TRABALLO FIN DE GRAO
GRAO EN ENXEÑARÍA INFORMÁTICA
MENCIÓN EN SISTEMAS DE INFORMACIÓN

Diseño e implementación de una solución de Business Intelligence para el análisis y estudio de pruebas atléticas

Estudante: Alexandre Pérez Paredes

Dirección: Susana Ladra González

A Coruña, June de 2023.

A mi familia por darme esta oportunidad y mantenerse siempre a mi lado. Este es un tributo a su paciencia, sacrificio y esfuerzo.

Agradecimientos

Quiero agradecer a todos los compañeros y profesores que compartieron su conocimiento y experiencia conmigo. En especial quiero destacar a mi tutora, cuya guía hábil y precisa me acompañó en cada etapa del proceso permitiéndome esquivar problemas, refinar mis ideas y orientar los esfuerzos para lograr el objetivo.

Esta dedicatoria representa el agradecimiento que siento hacia todos aquellos que, directa o indirectamente, contribuyeron a mi desarrollo como estudiante y como persona.

Resumen

El foco central de este trabajo es la creación de un data warehouse para el análisis de datos de pruebas de atletismo. Para conseguir este objetivo se diseña un sistema complejo que realiza todo el proceso completo desde la obtención de los datos en origen hasta su introducción en el data warehouse. Se integran diferentes fuentes de datos y se aplican diversos procesos ETL con los que garantizar la calidad de los datos, facilitando así su análisis. Una vez completado el data warehouse permitirá realizar consultas personalizadas, generar informes, analizar tendencias y comparar marcas. En resumen, este data warehouse beneficiará a entrenadores, atletas, clubes y federaciones brindándoles una mejor comprensión del rendimiento atlético.

Abstract

The central focus of this work is the creation of a data warehouse for the analysis of track and fields events data. In order to accomplish this goal, a complex system is designed to handle the entire process, starting from data acquisition at the source and concluding with its introduction into the data warehouse. Different data sources are integrated, and various ETL processes are applied to ensure data quality, thus facilitating its analysis. Once the data warehouse is completed, it will allow for personalized queries, reports and dahboards generation, trend analysis, and performance comparisons. In summary, this data warehouse will benefit coaches, athletes, clubs, and federations by providing them with a better understanding of athletic performance.

Palabras clave:

- Almacén de datos
- Proceso ETL
- World Athletics
- Pruebas atléticas
- Análisis de datos

Keywords:

- Data warehouse
- ETL process
- World Athletics
- Track and field events
- Data analysis

Índice general

1 Introducción **1**

1.1 Contexto y motivación . . . 1
1.2 Objetivos . . . 2
1.3 Estructura de la memoria . . . 2

2 Tecnologías y herramientas **4**

2.1 Pentaho Data Integration . . . 4
2.2 PostgreSQL . . . 4
2.3 DBeaver . . . 5
2.4 Python . . . 5
2.5 PHP . . . 5
2.6 Java . . . 6
2.7 Power BI . . . 6
2.8 Excel . . . 6
2.9 Otras herramientas auxiliares . . . 7
2.9.1 Kaggle . . . 7
2.9.2 Overleaf . . . 7
2.9.3 Git . . . 7
2.9.4 Taclia . . . 8
2.9.5 PyCharm . . . 8
2.9.6 PhpStorm . . . 8
2.9.7 Atom . . . 8
2.9.8 Draw.io . . . 8

3 Metodología y planificación **10**

3.1 Metodología de Kimball . . . 10
3.2 Planificación del proyecto . . . 11

4 Análisis previo **13**
4.1 Necesidades analíticas . . . 13
4.2 Fuentes de datos . . . 14
4.2.1 Datos de competiciones . . . 14
4.2.2 Geocodificación . . . 15
4.2.3 Elevación . . . 17
4.2.4 Código país . . . 17
4.2.5 Capitales y continentes . . . 18
4.2.6 Clima . . . 18
4.2.7 Desfase horario . . . 18
4.2.8 Cálculo de puntos húngaros . . . 19
4.2.9 Cálculo de distancias . . . 19

5 Diseño del Data Warehouse **20**
5.1 Modelado conceptual . . . 20
5.1.1 Hecho . . . 22
5.1.2 Dimensiones . . . 22
5.1.3 Métricas . . . 29
5.2 Diseño lógico . . . 29
5.2.1 Implementación física . . . 30

6 Diseño e implementación ETL **32**
6.1 Extracción . . . 32
6.1.1 Datos de competiciones . . . 33
6.1.2 Geocodificación, altitud y desfase horario . . . 33
6.1.3 Código país, capitales y continentes . . . 33
6.1.4 Clima . . . 35
6.1.5 Generación de fechas . . . 35
6.1.6 Cálculo de puntos húngaros . . . 35
6.2 Transformación . . . 36
6.2.1 Validación . . . 37
6.2.2 Atleta . . . 37
6.2.3 Recinto . . . 40
6.2.4 Entorno . . . 41
6.2.5 Competición . . . 44
6.2.6 Disciplina . . . 46
6.2.7 Resultado . . . 47
6.2.8 Clima . . . 48

6.3 Carga 49

7 Explotación **53**
7.1 Atleta 53
7.2 Competición y recinto 55
7.3 Resultados 57

8 Conclusiones **59**

Lista de acrónimos **60**

Glosario **61**

Bibliografía **63**

Índice de figuras

3.1 Ciclo de vida de la metodología Kimball [1] 10

4.1 Tabla con los datos de las competiciones. 15
4.2 Página con los resultados de una competición. 16

5.1 Diseño conceptual del Data Warehouse utilizando la notación DFM 21
5.2 Diseño lógico del Data Warehouse 31

6.1 Transformación validación y edad del atleta. 38
6.2 Transformación género y país del atleta. 39
6.3 Transformación para obtener el código del atleta. 40
6.4 Transformación para dividir el nombre del recinto. 41
6.5 Transformación para dividir el nombre del recinto. 41
6.6 Cálculo de un triángulo en una esfera con la fórmula de Haversine. [2] 42
6.7 Elipsoide de revolución empleado por WGS84 [3] 42
6.8 Transformación para calcular la diferencia de horas y distancia. 43
6.9 Transformación para calcular la diferencia de niveles de altitud. 44
6.10 Transformación de grupos, código y año de competición. 45
6.11 Transformación para obtener los tipos de competición. 45
6.12 Transformación para categorizar los tipos y subtipos de pruebas. 46
6.13 Transformación para corregir las marcas y obtener los datos de días. 47
6.14 Transformación para calcular las diferencias y los días restantes. 48
6.15 Transformación para calcular el rango de viento 49
6.16 Configuración SCD de la dimensión atleta 50
6.17 Inserción de resultados empleando Bulk Load 50
6.18 Inserción de competición con Búsqueda/actualización en combinación 50
6.19 Actualización para simular el cambio 51
6.20 Trabajo con todas las transformaciones y scripts aplicados en el ETL 52

7.1 Informe con los principales datos sobre el atleta 54
7.2 Informe con los principales datos sobre las competiciones y el recinto donde se albergan 56
7.3 Informe que contiene diferentes métricas sobre el resultado 58

Índice de cuadros

3.1 Planificación del proyecto 11
3.2 Seguimiento del proyecto 12
3.3 Tabla de costes 12

Capítulo 1

Introducción

En este documento se expondrá el trabajo realizado para diseñar e implementar una solución de Business Intelligence para el análisis y estudio de pruebas atléticas. Esto proporcionará un sistema completo capaz de obtener los datos desde la fuente de origen y tratarlos hasta obtener diversos informes, cuadros de mando y gráficas con las que poder obtener un conocimiento más profundo y ordenado de los resultados de atletismo.

1.1 Contexto y motivación

Los datos de pruebas atléticas son diversos, y pueden variar desde tiempos de carrera hasta distancias recorridas o la longitud de saltos o lanzamientos. Además, estos datos pueden ser registrados en diferentes formatos y lugares. Por ello, la recopilación y almacenamiento de estos datos en un Data Warehouse permitirá un espacio único donde integrar, consultar y analizar los datos de diferentes fuentes. La causa de la creación de este proyecto es, por tanto, intentar proveer una solución a la incapacidad de encontrar datos analíticos de calidad y fácilmente consultables sobre resultados de pruebas de atletismo. Esta solución tendrá forma de Data Warehouse con pruebas atléticas donde poder almacenar resultados de pruebas tales como los 100 metros lisos o la maratón entre otras.

En definitiva, la creación de este sistema permitirá resolver cuestiones hasta el momento poco accesibles como, por ejemplo, las que a continuación se enumeran:

- En qué zonas geográficas compiten con mayor frecuencia los atletas de un determinado país.
- Conocer los países que producen más atletas.
- La distribución de ciudades donde se realizan competiciones.

Proporcionar la capacidad de acceder a datos históricos permite a entrenadores, atletas y organizadores tomar decisiones más informadas en cuanto al rendimiento, la estrategia y la

planificación de competiciones para maximizar los resultados. Además, el análisis de los datos de desempeño puede revelar patrones y tendencias, lo que puede ser útil para el diseño de programas de entrenamiento más efectivos o al menos tener constancia de bajo qué condiciones el atleta se desempeña mejor.

En definitiva, la creación de un Data Warehouse para pruebas de atletismo proporciona información valiosa que mejora la toma de decisión. Potencialmente, también se podrá mejorar el desempeño de los atletas, provocando una mayor competitividad. Este aumento en el rendimiento podría a la larga atraer a nuevos aficionados y una mayor cantidad de personas se animen a la práctica del deporte. Estos eventuales beneficios son objetivos declarados por la misma World Athletics tanto en su último plan estratégico de crecimiento para los años 2020-2023 como en su plan global para el período 2022-2030. [4]

1.2 Objetivos

El objetivo principal de este trabajo es crear un Data Warehouse en el que almacenar información de las diferentes pruebas para posteriormente realizar visualizaciones e informes con los datos que este contiene. Los objetivos específicos del proyecto son los que siguen:

- Obtener los datos necesarios desde las diferentes fuentes mediante el uso de APIs o Web Scraping, tanto con la herramienta Octoparse como con código Python.
- Diseñar la estructura del Data Warehouse en el que se cargarán los datos.
- Realizar un proceso ETL que a su finalización obtenga el Data Warehouse previamente diseñado cargado con la información pertinente.
- Explotar el Data Warehouse mediante la aplicación de técnicas y herramientas de Business Intelligence para obtener como resultado informes de calidad.

1.3 Estructura de la memoria

En esta sección se detallará brevemente los temas que se tratarán en cada uno de los capítulos de esta memoria.

- Capítulo 1, **Introducción**: se indican las causas para seleccionar esta temática así como la adecuación de la propuesta realizada.
- Capítulo 2, **Tecnologías y herramientas**: se detallan cada una de las tecnologías, softwares y herramientas empleadas en la realización del proyecto.

- Capítulo 3, **Metodología y planificación**: se explica la metodología empleada y los motivos de su selección junto con la planificación y seguimiento final del proyecto.
- Capítulo 4, **Análisis del dominio y requisitos analíticos**: se detalla el proceso inicial para determinar qué características debe poseer el modelo.
- Capítulo 5, **Diseño del Data Warehouse**: se detallan las métricas y atributos seleccionados para estar presentes en el diseño final.
- Capítulo 6, **Diseño e implementación del proceso ETL**: se explica qué es el proceso ETL y los pasos seguidos en cada una de las fases.
- Capítulo 7, **Explotación**: en este capítulo se detallan los diferentes informes creados y la utilidad de sus gráficos y métricas.
- Capítulo 8, **Conclusiones**: en este capítulo final se indican los resultados generales obtenidos sobre el trabajo realizado.

Capítulo 2

Tecnologías y herramientas

En este capítulo se comentarán de forma breve las principales tecnologías y herramientas empleadas en el transcurso de la realización de este trabajo.

2.1 Pentaho Data Integration

Pentaho Data Integration[1], también conocido como PDI o Kettle, es una herramienta de integración y transformación de datos desarrollada por la empresa Hitachi Vantara, radicada en California, Estados Unidos. Este software dispone de una versión 'Community' de carácter gratuito que es la que se empleará en el marco de este proyecto. Concretamente se empleara su Interfaz gráfica de Usuario (GUI) Spoon. En esta interfaz conviven dos tipos de tareas, los trabajos y las transformaciones, donde las transformaciones son los subcomponentes de los diferentes trabajos. Estas tareas se representan mediante notación gráfica, lo que dota de una mayor legibilidad a la herramienta.

2.2 PostgreSQL

PostgreSQL[2] es un sistema de gestión de bases de datos relacional de código abierto y altamente escalable. Utiliza un modelo de datos basado en tablas y ofrece soporte para consultas complejas y transacciones ACID. Es conocido por su capacidad de manejar grandes volúmenes de datos y por su robustez lo que lo convierte en idóneo para este proyecto. Además, proporciona una amplia gama de características avanzadas. Es ampliamente utilizado en aplicaciones empresariales y científicas que requieren un almacenamiento de datos confiable y escalable.

[1] https://www.hitachivantara.com/en-us/products/pentaho-platform/data-integration-analytics.html
[2] https://www.postgresql.org/

2.3 DBeaver

DBeaver[3] es una herramienta de administración de bases de datos multiplataforma que proporciona una versión 'Community' de carácter gratuito. Dispone una interfaz gráfica intuitiva y potente para conectarse y administrar diversas bases de datos, como MySQL, PostgreSQL, Oracle y SQL Server. Ofrece características avanzadas como autocompletado de SQL facilitando el desarrollo y la administración de bases de datos. En general, permite realizar consultas SQL, administrar estructuras de bases de datos, importar y exportar datos, y llevar a cabo tareas de administración como la configuración de índices y ajuste de rendimiento además de soportar conexión simultánea a múltiples bases de datos. Es una herramienta ampliamente extendida en el sector del desarrollo y administración de bases de datos.

2.4 Python

Python[4] es un lenguaje de programación interpretado, de alto nivel y multiparadigma, caracterizado por su sintaxis clara y legible. Utiliza tipado dinámico y ofrece soporte para múltiples paradigmas de programación como programación orientada a objetos o programación funcional. Su biblioteca estándar es amplia y proporciona funcionalidades a tareas comunes, así como una gran cantidad de bibliotecas de terceros que facilitan el desarrollo en diversas áreas, como ciencia de datos, aprendizaje automático o desarrollo web. Su filosofía se basa en la legibilidad del código, fomentando el uso de indentación significativa y evitando la redundancia. En el marco del proyecto se empleará principalmente para la obtención de datos de diferentes APIs.

2.5 PHP

PHP es un lenguaje de programación de código abierto, interpretado y diseñado especialmente para el desarrollo web. Se utiliza principalmente para crear aplicaciones dinámicas y sitios web interactivos. Es un lenguaje de lado servidor, lo que significa que se ejecuta en el servidor web y genera contenido HTML que se envía al navegador del cliente. Es altamente flexible y compatible con diversos sistemas operativos y servidores web, ofreciendo una amplia gama de características, incluyendo una sintaxis similar a la de C, una gran cantidad de funciones integradas y una extensa comunidad que proporciona numerosas bibliotecas y frameworks para acelerar el desarrollo. En el marco del proyecto se usa única y exclusivamente para obtener resultados de una librería no existente en otros lenguajes.

[3] https://dbeaver.io/
[4] https://www.python.org/

2.6 Java

Java[5] es un lenguaje de programación orientado a objetos, de propósito general y altamente versátil. Es conocido por su portabilidad al poder ejecutar desde diferentes lugares en diferentes plataformas sin necesidad de recompilación. Java es un lenguaje de programación de alto nivel que se basa en la sintaxis de C++ pero elimina ciertas características propensas a errores. Ofrece una amplia biblioteca estándar y una gran comunidad de desarrolladores, lo que facilita la creación de aplicaciones robustas y escalables. Java es utilizado en una variedad de dominios, desde el desarrollo de aplicaciones de escritorio y móviles hasta sistemas empresariales y desarrollo de software embebido. Además, Java soporta programación concurrente y ofrece características como la gestión automática de la memoria mediante el recolector de basura. Es reconocido por su enfoque en la seguridad, su arquitectura robusta y su capacidad para manejar grandes proyectos de software. En el proyecto se emplea principalmente para calcular valores dentro de las transformaciones de Pentaho.

2.7 Power BI

Power BI[6] es un software creado por Microsoft y se destaca como una de las principales herramientas de Business Intelligence del mercado. Sus principales funciones son: creación de informes y cuadros de mando interactivos para la visualización de datos, integración de datos de diferentes fuentes y colaboración y compartición. Esto último se realiza gracias a Power BI Service, un servicio en la nube que permite colaborar y publicar informes. Además de Service también tiene otros dos componentes los cuales son: Power BI Desktop y Power BI Mobile, siendo estbe último esencialmente para consumir informes y cuadros de mando mientras que las otras están mas enfocadas al desarrollo de estos. En el marco de este proyecto solo se emplea la versión de escritorio del software.

2.8 Excel

Excel[7] es una aplicación de hoja de cálculo desarrollada por Microsoft para organizar, analizar y manipular datos de manera eficiente. Ofrece una interfaz intuitiva y funciones predefinidas que facilitan la realización de cálculos matemáticas y estadísticas complejas. Además, permite la automatización de tareas mediante el uso de fórmulas y macros. Su capacidad para manipular datos numéricos y textuales lo convierte en una herramienta poderosa para el análisis y la toma de decisiones basada en datos. Adicionalmente, en el marco de este proyecto

[5] https://www.java.com/es/
[6] https://powerbi.microsoft.com/es-es/
[7] https://www.microsoft.com/es-es/microsoft-365/excel

también se emplea la extensión 'xlwings' escrita en código Python y que permite entre otras muchas funcionalidades crear fórmulas de excel que consulten las tablas como si fuera SQL.

2.9 Otras herramientas auxiliares

2.9.1 Kaggle

Kaggle[8] es una plataforma competitiva para científicos de datos a la vez que comunidad tanto para temas de ciencia de datos como para aprendizaje automático. Dentro de la plataforma sus integrantes colaboran, compiten y comparten conocimientos. Esto se produce mediante la publicación de conjuntos de datos, desafíos de aprendizaje automático o herramientas para compartir trabajo y conocimientos. Dentro de la plataforma también destacan sus competiciones patrocinadas, donde los participantes resuelven problemas específicos. Otras posibilidades dentro de la plataforma es la colaboración en proyectos, publicar cuadernos con código y análisis o participar en discusiones en los foros. En el marco de este proyecto se emplea para buscar datasets con datos de deportes en general, y atletismo en particular de los que extraer posibles requisitos, tanto por los campos que poseen como por aquellos de los que carecen.

2.9.2 Overleaf

Overleaf es una plataforma en línea de edición de documentos LaTeX que posee una interfaz intuitiva y permite mostrar el código y su resultado al mismo tiempo. Su uso está enfocado en crear documentos científicos y académicos con facilidad.

2.9.3 Git

Es el sistema de control de versiones empleado en el proyecto. Este repositorio se empleó principalmente para gestionar las diferentes versiones de los scripts. En concreto, se emplea GitHub[9] por estar ampliamente extendida dentro del sector. Un sistema de control de versiones permite esencialmente realizar el seguimiento histórico de los cambios realizados dentro de un documento. Para esto registra los cambios en un repositorio local y los sincroniza con un repositorio remoto.

