, de forma que tengan un sentido a la vista del usuario. Éste se va a componer de 5 páginas, las cuales se describen a continuación indicando sus elementos.

4.1 Página de inicio

La página principal es la siguiente:

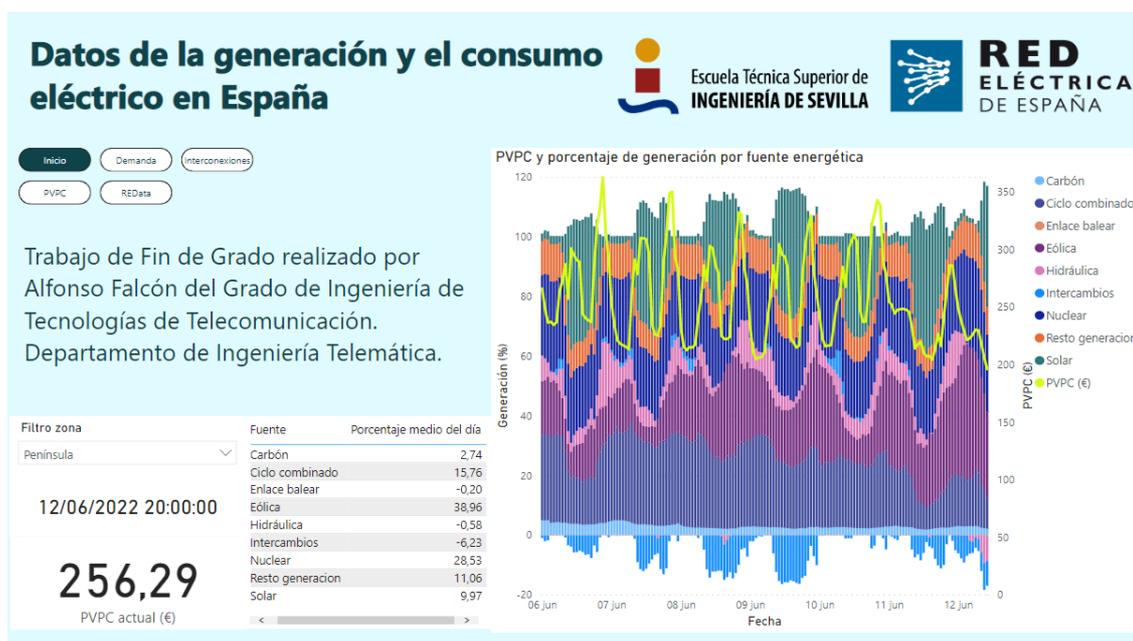


Figura 4-1. Panta inicial del cuadro de mando.

El título y el texto son simplemente cuadros de texto, y los logos son insertados desde archivos locales. Los botones de navegación son elementos propios de Power BI en el que se selecciona si navegar entre páginas o entre marcadores, siendo este caso entre las páginas del cuadro de mando. Empezando por la izquierda tenemos tres elementos juntos:



Figura 4-2. Elemento visual con PVPC actual o más reciente.

Consta de dos tarjetas de datos y un filtro para ellos. Este filtro selecciona una localización concreta perteneciente a la columna DS_CCAA de la vista creada para este elemento V_PVPC_ACTUAL. Para ello, se añade el elemento 'Segmentación de datos' y se incluye dicha columna. Esta vista nada más que tiene un registro por cada localización, por lo que seleccionando una de estas, obtenemos el N_VALOR del PVPC de la D_FECHA_HORA de la hora peninsular.

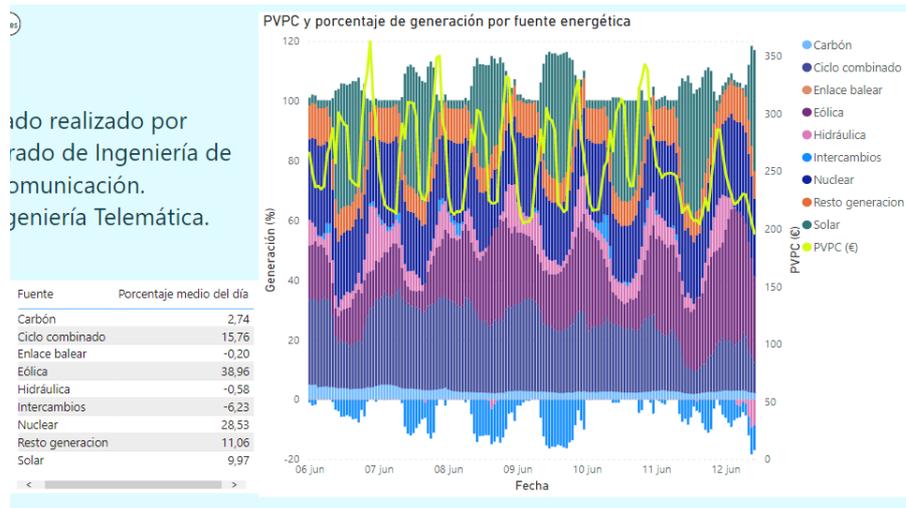


Figura 4-3. Elementos visuales de porcentaje de generación energética.

En el resto de la página se encuentran dos elementos más, una tabla y una gráfica de barras y líneas. Los datos de la tabla pertenecen al día relativo en el que se está visualizando. Hay dos columnas, la DS_INDICADOR_CORTO, renombrado a 'Fuente', y el PORCENTAJE, renombrado a 'Porcentaje medio del día', de la vista V_PORCENTAJE_GENERACION. De esta última columna se hace un promedio con los datos disponibles del día.

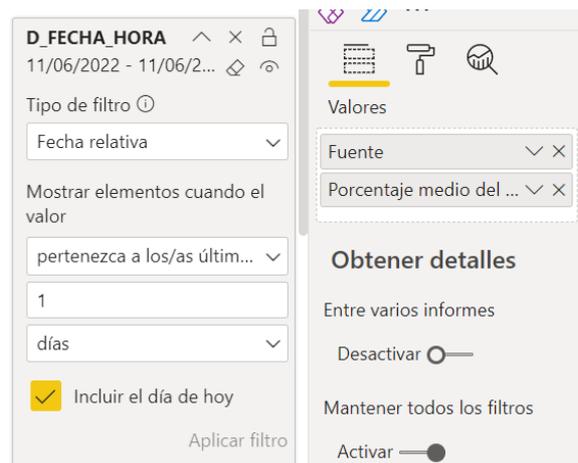


Figura 4-4. Configuración de la tabla de generación energética.

El siguiente gráfico es uno de los más completos del proyecto. Se trata de la representación del porcentaje de generación energética según su fuente comparado con el PVPC, ya que este último depende de dónde provenga la energía, pues abaratará o encarecerá los costes de producción.

En esta gráfica se usan dos vistas, V_INDICADOR_1001 y V_PORCENTAJE_GENERACION, los cuales se han relacionado anteriormente por la columna D_FECHA_HORA para mostrar los datos pertenecientes al mismo día y hora. En el eje X tenemos la fecha, cuyos valores corresponden a D_FECHA_HORA, en este caso de V_PORCENTAJE_GENERACION, aunque es lo mismo que colocar la misma columna de V_INDICADOR_1001. Este eje está filtrado a los últimos siete días desde el día que se visualiza. El eje Y es compartido por el porcentaje, a la izquierda, y por el PVPC, a la derecha. Para ello, indicamos en 'Valores de columnas' el campo PORCENTAJE de V_PORCENTAJE_GENERACION renombrado a 'Generación (%)', y en 'Valores de líneas' el campo N_VALOR de V_INDICADOR_1001 renombrado a 'PVPC (€)'. Para indicar a qué fuente pertenece cada porcentaje, se añade en 'Serie de columnas' el campo DS_INDICADOR_CORTO de V_PORCENTAJE_GENERACION, visualizando también la leyenda de colores de esta a la derecha. Además, en 'Información sobre herramientas', se incluye el N_VALOR de V_PORCENTAJES_GENERACION renombrado a 'Generación (MW)' para que indique el valor de este al pasar el ratón por encima de alguna fuente en una fecha y hora concretas. Se ha filtrado el PVPC para mostrar los datos de la península, ya que los datos en tiempo real de estos indicadores pertenecen a esta zona y de

donde se genera la gran mayoría de energía.