[8] https://www.kaggle.com/
[9] https://github.com/

2.9.4 Taclia

Taclia[10] es una pequeña web que permite registrar y controlar los turnos que se realizan, permitiendo anotar la temática del turno y llevar la cuenta de las horas diarias, semanales y registra trabajadas.

2.9.5 PyCharm

PyCharm[11] es un Entorno de desarrollo integrado (IDE) completo de la empresa JetBrains para programar en lenguaje Python. Ofrece características y herramientas potentes, incluyendo asistencia inteligente para contribuir a la mejora de la calidad del código escrito. En definitiva, es una solución completa para el desarrollo de aplicaciones Python y tecnologías web relacionadas.

2.9.6 PhpStorm

PhpStorm[12] es un IDE avanzado de la empresa JetBrains para programar en PHP. Ofrece una amplia gama de características y herramientas así como una interfaz atractiva y personalizable que mejoran la productividad en el desarrollo de aplicaciones PHP.

2.9.7 Atom

Es un editor de texto que posee soporte para una gran variedad de archivos y lenguajes siendo tremendamente versátil y útil. Adicionalmente, en el marco del proyecto se emplea para revisar los archivos de datos cuando se producian errores durante el desarrollo de las diferentes transformaciones por no tener límite en cuanto al número de filas que se podían mostrar. Atom[13] es open source y esta desarrollado por la empresa GitHub integrando el control de versiones Git lo cual le dota de una mayor solidez. En general Atom es un editor de texto sencillo pero completo que posee las características necesarias para realizar pequeños códigos o cambios rápidos en ficheros. Lamentablemente, GitHub descontinuó el proyecto recientemente para centrarse en editores en la nube con GitHub Codespaces.

2.9.8 Draw.io

Draw.io[14] es una herramienta versátil para crear diagramas y gráficos. Permite representar la gran mayoría de tipos de diagramas existentes proporcionando soporte para entre otros:

[10] https://www.taclia.com/
[11] https://www.jetbrains.com/es-es/pycharm/
[12] https://www.jetbrains.com/es-es/phpstorm/
[13] https://github.com/atom/atom
[14] https://app.diagrams.net/

- Diagramas UML.
- Diagramas de secuencia.
- Entidad-Relación.

Esto lo permite a través de una interfaz intuitiva y un amplio conjunto de formas y símbolos. Por los motivos comentados y por poseer tanto versión de escritorio como web lo convierten en una herramienta útil y flexible para presentar de forma sencilla y efectiva documentación técnica.

Capítulo 3

Metodología y planificación

3.1 Metodología de Kimball

Para la realización organizada y estructurada del proyecto se opta por seguir un procedimiento conocido para minimizar los errores cometidos. Teniendo esto en mente, se selecciona la conocida y extendida metodología de Kimball [1] por ser la más difundida de entre las existentes para la realización de Data Warehouses. Está principalmente basada en lo que el autor denomina como Ciclo de Vida Dimensional del Negocio (Business Dimensional Lifecycle) representado en la figura 3.1.

Esta metodología está centrada en el negocio y se fundamenta sobre tres principios esenciales enumerados a continuación:

- Enfoque en aportar valor al negocio.
- Estructuración dimensional de los datos entregados.
- Desarrollo iterativo en incrementos manejables en lugar de una estrategia Big Bang.

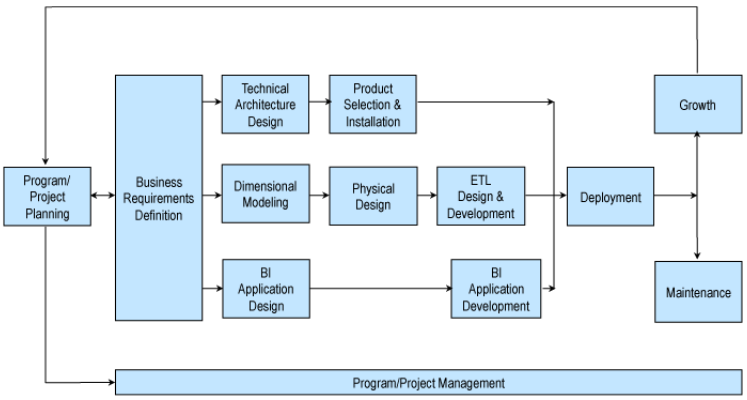

Figura 3.1: Ciclo de vida de la metodología Kimball [1]

A continuación se resumirán algunas de las principales fases del ciclo de vida:

- Planificación del proyecto: en esta fase se definen los diferentes objetivos a conseguir a la finalización del proyecto y se establecen prioridades y asignación de tareas.
- Definición de requisitos del negocio: en esta fase se determinan los requisitos del almacén de datos y cuáles son las necesidades de información que necesitan.
- Modelado dimensional: se diseña a nivel conceptual cómo debe ser el modelo de datos.
- Diseño físico: se transforma el modelado dimensional en la verdadera forma en que se va a insertar en la base de datos.
- Diseño e implementación del proceso Extracción, transformación y carga (ETL): se transforman los datos de entrada en datos listos para ser explotados.
- Explotación: se generan informes y visuales de diferente índole con los datos del Data Warehouse para poder apoyar las diferentes tomas de decisión de la organización.

3.2 Planificación del proyecto

Fase	Período	Horas
Planteamiento del proyecto	02/11/22 - 29/12/22	6
Análisis de requisitos	02/11/22 - 25/02/23	24
Búsqueda de fuentes de datos	02/12/22 - 10/03/23	40
Diseño dimensional DW	25/01/23 - 27/02/23	25
Implementación física del DW	27/02/23 - 10/03/23	15
Diseño e implementación proceso ETL	10/03/23 - 21/04/23	100
Diseño e implementación de informes en Power BI	21/04/23 - 26/05/23	90

Cuadro 3.1: Planificación del proyecto

Adicionalmente a las tablas de planificación 3.1 y seguimiento 3.2 se debe contabilizar el número de horas de dirección para las cuales se obtiene un total de 10 horas. En la tabla 3.3 se muestran los diferentes costes para la planificación, el seguimiento y la dirección.

Una vez calculada la tabla de costes 3.3, se realiza el cálculo del coste total y se obtiene como resultado 9525 €. Esto provoca incurrir en un sobrecoste de 2175 € respecto del presupuesto inicial, o lo que es lo mismo, una desviación del 22,83 % en términos de coste.

Fase	Período	Horas
Planteamiento del proyecto	02/11/22 - 29/12/22	6
Análisis de requisitos	02/11/22 - 25/02/23	20
Búsqueda de fuentes de datos	02/12/22 - 10/03/23	35
Diseño dimensional DW	25/01/23 - 27/02/23	32
Implementación física del DW	27/02/23 - 10/03/23	12
Diseño e implementación proceso ETL	10/03/23 - 21/04/23	252
Diseño e implementación de informes en Power BI	21/04/23 - 26/05/23	10

Cuadro 3.2: Seguimiento del proyecto

	Horas	Coste (€/h)	Presupuesto (€)
Planificación del proyecto	300	25	7500
Seguimiento del proyecto	367	25	9175
Dirección del proyecto	10	35	350

Cuadro 3.3: Tabla de costes

Capítulo 4

Análisis previo

En esta sección se explicará el paso inicial del proceso y uno de los más importantes por ser el pilar sobre el que se asienta la totalidad del sistema. Para poder realizar un buen análisis, se necesita comprender en profundidad las diversas necesidades, objetivos y requerimientos del dominio. En resumen, este estudio tiene como objetivo la alineación de los requisitos del negocio con el posterior diseño e implementación del Data Warehouse.

4.1 Necesidades analíticas

El dominio de este trabajo son las pruebas atléticas y poder observar cómo las diferentes condiciones afectan a los resultados. El principal objetivo de un atleta es mejorar sus marcas primero y ganar competiciones segundo. Para conseguir cumplir sus objetivos el atleta busca de forma constante y meticulosa arañar las pocas milésimas o centímetros que pueden suponer la clasificación para un campeonato, un podio o incluso una victoria. Como consecuencia de esta búsqueda constante de mejora, los atletas, entrenadores, clubes y organizaciones buscan profesionalizar en mayor medida el deporte haciendo un mayor uso de estudios médicos o científicos con los que optimizar el entrenamiento y la técnica para favorecer un incremento progresivo del nivel global. En este contexto, es fundamental poseer un excelso conocimiento propio no basado en sensaciones, sino en fríos datos alejados de sentimientos o sesgos que favorezca una mejora orgánica y constante con la que contribuir entre otros muchos factores a la mejora del atleta.

En este contexto de mejora continua y competitividad donde los atletas cada vez acortan más las diferencias entre ellos se necesita analizar datos y crear informes con estos para apoyar la toma de decisiones basadas en conocimiento facilitando la mejora de resultados.

La definición de requisitos es un paso de especial importancia en proyectos de cualquier índole, por tanto se le debe prestar siempre un especial cuidado para que sea lo más preciso posible. En combinación con el estudio de negocio de la sección 4.1 constituye la base del

proyecto que debe ser lo más consistente y sólida posible para evitar la mayor cantidad de errores posible. Una vez tenemos este conocimiento, se deben plantear los diferentes requisitos que, en este caso se plantearán como diversas consultas a realizar:

- ¿Influye el continente en la generación de atletas?
- ¿Influye el país en el promedio de puntos obtenido por un atleta?
- ¿Influye la diferencia de altitud en el promedio de puntos de un atleta?
- ¿Influye la edad en la posición en la que se finaliza la prueba?
- ¿Influye la velocidad del viento en la posición resultado de la prueba?
- ¿Influye la ciudad en el número de atletas que participan?
- ¿Influye el país en el rango de edad de los atletas?
- ¿Influye el continente en el número de competiciones?
- ¿Cómo influye la altitud en cada disciplina?

Para comparar los resultados no sólo se emplea la propia marca, sino que se calculan los Puntos húngaros. Estos son de gran utilidad debido a que permiten comparar las marcas de diferente disciplinas. Pudiendo comparar por ejemplo si el récord mundial de 100 metros lisos de Usain Bolt es más o menos extraordinario que el récord mundial de Yulimar Rojas en triple salto femenino.

4.2 Fuentes de datos

En esta sección se comentarán las necesidades de datos y las fuentes valoradas para su extracción.

4.2.1 Datos de competiciones

Para poder llevar a cabo el sistema el primer componente indispensable es poseer datos con una cierta calidad y completitud sobre los que poder trabajar. Con este objetivo en mente, se buscan fuentes de información que contengan datos sobre pruebas de atletismo. En un primer instante, se pensó en la posibilidad de emplear un dataset ya existente sobre el cual obtener datos adicionales y realizar las transformaciones pertinentes. Esta idea se descartó rápidamente debido a la práctica nula existencia de datasets públicos con información suficiente de competiciones y pruebas de atletismo. El más relevante de entre los consultados,

atendiendo a motivos de consistencia y tamaño, es encontrado en la plataforma Kaggle[1], siendo descartado por contener tan solo datos sobre los medallistas en Juegos Olímpicos (JJOO). Una vez descartada esta vía por considerarse que proporciona una cantidad insuficiente de datos, se opta por aplicar técnicas de Web Scraping sobre la página de la World Athletics[2]. En primer lugar con el software Octoparse para posteriormente decantarse por el uso de código en lenguaje Python. Se emplea código Python por permitir una mayor flexibilidad para un uso avanzado además de no tener limitación en la cantidad de datos extraída, ya que Octoparse limita a 10.000 filas de datos por extracción en su versión gratuita y no permite automatización.

Como se acaba de comentar, la principal página empleada como fuente de datos será la propia de la World Athletics por ser la de mayor fiabilidad y completitud, conteniendo datos de competiciones desde 1934 y actualizándose prácticamente a diario. Para poder extraer los datos que esta contiene se debe realizar previamente un análisis de la estructura html de la página para poder leerla de forma adecuada y capturar los datos necesarios. Esencialmente la página se divide en una principal mostrada en la figura 4.1, donde se capturan los datos de cada competición exceptuando la información de contacto, y, otra página concreta de la competición son sus resultados que se muestra en la figura 4.2 a la cual se accede mediante el enlace asociado de cada competición en la página principal.

DATE ⇅	NAME ⇅	⇅	VENUE ⇅	COUNTRY ⇅	CAT. ⇅	DISCIPLINE ⇅	COMPETITION GROUP ⇅	
23 APR 2023	Campionato Regionale di Staffette	ⓘ	Stadio Narciso Soldan, Conegliano (ITA)	ITA	F	Stadium Outdoor		Result
23 APR 2023	TCS London Marathon		London (GBR)	GBR	GW	Road Running	World Athletics Label Road Races – Platinum	Result
23 APR 2023	Argentinian Marathon Championships		Buenos Aires (ARG)	ARG	B	Road Running		Result
23 APR 2023	Zürich Half Marathon & 10km	ⓘ	Zürich (SUI)	SUI	F	Road Running		Result
23 APR 2023	Competitie U20/U18 Poule 31 R1 [2023]	ⓘ	Pim Mulier Sportpark, Haarlem (NED)	NED	F	Stadium Outdoor		Result
23 APR 2023	Belgrade Marathon	ⓘ	Beograd (SRB)	SRB	E	Road Running	National Senior Marathon Championships, World Athletics Label Road Races – Label	Result
23 APR 2023	Kaunas Marathon		Kaunas (LTU)	LTU	E	Road Running	World Athletics Label Road Races – Label	Result
23 APR 2023	Zürich Marathon	ⓘ	Zürich (SUI)	SUI	F	Road Running	National Senior Marathon Championships	Result

Figura 4.1: Tabla con los datos de las competiciones.

4.2.2 Geocodificación

Debido a la necesidad de representar de forma adecuada tanto atletas como recintos en el mapamundi se deben obtener sus coordenadas.Para la obtención de estas a partir del nombre de una ubicación se debe realizar un proceso de Geocodificación. Para esto se debe seleccionar

[1] https://www.kaggle.com/datasets/jayrav13/olympic-track-field-results
[2] https://worldathletics.org/competition/calendar-results?hideCompetitionsWithNoResults=true

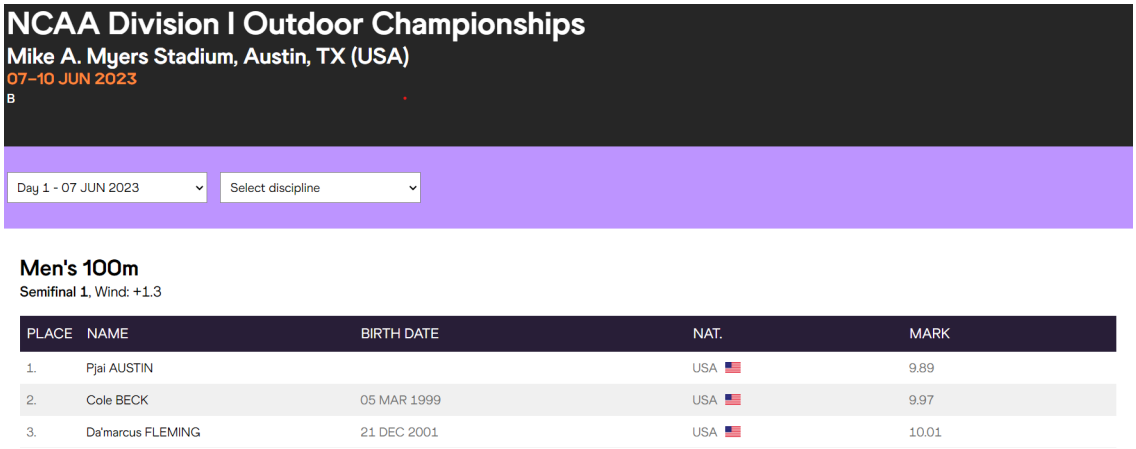

Figura 4.2: Página con los resultados de una competición.

algún servicio que ofrezca las características necesarias y la mayor sencillez posible. A continuación se enumeran algunos de los servicios candidatos y sus principales características.

- **Geopy**: librería Python[3] que permite el uso de los principales servicios de Geocodificación como son Google Geocoding API, Bing Maps API o Nominatim entre otros.
- **Nominatim**: librería que hace uso de OpenStreetMap[4]. Permite crear una instancia en local o usar como servicio mediante una API.
- **Open-meteo Geocoding API**: API[5] Open Source gratuita para uso no comercial que proporciona una interfaz simple donde indicar el nombre de la ciudad a buscar, el número de resultados y el formato del resultado. Permite un máximo de 10.000 llamadas diarias.
- **API Ninjas Geocoding**: API propietaria que proporciona una interfaz sencilla donde indicar la ciudad y el código del país al que pertenece. Proporciona una API-key gratuita para realizar hasta 50.000 peticiones mensuales[6].

Se selecciona API Ninjas por su mayor adecuación con el formato en el que se dispone de los datos además de permitir la posibilidad de precisar el país donde se quiere buscar. Como segunda opción se considera a la solución perteneciente a Open-meteo, pero por motivos de seguridad y diversificación en cuanto al número de proveedores se descarta. De esta forma, se añade una mayor seguridad al tener diversos orígenes y en caso de fallo de uno de estos servicios el resto de fuentes sí que estén aseguradas.

[3] https://pypi.org/project/geopy/
[4] https://nominatim.org/
[5] https://open-meteo.com/en/docs/geocoding-api
[6] https://api-ninjas.com/api/geocoding

Para aquellos lugares que no encuentra de forma automática la API se deben resolver de forma manual para lo que se emplea Google Maps[7].

4.2.3 Elevación

Con el objetivo de representar la altitud del país del atleta y la altitud a la que se sitúa el recinto se busca una fuente con la que conseguirla para unas coordenadas dadas. Para ello al igual que en el caso de la Geocodificación de la sección 6.1.2 se busca una librería o API con la que lograrlo. Algunas de las posibilidades consultadas son:

- **Elevation**: permite descargar, cachear y acceder[8] al dataset SRTM 30m Global 1 arc second V003[9] elaborado por la NASA.
- **Open-meteo Elevation API**: API Open Source gratuita para uso no comercial que proporciona una interfaz simple donde indicar las coordenadas de latitud y longitud. Permite un máximo de 10.000 llamadas diarias[10].

Se selecciona la solución de Open-meteo por ser más sencilla y no requerir la necesidad de descargar bases de datos. Esto nos sirve para poder obtener los datos de altitud de cualquier lugar del globo mediante sus coordenadas, pero para el caso de la altitud media de un país no es válida. Para la obtención de este dato escasean las fuentes, especialmente aquellas de gran fiabilidad. Teniendo esto en cuenta y la no necesidad de una gran precisión en el cálculo por su clasificación posterior en rangos, finalmente se opta por emplear un artículo de la Wikipedia sobre altitudes medias de los diferentes países[11].

Por último, para aquellas coordenadas que no encuentra de forma automática la API y se resuelven de forma manual, se emplea topographic-map[12].

4.2.4 Código país

Este código se necesita para poder obtener toda la información del país ya que la World Athletics lo indica con un código corto. Para su traducción se combinan los códigos de la ISO 3166-1 alfa-3[13] y los del COI[14]. Estos se obtienen mediante Web Scraping con la herramienta Octoparse en los respectivos enlaces.

[7] https://www.google.es/maps/
[8] https://pypi.org/project/elevation/
[9] https://doi.org/10.5067/MEaSUREs/SRTM/SRTMGL1N.003
[10] https://open-meteo.com/en/docs/elevation-api
[11] https://en.wikipedia.org/wiki/List$_o f_c ountries_b y_a verage_e levation$
[12] https://es-es.topographic-map.com/
[13] https://www.iso.org/iso-3166-country-codes.html
[14] https://odf.olympictech.org/2022-Beijing/codes/HTML/og$_c c/NOC.htm$

4.2.5 Capitales y continentes

La capital se necesita por ser el punto de referencia para su país en términos de distancias y desfase horario. Por otro lado el continente se obtiene para realizar un nivel de agregación superior al país. Para este último, se asigna el de la federación de área bajo la que compite, por ejemplo en el caso de España sería la European Athletics y por tanto se asigna Europa. En este caso, coinciden continente geográfico y deportivo, pero no tendría por qué suceder. Para obtener este listado y una vez comprobado que ni en las páginas de todas las federaciones ni en la de la World Athletics se indica directamente los países que componen la federación, aunque en algunas se indica en mapas dificultando la extracción. Se opta por emplear una fuente con una menor fiabilidad pero igual para todas las federaciones, extrayéndose por tanto de la Wikipedia[15].

En cuanto a las capitales, tampoco se encontró una fuente unificada donde encontrar las capitales de todos los países del mundo por lo que nuevamente se acaba recurriendo a la Wikipedia[16].

4.2.6 Clima

La siguiente fuente necesaria es una amplia base de datos históricos acerca del clima para comprobar cómo afecta a los diferentes resultados. La única alternativa gratuita o con un plan gratuito usable encontrada es la API proporcionada por Open-meteo[17], la cual contiene información desde 1940 y se actualiza con unos 6 a 7 días de retraso lo cual no es una gran limitación para este caso de estudio. La principal alternativa encontrada es Weather API OpenWeatherMap[18] pero tan solo contiene datos desde 1979 y además requiere crear una cuenta y un límite de 1000 llamadas diarias. Otra alternativa sería el uso de Historical Weather API de la empresa Tomorrow.io[19], pero tan solo posee los datos históricos desde el año 2000 y es de pago.

4.2.7 Desfase horario

Para esta métrica es necesario obtener una librería o servicio que dadas unas coordenadas, sea capaz de calcular el huso horario correspondiente. Para esto se encuentran las siguientes dos alternativas:

- **Timezonefinder**: librería Python que dadas las coordenadas de latitud y longitud de

[15] https://es.wikipedia.org/wiki/World$_{A}thletics$
[16] https://en.wikipedia.org/wiki/List$_{o}f_{n}ational_{c}apitals$
[17] https://open-meteo.com/en/docs/historical-weather-api
[18] https://openweathermap.org/api
[19] https://docs.tomorrow.io/reference/historical-overview

un lugar calcula su huso horario correspondiente[20].

- **TimeZoneDB**: API gratuita previo registro que proporciona una interfaz con varias configuraciones. Para la que nos interesa se deben indicar las coordenadas de latitud y longitud para obtener su huso[21].

Se elige timezonefinder porque no esta expuesto a fallos externos y por tanto la convierte en más fiable.

4.2.8 Cálculo de puntos húngaros

Para el cálculo de los puntos húngaros se necesitan 3 elementos: la marca del atleta, la disciplina y el género. Sabiendo esto se buscan alternativas que realicen este cálculo, encontrándose las siguientes.

- **athleticscalculator**: página web con una calculadora donde obtener los puntos. Para su uso sería necesario crear un código que simule la interacción[22].
- **scoringtablescalculator**: página web[23] que permite el cálculo de los puntos húngaros en su versión de 2017. Podría emplearse de forma similar a una API por el uso de parámetros de consulta dentro de la URL.
- **Librería de PHP**: librería del lenguaje PHP empleada por la Federación Letona para realizar los cálculos dentro de su página web. [5]
- **Generación del algoritmo con los pdf's de la World Athletics**: consistiría en transformar los pdf's de la World Athletics a excel y crear un algoritmo que para cada prueba y género busque el par marca - puntos. Sería bastante complejo, debido a la opacidad de la World Athletics y el formato de los pdf's. [6]

Se selecciona la librería PHP por ser la única con respaldo oficial.

4.2.9 Cálculo de distancias

Una vez tenemos los datos de altitudes, coordenadas y husos horarios se debe planificar la obtención de sus diferencias. Para los casos de las altitudes u horarios no hay problema porque es una sencilla resta. Pero en el caso de las coordenadas debemos calcular la distancia entre ellas empleando un algoritmo de cálculo de distancias sobre una superficie esférica. Esta última transformación se explica en detalle en la sección 6.2.4.

[20] https://pypi.org/project/timezonefinder/
[21] https://timezonedb.com/references/get-time-zone
[22] https://athleticscalculator.com/rns/ranking
[23] https://scoringtablescalculator.com/calculator/

Capítulo 5

Diseño del Data Warehouse

5.1 Modelado conceptual

El modelado conceptual es el proceso mediante el cual obtener una intuitiva representación estructurada de los datos de una organización para una sencilla comprensión aun sin tener conocimientos avanzados en la materia. En términos técnicos, el modelado dimensional se basa en el diseño y creación de esquemas que capturen y relacionen aquellas dimensiones y métricas relevantes para la organización. Las dimensiones representan características, mientras que las métricas son los valores cuantitativos que se pueden analizar o agregar en función de las diferentes dimensiones.

Este enfoque permite una fácil navegación y consulta, pudiendo sus usuarios realizar análisis multidimensionales y aplicar filtros o desgloses en función de diferentes categorías. Además, es un un modelado flexible y escalable que permite la evolución a la vez que la organización crece o modifica sus requisitos.

En este caso se selecciona para su representación el Dimensional Fact Model (DFM), un diagrama enfocado en la captura de métricas relevantes agrupadas en una tabla que contiene los hechos. Adicionalmente, cada hecho tiene asociadas diversas cualidades agrupadas en diferentes dimensiones. A continuación se muestra la figura 5.1 con el diseño final del DFM realizado.

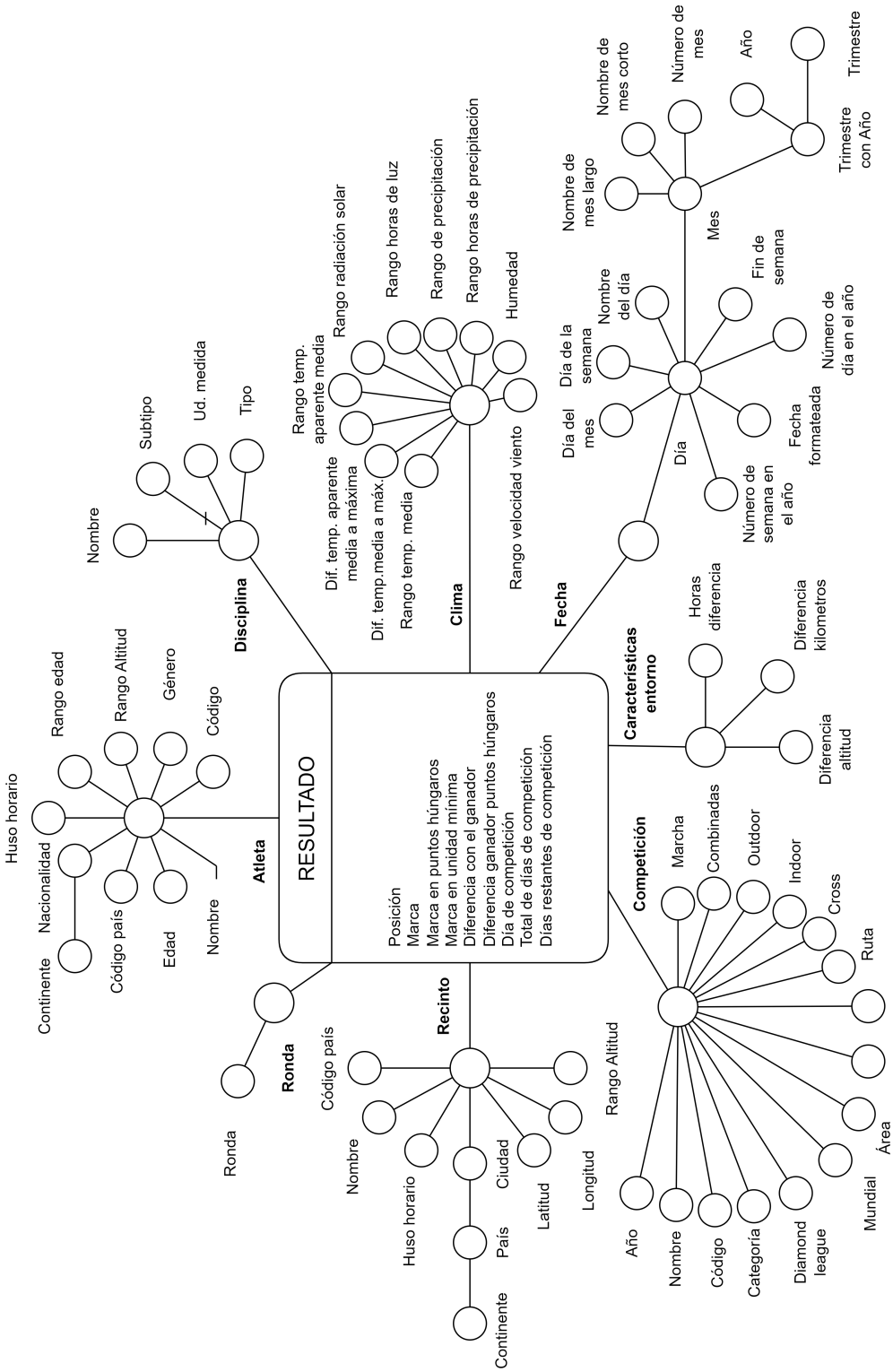

Figura 5.1: Diseño conceptual del Data Warehouse utilizando la notación DFM

5.1.1 Hecho

El hecho serán los resultados de un atleta en una competición y ronda, sin ningún tipo de agrupación para representar la granularidad más fina posible con los datos disponibles. Por tanto, no se realiza ningún tipo de agregación previa.

5.1.2 Dimensiones

Atleta

Dimensión que permite conocer con detalle al atleta que participa en la prueba medida. A continuación se indican los diferentes campos de esta dimensión:

- **Código**: código asignado por la federación internacional a cada atleta, sirve para identificarlo unívocamente. Es la clave natural empleada para la realización del control de cambios. En caso de no tener un código asignado por la World Athletics, se le asigna un código de igual formato discrecionalmente, con la salvedad de que para poder diferenciar estos tienen como primera cifra un 9.
- **Nombre**: atributo descriptivo que almacena el nombre completo del atleta.
- **Edad**: años transcurridos desde el año de nacimiento del atleta, permite agrupar y comparar con otros atletas.
- **Nacionalidad**: nombre del país del atleta, indica la procedencia del atleta y permite agrupar a los atletas.
- **Continente**: nombre del continente, indica la procedencia del atleta y permite agrupar a los atletas.
- **Código país**: nombre del país indicado con el código de 3 letras empleado por la World Athletics, esta codificación se obtiene de hibridación entre las codificaciones del Comite Olímpico Internacional (COI) y los correspondientes al estándar internacional ISO 3166-1 alfa-3. Adicionalmente, existen ciertos códigos que no se encuentran en ninguno de los dos organismos y son propios de la World Athletics.
- **Huso horario**: permite agrupar por el horario empleado los diferentes atletas. En el caso de los países que emplean varios, se emplea como referencia el de la capital del país ya que por norma general suele ser el empleado por la mayoría de la población. Por ejemplo en el caso de España, se asigna el UTC+2 a la totalidad del territorio a pesar de que no es el único en la totalidad del mismo.

- **Rango edad**: rango de edad del atleta, se categoriza en los siguientes rangos: <20, [20, 24], [25, 29], [30, 35], [36, 40], 40<.
- **Rango altitud**: se toma como referencia la altitud media del país al que pertenece. Se categoriza en cuatro niveles. Baja <200, media [200, 600), alta [600, 1000), extrema 1000<.
- **Género**: sexo del atleta, hombre en caso de ser masculino y mujer en caso de ser femenino.

En esta dimensión es la única en la que es necesario realizar un control de cambios por tener atributos que cambian en el tiempo. Para esto, se emplea Slowly Changing Dimension (SCD) que es una política de actualización que controla cómo reflejar los diferentes cambios producidos en un almacén de datos a lo largo del tiempo. Tiene varios tipos, pero los empleados en la realización de este trabajo son el de tipo 1 y el de tipo 2, que cuando se emplean en combinación son conocidos como SCD híbrido. [7]

- SCD-1: es la solución más drástica y consiste en directamente sobrescribir el valor antiguo, eliminando cualquier rastro de que este existiera.
- SCD-2: añade una nueva fila con el nuevo valor del dato, esto se conoce como versionado.

En este caso se aplicará SCD-2 para el campo edad y rango edad mientras que para el campo nombre y el campo género se empleará SCD-1. El resto de campos simplemente se actualizan en la versión activa. En el caso del campo rango edad se aplica el control de tipo SCD-2 implícitamente por ir ligado al campo edad.

Recinto

En esta dimensión se agrupan los diferentes atributos que tiene cada uno de los recintos donde se realizan las diferentes competiciones del calendario de la World Athletics.

- **Nombre**: en combinación con la ciudad permite identificar un recinto de forma sencilla.
- **Ciudad**: indica el espacio geográfico donde se realiza la competición. Junto con el nombre permite identificar el recinto.
- **País**: indica el espacio geográfico delimitado por fronteras donde se realiza la competición.
- **Continente**: indica el espacio geográfico bajo el que se agrupan los diferentes países del mundo. Junto con país y ciudad constituye una jerarquía.

- **Latitud**: medida angular determina la ubicación relativa de un punto en la superficie de la Tierra en sentido norte o sur respecto del ecuador. Varía entre 0° y 90° y se almacena para poder realizar visualizaciones con mapas.
- **Longitud**: medida angular determina la ubicación relativa de un punto en la superficie de la Tierra en sentido este oeste respecto al meridiano de referencia. Varía entre 0° y 180° y se almacena para poder realizar visualizaciones con mapas.
- **Código país**: código corto del país donde se realiza la prueba.
- **Rango altitud**: se categoriza de igual forma que para el atleta y permite agrupar en función de la altura a la que se realiza la prueba. Para esta medición se toma la altitud de la ciudad donde se lleva a cabo la competición.
- **Huso horario**: permite agrupar por la zona horaria empleada en los diferentes recintos. A diferencia del mismo campo en la dimensión atleta, se emplea el huso concreto de la ubicación del estadio. Por ejemplo en el caso de un recinto situado en Canarias, se asigna el huso UTC+1 a pesar de no ser el empleado en la capital del país al que pertenece, en este caso Madrid, donde se emplea UTC+2.

Competición

En esta dimensión se agrupan los atributos que caracterizan el torneo.

- **Año**: junto con el nombre de la prueba nos permite diferenciar la prueba. Se separa del nombre para poder agrupar de una forma sencilla las diferentes ediciones de una competición.
- **Nombre**: denominación por la que se conoce la competición y que permite reconocerlo de forma sencilla.
- **Marcha**: campo de carácter booleano que indica si se acogen pruebas de marcha dentro de la competición, los valores aceptados son: Verdadero o Falso.
- **Cross**: campo de carácter booleano que indica si se acogen pruebas de cross dentro de la competición, los valores aceptados son: Verdadero o Falso.
- **Ruta**: campo de carácter booleano que indica si se acogen pruebas en carretera dentro de la competición, los valores aceptados son: Verdadero o Falso.
- **Aire libre**: campo de carácter booleano que indica si la competición alberga pruebas realizadas en una pista larga de 400m, los valores aceptados son: Verdadero o Falso.

- **Pista cubierta**: campo de carácter booleano que indica si la competición alberga pruebas realizadas en una pista corta de 200m, los valores aceptados son: Verdadero o Falso.
- **Categoría**: categorización realizada a partir de 2018 por la World Athletics para identificar el nivel de una competición. Los posibles valores son: OW, DF, GW, GL, A, B, C, D, E y F en este orden de calidad. Para las competiciones previas a 2018 también se realiza esta categorización y se indica con los mismos valores.
- **Nacional**: campo de carácter booleano que indica si la competición es un campeonato nacional. Admite dos valores: Verdadero o Falso.
- **Área**: campo de carácter booleano que indica si la competición es un campeonato de área[1]. Admite dos valores: Verdadero o Falso.
- **Mundial**: campo de carácter booleano que indica si la competición es un campeonato mundial. Admite dos valores: Verdadero o Falso.
- **Diamond League**: campo de carácter booleano que indica si la competición es una prueba de la Diamond League [2]. Admite dos valores: Verdadero o Falso.
- **Juegos Olímpicos**: campo de carácter booleano que indica si la competición esta dentro de los Juegos Olímpicos (JJOO). Admite dos valores: Verdadero o Falso.

Aunque de forma general no coinciden los campos relativos al tipo de competición, una competición no puede pertenecer a dos a la vez. La única excepción es en las ocasiones en que un campeonato nacional se recoge a partir del resultado del campeonato de área, por lo que podrían llegar a coincidir. Esta extraña casuística suele darse especialmente en pruebas de gran fondo como medias maratones o maratones. En caso de ocurrir, el campeonato nacional no será considerado.

Características entorno

Dimensión con diversos indicadores relacionados con las diferencias existentes entre el origen del atleta y el lugar de realización de la prueba.

- **Diferencia altitud**: acepta el rengo de valores discretos comprendido en [-3, 3], representando así la diferencia de rango entre el correspondiente al atleta y el del recinto. Se

[1] Competiciones de carácter continental organizadas por el organismo regional, destacar que las federaciones no se agrupan exactamente por continente, de forma que se producen casos como que Israel compita en campeonatos Europeos.

[2] Competición organizada por la World Athletics desde 2010 que agrupa las 15 competiciones más importantes del mundo. Los atletas suman puntos durante la temporada hasta llegar a la final en Zúrich, Suiza donde la puntuación es doble.

toma como referencia el del atleta, es decir, -1 se produce, por ejemplo, con altitud del atleta 'Media' y altitud del recinto 'Baja'.

- **Diferencia kilómetros**: diferencia en kilómetros desde el lugar de celebración del evento y el punto de referencia tomado para el atleta, la capital de su país. Se categoriza en los siguientes rangos: Cercana <750, Media [750, 3000), Lejana [3000, 7500] y Extrema 7500<.
- **Horas diferencia**: desfase horario entre el lugar de origen del atleta y el lugar de celebración de la prueba. De forma general, un desfase negativo implica atrasar la hora de sueño, mientras que uno positivo provoca adelantarla.

Fecha

Dimensión que contiene diferentes datos y formatos sobre la fecha en que se realiza la prueba.