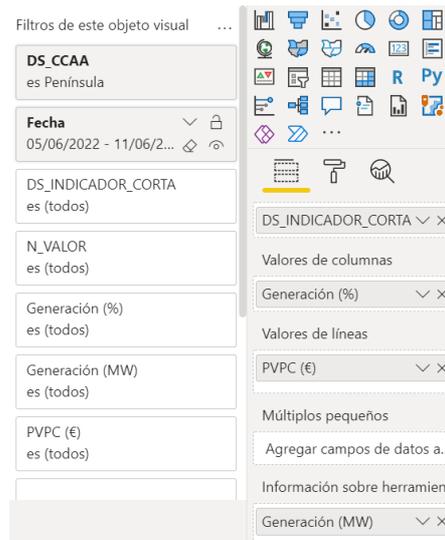


Figura 4-5. Configuración del gráfico de porcentajes de generación energética.

4.2 Demanda

Dando click en el botón de navegación, se pasa a la siguiente página que es la de Demanda. Esta página consta de cinco elementos incluyendo los botones de navegación ya vistos.

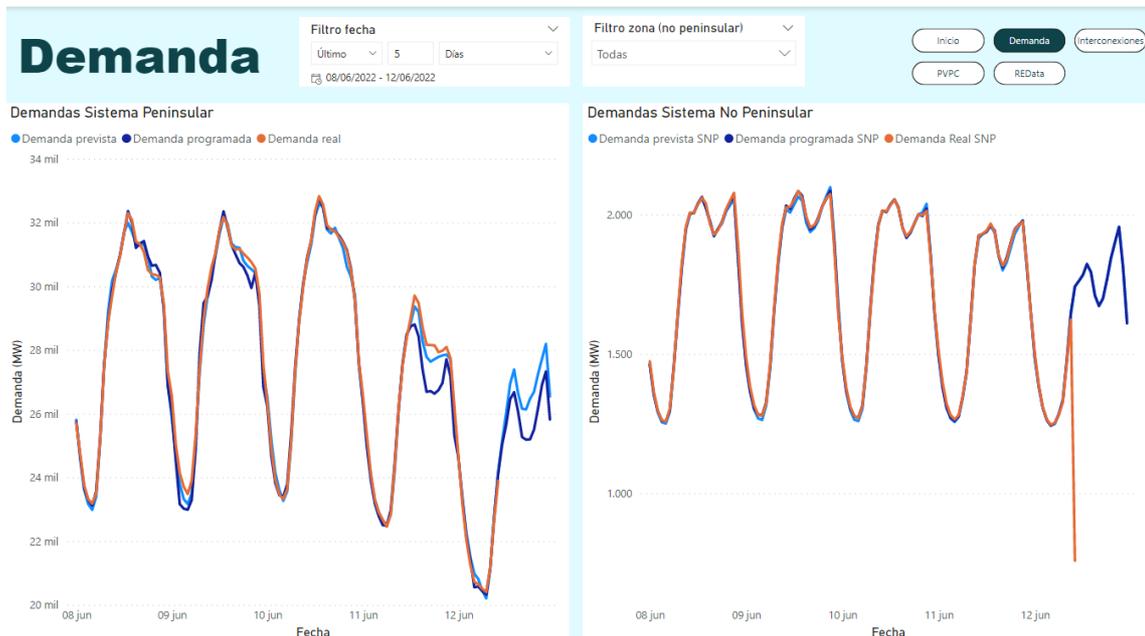


Figura 4-6. Pantalla de gráficas de demanda.

Los primeros elementos son dos filtros de datos. El primero de ellos es un filtro común para seleccionar los últimos 5 días desde el día en el que se visualiza. Para que sea común, se debe relacionar dentro de Power BI las columnas D_FECHA_HORA de las vistas V_DEMANDA_SP y V_DEMANDA_SNP, incluyendo en el filtro cualquiera de estas columnas, siendo en este caso la de V_DEMANDA_SNP.

Con este filtro aplicado ya está disponible los datos de los distintos tipos de demanda del sistema peninsular, que es el gráfico de la izquierda. Para este se ha usado el tipo gráfico de líneas, donde en el eje se coloca el

campo D_FECHA_HORA renombrado a 'Fecha', en valores el N_VALOR renombrado a 'Demanda (MW)', y en la leyenda el campo DS_INDICADOR_CORTA para que muestre las distintas demandas con colores diferentes y poder comparar sus valores. Todos estos campos son extraídos de la vista V_DEMANDA_SP.



Figura 4-7. Configuración de las gráficas de demanda.

A su derecha tenemos la gráfica de los tipos de demanda del sistema no peninsular. Este tiene los mismos campos que el anterior, pero procedentes de la vista V_DEMANDA_SNP. La diferencia con el anterior es que el sistema no peninsular se compone de varias zonas, por lo que se ha incluido encima de esta gráfica un filtro de DS_CCAA de la vista V_DEMANDA_SNP para seleccionar la localización de la cual mostrar los datos, renombrándola a 'Filtro zona (no peninsular)'.

4.3 Interconexiones internacionales

Pasando a la siguiente página, se muestran varios elementos que hacen referencia a los intercambios de energía que hace España con los países vecinos.

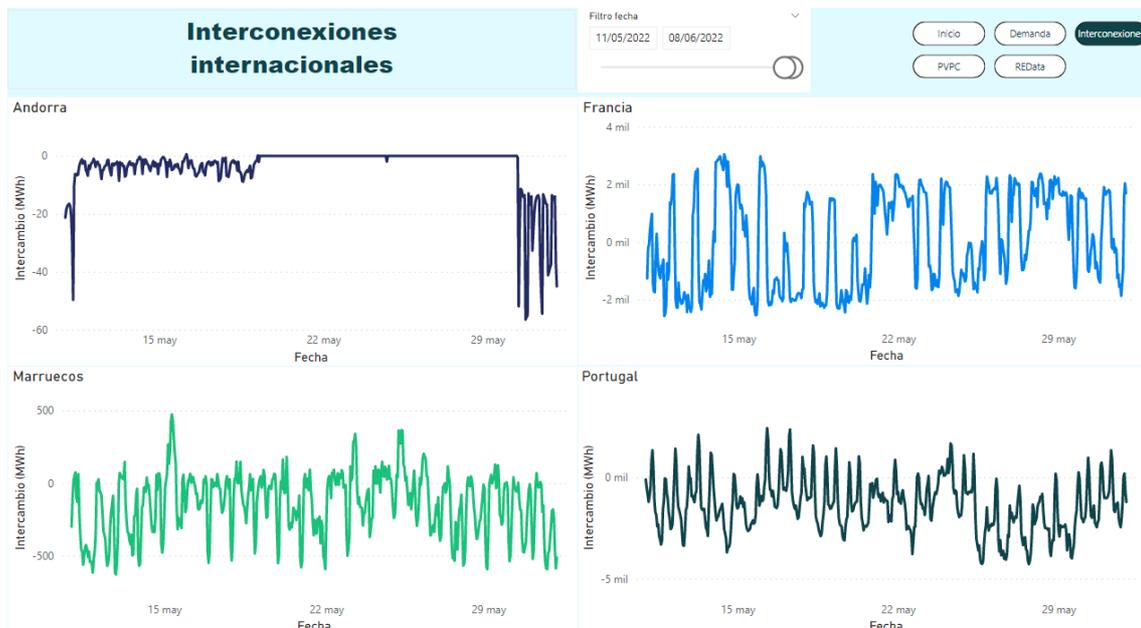


Figura 4-8. Pantalla de gráficas de interconexiones.

Se muestran cuatro gráficas, las cuales usan los mismos campos. En el eje X se añade el campo D_FECHA_HORA renombrado a 'Fecha' y en el campo de 'Valores' se añade N_VALOR renombrado a 'Intercambio (MWh)', ambas columnas de la vista V_INTERCAMBIOS. Se cambia el color de línea de cada gráfica en el sector de Formato, en el apartado de 'Colores de datos'. La diferencia de cada gráfica es que se le

ha añadido un filtro estático a cada una, seleccionando un DS_CCAA específico para mostrar los datos de los intercambios energéticos con cada país específico, teniendo los valores de Andorra, Francia, Marruecos y Portugal.