- **Fecha**: momento de realización de la prueba con granularidad diaria. Se indica en formato yyyy-mm-dd.
- **Fecha formateada**: campo fecha con formato yyyy MMM dd. Donde MMM es la abreviatura del nombre del mes en inglés. Por ejemplo, para el mes de enero se indica JAN.
- **Mes con año**: campo que indica el mes y año de realización con formato mm-yyyy.
- **Año**: campo que indica el año de realización de la prueba con formato yyyy.
- **Nombre del día**: campo que indica el nombre del día de la semana en inglés.
- **Nombre del mes largo**: campo que indica el nombre completo del mes en inglés.
- **Nombre del mes corto**: campo que indica el nombre abreviado del mes en inglés.
- **Día de la semana**: campo que indica el número del día de la semana, toma valores en [1, 7].
- **Día del mes**: campo que indica el número de día en el mes, los valores que puede albergar varían en el rango [1, 31].
- **Número de semana en el año**: campo que indica el número de la semana en el año, los posibles valores que puede tomar comprenden el intervalo [0, 52].
- **Número de día en el año**: campo que indica el número de día en el año, toma valores en el intervalo [1, 366].

- **Fin de semana**: indicador booleano con el que conocer si la fecha de competición coincidió en Sábado o Domingo.
- **Número de mes**: número de mes en el año, toma valores en [1, 12].
- **Trimestre con año**: campo que indica el trimestre y el año con formato q-yyyy
- **Trimestre**: número de trimestre en el año, puede tomar los valores en [1, 4].

Clima

Dimensión con los datos del estado del tiempo en el momento de realización de la prueba.

- **Rango de temperatura media**: clasificación en rangos de la temperatura media en grados centígrados registrada en las coordenadas del estadio el día de la prueba. Se clasifica en cuatro rangos: Frío para temperaturas menores o iguales a 10 ℃, templado para temperaturas en el rango (10, 20], cálido para temperaturas en el intervalo (20, 30] y calor para aquellos registros de temperatura que excedan los 30 ℃.
- **Diferencia de temperatura media a máxima**: proporciona el número de grados de diferencia entre el máximo diario alcanzado y el valor medio de temperatura.
- **Rango de temperatura aparente media**: se categoriza de igual forma que la temperatura media e indica la sensación térmica mediante una combinación del nivel de humedad y la temperatura real del aire.
- **Diferencia de temperatura aparente media a máxima**: indica el número de grados de diferencia entre el máximo diario alcanzado y el valor medio de temperatura aparente.
- **Rango de radiación solar**: indica la potencia con la que se mostró el sol en MJ/m^2 [3]. Se categoriza en cuatro niveles o rangos con los siguientes valores: baja, moderada, alta y muy alta para valores hasta 10, 20, 30 y a partir de 30 respectivamente.
- **Rango de horas de luz**: atributo que indica la duración del día, en conjunto con el resto de datos permite tener un conocimiento profundo sobre el tipo de día bajo el que se realizó la prueba. Se categoriza en tres rangos, corto para menos de 9 horas de luz, estándar para el rango comprendido de 9 a 15 horas y por último largo para aquellos días con más de 15 horas de luz.

[3] Megajulios por metro cuadrado

- **Rango de precipitación**: volumen de agua acumulado durante el día de la medición en la ciudad donde se realiza. Se mide en mm que es equivalente a la unidad l/m²[4]. Se establecen cuatro rangos: Bajo para mediciones hasta 10 mm, moderado para registros en el rango [10, 30), alto para mediciones en el intervalo [30, 50] y muy alto para medidas superiores a los 50 mm.
- **Rango de horas de precipitación**: indica el número de horas que llueve permitiendo una aproximación del tipo de día que hizo. Nuevamente, se clasifica en cuatro rangos: Sin lluvia para los valores en el intervalo [0, 1], llovizna para el intervalo (1, 3], lluvioso'para el intervalo (3, 6] y tormenta para todos los valores superiores a 6.
- **Humedad**: promedio del porcentaje de humedad registrado a lo largo del día de realización de la prueba. Se categoriza en cuatro rangos diferentes inspirados en el nombrado de la clasificación climática de Thornthwaite [8] para el índice de humedad. Estos rangos son continuos y poseen igual tamaño, sus nombres son, en orden de menor a mayor porcentaje: árido, seco, húmedo y perhúmedo.
- **Rango de velocidad del viento**: velocidad del viento en m/s. Solo se mide en las pruebas al aire libre con distancia menor a 400 metros y los saltos horizontales. Se categoriza en hasta siete rangos diferentes medidos en m/s: Viento desfavorable extremo para velocidades menores a -2.0, Viento muy desfavorable para velocidades en el intervalo [-2,0, -1,2), viento en contra para el intervalo [-1,2, -0,4), baja velocidad para el intervalo [-0,4, 0,5), viento a favor para el intervalo [0,5, 1,3), viento muy favorable para valores en el intervalo [1,3, 2,0] y viento ilegal para velocidades que sobrepasen los 2 m/s.

Ronda

- **Ronda**: fase de la competición en la que se realiza la prueba.

Disciplina

Dimensión que contiene la información relativa a la prueba que realiza el atleta.

- **Nombre**: denominación con la que se identifica la prueba.
- **Unidades de medida**: magnitud en la que se mide el resultado de la prueba. Se almacena para poder representar correctamente.
- **Tipo**: categorización realizada para agrupar pruebas similares entre sí. Los grupos son: saltos, lanzamientos, velocidad, medio fondo, fondo, combinadas, cross y marcha.

[4] Litros por metro cuadrado

- **Subtipo**: categorización realizada para agrupar con mayor precisión las diferentes pruebas en función de su grado de similitud. En dependencia del tipo pueden tomar unos valores u otros, pero los diferentes valores posibles son: Ruta, pista, relevos, individual, horizontal, vertical, vallas, lisos y obstáculos.

5.1.3 Métricas

En esta sección se indican de forma detallada cuales son las métricas analizadas para el hecho medido.

- **Posición**: orden en el que el atleta llega a la meta.
- **Marca**: tiempo, longitud o altura con la que finaliza el atleta la prueba.
- **Marca en puntos húngaros**: transformación que asigna un valor numérico a los registros de las pruebas que se celebran en los grandes campeonatos. Esta asignación numérica permite comparar los resultados de las diferentes pruebas atléticas entre sí.
- **Diferencia con el ganador**: distancia en tiempo, longitud o altura respecto al atleta que cruza la meta en primer lugar. En caso de ser el ganador o registrar la misma marca se indica una diferencia de 0.
- **Diferencia con el ganador en puntos húngaros**: déficit de puntos respecto al atleta que finaliza la prueba en primer lugar. En caso de obtener igual puntuación o ser el ganador se indica 0.
- **Día de competición**: diferencia en días desde el inicio de la competición, es un número entre 1 y el total de días de la competición.
- **Total de días de competición**: duración en días de la competición.
- **Días restantes de competición**: diferencia en días hasta el final de la competición. Se obtiene de la resta entre el número de días de la competición y el día de competición.

En lo que respecta a la aditividad todas las métricas son no aditivas al carecer de sentido la suma por alguna de las diferentes dimensiones. Por ejemplo, no tiene sentido sumar los valores de las diferentes posiciones en las que finaliza una prueba un atleta.

5.2 Diseño lógico

Debido al uso del gestor de base de datos relacional PosgreSQL, se opta por seleccionar el modelo relacional para realizar el diseño lógico del Data Warehouse. El modelado relacional es una técnica empleada para organizar y estructurar información en una base de datos

mediante tablas relacionadas. Este enfoque surgió en la década de 1970 en los laboratorios de la empresa IBM en California, Estados Unidos, de la mano de Edgar Frank Codd en respuesta a la necesidad de un método más sencillo y eficiente para almacenar y acceder a grandes volúmenes de datos. Desde su creación y debido a su sencillez, este modelo se impuso sobre sus competidores convirtiéndose en un estándar de facto en lo referente al diseño de bases de datos. [9]

Para la realización de este modelo lógico se selecciona el eficiente pero sencillo **esquema en estrella**, compuesto de un elemento central conocido como tabla de hechos y diversas tablas auxiliares conocidas como dimensiones. Este sencillo esquema permite un análisis eficiente y una sencilla comprensión de los datos almacenados. En este caso se elige este esquema sobre otros de mayor complejidad como el esquema en copo de nieve donde las dimensiones tienen a su vez subdimensiones eliminando una mayor redundancia pero haciendo las consultas menos eficientes al tener estas que realizar más uniones entre tablas. En lo referente al diagrama como tal, los principales cambios respecto del diseño conceptual son:

- Degeneración en la tabla de hechos de la dimensión Ronda debido a que tan solo posee un atributo, esto se conoce de forma genérica como Dimensión Degenerada.
- Adición de un identificador subrogado a todas las dimensiones exceptuando la fecha. Este conjunto de identificadores conforma la clave de la tabla resultado.
- En el caso de la fecha el identificador es la propia fecha en formato yyyymmdd. El cual es habitualmente empleado para este fin por poseer la cualidad de corresponderse con el orden natural de las fechas, facilitando la búsqueda e indexación.
- También se añaden columnas de auditoría a causa del SCD-2. Estas columnas son, fecha de inicio y fin de validez junto con el número de versión.

El esquema resultante puede ser consultado en la figura 5.2.

5.2.1 Implementación física

Una vez terminado el modelado relacional es momento de convertir el esquema en un script para poder crear las diferentes tablas en la base de datos. Para esto se emplea una base de datos PostgreSQL y una conexión a la misma desde la herramienta de administración de base de datos DBeaver. El trabajo de transformar el modelo lógico en un script consiste esencialmente en la selección de tipos y restricciones diferentes a las claves primarias o foráneas ya definidas. En el momento que se finaliza la creación del script este puede ser ejecutado en el editor de la herramienta DBeaver resultando en la creación de 8 tablas.

Adicionalmente a los atributos comentados y reflejados en la figura se duplican algunos de ellos por motivos de internacionalización.

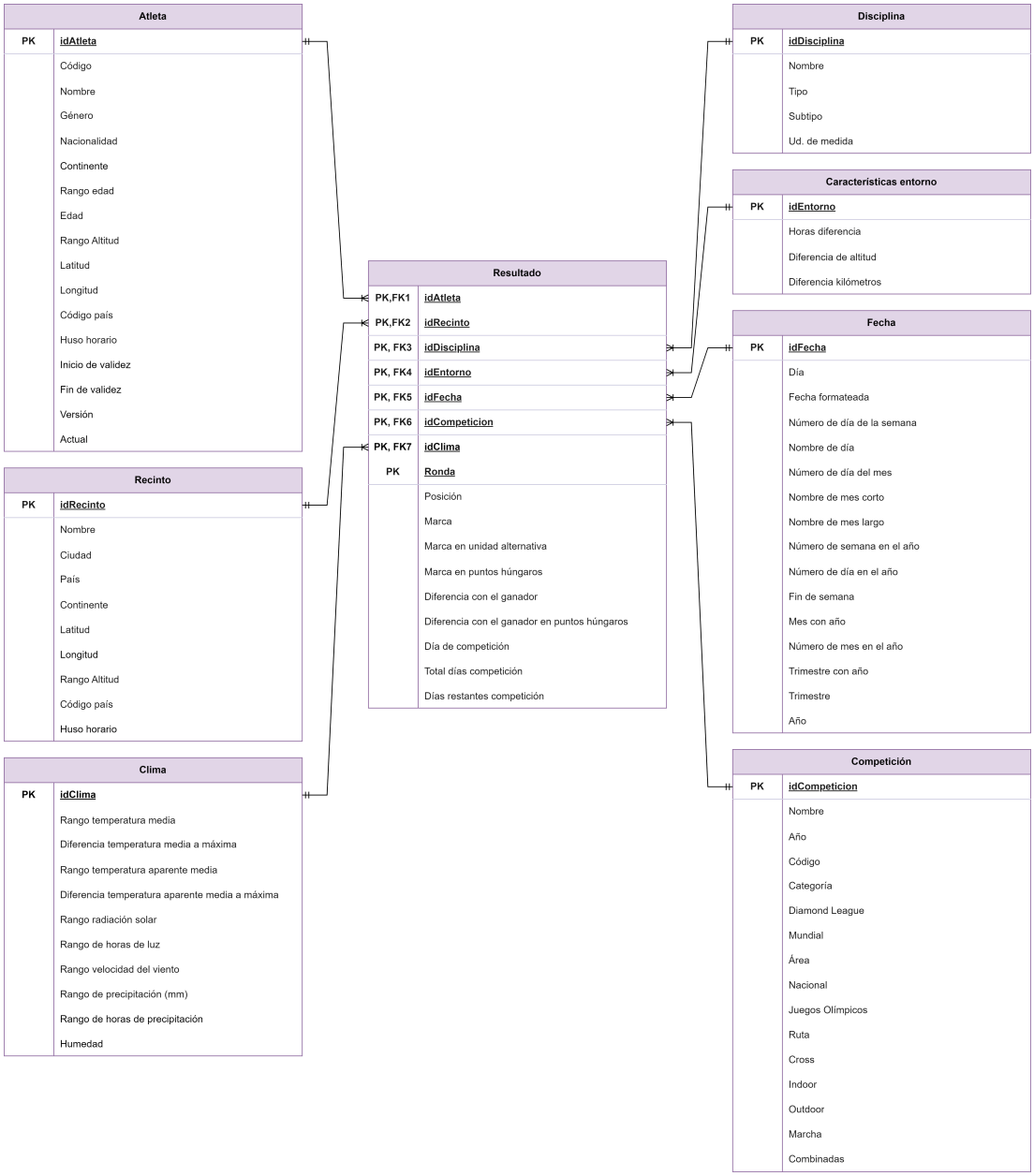

Figura 5.2: Diseño lógico del Data Warehouse

Capítulo 6

Diseño e implementación ETL

El proceso Extracción, transformación y carga (ETL) es un proceso consistente en la integración de diversas fuentes de datos junto con su correspondiente limpieza y transformación para lograr una calidad, cohesión y coherencia que permita al sistema resultado ser considerado como confiable y óptimo para proporcionar soporte a los diferentes empleados encargados de realizar la toma de decisión. Esto posibilita reducir el margen de error y contribuye de forma clave en el porvenir de la empresa. Para poder realizar esta tarea el proceso se divide en tres fases fundamentales:

- **Extracción**: etapa inicial en donde se capturan y obtienen los diferentes datos que conformarán el sistema. Estos datos pueden obtenerse de diversas fuentes como bases de datos ya existentes, servicios web, ficheros de texto u hojas de cálculo entre otros. En el caso de este proyecto, la fuente de datos principal será la página de la federación internacional de atletismo World Athletics como se comentó en la sección 4.1.
- **Transformación**: segunda etapa del proceso consistente en la realización de diferentes modificaciones y manipulaciones en los datos para conseguir la normalización y eliminación de repetidos. Además, en esta fase también se realiza la aplicación de reglas de negocio y obtención de nuevos datos mediante la aplicación de diversos cálculos.
- **Carga**: en esta última fase del proceso se realiza el mapeo de valores a la base de datos o sistema correspondiente y se vuelcan los datos sobre él.

6.1 Extracción

En esta sección se detallarán los diferentes pasos realizados para extraer los datos de las fuentes seleccionadas en la fase de análisis realizada en el capítulo 4.

6.1.1 Datos de competiciones

Para poder extraer los datos que la página de la World Athletics se genera un código Python que esencialmente itera a través de la tabla mostrada en la figura 4.1 obteniendo sus campos. La página mezcla scroll con paginación por lo que al llegar al final de la página, salta a la siguiente para continuar con la extracción.

Una vez obtiene estos campos, accede al enlace del campo 'Result' e itera sobre las diferentes tablas con resultados de cada disciplina y ronda, como se ve en la figura 4.2. En caso de que la competición tenga varios días, se itera sobre el menú desplegable. En estas tablas es importante tener en cuenta que en caso de que la prueba sea un Salto horizontal tendrá una columna adicional con el valor medido de viento en el momento del salto. Otro elemento a tener en cuenta es la categoría, que si bien aparece en el listado de competiciones, en caso de que la competición sea previa a 2018 simplemente se indica como 'Pre2018' en el listado, mientras, en la vista en detalle de la competición sí se indica su categorización y por tanto se recoge.

6.1.2 Geocodificación, altitud y desfase horario

Los datos de estas dimensiones y sus respectivas llamadas a las APIs seleccionadas en la sección 4.2 se tratan en un mismo script Python donde se aplican políticas de reintento, especialmente en las llamadas a la API de geocodificación. Como salida de este script se genera un fichero actualizado de todas las ciudades. En el caso concreto de los países al emplearse la altitud media se modifica el script para que no calcule este parámetro. Como salida se obtiene el fichero de entrada actualizado con los campos de latitud, longitud y desfase horario.

6.1.3 Código país, capitales y continentes

Una vez se extraen los datos necesarios de cada una de las fuentes se deben aplicar algunas correcciones. En primer lugar se revisa la lista de códigos de país obtenida y se compara con los códigos empleados por la World Athletics. Una vez tenemos la lista de códigos sin asociación se realizan algunos ajustes manuales para obtener la traducción al código empleado. Este es el caso de Norfolk Island. La cual tiene el código Organización Internacional de Estandarización (ISO) NFK que no aparece en la lista del COI y la World Athletics asigna el código NFI.

Para la extracción de los países por federación tan sólo se realiza una corrección, cambiando el nombre de Birmania por el de Myanmar, nombre actual del país.

En cuanto al caso de las capitales es importante destacar que para algunos países se indican varias, conflicto que se resuelve manualmente aplicando las siguientes reglas mostradas en orden de prioridad.

- **Distingue oficial**: la que se indican como oficial es la seleccionada.

- **Territorio ocupado**: en caso de no coincidir con la del país ocupante se mantiene la oficial, en caso contrario se asigna la considerada De Facto.
- **No distingue oficial**: se selecciona en función de varios parámetros subjetivos:
  - Existe una más reconocida que la otra.
  - Existe una más centrada que la otra.

A continuación se muestran algunos ejemplos de decisión:

- **Países Bajos**: se selecciona Ámsterdam sobre La Haya por ser la oficial.
- **Sáhara Occidental y Palestina**: en el caso del Sáhara se selecciona El Aaiún por encima de Tirifati por ser la oficial aún estando ocupada por Marruecos. Mientras que en el caso de Palestina, se selecciona Ramallah sin ser la oficial por motivos de representación en mapas ya que Jerusalem es a su vez capital de Israel.
- **Bolivia y República Checa**: se seleccionan La Paz y Praga sobre Sucre y Brno por ser más reconocibles.
- **Sudáfrica**: se selecciona Bloemfontein por encima de Ciudad del Cabo y Pretoria por estar más centrada en el territorio.
- **Santa Helena, Ascensión y Tristán de Acuña**: para estas dependencias británicas de ultramar, se selecciona Jamestown en Santa Helena como capital por un doble motivo, es la ciudad más poblada y, aunque ahora comparten igual estatus, hasta 2009 se conocía a estos territorios como Santa Helena y dependencias, siendo esta la isla más importante.

Adicionalmente, existen cinco categorías para atletas que no compiten representando a su país.

- **Internacional, INT**: se produce en las pruebas de relevos cuando los atletas tienen diferentes procedencias. Se le asigna los valores de coordenadas (0, 0)
- **Indeterminado, UND**: se produce en las pruebas de relevos cuando los atletas tienen diferentes procedencias. Se le asigna los valores de coordenadas (0, 0)
- **Atleta Neutral Autorizado, ANA**: categoría para aquellos atletas que no pueden representar a su país porque este esta sancionado. Se le asigna los valores de coordenadas de la sede de la World Athletics situada en Mónaco.
- **Equipo de Atletas Refugiados, ART**: categoría asignada para aquellos atletas cuyos países están envueltos en conflictos y no participan en pruebas de la World Athletics. Se le asigna los valores de coordenadas de la sede de la Organización de las Naciones Unidas (ONU) situada en Nueva York, Estados Unidos.

- **Equipo Olímpico de Refugiados, ROT**: categoría asignada para aquellos atletas cuyos países están envueltos en conflictos y no participan en los JJOO. Se le asigna los valores de coordenadas de la sede de la ONU situada en Nueva York, Estados Unidos.

Una vez aplicadas las correcciones se une con la tabla ya existente de países para añadir estos tres nuevos campos a la misma. Este proceso se realiza tanto para el archivo de países como para el de ciudades. En el momento que se realiza este paso, los ficheros quedan listos para su uso y no se actualizarán salvo aparición de nuevos países o ciudades.

6.1.4 Clima

Para obtener los datos necesarios de esta dimensión, se crea un simple script en lenguaje Python en el que se consulta la API Open-meteo Historical Data. Estos datos se reciben en términos diarios exceptuando la humedad, la cual se recibe de forma horaria y es necesario calcular la media. Una vez recibidos los datos y calcula la humedad media se categorizan los valores de las dimensiones mostradas en rangos, los cuales están descritos en el apartado 5.1.2.

6.1.5 Generación de fechas

Para generar las fechas se emplean las librerías datetime [1] y calendar [2] del lenguaje Python además de diversas operaciones básicas. Se generan todas las fechas desde el 1 de Enero del 1934 hasta el 31 de Diciembre del 2030.

6.1.6 Cálculo de puntos húngaros

Para el cálculo de los Puntos húngaros se necesitan tres elementos: la marca del atleta, la disciplina y el género. Se creó un pequeño código que recibe los parámetros necesarios para ejecutar la librería y devuelve el resultado.

En cuanto al script 'padre' se elige el uso de Python por su mayor facilidad para el manejo de datos. En primer lugar si la marca está medida en unidades de tiempo debe ser convertida a segundos. El segundo paso consiste en realizar el mapeo de valores entre el nombre de la disciplina asignado por la World Athletics y el formato en el que se indica para la librería escogida. En caso de que la disciplina no tenga soporte para el cálculo de puntos se asigna un valor de 0. Una vez realizado el mapeo, se llama al script PHP mediante el uso de la librería subprocess de Python[3] y se recoge su resultado.

[1] https://docs.python.org/3/library/datetime.html
[2] https://docs.python.org/3/library/calendar.html
[3] https://docs.python.org/3/library/subprocess.html

6.2 Transformación

En esta sección se detallarán los diferentes procesos seguidos para obtener un conjunto de datos normalizado y limpio para poder cargarlos dentro de la base de datos objetivo. Para una explicación lo más clara posible se dividirán las transformaciones por cada una de las dimensiones del modelo. Es destacable mencionar que para poder enviar los datos entre transformaciones se generan una serie de archivos intermedios.

Es conveniente para la comprensión de esta sección mostrar la estructura de datos en que se extraen los diferentes resultados de la página de la World Athletics junto con algunos ejemplos para comprender mejor el formato.

- Enlace a la competición:
  https://worldathletics.org/competition/calendar-results/results/7192896
- Período de la competición: 17–18 JUN 2023 | 18 JUN 2023
- Nombre de la competición; Swiss Combined Events Championships
- Nombre completo del recinto: Stadion Schützenmatte, Basel (SUI)
- Categoría de la competición: B
- Grupo de la competición: National Senior Outdoor Combined Events Championships
- Tipo de la competición: Combined Events
- Día de la competición: 17 JUN 2023
- Nombre completo de la disciplina: Women's Shot Put
- Ronda: Final
- Viento: ', Wind: +1.4'
- Posición. '1.'
- Marca: 14.30
- Enlace del atleta:
  https://worldathletics.org/athletes/switzerland/mathilde-rey-14655511
- Nombre del atleta: Mathilde REY
- Fecha de nacimiento del atleta: 10 NOV 2023
- Nacionalidad del atleta: SUI

6.2.1 Validación

En primer lugar y antes de realizar ninguna transformación se realiza una pequeña validación de los datos de entrada con el fin de descartar las filas no válidas. Esta consiste esencialmente en los siguientes pasos en el orden en que se describen:

- **Eliminación de repetidos**: aunque en un inicio no deberían existir, se comprueba que no existen dos filas idénticas, y en caso de existir estas se retiran.
- **Eliminación de filas incongruentes**: se eliminan aquellas filas que tienen el puesto de finalización del atleta pero no tienen la marca realizada.
- **Eliminación de filas con nulos**: solo se aceptan filas con nulos en los campos a continuación listados, en caso de tener uno o más nulos en algún otro campo la fila será eliminada.
  - Viento
  - Posición
  - Fecha de nacimiento
  - Enlace del atleta
  - Marca
- **Nulos en posición**: en caso de pertenecer a una prueba del tipo Salto horizontal y tener el puesto nulo se reasignan a Fuera de competición (OC).

Adicionalmente para cada dimensión se realizarán modificaciones en sus datos, pero ya no se eliminarán más filas.

6.2.2 Atleta

Las principales transformaciones necesarias para esta dimensión son:

- Cálculo de la edad y el rango edad desde el parámetro fecha.
- Obtención de género y los datos del país.
- Obtención del código.

A continuación se entra más en detalle en estas transformaciones.

Cálculo edad

Esta es la primera transformación y tiene por objetivo obtener la edad del atleta en el momento de realización de la prueba así como categorizarla en rangos. En esta misma transformación también se encuentra la validación inicial previamente comentada en la sección 6.2.1. Se realiza de esta forma por motivos de eficiencia, generando un menor número de archivos intermedios.

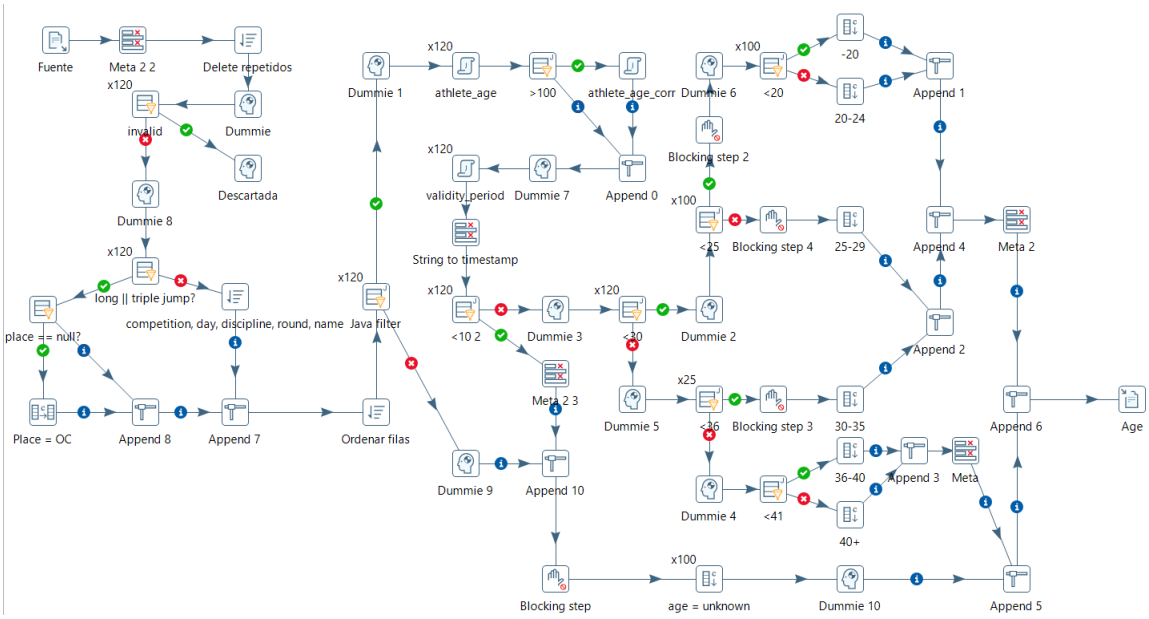

Figura 6.1: Transformación validación y edad del atleta.

Como se observa en la figura 6.1, el primer paso es separar los atletas sin fecha de nacimiento a los que se les asignan los valores 'Unknown' tanto en el campo edad como en el campo rango edad. Para los atletas que sí tienen una fecha de nacimiento, se calcula restando el año de la competición y el año de nacimiento[4]. Como de forma general no van a haber atletas con edad menor a 10, si existe alguno con edad inferior a 10, se le tratará como si careciera de fecha de nacimiento. A continuación, se asigna el rango de edad en el que está comprendido con diversos ifs anidados, nuevamente en el caso de los atletas sin fecha de nacimiento o los menores de 10 se asigna el valor 'Unknown'. Por último, se crean dos valores para poder simular posteriormente las actualizaciones pasadas y son las fechas de inicio y fin del año en que se realiza la prueba. dan nueva tan solo se calcula para los atletas con fecha de nacimiento y edad mayor a 10 años.

[4] Nótese que no se tiene en cuenta el mes, por lo que más que la edad en el momento de la competición el campo es edad a cumplir en el año de competición.

Género, país y continente

En la figura 6.2 mostrada a continuación se puede observar que en primer lugar se separa el campo disciplina en dos, el género y el nombre. Para unificar el género, asignamos a 'Boy' y 'Girl' 'Men' y 'Women' respectivamente. Prosigue uniendo el atleta con la tabla auxiliar generada con todos los datos de los países por el campo Nacionalidad del atleta. Una vez unido se calcula el rango de altitud en caso de existir valor para esta, indicándose 'Unknown' en caso contrario. Similar pasa para el desfase horario, en caso de no existir se asigna el valor 'Unknown'.

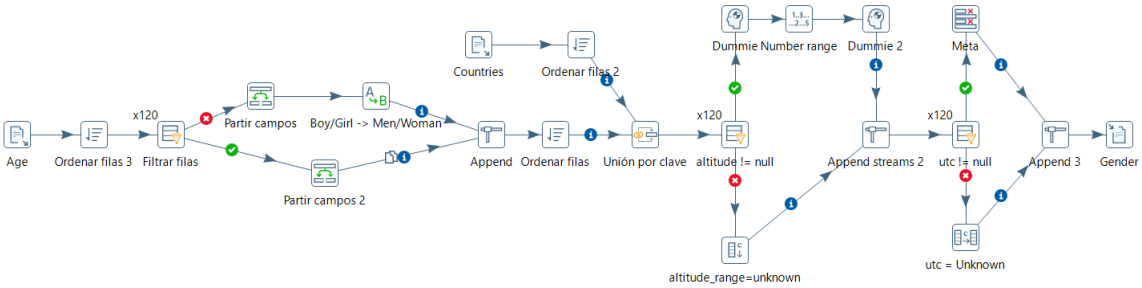

Figura 6.2: Transformación género y país del atleta.

Código

El código del atleta se obtiene del enlace para aquellos que compiten individualmente, salvo aquellos que no tienen. En el caso de los relevos siempre va a ser un código generado. Por tanto, el primer paso es separar el flujo diferenciando entre las pruebas de relevos y las individuales. Para las pruebas de relevos, lo primero es obtener el nombre del equipo ya que viene con este formato: ClubAPELLIDO1 Nombre1, APELLIDO2 Nombre2.... A pesar de que pueda parecer fácil a simple vista tan solo con aplicar un regex, no es tan sencillo y existen varias excepciones al regex general. Por la forma en que está configurado el regex, los mayores problemas los dan apellidos que contienen la segunda letra en minúscula, como los aquellos que tienen la raíz 'Mc'. Una vez obtenido el nombre, se comprueba si para los valores nombre - nacionalidad - género, ya existen en la base de datos. En caso de ser afirmativo se le asigna el valor del código y se termina la transformación para él, pero en caso de no tener se le asignará a todos los valores nombre - nacionalidad - género iguales un código único generado. Para generar este código se usa una secuencia con valor inicial 80000000 y aumentos de 1 unidad. En el caso de los atletas que participan en pruebas individuales, en caso de tener enlace se obtiene el trozo final de la cadena donde se encuentra el código y, en caso de no tener, se realiza el mismo proceso que con los clubes pero buscando por los pares nombre - nacionalidad.

Las diferentes fases explicadas se ilustran de forma gráfica con la figura 6.3.

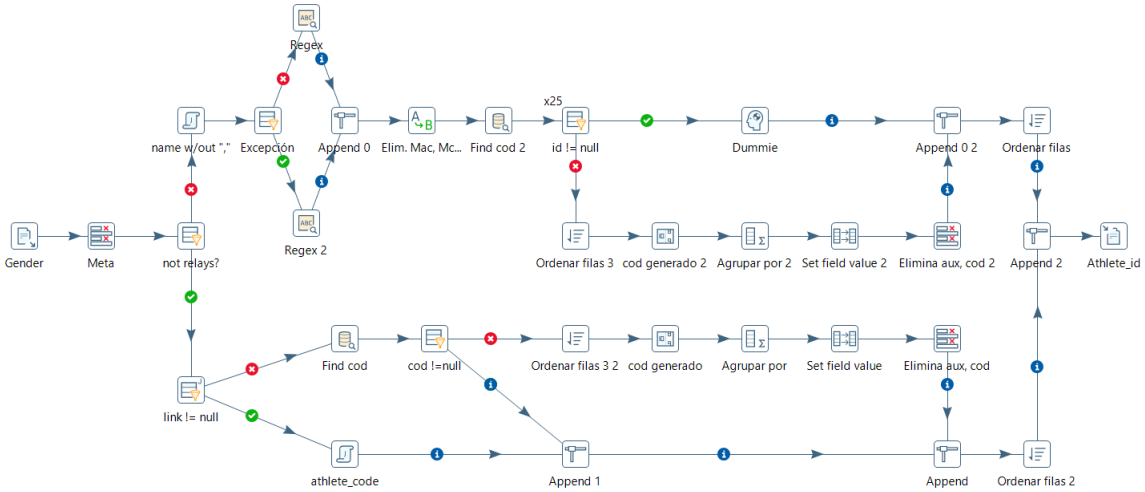

Figura 6.3: Transformación para obtener el código del atleta.

6.2.3 Recinto

En la dimensión recinto se deben realizar 3 transformaciones, la primera de ellas leer el archivo completo y para cada fila obtener la Geocodificación de la ciudad. Este script se ejecuta desde Pentaho como un archivo bash que a su vez ejecuta el script de Python donde se aplican técnicas de cacheado para minimizar el número de consultas a realizar a la API. Adicionalmente, se deben realizar dos transformaciones, estas son:

- Dividir el campo nombre recinto en los nombres del recinto y la ciudad.
- Obtener los datos de la ciudad.

División nombre recinto

En esta transformación representada en la figura 6.4, el objetivo es encontrar un algoritmo que procese los diferentes formatos en los que está escrito el nombre del recinto con el fin de obtener por separado el nombre del recinto y el de la ciudad donde se realiza. En caso de no existir un recinto como tal este se indica con el valor 'Unknown'. Los formatos en los que se puede encontrar el campo nombre recinto y como se desean dividir son los siguientes:

- New York, NY (USA) -> Unknown | New York
- Armory Track, New York, NY (USA) -> Armory Track | New York
- Madrid (ESP) -> Unknown | Madrid
- Estadio Vallehermoso, Madrid (ESP) -> Estadio Vallehermoso | Madrid

- Estadio Vallehermoso, UCM, Madrid (ESP) -> Estadio Vallehermoso, UCM | Madrid

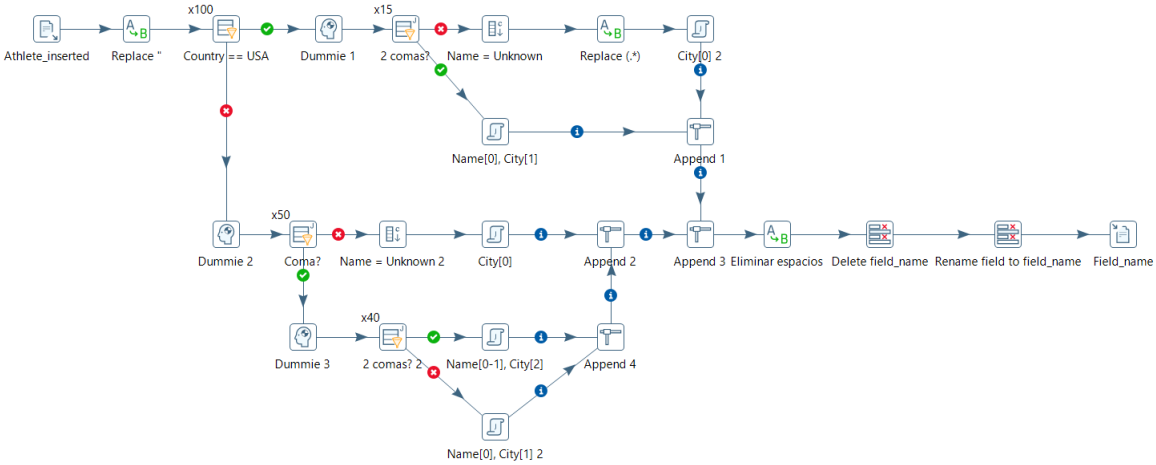

Figura 6.4: Transformación para dividir el nombre del recinto.

País y continente

Transformación prácticamente idéntica a la segunda parte de la figura 6.2. En la figura 6.5 se aprecia como la única diferencia es la inclusión del prefijo 'field' para que se una de forma correcta.

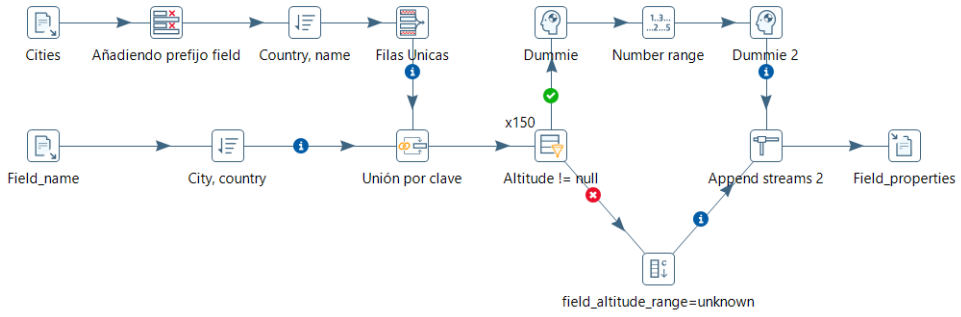

Figura 6.5: Transformación para dividir el nombre del recinto.

6.2.4 Entorno

En esta dimensión que almacena datos comparativos entre el lugar de realización del evento y el lugar de procedencia del atleta.