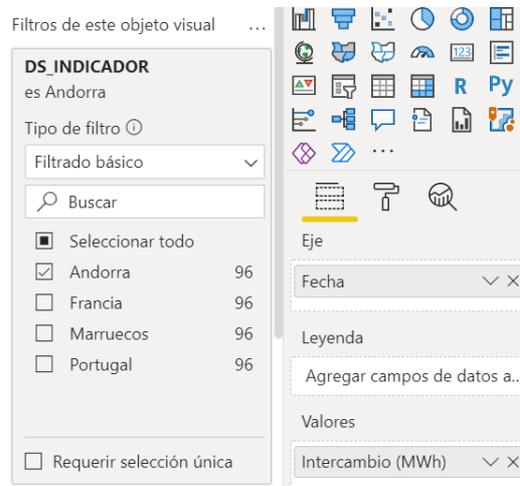


Figura 4-9. Configuración de la gráfica de interconexión con Andorra.

Estos datos los proporciona la API de e-sios al mes siguiente de su medida, por lo que se ha optado por añadir un filtro de fecha en el que se pueda seleccionar un rango de días en vez de filtrar por un número de días relativos al de visualización. Para seleccionar este tipo de filtro, una vez añadido el campo D_FECHA_HORA renombrado a 'Filtro fecha' de la vista V_INTERCAMBIOS, se clicca en la flecha de la esquina superior derecha y se selecciona el tipo de filtro 'Entre'.



Figura 4-10. Configuración del filtro de la fecha.

4.4 PVPC

La siguiente página es de las más interesantes, ya que se muestran los datos referentes al 'Precio voluntario para el pequeño consumidor'.



Figura 4-11. Pantalla de gráficas del PVPC.

Para empezar, se ha colocado dos filtros en la parte superior de la página, el primero selecciona datos de los últimos 7 días haciendo referencia a la columna D_FECHA_HORA y el segundo selecciona una zona dentro de DS_CCAA, ambos de la vista V_INDICADOR_1001.

El siguiente elemento es un gráfico de barras en el que se muestra el valor del PVPC por día y hora, diferenciándose entre los tramos horarios establecidos, viendo que se cumple que los valores máximos se establecen en los tramos de hora punta y los mínimos en los de hora llana. Para ello, se ha indicado en el eje el campo D_FECHA_HORA renombrado a 'Fecha', en 'Valores' el campo N_VALOR renombrado a 'PVPC (€)' y, para diferenciar entre tramos horarios, se incluye en 'Leyenda' el campo de DS_TRAMO, todas estas columnas pertenecientes a la vista V_INDICADOR_1001. Para asociar los colores a cada tramo, en el sector de Formato se expande el apartado de 'Colores de datos', seleccionando un color para cada uno.

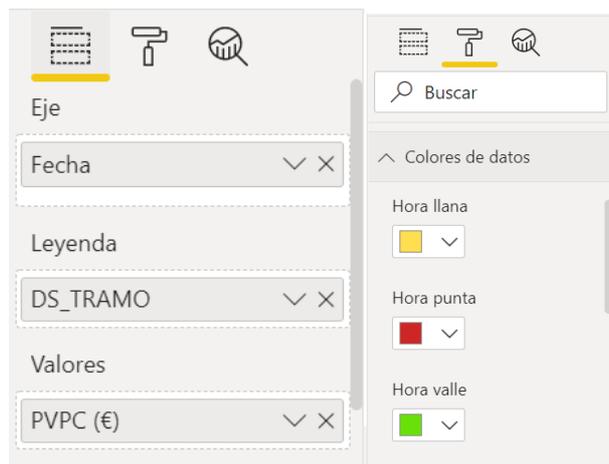
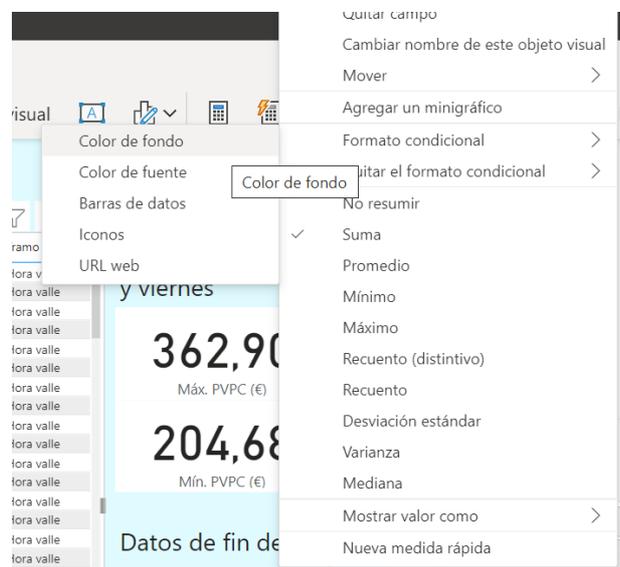


Figura 4-12. Configuración de la gráfica de barras del PVPC.

A su derecha se encuentran los datos de la gráfica de barras, pero detallados en una tabla. Para ello, se han añadido la columna D_FECHA_HORA, renombrada a 'Fecha', la columna N_VALOR, renombrada a 'PVPC (€)', y la columna DS_TRAMO, renombrada a 'Tramo'. Además, se ha añadido un código de colores para el PVPC en el que cuanto más oscuro y rojo, más elevado es el valor. Esto se configura dando click en la columna de 'PVPC (€)' dentro de Valores y se selecciona Formato condicional > Color de fondo. A partir de aquí aparece una pantalla en la que se puede seleccionar el valor en el que se quiere basar el degradado de color y los colores que lo formarán.



Figura 4-13. Campos de la tabla de PVPC.



Color de fondo - PVPC (€)

Estilo de formato: Degradado | Aplicar a: Solo valores

¿En qué campo debemos basar esto? Suma de N_VALOR | Resumen: Suma | ¿Cómo se deben dar formato a los valores vacíos? Sin formato

Mínimo: Valor más bajo | Centro: Valor medio | Máximo: Valor más alto

Agregar un color medio

[Más información sobre el formato condicional](#) Aceptar Cancelar

Figura 4-14. Configuración de la tabla de PVPC.

Para finalizar, se han añadido cuatro elementos, el valor máximo y mínimo del PVPC tanto entre semana como en fin de semana. Se han colocado cuatro tarjetas de datos en los que se ha seleccionado el máximo y mínimo del campo N_VALOR, renombrados a 'Máx. PVPC (€)' y 'Mín. PVPC (€)' respectivamente, de la vista V_INDICADOR_1001, filtrando por un lado que el DS_DIA_SEMANA tuviera el valor 'Lunes', 'Martes', 'Miércoles', 'Jueves' o 'Viernes', y por otro lado 'Sábado' o 'Domingo'. Esto es así ya que en los fines de semana no hay tramos horarios, siempre se supone 'hora valle', de ahí el diferenciar estos días.

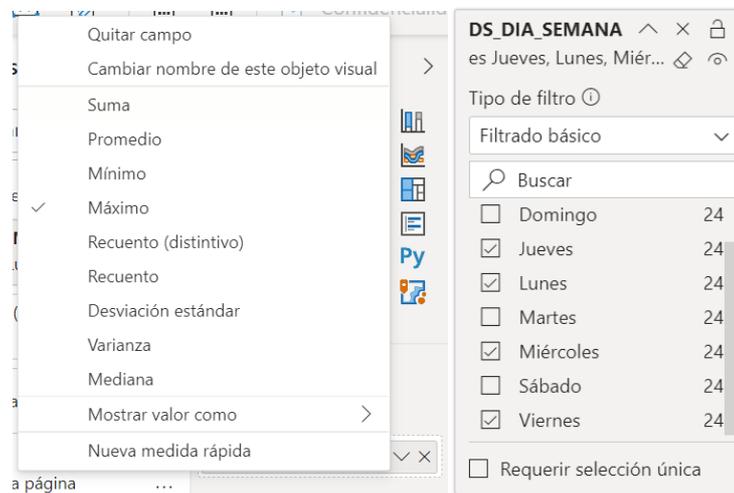


Figura 4-15. Configuración del cuadro de dato del máximo valor de PVPC entre semana.

4.5 Mapas de RData

En la siguiente página hay cuatro pestañas por las cuales se pueden acceder a los distintos mapas con datos de la API de RData. Inicialmente, la opción de 'Mapas de formas' no está disponible en Power BI. Para poder utilizar esta herramienta, es necesario acceder a Archivo>Opciones>Características de versión preliminar, y activar la opción 'Objeto visual Mapas de formas' [17].