En primer lugar se calcula la distancia para la cual se busca un algoritmo que realice este cálculo. Se descubren las dos fórmulas más relevantes, Haversine y Vincenty. La fórmula de

Haversine o del semiverseno, es una fórmula trigonométrica basada en la ley de los senos que calcula la distancia angular entre dos puntos de la superficie de una esfera perfecta. Es una fórmula sencilla y eficiente, pero no contempla el ligero achatamiento de la tierra, lo cual le hace perder precisión especialmente en latitudes cercanas a los polos y en distancias muy largas. Aún con ello, demuestra ser una buena aproximación, con un margen de error menor al 1% [10]. En la figura 6.6 se aprecia visualmente como se calcularían las distancias entre tres puntos en la esfera.

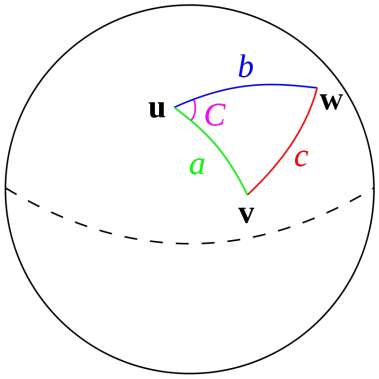

Figura 6.6: Cálculo de un triángulo en una esfera con la fórmula de Haversine. [2]

Mientras, la fórmula de Vincenty, emplea un elipsoide de revolución, concretamente el empleado como referencia en WGS84, en la figura 6.7 se encuentra una visualización gráfica del elipsoide. Debido a la mayor complejidad del elipsoide de revolución se obtiene una fórmula más compleja y con un mayor número de operaciones que la fórmula de Haversine. En consecuencia, también obtiene resultados con mayor precisión.

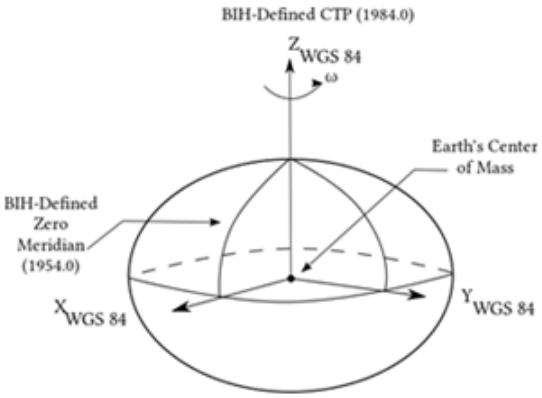

Figura 6.7: Elipsoide de revolución empleado por WGS84 [3]

En este contexto en el cual la precisión no es crítica y además el resultado va a ser categorizado se selecciona la formula de Haversine por su mayor sencillez de cálculo. Para poder

realizar este cálculo se empleará la librería Haversine[5] de Python. Para realizar este cálculo se emplea un script Python ejecutado en Pentaho mediante un archivo bash. Este archivo genera un fichero auxiliar con las coordenadas del atleta y las coordenadas del lugar de competición así como la distancia en kilómetros entre estas. Adicionalmente a esta transformación se realizan dos más para completar la dimensión:

- Horas de diferencia y categorización distancia.
- Cálculo de la diferencia de altitud.

Horas y distancia

En esta transformación se calculan las horas de diferencia entre el lugar de realización de la prueba y el lugar de procedencia del atleta. En caso de alguno de los dos campos no tener un valor numérico, se indica el el valor 'Unknown'. A continuación, se divide el flujo en las filas que tienen las coordenadas de origen y de la prueba diferentes de nulo. Seguidamente, se unen con las filas generadas por el script de cálculo de distancias comentado. En cambio, para las filas que tienen alguna coordenada a nulo se indica nuevamente el valor 'Unknown'. Todo este proceso se puede apreciar de forma gráfica en la figura 6.8

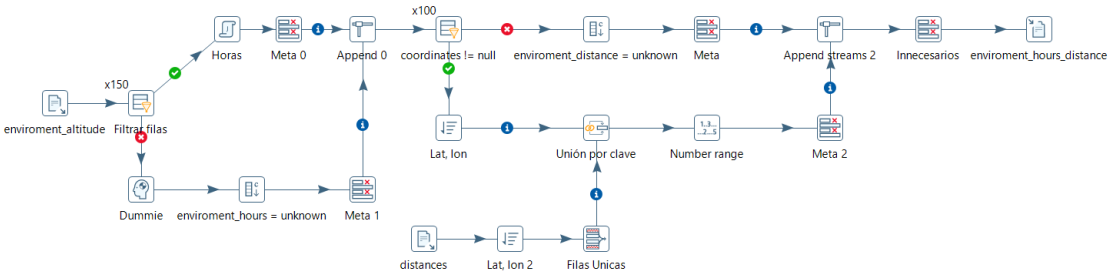

Figura 6.8: Transformación para calcular la diferencia de horas y distancia.

Altitud

En esta compleja transformación representada en la figura 6.9, se calculan los niveles de diferencia entre la altitud de procedencia del atleta y la de la competición. El resultado puede ser cualquier valor discreto en el intervalo [-3, 3] y adicionalmente el valor 'Unknown' en caso de que alguna de las dos altitudes sea nula.

[5] https://pypi.org/project/haversine/

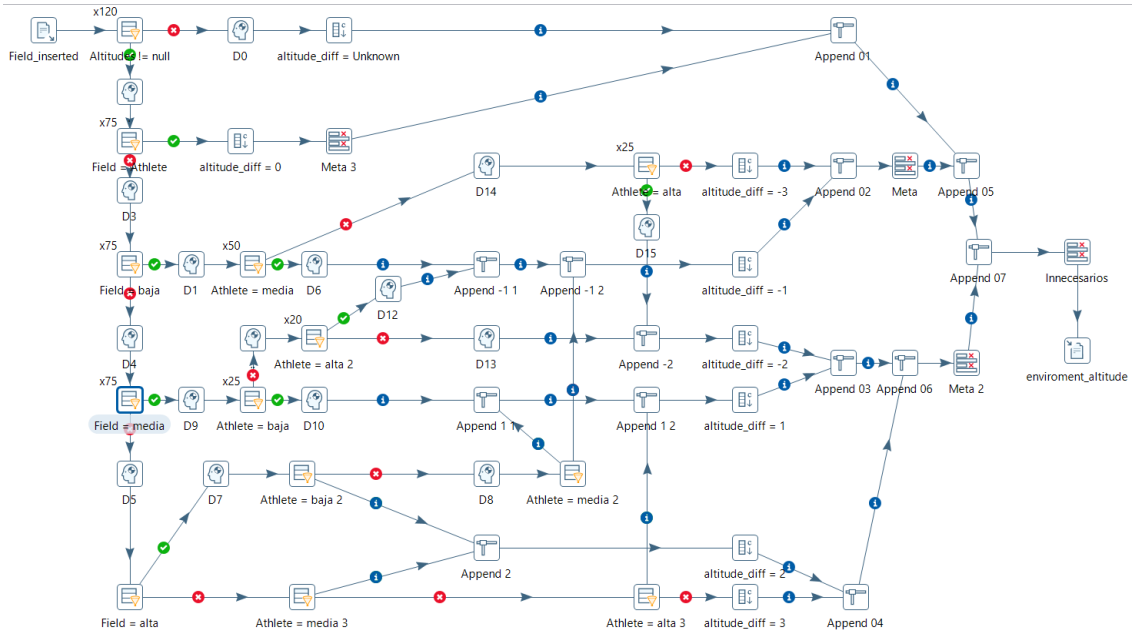

Figura 6.9: Transformación para calcular la diferencia de niveles de altitud.

6.2.5 Competición

En esta dimensión que almacena la información relativa al torneo donde se realiza la prueba, se necesitan esencialmente dos transformaciones.

- Segmentación de los valores de grupo de competición en los diferentes valores booleanos asociados.
- Segmentación de los valores del tipo de competición en los diferentes valores booleanos asociados.

Adicionalmente también se obtienen el año y el código de la competición. Pero no constituyen una transformación propia por motivos de eficiencia. Incluyéndose por tanto en la transformación de grupo.

Segmentación de grupo

En primer lugar se parte el atributo grupo de competición en hasta 5 campos separando por el carácter ‘,’. Para a continuación comprobar si aparece en alguno de estos los campos almacenados. En caso de aparecer se asigna el campo a verdadero y ya no se comprueban el resto de campos ya que no son compatibles, asignándose por tanto a falso. Una vez realizado este cálculo se extrae el código de la competición del enlace y el año de la fecha. La figura 6.10 muestra de forma gráfica y accesible los pasos que se acaban de comentar.

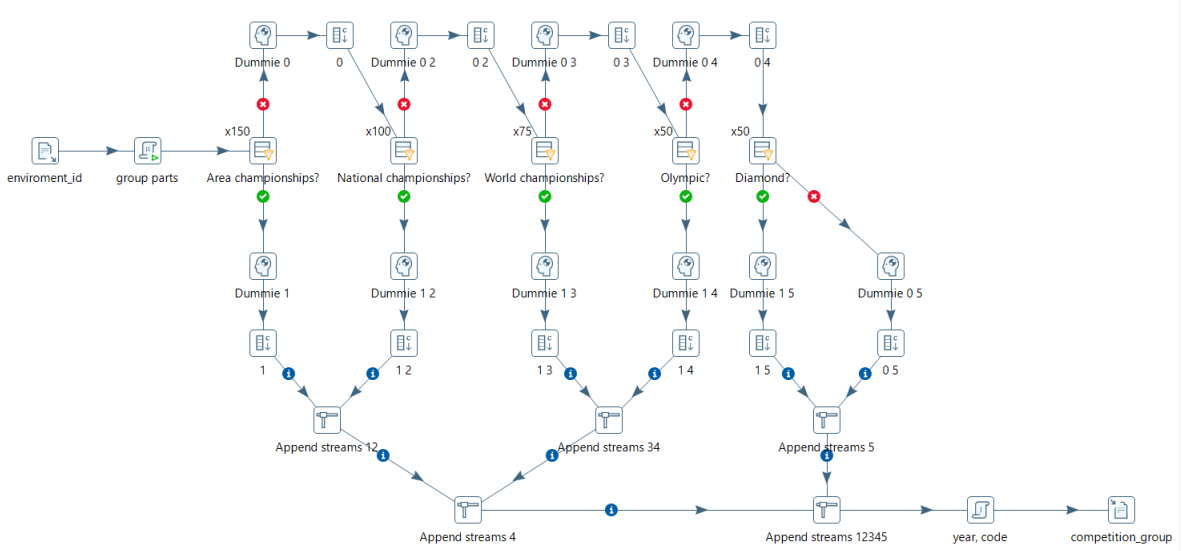

Figura 6.10: Transformación de grupos, código y año de competición.

Segmentación de tipo

En esta transformación se busca extraer los diferentes tipos de pruebas que se albergan en una competición. Para esto se comprueba secuencialmente si el valor de cada uno de los 6 valores posibles aparece en el campo tipo extraído. Para cada uno se asigna el valor verdadero o falso en función de si se encuentra. Esta transformación se aprecia de forma gráfica y sencilla en la figura 6.11.

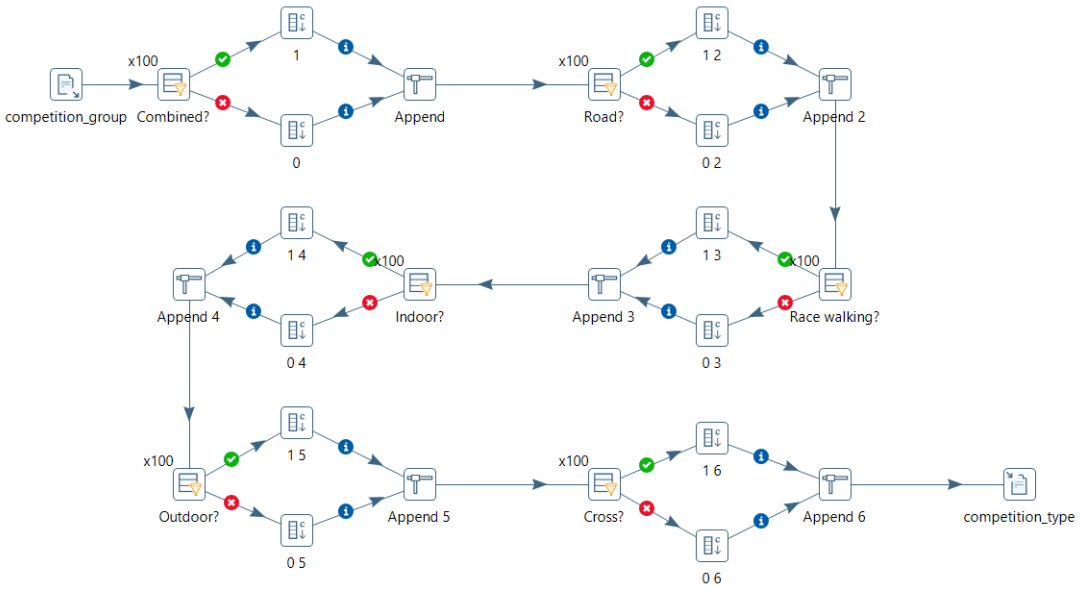

Figura 6.11: Transformación para obtener los tipos de competición.

6.2.6 Disciplina

En esta dimensión donde se almacena información sobre la prueba en la que participa el atleta. Son necesarias dos transformaciones, la primera de ellas ya fue realizada para la obtención del género de atleta:

- Separación del nombre de la disciplina en el nombre como tal y el género.
- Categorización de la prueba en tipo, subtipo y unidades de medida.

La primera de estas transformaciones ya fue realizada para obtener el género del atleta en la figura 6.2. Por tanto, en esta sección solo se comentará la realización de la segunda de las transformaciones.

Categorización

Para completar esta categorización, en primer lugar se comprueba el el tipo principal de forma secuencial, separando el flujo en cada paso. Una vez separados los flujos según el tipo de prueba, se asigna el nombre del tipo y las unidades en que se mide, comunes a todas las pruebas del tipo. Por último, se categoriza en subtipos para las pruebas en las que existen, asignándose el valor 'Unknown' en aquellas pruebas que no disponen de subtipo. Esta transformación y su compleja representación gráfica se muestran en la figura 6.12.

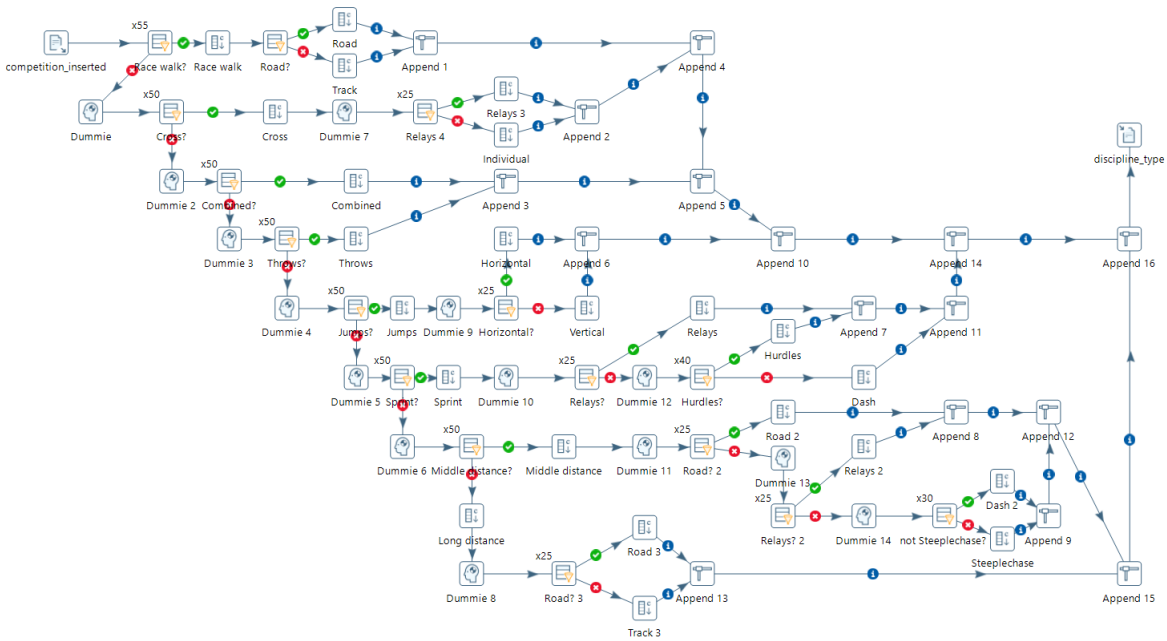

Figura 6.12: Transformación para categorizar los tipos y subtipos de pruebas.

6.2.7 Resultado

Para esta dimensión se realizan diferentes transformaciones para asegurar la calidad de los datos a insertar en la base de datos. En total se registran 2 transformaciones además del script Python para calcular los Puntos húngaros ejecutado Pentaho mediante un archivo bash. Como resultado de este script obtenemos el dataset de entrada con dos nuevos campos, el número de Puntos húngaros obtenido y la marca en la Unidad mínima. Transformaciones adicionales:

- Posición, marca y días.
- Cálculo de diferencias y días restantes.

Cabe destacar que el script se ejecuta intercalado entre las dos transformaciones mencionadas.

Posición, marca y días

En primer lugar se modifica las marcas que contienen el carácter 'h'[6]. A continuación, se busca si existe un carácter '.' dentro de la posición y en caso de existir se retira. Por último, se modifica el campo día de competición. En este se realizan las siguientes modificaciones:

- Separación del campo día de competición por el carácter '-', obteniendo así el día de competición y la fecha por separado. En la figura 6.13 se muestra este paso aplicando una transformación de tipo 'split'.
- En el campo día se elimina la cadena 'Day ' quedándonos tan solo con el número de día dentro de la competición.
- Obtención del máximo número de día agrupando por cada competición. De esta forma se obtiene la duración en días de la competición.

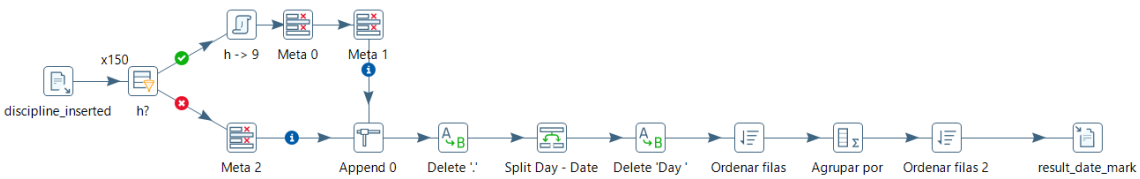

Figura 6.13: Transformación para corregir las marcas y obtener los datos de días.

[6] Tiene formato similar a '10.3h', se produce con las resultados medidos con un solo dígito de precisión. Como es imposible determinar el valor, se indica el de mayor tamaño es decir, 9. De esta forma para '10.3h' se obtiene '10.39'

Diferencias y días restantes

En esta última transformación de la dimensión resultado representada en la figura 6.14, se obtienen los valores del primer clasificado de la prueba para aquellos que tienen una marca diferente a DNS, DNF, NM, NT, DQ o VST. Una vez se obtiene el resultado del primer clasificado, se calcula el diferencial respecto al resultado del atleta en cuestión. Esta diferencia se calcula en la Unidad mínima y en Puntos húngaros. En algunos casos podría darse la situación de que sean negativas por ser marcas que no cuentan para el resultado final de la competición. Una vez el valor de la posición no se va a volver a emplear podemos modificar los valores nulos para asignarles la cadena 'Unknown'. Una vez realizado, calculamos los días restantes de competición mediante la resta del total de días y el día actual. Por último y para finalizar la transformación, se une la fecha de la competición extraída en la sección 6.2.7 con la fecha formateada para obtener el id de fecha correspondiente.