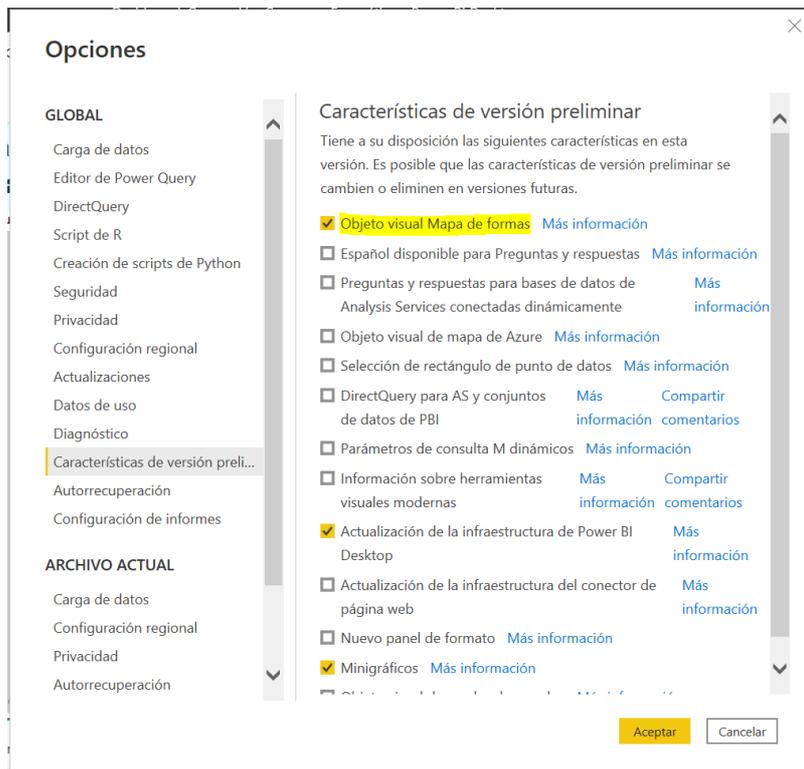


Figura 4-16. Configuración para mostrar Mapa de formas.

Como se explicó anteriormente, los botones se pueden usar para navegar entre marcadores. Los marcadores son accesos directos a una página con sus elementos configurados de forma concreta. Más adelante se explicará cómo usarlos. A continuación, se muestra el formato de esta página.

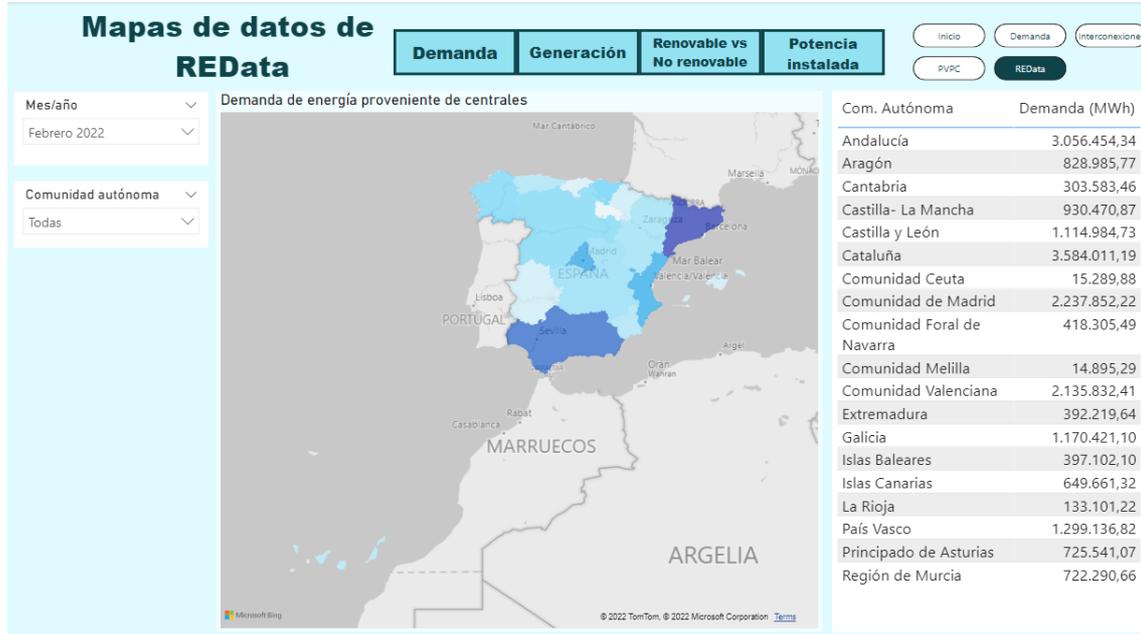


Figura 4-17. Pantalla de mapa de REData correspondiente a la demanda energética.

Se trata de cuatro elementos, en los que se distinguen dos filtros, un mapa y una tabla. Todos los campos usados para esta página provienen de la vista V_INDICADOR_1, que son los datos pertenecientes a la demanda de energía. El primer filtro es el de fecha, seleccionando la columna DS_FECHA y renombrándola a 'Mes/año', donde se puede elegir un mes de un año en concreto para mostrar datos. Además, se incluye el filtro de zona en el que se selecciona el campo DS_CCAA y se renombra a 'Comunidad autónoma'.

El siguiente elemento es un mapa donde se representan los datos de la demanda de energía visualmente mediante un degradado de color dependiendo del valor que tenga. En el campo 'Información sobre herramienta' se inserta N_VALOR renombrado a 'Demanda (MWh)'. Esto es para indicar el valor de la demanda al pasar el ratón por una comunidad autónoma. En el campo 'Ubicación' se inserta la columna DS_CCAA, la cual se renombra a 'Com. Autónoma'. El elemento de mapa reconoce los nombres de las comunidades autónomas, las cuales tienen el mismo nombre que en los mapas de Bing. Para mostrar el degradado de colores dependiendo del valor, se selecciona en el sector 'Formato' el apartado 'Colores de datos' y se clickea sobre el cuadrado fx. Tras ello, se abrirá una ventana en la que se selecciona Degradado como estilo, y N_VALOR como el campo del cual depende este degradado.

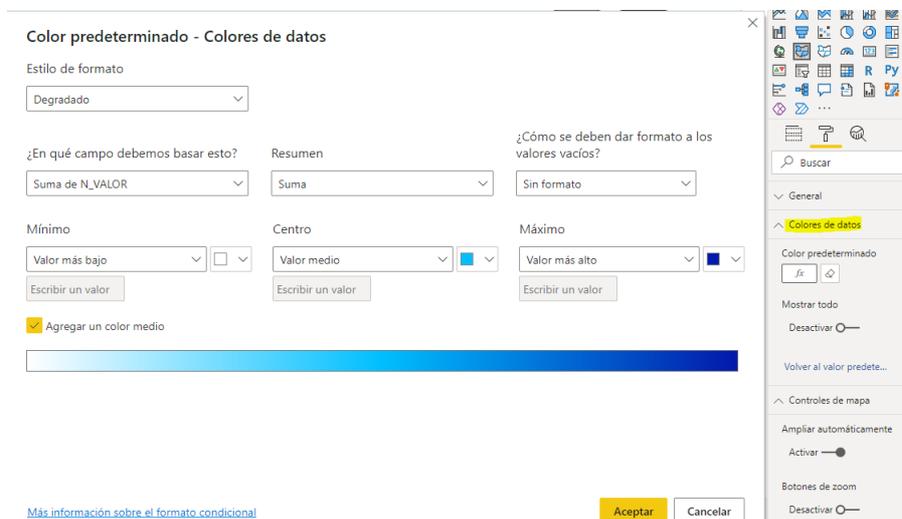


Figura 4-18. Configuración de colores del mapa según el valor del indicador.

A la derecha del mapa se añade un mapa en el que se ven los datos en el formato de tabla. En la tabla se han añadido los campos DS_CCAA renombrado a 'Com. Autónoma' y el campo N_VALOR con el nombre de 'Demanda (MWh)'.

Com. Autónoma	Demanda (MWh)
Andalucía	3.056.454,34
Aragón	828.985,77
Cantabria	303.583,46
Castilla- La Mancha	930.470,87
Castilla y León	1.114.984,73
Cataluña	3.584.011,19
Comunidad Ceuta	15.289,88
Comunidad de Madrid	2.237.852,22
Comunidad Foral de Navarra	418.305,49
Comunidad Melilla	14.895,29
Comunidad Valenciana	2.135.832,41
Extremadura	392.219,64
Galicia	1.170.421,10
Islas Baleares	397.102,10
Islas Canarias	649.661,32
La Rioja	133.101,22
País Vasco	1.299.136,82
Principado de Asturias	725.541,07
Región de Murcia	722.290,66

Figura 4-19. Campos de la tabla de demanda energética.

Estos elementos van a ser comunes en las demás pestañas, por lo que en el panel de Selección se agrupan estos elementos y se renombra el conjunto a 'Demanda'. Se copia y pega dicho conjunto tres veces, cambiando sus nombres a 'Generación', 'Renovables' y 'Potencia instalada'. Para poder configurar cada uno de ellos correctamente, se quita la visibilidad de los demás.



Figura 4-20. Agrupación de elementos visuales.

La diferencia entre las de 'Generación' y 'Potencia instalada' con la de 'Demanda' son los colores y que a éstas se les añade otro filtro, por lo demás usan los mismos campos, pero de las vistas V_INDICADOR_2 y V_INDICADOR_4 respectivamente. El otro filtro que se le añade es el campo DS_TÍPO renombrado a 'Fuente energética', tanto para Generación como para Potencia.



Figura 4-21. Pantalla de mapa de REData correspondiente a la generación energética.



Figura 4-22. Pantalla de mapa de REData correspondiente a la potencia instalada.

Por último, se configura la pestaña de 'Renovable vs No renovable'. En el mapa de esta pestaña se van a mostrar varias columnas de la vista V_INDICADOR_3 en vez de N_VALOR, y son N_RENOVABLE renombrado a 'Renovable (MWh)', N_NO_RENOVABLE renombrado a 'No renovable (MWh)', PORCENTAJE_RENOVABLE renombrado a '% Renovable' y PORCENTAJE_NO_RENOVABLE renombrado a '% No renovable'.



Figura 4-23. Configuración del mapa de REData correspondiente al porcentaje de energía generada a partir de fuentes renovables vs no renovables.

Además, se ha cambiado la escala de colores en función de PORCENTAJE_RENOVABLE, siendo rojo un 0% y verde un 100%.



Figura 4-24. Pantalla de mapa de REData correspondiente al porcentaje de energía generada a partir de fuentes renovables vs no renovables.

La tabla de la derecha tiene los campos DS_CCAA renombrado a 'Com. Autónoma', N_RENOVABLE renombrado a 'Renovable (MWh)' y N_NO_RENOVABLE renombrado a 'No renovable (MWh)'.

Una vez configurados y agrupados todos los elementos, se procede a realizar los marcadores. Mostrando únicamente uno de los conjuntos, es decir, marcando los otros como no visibles, agregamos un marcador con el nombre de ese conjunto para identificarlo. Hay que repetir este proceso para cada conjunto.

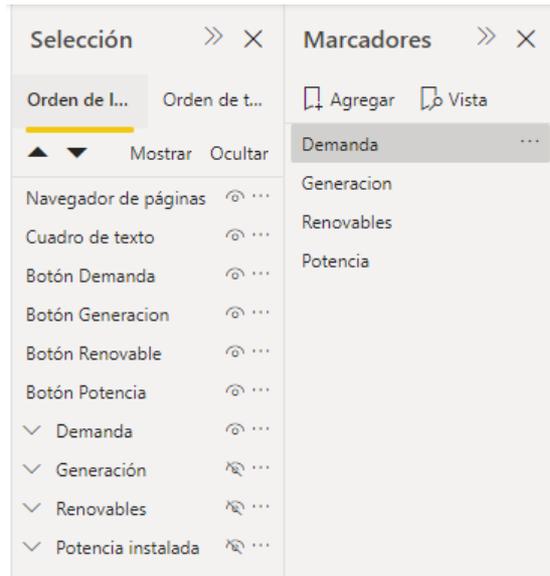


Figura 4-25. Creación de marcadores.

Creados los marcadores, añadimos cuatro botones en blanco. Se activa el apartado ‘Acción’, se selecciona el tipo Marcador y el marcador al cual se quiere acceder al presionar este botón. Una vez configurado, se cambia el texto del botón al marcador al cual hace referencia.

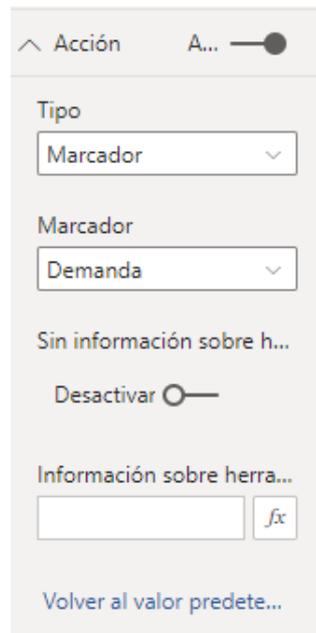


Figura 4-26. Configuración del marcador de Demanda.

REFERENCIAS

- [1] https://es.wikipedia.org/wiki/Inteligencia_empresarial
- [2] <https://www.ree.es/es/apidatos>
- [3] <https://www.ree.es/es/datos/publicaciones/noticias-aldia/nace-redata-tu-espacio-de-datos-estadisticos-del-sistema-electrico-espanol>
- [4] <https://www.ree.es/es/actividades/operacion-del-sistema-electrico/e-sios>
- [5] <https://www.oracle.com/es/autonomous-database/>
- [6] <https://docs.oracle.com/en/cloud/paas/autonomous-database/adbsa/connect-download-wallet.html#GUID-B06202D2-0597-41AA-9481-3B174F75D4B1>
- [7] https://help.hitachivantara.com/Documentation/Pentaho/7.1/0D0/Pentaho_Data_Integration
- [8] <https://pentaho-community.atlassian.net/wiki/spaces/EAI/pages/372704158/Carre+User+Documentation#CarreUserDocumentation-WhatisCarre?>
- [9] <https://holowczak.com/configuring-pentaho-data-integration-to-use-the-oracle-wallet-for-oracle-cloud/>
- [10] <https://powerbi.microsoft.com/es-es/what-is-power-bi/>
- [11] <https://www.oracle.com/database/technologies/dotnet-odacdeploy-downloads.html>
- [12] <https://www.oracle.com/es/database/technologies/instant-client/downloads.html>
- [13] <https://www.oracle.com/a/ocom/docs/database/microsoft-powerbi-connection-adw.pdf>
- [14] <https://blogs.oracle.com/lad-cloud-experts-es/post/microsoft-power-bi-server-oracle-autonomous-database>
- [15] <https://www.oracle.com/es/tools/technologies/whatis-sql-developer.html>
- [16] <https://www.cnmc.es/la-nueva-factura-de-la-luz>
- [17] <https://docs.microsoft.com/es-es/power-bi/visuals/desktop-shape-map>

ANEXO A. DIMENSIONES

A continuación se muestran los valores de las tablas de dimensiones que han sido insertadas mediante SQL. La sentencia para ello es la siguiente:

INSERT INTO TABLA (COLUMNA1, COLUMNA2, ...) VALUES (VALOR1, VALOR2, ...)

Los valores son:

DIM_CCAA	
ID_CCAA	DS_CCAA
0	Todas
4	Andalucía
5	Aragón
6	Cantabria
7	Castilla- La Mancha
8	Castilla y León
9	Cataluña
10	País Vasco
11	Principado de Asturias
12	Comunidad Ceuta
13	Comunidad de Madrid
14	Comunidad de Navarra
15	Comunidad Valenciana
16	Extremadura
17	Galicia
18	Islas Baleares
19	Islas Canarias
20	La Rioja
21	Región de Murcia
8746	Comunidad Melilla
8741	Península

Tabla 8. Contenido de DIM_CCAA.