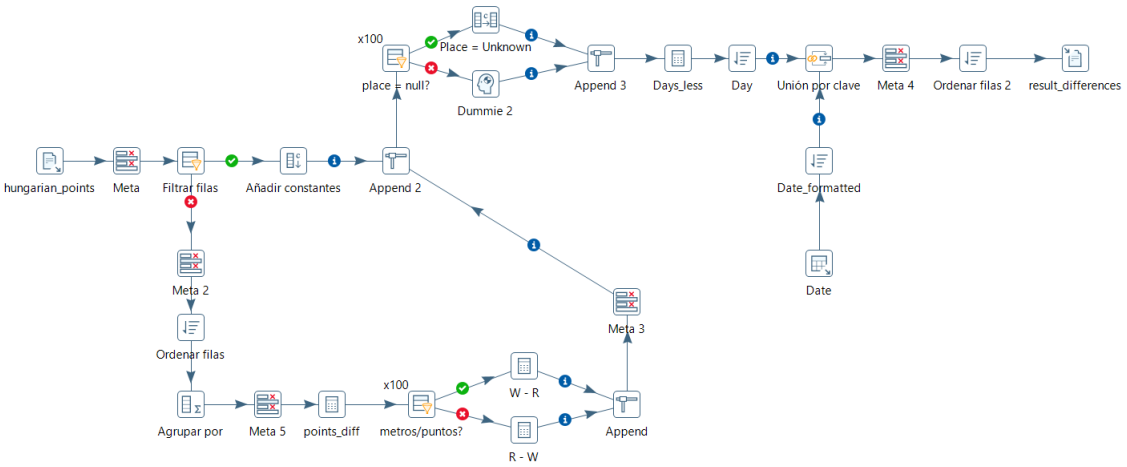

Figura 6.14: Transformación para calcular las diferencias y los días restantes.

6.2.8 Clima

En esta última dimensión, se obtienen los datos de clima mediante un script bash ejecutado en Pentaho que a su vez ejecuta el script de Python donde se aplican técnicas de cacheado para minimizar el número de consultas a realizar a la API. [11] Este script genera un archivo con los diferentes datos de clima obtenidos.

Una vez obtenido este archivo, se debe realizar una transformación adicional para categorizar en rangos la velocidad del viento. Esta transformación reflejada en la figura 6.15 también une las filas con el archivo de generado por el script. Con la excepción de aquellos con fecha previa a 1940 debido a que la API no contiene datos para esas fechas. Para los datos de estas fechas se asigna un clima con todos los valores a 'Unknown'.

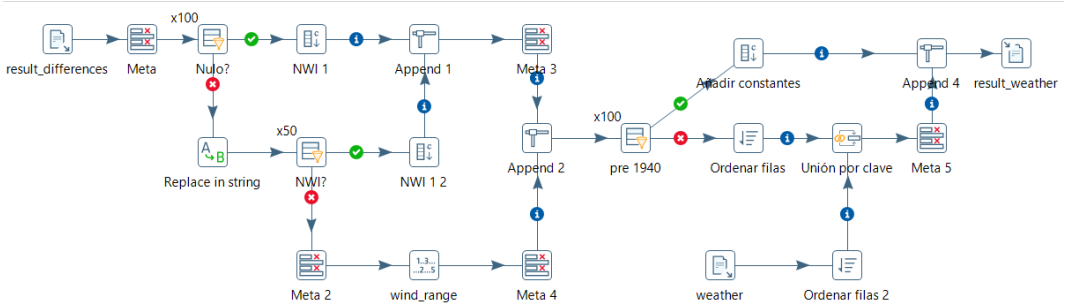

Figura 6.15: Transformación para calcular el rango de viento

6.3 Carga

En esta última fase del proceso ETL se explicarán las diferentes estrategias de carga y actualización llevadas a cabo para cada una de las dimensiones. Principalmente se emplean tres enfoques diferentes para realizar la carga:

- **Bulk load**: esta es una forma de carga eficiente y optimizada especialmente pensada para la inserción o actualización de grandes volúmenes de datos.
- **Búsqueda/actualización en combinación**: este tipo de carga comprueba si existe una fila idéntica a la que se intenta insertar. En caso de existir devuelve el id de la fila ya insertada y en caso de no existir se inserta.
- **Búsqueda/actualización en dimensión**: este tipo de carga es la que permite emplear estrategias de control de cambios. Permite tanto la política de actualización SCD-1 como SCD-2. En la figura 6.16 se aprecia la configuración de este paso en la dimensión atleta.

La inserción de tipo bulk se emplea en las dimensiones fecha y resultado, en estas se puede aplicar porque se tiene la certeza de que no existen repetidos. Para la dimensión resultado se tiene además la particularidad de generar una secuencia en la base de datos para obtener un valor secuencial empleado como identificador. Esta transformación queda reflejada paso a paso en la figura 6.17.

Para el resto de dimensiones, se emplea la estrategia Búsqueda/actualización en combinación y la transformación para insertarlas es similar si no idéntica a la figura 6.18.

Por último, para poder reflejar la periodo de validez del atleta de forma correcta, se crea una transformación update que se muestra en la figura a 6.19. Esta transformación se bifurca para reflejar si es la fila con la versión actual o no. En caso de ser una la fila con la versión

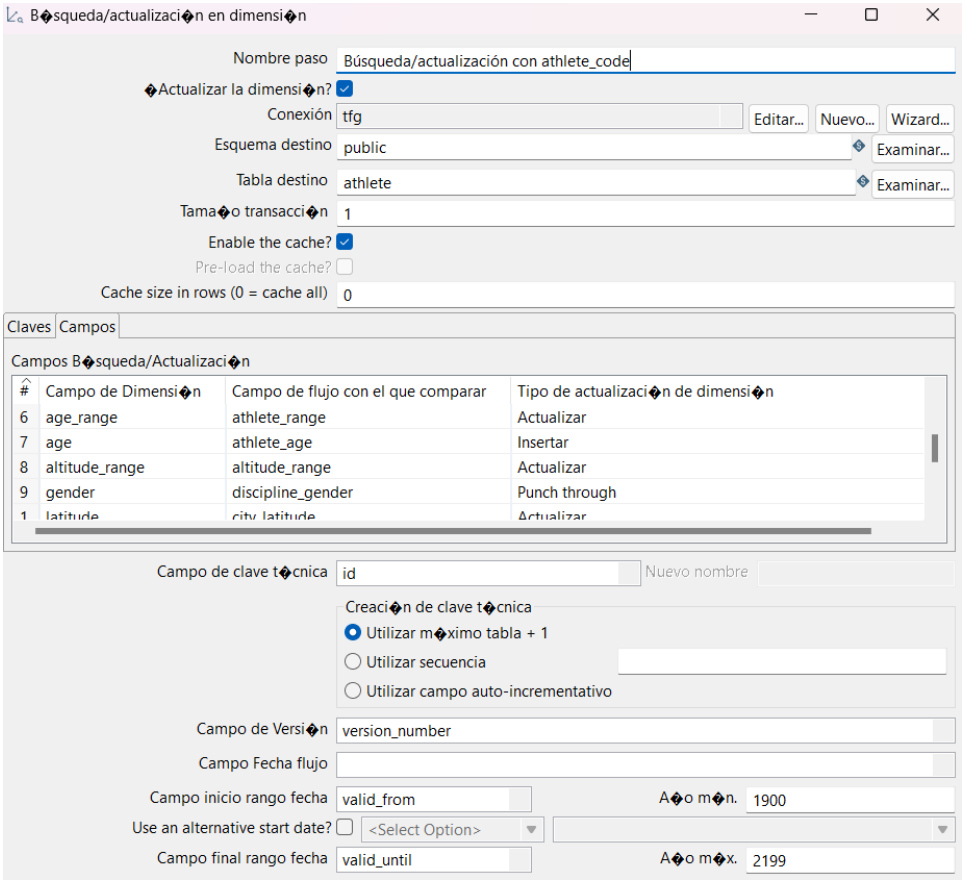

Figura 6.16: Configuración SCD de la dimensión atleta

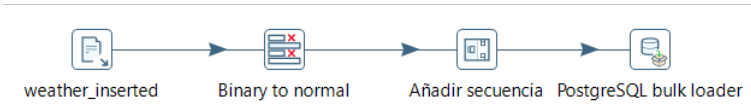

Figura 6.17: Inserción de resultados empleando Bulk Load

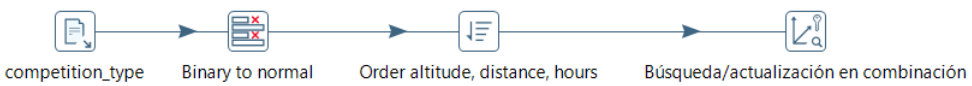

Figura 6.18: Inserción de competición con Búsqueda/actualización en combinación

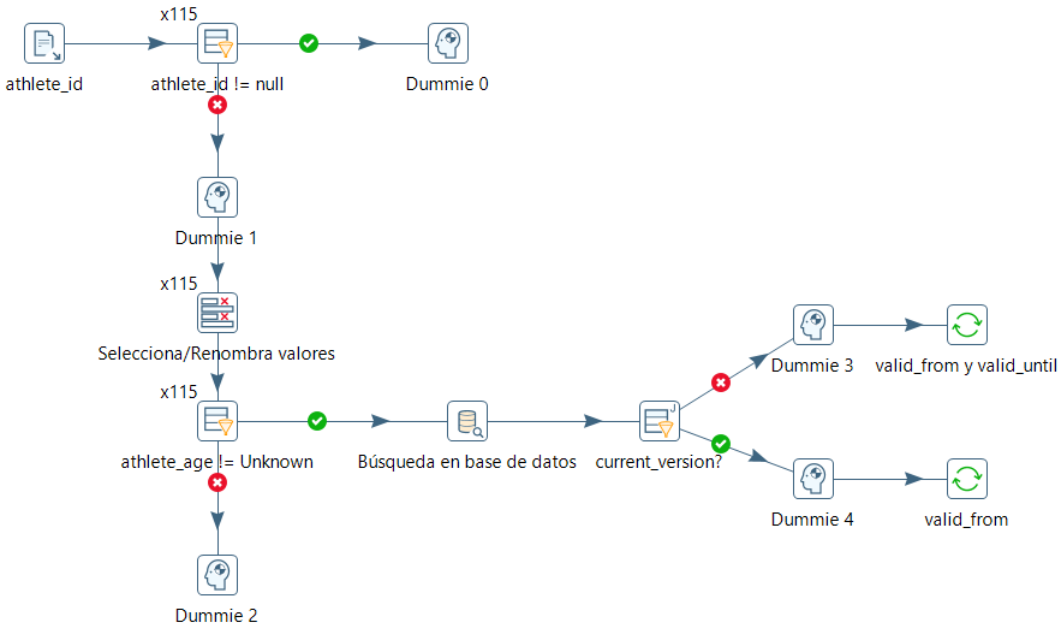

Figura 6.19: Actualización para simular el cambio

actual, solo se actualiza el campo 'válido desde' mientras que si la fila no es actual se actualizan tanto el campo 'válido hasta' como el 'válido desde'.

Finalmente, destacar las ligeras variaciones entre el trabajo de la primera inserción respecto a los incrementales posteriores. Esencialmente son los tres que se indican a continuación:

- En el incremental los datos se recogen previamente a iniciarse el proceso en Pentaho y por tanto no tiene ese paso ejecutando el script.
- En el incremental no es necesario realizar la comprobación de la figura 6.15 sobre si es una fecha previa a 1940. Por tanto se modifica la transformación para que en los incrementales no tenga ese dato.
- Una vez cargadas las fechas no es necesario volverlas a cargarlas hasta que se acerque la última fecha generada.

A continuación se puede observar la figura 6.20 que muestra el trabajo con sus diferentes transformaciones y scripts realizados para la primera carga.

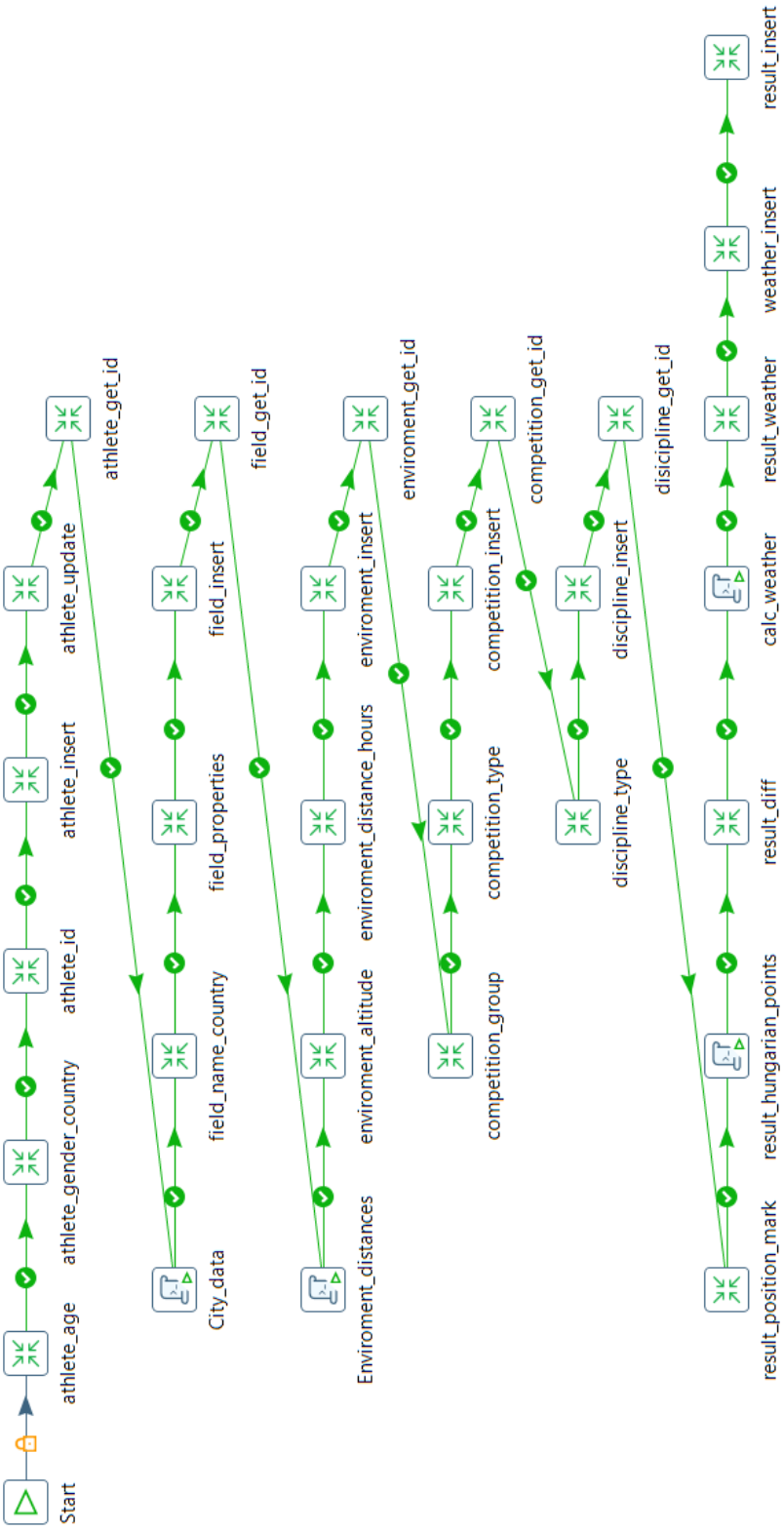

Figura 6.20: Trabajo con todas las transformaciones y scripts aplicados en el ETL

Capítulo 7

Explotación

Una vez completado el proceso ETL y obtenido el Data Warehouse cargado, es momento de comenzar a explotarlo. En este caso la explotación está centrada en la obtención de diversos informes y métricas empleando Power BI. La bondad de la aproximación es que se pueden ampliar las gráficas fácilmente, e incluir nuevas sin coste alguno. A continuación, se expondrán los informes realizados como muestra de la utilidad del sistema.

7.1 Atleta

En esta pestaña del informe se recogen los principales datos del atleta. En la figura 7.1 se muestran, por tanto, el rango de edad, el género, el continente y los principales países por número de atletas. Si bien estos gráficos son interesantes, el principal gráfico de la pestaña es el gran mapa central donde se encuentra la distribución mundial de los atletas. Por último, se incluye un filtro textual con el que poder buscar un atleta concreto. Destacar que tanto el mapa como el resto de gráficos permiten el filtrado por ese valor clicando sobre él. Adicionalmente, se crea la métrica edad media. Para su cálculo, se retiran aquellos valores no numéricos como 'Unknown' y se transforma el campo en un entero al que se le calcula la media.