DIM_INDICADORES

ID_INDICADOR	DS_INDICADOR_CORTA	DS_INDICADOR	DS_ORIGEN
1	Demanda	Demanda de energía eléctrica por CCAA	REData
2	Generación	Generación eléctrica por CCAA y por fuente de energía	REData
3	Renovable/no renovable	Comparativa de generación renovable vs no renovable	REData
4	Potencia instalada	Potencia eléctrica instalada por CCAA	REData
544	Demanda prevista	Demanda de la energía prevista	ESIOS
545	Demanda programada	Demanda de la energía programada	ESIOS
546	Hidráulica	Generación medida en tiempo real del tipo de producción hidráulica	ESIOS
547	Carbón	Generación medida en tiempo real del tipo de producción carbón	ESIOS
549	Nuclear	Generación medida en tiempo real del tipo de producción nuclear	ESIOS
550	Ciclo combinado	Generación medida en tiempo real del tipo de producción ciclo combinado	ESIOS
551	Eólica	Generación medida en tiempo real del tipo de producción eólica	ESIOS
10206	Solar	Generación medida en tiempo real del tipo de producción solar	ESIOS
553	Intercambios	Generación medida en tiempo real producida como consecuencia de intercambios internacionales	ESIOS
554	Enlace balear	Generación medida en tiempo real en el enlace Península-Baleares	ESIOS
1297	Resto generación	Generación medida en tiempo real de los tipos de producción térmica renovable, cogeneración y resto de generación no convencional	ESIOS
1001	PVPC	Precio horario del término de energía que se aplican en la factura eléctrica de los pequeños consumidores	ESIOS
1293	Demanda real	Demanda de la energía real	ESIOS
10044	Portugal	Generación medida en la interconexión con Portugal	ESIOS
10045	Francia	Generación medida en la interconexión con Portugal	ESIOS
10046	Marruecos	Generación medida en la interconexión con Portugal	ESIOS
10047	Andorra	Generación medida en la interconexión con Portugal	ESIOS

Tabla 9. Contenido de DIM_INDICADORES.

DIM_TIPO_FUENTE	
ID_TIPO	DS_TIPO
1	Carbón
2	Ciclo combinado
3	Cogeneración
4	Demanda
5	Eólica
6	Fuel + Gas
7	Generación total
8	Hidráulica
9	Hidroeólica
10	Motores diésel
11	No renovable
12	Nuclear
13	Otras renovables
14	Potencia total
15	Renovable
16	Residuos no renovables
17	Residuos renovables
18	Solar fotovoltaica
19	Solar térmica
20	Turbina de gas
21	Turbina de vapor
22	Turbinación bombeo

Tabla 10. Contenido de DIM_TIPO_FUENTE.

DIM_TRAMOS	
ID_TRAMO	DS_TRAMO
1	Hora valle
2	Hora llana
3	Hora punta

Tabla 11. Contenido de DIM_TRAMOS.

ANEXO B. INDICADORES USADOS

A continuación se describen los indicadores usados en este proyecto de la API de REData:

- Demanda: demanda mensual de energía proveniente de las centrales medido en MWh. Estos datos se muestran divididos en comunidades autónomas y en meses, siendo los últimos datos definitivos, a día de redacción de este documento, de mayo de 2021 y los últimos provisionales los de febrero de 2022.
- Generación: estructura de la generación eléctrica mensual medida en MWh. Estos datos se muestran divididos en comunidades autónomas, en fuentes energéticas y en meses, siendo los últimos datos definitivos, a día de redacción de este documento, de mayo de 2021 y los últimos provisionales los de febrero de 2022.
- Potencia instalada: capacidad de generación de energía medida en MW. Estos datos se muestran divididos en comunidades autónomas, en fuentes energéticas y en meses, siendo los últimos datos definitivos, a día de redacción de este documento, de junio de 2022.
- Renovables/no renovables: evolución de la generación eléctrica a partir de fuentes renovables y no renovables. Estos datos se muestran divididos en comunidades autónomas y en meses, siendo los últimos datos definitivos, a día de redacción de este documento, de mayo de 2021 y los últimos provisionales los de febrero de 2022.

Además de estos, se hacen peticiones a la API de e-sios para obtener datos de los siguientes indicadores:

- Demandas: se obtienen las demandas prevista, programada y real del sistema peninsular y del sistema no peninsular con valores agrupados por horas. La prevista y la programada se publican en e-sios diariamente a las 0:00 horas con la información ese día y a las 13:30 horas con las tres primeras horas del día siguiente, mientras que la real se publica cada 10 minutos con la información de los 10 minutos anteriores. La demanda prevista es una estimación de la curva de demanda para ese día determinado, la programada es un modelado a partir de la prevista para asegurar una generación que se aproxime lo más posible a esta, y la real es la final, la que se ha medido para ese periodo. Los valores de los indicadores son, dentro del sistema peninsular, para la demanda prevista el 544, para la demanda programada el 545, y para la demanda real el 1293. Para el sistema no peninsular, los valores son el 1740 para la demanda real, el 1741 para la demanda programada y el 1742 para la demanda prevista.
- Generación: generación medida en tiempo real de la energía según la fuente de origen con valores agrupados por horas. Estos datos se publican cada 10 minutos con la información de las tres últimas horas del día anterior hasta la hora actual del este día. El valor del indicador es el 546 para la producción hidráulica, 547 para la producción con carbón, 549 para la producción nuclear, 550 para la producción con ciclo combinado, 551 para la producción eólica, 553 para los intercambios internacionales, 554 para el intercambio en el enlace Península-Balear, 10206 para la producción solar y 1297 para el resto de generación, el cual incluye térmica renovable, cogeneración y resto de generación no convencional.
- PVPC: este es el precio horario que se aplica a los consumidores que tengan una potencia contratada inferior a 15kW y que estén acogidos al Precio Voluntario para el Pequeño Consumidor, donde no se incluyen impuestos, divididos por zona a la que aplica. Estos datos se publican en la API de e-sios diariamente a las 20h con los datos del día siguiente. Su número de indicador es el 1001.
- Intercambios internacionales: proporciona las medidas de generación en las interconexiones con los países colindantes en ambos sentidos. Se publican diariamente cada vez que llegan nuevas medidas a partir del siguiente mes. Los valores de los indicadores son el 10044 para el intercambio con Portugal, 10045 para el intercambio con Francia, 10046 para el intercambio con Marruecos y el 10047 para el intercambio con Andorra.

ANEXO C. VISTAS DE BASE DE DATOS

Para facilitar la creación del cuadro de mando y, además, no aumentar el almacenamiento ocupado en la base de datos con tablas, se ha optado por la creación de vistas para obtener únicamente los datos necesarios para los elementos del cuadro de mando. Se procede a describir estas vistas:

- **V_PVPC_ACTUAL:** la función de esta vista es obtener el valor actual del PVPC o, si no está cargado, el de la última hora que hay en la tabla. Para ello, primero se selecciona de la tabla TH_INDICADORES el dato que tenga el día y la fecha de hoy, y por otro lado se selecciona el dato que tenga mayor ID_FECHA, todo ello de cada localización. Se hace un left join entre la consulta con la máxima fecha y la de la fecha actual uniéndola por el ID_LOCALIZACION. Si para esa localización no está el dato de la hora y fecha actual, se muestra el registro del máximo ID_FECHA y su máximo correspondiente de ID_HORA. Una vez escogidos el ID_FECHA e ID_HORA, se seleccionan de la tabla de hechos los registros correspondientes a estos identificadores para, posteriormente, unirlos a las dimensiones y obtener los datos literales de las columnas. A continuación, se muestra la sentencia de creación de la vista:

```
CREATE OR REPLACE VIEW V_PVPC_ACTUAL (D_FECHA_HORA, DS_CCAA, N_VALOR,
DS_INDICADOR_CORTA) AS

SELECT TO_DATE(CONCAT(TO_CHAR(F.D_FECHA, 'DD/MM/YYYY'), CONCAT('
', H.DS_HORA)), 'DD/MM/YYYY HH24:MI') AS D_FECHA_HORA, DS_CCAA, N_VALOR,
DS_INDICADOR_CORTA FROM TH_INDICADORES TH JOIN

(SELECT CASE WHEN ACT.ID_FECHA_ACT IS NULL THEN M.MAX_FECHA ELSE
ACT.ID_FECHA_ACT END AS ID_FECHA,

CASE WHEN ACT.ID_HORA_ACT IS NULL THEN M.MAX_HORA ELSE ACT.ID_HORA_ACT
END AS ID_HORA, M.ID_LOCALIZACION FROM

(SELECT MAX(ID_FECHA) AS MAX_FECHA, MAX(ID_HORA) AS MAX_HORA,
ID_LOCALIZACION FROM TH_INDICADORES WHERE ID_INDICADOR=1001 GROUP BY
ID_INDICADOR, ID_LOCALIZACION) M

LEFT JOIN (SELECT ID_FECHA AS ID_FECHA_ACT, ID_HORA AS ID_HORA_ACT,
ID_LOCALIZACION FROM TH_INDICADORES WHERE
TO_CHAR(SYSDATE, 'YYYYMMDD')=ID_FECHA AND TO_CHAR(SYSDATE, 'HH24')=ID_HORA-
2 AND ID_INDICADOR=1001) ACT

ON M.ID_LOCALIZACION=ACT.ID_LOCALIZACION) FH ON TH.ID_FECHA=FH.ID_FECHA
AND TH.ID_HORA=FH.ID_HORA AND TH.ID_LOCALIZACION=FH.ID_LOCALIZACION AND
ID_INDICADOR=1001

JOIN DIM_FECHA F ON F.ID_FECHA=TH.ID_FECHA

JOIN DIM_HORAS H ON H.ID_HORA=TH.ID_HORA

JOIN DIM_CCAA C ON C.ID_CCAA=TH.ID_LOCALIZACION

JOIN DIM_INDICADORES IND ON TH.ID_INDICADOR=IND.ID_INDICADOR;
```

- **V_PORCENTAJES_GENERACIÓN:** esta vista muestra el porcentaje de generación eléctrica que ha tenido cada fuente en una hora. Para este cálculo, primero se obtiene el total de energía generada en esa hora con todas las fuentes. Con este dato, se puede hacer la división de lo que ha generado cada fuente entre el total y multiplicarlo por 100, de forma que obtenemos el porcentaje de este. La query de creación es la siguiente:

```

CREATE OR REPLACE VIEW V_PORCENTAJES_GENERACION (ID_INDICADOR,
DS_INDICADOR_CORTA, D_FECHA_HORA, D_FECHA, PORCENTAJE, N_VALOR)
SELECT SUM(PORCENTAJE), D_FECHA_HORA FROM (SELECT TH.ID_INDICADOR,
DS_INDICADOR_CORTA,
TO_DATE (CONCAT (TO_CHAR (F.D_FECHA, 'DD/MM/YYYY'), CONCAT (' ', H.DS_HORA)),
'DD/MM/YYYY HH24:MI') AS D_FECHA_HORA,
F.D_FECHA, ROUND (N_VALOR/TOTAL_HORA*100,3) AS PORCENTAJE, N_VALOR FROM
TH_INDICADORES TH
JOIN (SELECT ID_HORA, ID_FECHA, SUM (N_VALOR) AS TOTAL_HORA FROM
TH_INDICADORES
WHERE ID_INDICADOR IN (546,547,549,550,551,10206,553,554,1297)
GROUP BY ID_HORA, ID_FECHA ORDER BY ID_FECHA, ID_HORA) TOTAL
ON TH.ID_FECHA=TOTAL.ID_FECHA AND TH.ID_HORA=TOTAL.ID_HORA
JOIN DIM_FECHA F ON TH.ID_FECHA=F.ID_FECHA
JOIN DIM_HORAS H ON TH.ID_HORA=H.ID_HORA
JOIN DIM_INDICADORES I ON TH.ID_INDICADOR=I.ID_INDICADOR
WHERE TH.ID_INDICADOR IN (546,547,549,550,551,10206,553,554,1297)) GROUP BY
D_FECHA_HORA;

```

- **V_INDICADOR_1001:** con esta vista se muestran los literales de los datos de la tabla de hechos, pero filtrando por el indicador 1001, que es el PVPC. La consulta es la siguiente:

```

CREATE OR REPLACE VIEW V_INDICADOR_1001 (DS_INDICADOR, N_VALOR, DS_CCAA,
D_FECHA, DS_DIA_SEMANA, DS_MES, ID_HORA, DS_HORA, DS_TRAMO, D_FECHA_HORA,
F_CREACION) AS
SELECT IND.DS_INDICADOR_CORTA AS DS_INDICADOR, TH.N_VALOR, C.DS_CCAA,
F.D_FECHA, F.DS_DIA_SEMANA, F.DS_MES, H.ID_HORA, H.DS_HORA, T.DS_TRAMO,
TO_DATE (CONCAT (TO_CHAR (F.D_FECHA, 'DD/MM/YYYY'), CONCAT (' ', H.DS_HORA)),
'DD/MM/YYYY HH24:MI') AS D_FECHA_HORA, F_CREACION FROM TH_INDICADORES TH
JOIN DIM_INDICADORES IND ON TH.ID_INDICADOR=IND.ID_INDICADOR
JOIN DIM_FECHA F ON TH.ID_FECHA=F.ID_FECHA
JOIN DIM_HORAS H ON TH.ID_HORA=H.ID_HORA
JOIN DIM_CCAA C ON TH.ID_LOCALIZACION=C.ID_CCAA
JOIN DIM_TRAMOS T ON TH.ID_TRAMO=T.ID_TRAMO
WHERE TH.ID_INDICADOR=1001;

```

- **V_DEMANDA_SP:** esta vista muestra los literales de los datos de TH_INDICADORES, filtrando por los indicadores de demanda del sistema peninsular, cuyos indicadores son 544, 545 y 1293, siendo así la query:

```

CREATE OR REPLACE VIEW V_DEMANDA_SP (DS_INDICADOR_CORTA, N_VALOR, D_FECHA,
DS_DIA_SEMANA, DS_MES, ID_HORA, DS_HORA, DS_TRAMO, D_FECHA_HORA,
F_CREACION) AS

SELECT D.DS_INDICADOR_CORTA, TH.N_VALOR, F.D_FECHA,
F.DS_DIA_SEMANA, F.DS_MES, H.ID_HORA, H.DS_HORA, T.DS_TRAMO,
TO_DATE(CONCAT(TO_CHAR(F.D_FECHA,'DD/MM/YYYY'), CONCAT(' ',H.DS_HORA)),
'DD/MM/YYYY HH24:MI') AS D_FECHA_HORA, F_CREACION FROM TH_INDICADORES TH
JOIN DIM_INDICADORES D ON TH.ID_INDICADOR=D.ID_INDICADOR
JOIN DIM_FECHA F ON TH.ID_FECHA=F.ID_FECHA
JOIN DIM_HORAS H ON TH.ID_HORA=H.ID_HORA
JOIN DIM_TRAMOS T ON TH.ID_TRAMO=T.ID_TRAMO WHERE TH.ID_INDICADOR IN
(544,545,1293);

```

- **V_DEMANDA_SNP:** al igual que la vista anterior, esta selecciona los datos de los indicadores de demanda, pero del sistema no peninsular, cuyos identificadores son 1740, 1741 y 1742. La sentencia de creación queda así:

```

CREATE OR REPLACE VIEW V_DEMANDA_SNP (DS_INDICADOR_CORTA, N_VALOR, D_FECHA,
DS_CCAA, DS_DIA_SEMANA, DS_MES, ID_HORA, DS_HORA, DS_TRAMO, D_FECHA_HORA,
F_CREACION) AS

SELECT D.DS_INDICADOR_CORTA, TH.N_VALOR, F.D_FECHA, C.DS_CCAA,
F.DS_DIA_SEMANA, F.DS_MES, H.ID_HORA, H.DS_HORA, T.DS_TRAMO,
TO_DATE(CONCAT(TO_CHAR(F.D_FECHA,'DD/MM/YYYY'), CONCAT(' ',H.DS_HORA)),
'DD/MM/YYYY HH24:MI') AS D_FECHA_HORA, F_CREACION FROM TH_INDICADORES TH
JOIN DIM_INDICADORES D ON TH.ID_INDICADOR=D.ID_INDICADOR
JOIN DIM_FECHA F ON TH.ID_FECHA=F.ID_FECHA
JOIN DIM_HORAS H ON TH.ID_HORA=H.ID_HORA
JOIN DIM_CCAA C ON C.ID_CCAA=TH.ID_LOCALIZACION
JOIN DIM_TRAMOS T ON TH.ID_TRAMO=T.ID_TRAMO WHERE TH.ID_INDICADOR IN
(1740,1741,1742);