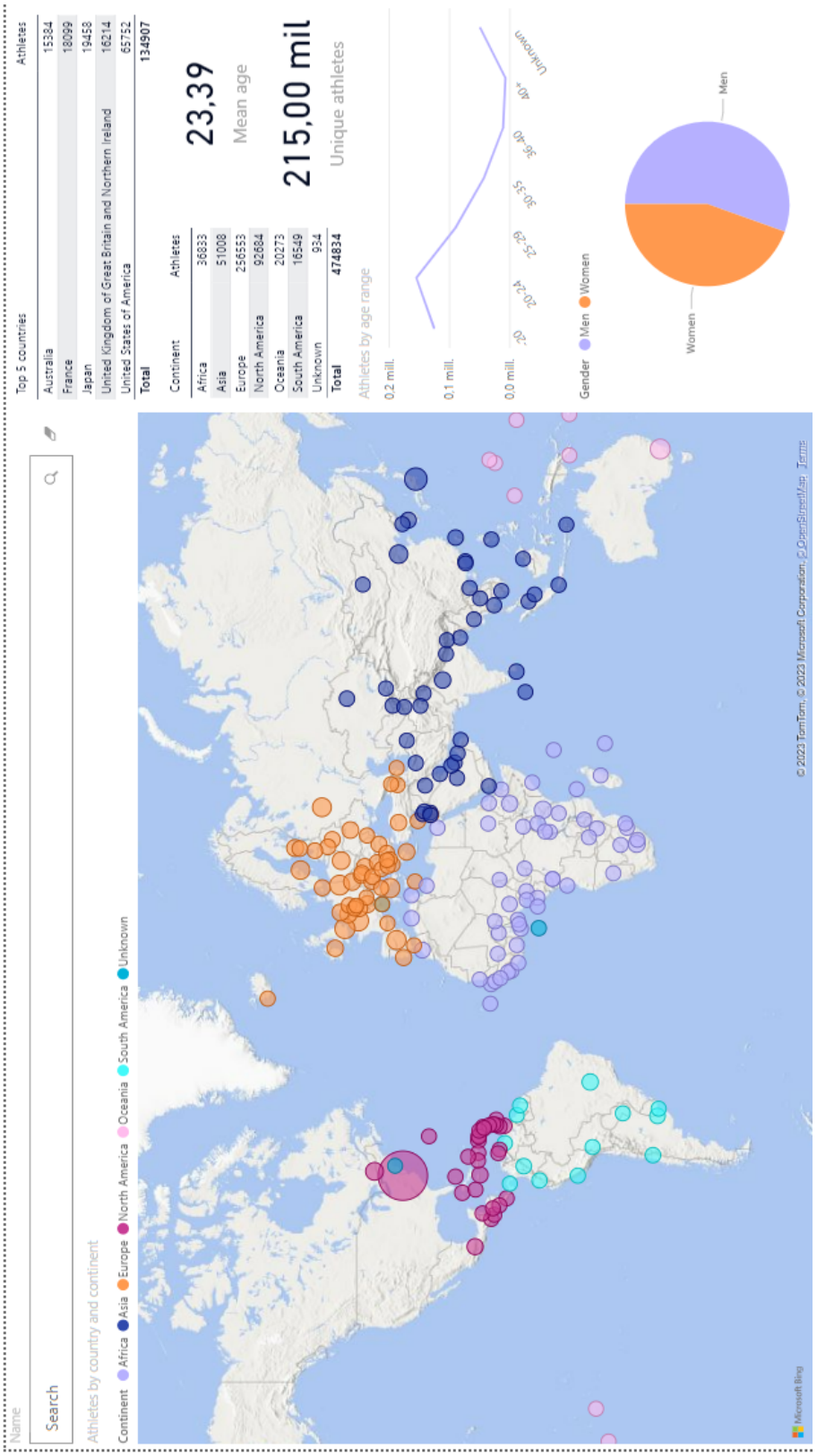

Figura 7.1: Informe con los principales datos sobre el atleta

7.2 Competición y recinto

En esta segunda pestaña del informe se muestran diferentes gráficas para reflejar los datos del lugar de realización así como del torneo asociado. En la figura 7.2 destaca principalmente el gráfico con el mapa donde se muestran los diferentes recintos donde en algún momento se albergó una competición. Para poder obtener la información requerida sobre este mapa se generan múltiples filtros con los que detallar más o menos el informe. Algunos de estos filtros son:

- País y continente de origen del atleta.
- Año de realización: acota las competiciones a las comprendidas en el intervalo indicado.
- Zona horaria: reduce las mostradas a solo aquellas en el rango indicado.
- País y continente donde se realiza la competición.
- Ciudad donde se realizan las diversas pruebas.
- Categoría de la competición.

Estos filtros pueden combinarse al gusto del usuario, pudiendo emplear cuantos quiera al mismo tiempo. Adicionalmente, se añade una métrica para conocer la cantidad media de competiciones y se permite la búsqueda de un atleta concreto, pudiendo visualizando así su historial completo de competiciones.

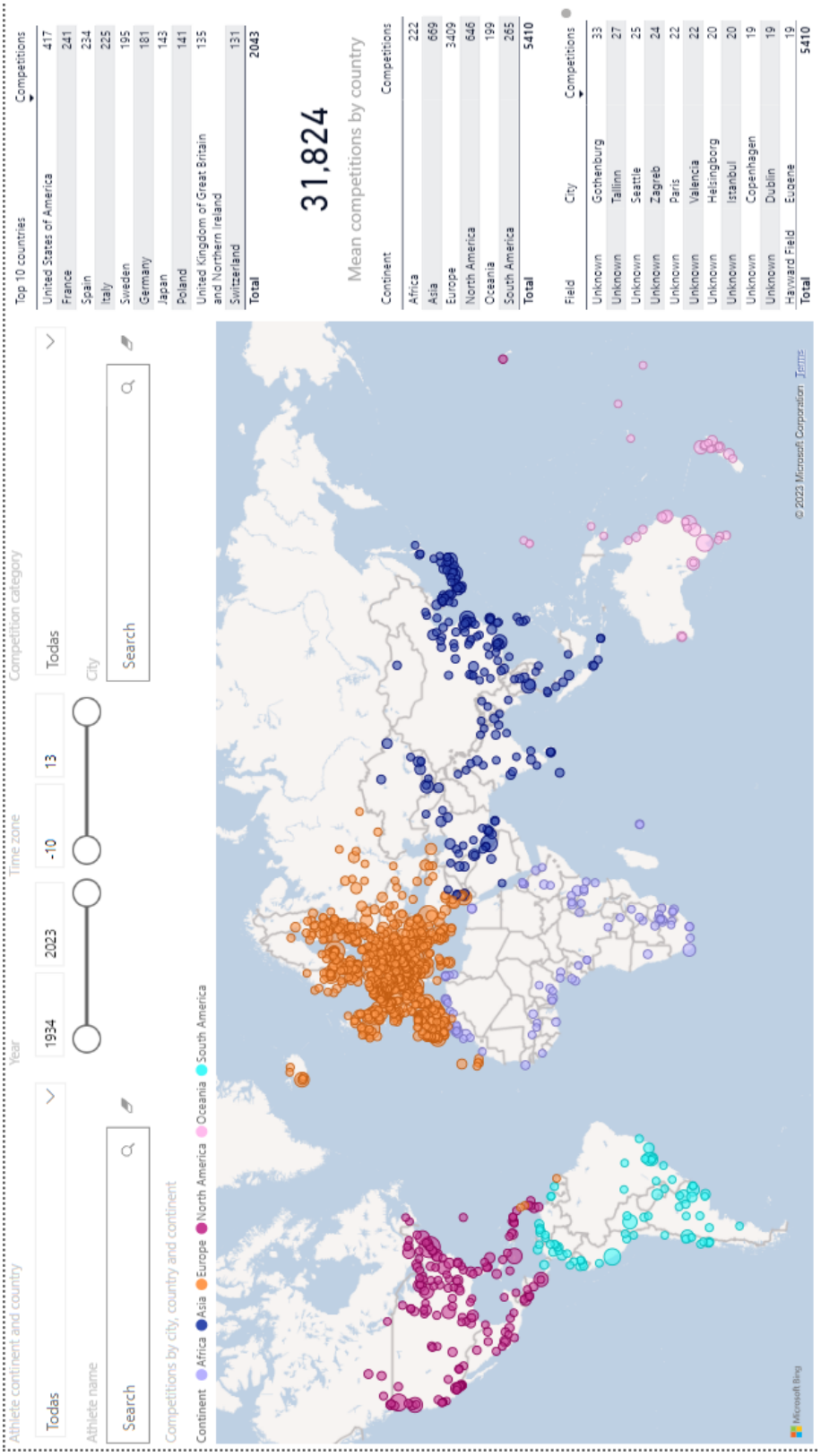

Figura 7.2: Informe con los principales datos sobre las competiciones y el recinto donde se albergan

7.3 Resultados

En esta última y compleja pestaña del informe mostrada en la figura 7.3 se representan diferentes gráficas para reflejar la mayor cantidad de datos sobre las condiciones en las que se realiza cada una de las pruebas. Se crean diferentes representaciones y gráficos con los que representar las condiciones climáticas más importantes así como las características del entorno o el género. Otros gráficos destacables son la media de puntuación, la cual se indica en un rango de 0 a 1.400 por ser los valores mínimo y máximo, y el gráfico con las métricas de puntos[1] con su evolución en el tiempo. Para realizar este segundo gráfico, se generan dos medidas, la media de diferencias con el ganador y la media del ganador. La primera se obtiene filtrando los valores no numéricos, cambiando y tipo y calculando la media, mientras que la segunda se obtiene de la suma de la puntuación media con la primera medida calculada. datos del lugar de realización así como del torneo asociado. De igual forma, para poder representar el gráfico con la posición media, se obtiene la media filtrando los valores que no contienen valores numéricos. Esta gráfica se representa entre 1 y 10 por ser la franja de resultados más habitual, aunque pueden obtenerse valores superiores a 10, especialmente en caso de estar involucrada alguna prueba de gran fondo como podría ser la maratón o los 35 km marcha.

Por último, se tiene el mapa con la localización de las diferentes pruebas para una mayor comprensión. Como no podía ser de otra forma se añaden diversos filtros con los que poder precisar la información que se quiere consultar. Estos filtros son, además de las propias gráficas, los siguientes:

- Continente y país del atleta.
- Tipo, subtipo y prueba realizada.
- Año de competición.
- Zona horaria del recinto donde se celebra la competición.
- Buscador del nombre del atleta.

[1] Media del ganador, media general y diferencia entre ambas.

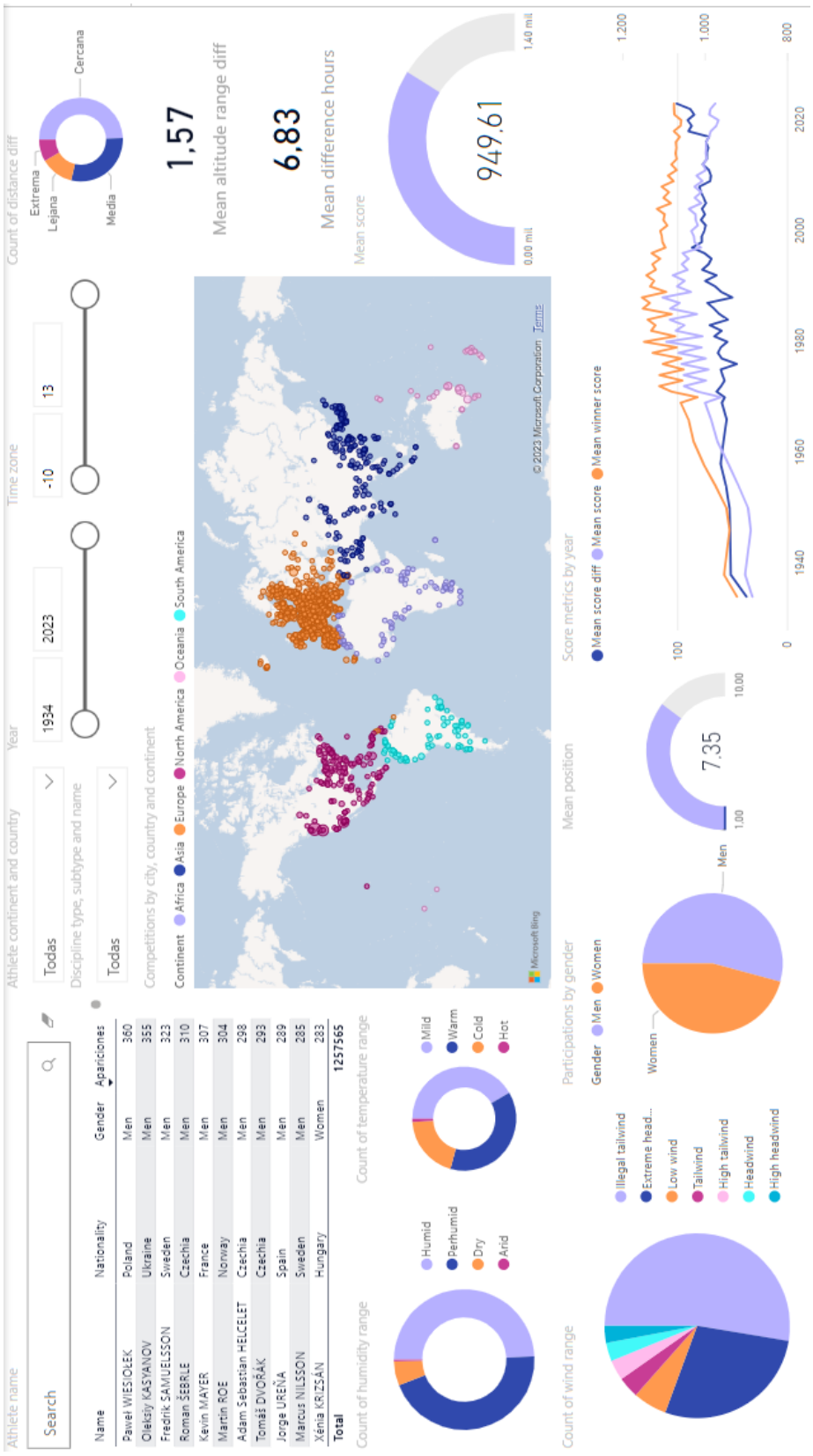

Figura 7.3: Informe que contiene diferentes métricas sobre el resultado

Capítulo 8

Conclusiones

En conclusión, este trabajo fin de grado ha demostrado la viabilidad y beneficios de implementar un Data Warehouse para el análisis de pruebas de atletismo. A través de un enfoque riguroso en la recopilación, transformación y almacenamiento de datos, se logró desarrollar una solución efectiva con la que poder analizar datos de pruebas de atletismo

La implementación del Data Warehouse permite centralizar y consolidar los datos provenientes de diversas fuentes, como páginas webs con estadísticas, bases de datos con información histórica o obtener el cálculo de la compleja asignacion de puntos de la World Athletics. También se establece un proceso ETL para garantizar la integridad y consistencia de la información. Estos conceptos facilitan el acceso a la información de manera eficiente y mejorando la calidad de los análisis.

La estructura y diseño del data warehouse fueron cuidadosamente planificados considerando los requerimientos concretos del negocio. Gracias a este diseño centrado en el negocio de pueden generar informes con gran utilidad proporcionando tanto a atletas como federaciones la posibilidad de soportar sus decisiones en datos específicos. Esto posibilita una mejor comprensión de las causas del rendimiento atlético, permitiendo identificar las áreas de mejora del atleta.

En resumen, este trabajo destaca la importancia del data warehouse en el contexto de las pruebas de atletismo al proporcionar una solución sólida para la gestión y análisis de datos.

En el futuro se prevé continuar mejorando el sistema especialmente en lo relativo a la explotación a medida que se descubran nuevas necesidades de uso. Adicionalmente se intentará integrar datos fisiológicos tales como las pulsaciones o los niveles de ácido láctico pre y post competición. También se planea aprovechar el conocimiento y experiencia adquiridos para realizar análisis más profundos y estimar el rendimiento futuro de los atletas.

Lista de acrónimos

ACID Atomicidad, consistencia, aislamiento y durabilidad. 4

COI Comite Olímpico Internacional. 17, 22, 33

DFM Dimensional Fact Model. iv, 20, 21

DNF El atleta no finaliza la competición. 48

DNS El atleta no inicia la competición. 48

DQ Descalificado. 48

ETL Extracción, transformación y carga. iv, 2, 3, 11, 32, 49, 52, 53, 59

GUI Interfaz gráfica de Usuario. 4

IDE Entorno de desarrollo integrado. 8

ISO Organización Internacional de Estandarización. 17, 22, 33

JJOO Juegos Olímpicos. 15, 25, 35

NM Sin marca. 48

NT Sin tiempo. 48

OC Fuera de competición. 37

ONU Organización de las Naciones Unidas. 34, 35

SCD Slowly Changing Dimension. iv, 23, 30, 49, 50

UTC Tiempo Universal Coordinado. 22, 24

VST Desconocido. 48

Glosario

API Interfaz de programación de aplicaciones por sus siglas, es una interfaz proporcionada por un software que permite una comunicación estandarizada y sencilla con este. Tiene un comportamiento similar a un contrato de servicio. 2, 5, 16–19, 33, 35, 40, 48

API-key Código único proporcionado para poder autenticar y autorizar el acceso a una . Puede enviarse como cabecera HTTP o como parámetro en la propia URL. 16

Data Warehouse Base de datos centralizada que contiene datos estructurados y consolidados para su análisis. iv, 1–3, 10, 11, 13, 21, 29, 53, 59

dataset Conjunto de datos estructurado y almacenado para su posterior análisis o uso. Normalmente es una tabla o matriz de datos. 14

Geocodificación Proceso que mediante el uso de algoritmos y bases de datos transforma una ubicación descriptiva en su ubicación geoespacial, es decir, sus coordenadas de latitud y longitud. 15–17, 40

PHP Lenguaje de programación de código abierto utilizado principalmente para el desarrollo de aplicaciones web dinámicas ejecutandose en lado servidor. 19

Puntos húngaros Asignación numérica que establece el mérito de una marca en función del género y la disciplina. Oscila en un rango entre 0 y 1400. 14, 35, 47, 48

Python Lenguaje de programación interpretado, de alto nivel y multiparadigma. Destaca por su legibilidad y sintaxis sencilla. 2, 7, 8, 15, 16, 18, 33, 35, 40, 43, 47, 48

Salto horizontal Tipo de prueba que engloba las disciplinas en las que se realiza un salto paralelo al suelo. Estas son, triple salto y salto de longitud. 33, 37

Unidad mínima En el caso de las medidas en unidades de tiempo en segundos y para las medidas en unidades de distancia en metros. 47, 48

Web Scraping Proceso automatizado de extracción de información de páginas web mediante software específico o código donde se analiza la estructura de la página web con el fin de capturar la información que esta contiene. 2, 15, 17

WGS84 Sistema Geodésico Mundial 1984. Es un sistema de coordenadas habitual de los sistemas de información geográficos, es el sistema de coordenadas de referencia para el GPS. iv, 42

World Athletics Organismo regulador del atletismo a nivel mundial. 2, 15, 17–19, 22, 23, 25, 32–36, 59, 63

Bibliografía

[1] R. Kimball, M. Ross, W. Thornthwaite, J. Mundy, and B. Becker, *The Data Warehouse Lifecycle Toolkit*, 2nd ed. Wiley, 2008.

[2] S. G. Johnson, "Law of haversine," 18 Marzo 2006, consultado el 06-06-2023. [En línea]. Disponible en: https://commons.wikimedia.org/wiki/File:Law-of-haversines.png

[3] J. Espiago, "Fundamentos de la georreferenciación," En línea, Universidad Autónoma de Madrid, Julio 2017, consultado el 10-06-2023. [En línea]. Disponible en: http://guiadigital.uam.es/SCUAM/documentacion/fundamentos_georref.php

[4] "Strategic plans and reports," 18 Noviembre 2022, consultado el 14-06-2023. [En línea]. Disponible en: https://worldathletics.org/about-iaaf/documents/strategic-plans-and-reports

[5] GlaivePro, "Iaafpoints," 5 Abril 2023, consultado el 26-05-2023. [En línea]. Disponible en: https://github.com/GlaivePro/IaafPoints

[6] World Athletics, "Scoring tables 2022," 2022, consultado el 26-05-2023. [En línea]. Disponible en: https://worldathletics.org/news/news/scoring-tables-2022

[7] R. Kimball and J. Caserta, *The Data Warehouse ETL Toolkit*. Wiley, 2004.

[8] C. W. Thornthwaite, "An approach toward a rational classification of climate," *Geographical Review*, vol. 38, no. 1, pp. 55–94, 1948. [En línea]. Disponible en: http://www.jstor.org/stable/210739

[9] Codd, *The Relational Model for Database Management*, 2nd ed. Addison Wesley Publishing Company, 1990.

[10] "Scikit-learn haversine distance," Scikit-learn, consultado el 10-06-2023. [En línea]. Disponible en: https://scikitlearn.org/stable/modules/generated/sklearn.metrics.pairwise.haversine_distances.html

[11] B. P. B. G. H. A. Hersbach H., Bell B., "Open-meteo historical weather api," 2023, consultado el 10-06-2023. [En línea]. Disponible en: https://open-meteo.com/en/docs/historical-weather-api