```

- **V_INTERCAMBIOS:** esta vista selecciona de la tabla de hechos los registros que pertenecen a los intercambios energéticos en las interconexiones internacionales, que son los identificadores 10044, 10045, 10046 y 10047, por lo que la query de creación es:

```

CREATE OR REPLACE VIEW V_INTERCAMBIOS (DS_INDICADOR_CORTA, N_VALOR,
D_FECHA, DS_DIA_SEMANA, DS_MES, ID_HORA, DS_HORA, DS_TRAMO, D_FECHA_HORA,
F_CREACION) AS

SELECT D.DS_INDICADOR_CORTA, TH.N_VALOR, F.D_FECHA,
F.DS_DIA_SEMANA, F.DS_MES, H.ID_HORA, H.DS_HORA, T.DS_TRAMO,
TO_DATE(CONCAT(TO_CHAR(F.D_FECHA,'DD/MM/YYYY'), CONCAT(' ',H.DS_HORA)),
'DD/MM/YYYY HH24:MI') AS D_FECHA_HORA, F_CREACION FROM TH_INDICADORES TH
JOIN DIM_INDICADORES D ON TH.ID_INDICADOR=D.ID_INDICADOR

```

```

JOIN DIM_FECHA F ON TH.ID_FECHA=F.ID_FECHA
JOIN DIM_HORAS H ON TH.ID_HORA=H.ID_HORA
JOIN DIM_CCAA C ON C.ID_CCAA=TH.ID_LOCALIZACION
JOIN DIM_TRAMOS T ON TH.ID_TRAMO=T.ID_TRAMO WHERE TH.ID_INDICADOR IN
(10044,10045,10046,10047);

```

- **V_INDICADOR_1:** las siguientes vistas son las correspondientes a REData y estas no tienen en la API un identificador asociado, por lo que en DIM_INDICADORES se les ha asociado el 1, 2, 3 y 4. Esta en concreto selecciona de la tabla de hechos el indicador 1, que es el correspondiente a la demanda energética. Además, se ha usado la cláusula CASE WHEN para mapear DS_CCAA de Navarra, ya que tiene que tener el mismo valor que en Bing Maps para que Power BI lo detecte:

```

CREATE OR REPLACE VIEW V_INDICADOR_1 (DS_INDICADOR, N_VALOR, DS_CCAA,
D_FECHA, DS_FECHA, F_CREACION) AS
SELECT IND.DS_INDICADOR_CORTA AS DS_INDICADOR, TH.N_VALOR, CASE WHEN
C.DS_CCAA='Comunidad de Navarra' THEN 'Comunidad Foral de Navarra' ELSE
C.DS_CCAA END AS DS_CCAA,
F.D_FECHA, CONCAT(F.DS_MES,CONCAT(' ',F.N_ANYO)) AS DS_FECHA,F_CREACION
FROM TH_INDICADORES TH
JOIN DIM_INDICADORES IND ON TH.ID_INDICADOR=IND.ID_INDICADOR
JOIN DIM_FECHA F ON TH.ID_FECHA=F.ID_FECHA
JOIN DIM_CCAA C ON TH.ID_LOCALIZACION=C.ID_CCAA
WHERE TH.ID_INDICADOR=1
ORDER BY F.D_FECHA DESC;

```

- **V_INDICADOR_2:** esta vista es similar a la anterior, con la diferencia que filtra por el ID_INDICADOR=2, que es el asociado a la generación, pero con la diferencia de que se añade el DS_TIPO para poder mostrar la fuente de generación eléctrica, quedando así la sentencia:

```

CREATE OR REPLACE VIEW V_INDICADOR_2 (DS_INDICADOR, N_VALOR, DS_CCAA,
D_FECHA, DS_FECHA, DS_TIPO, F_CREACION) AS
SELECT IND.DS_INDICADOR_CORTA AS DS_INDICADOR, TH.N_VALOR, CASE WHEN
C.DS_CCAA='Comunidad de Navarra' THEN 'Comunidad Foral de Navarra' ELSE
C.DS_CCAA END AS DS_CCAA,
F.D_FECHA, CONCAT(F.DS_MES,CONCAT(' ',F.N_ANYO)) AS DS_FECHA, TF.DS_TIPO,
F_CREACION FROM TH_INDICADORES TH
JOIN DIM_INDICADORES IND ON TH.ID_INDICADOR=IND.ID_INDICADOR
JOIN DIM_FECHA F ON TH.ID_FECHA=F.ID_FECHA
JOIN DIM_CCAA C ON TH.ID_LOCALIZACION=C.ID_CCAA
JOIN DIM_TIPO_FUENTE TF ON TH.ID_TIPO=TF.ID_TIPO
WHERE TH.ID_INDICADOR=2
ORDER BY F.D_FECHA DESC;

```

- **V_INDICADOR_3:** el indicador 3 es el asociado a la generación de energía a partir de fuentes renovables vs. no renovables. En esta vista se han hecho cálculos, seleccionando por una parte los datos de las fuentes renovables y por otra parte las no renovables, con respecto a los porcentajes de cada tipo de fuente además de añadir el valor de la energía generada. La composición de la sentencia es la siguiente:

```

CREATE OR REPLACE VIEW V_INDICADOR_3 (DS_INDICADOR, N_RENOVABLE,
N_NO_RENOVABLE, PORCENTAJE_RENOVABLE, PORCENTAJE_NO_RENOVABLE, DS_CCAA,
D_FECHA, DS_FECHA, F_CREACION) AS

SELECT IND.DS_INDICADOR_CORTA AS DS_INDICADOR, R.N_RENOVABLE,
NR.N_NO_RENOVABLE, ROUND(100*N_RENOVABLE/(N_RENOVABLE+N_NO_RENOVABLE),2) AS
PORCENTAJE_RENOVABLE,
ROUND(100*N_NO_RENOVABLE/(N_RENOVABLE+N_NO_RENOVABLE),2) AS
PORCENTAJE_NO_RENOVABLE,

CASE WHEN C.DS_CCAA='Comunidad de Navarra' THEN 'Comunidad Foral de
Navarra' ELSE C.DS_CCAA END AS DS_CCAA, F.D_FECHA,

CONCAT(F.DS_MES,CONCAT(' ',F.N_ANYO)) AS DS_FECHA, F_CREACION FROM

(SELECT DISTINCT N_VALOR AS N_RENOVABLE, ID_INDICADOR, ID_FECHA,
ID_LOCALIZACION, F_CREACION FROM TH_INDICADORES TH

JOIN DIM_TIPO_FUENTE TF ON TH.ID_TIPO=TF.ID_TIPO WHERE ID_INDICADOR=3 AND
DS_TIPO='RENOVABLE') R

JOIN

(SELECT DISTINCT N_VALOR AS N_NO_RENOVABLE, ID_FECHA, ID_LOCALIZACION FROM
TH_INDICADORES TH

JOIN DIM_TIPO_FUENTE TF ON TH.ID_TIPO=TF.ID_TIPO WHERE ID_INDICADOR=3 AND
DS_TIPO='NO RENOVABLE') NR ON R.ID_FECHA=NR.ID_FECHA AND
R.ID_LOCALIZACION=NR.ID_LOCALIZACION

JOIN DIM_INDICADORES IND ON R.ID_INDICADOR=IND.ID_INDICADOR

JOIN DIM_FECHA F ON R.ID_FECHA=F.ID_FECHA

JOIN DIM_CCAA C ON R.ID_LOCALIZACION=C.ID_CCAA

ORDER BY F.D_FECHA DESC;

```

- **V_INDICADOR_4:** esta vista es similar al V_INDICADOR_2, con la diferencia de que el indicador que se selecciona es el de potencia instalada, cuyo identificador es el 4. Finalmente, la query de creación de la vista será:

```

CREATE OR REPLACE VIEW V_INDICADOR_4 (DS_INDICADOR, N_VALOR, DS_CCAA,
D_FECHA, DS_FECHA, DS_TIPO, F_CREACION) AS

SELECT IND.DS_INDICADOR_CORTA AS DS_INDICADOR, TH.N_VALOR, CASE WHEN
C.DS_CCAA='Comunidad de Navarra' THEN 'Comunidad Foral de Navarra' ELSE
C.DS_CCAA END AS DS_CCAA,

F.D_FECHA, CONCAT(F.DS_MES,CONCAT(' ',F.N_ANYO)) AS DS_FECHA, TF.DS_TIPO,
F_CREACION FROM TH_INDICADORES TH

JOIN DIM_INDICADORES IND ON TH.ID_INDICADOR=IND.ID_INDICADOR

JOIN DIM_FECHA F ON TH.ID_FECHA=F.ID_FECHA

JOIN DIM_CCAA C ON TH.ID_LOCALIZACION=C.ID_CCAA

JOIN DIM_TIPO_FUENTE TF ON TH.ID_TIPO=TF.ID_TIPO

```

```
WHERE TH.ID_INDICADOR=4  
ORDER BY F.D_FECHA DESC;
```