



UNIVERSIDAD DE MÁLAGA



INGENIERÍA INFORMÁTICA

**DISEÑO, DESARROLLO Y PUESTA EN MARCHA DE UNA
BASE DE DATOS PARA EL ESTUDIO Y ANÁLISIS DE
LLUVIAS DE METEOROS**

**DESIGN, DEVELOPMENT AND IMPLEMENTATION
OF A DATABASE FOR THE STUDY AND ANALYSIS
OF METEOR SHOWERS**

Realizado por
JOSÉ IGNACIO GARCÍA ESCOBAR

Tutorizado por
**BEATRIZ BARROS BLANCO
ALBERTO CASTELLÓN SERRANO**

Departamento
**LENGUAJES Y CIENCIAS DE LA COMPUTACIÓN
UNIVERSIDAD DE MÁLAGA**

MÁLAGA, junio de 2023

Abstract

Astronomy studies events such as fireballs, fireballs, and meteors. These phenomena allow astronomers to perform studies such as assigning them to meteor showers. To do so, they perform complex calculations that reports from images and videos taken by several stations that integrate the Network of the University of Malaga and the Malaga Society of Astronomy (SMA/UMA). These reports are generated for several years and are stored in text format.

This TFG is focused on the design and implementation of a database with Web access in which thousands of available reports have been loaded automatically. A remarkable aspect of this development and its results is the opportunity to work on a real problem, with users who have expressed their needs and concerns about the project from the beginning.

The advantages that the project offers its users are remarkable by the use of a database that reorganizes all the information. First of all, the consultation and sharing of data is facilitated both between events of different dates and between institutions within the same context. Another contribution of the work is that, thanks to the public web, a wide community of users will be able to have open access to the data. Thirdly, it provides a tool to perform complex queries, which were impossible in the previous file system. In addition, the use of a database guarantees the maintenance of the data in an orderly and efficient manner. Finally, this work is the basis for other scientific studies that will possibly allow faster progress in this area of research.

Keywords: Meteors, meteor showers, ETL, Database, WWW

Resumen

En Astronomía se estudian bolas de fuego, bólidos, y meteoros. Estos fenómenos permiten a los astrónomos realizar estudios con los que asignación éstos a lluvias de estrellas. Para ello, se realizan complejos cálculos a partir de imágenes y vídeos tomados por varios observatorios. Así la Red de la Universidad de Málaga y de la Sociedad Malagueña de Astronomía (SMA/UMA) disponen de gran cantidad de datos después muchos años de observación del cielo. También disponen de programas informáticos que, a partir de vídeos e imágenes, generan informes que se guardan en formato de texto.

En este TFG se ha realizado en el diseño e implantación de una base de datos con acceso a Web en la que se han cargado miles informes que estaban almacenados en un complejo árbol de directorios como ficheros txt. Un aspecto destacable de este desarrollo y sus resultado es la oportunidad de trabajar en un problema real, con usuarios que han manifestado sus necesidades e inquietudes acerca del proyecto desde el primer momento.

Las ventajas que el resultado de este TFG a sus usuarios son notables gracias al uso de la nueva base de datos. En primer lugar se facilita la consulta y compartición de los datos tanto entre eventos de diferentes fechas, como entre instituciones dentro del mismo contexto. Otra aportación del trabajo es que, gracias a la web y el acceso abierto, se va a dar difusión de toda la información a una amplia comunidad de usuarios. En tercer lugar, se proporciona una herramienta con la que el científico podrá realizar consultas ciertamente complejas, siendo éstas imposibles en el previo sistema de ficheros. Además, el uso de la base de datos garantiza el mantenimiento de los datos de una forma ordenada y eficiente. Por último, destacar que el resultado de este TFG es el punto de partida para otros estudios científicos que posiblemente permitan avanzar en este área de investigación con más rapidez.

Palabras clave: Meteoros, lluvias de meteoros, ETL, Base de Datos, Web

Índice

1. Introducción	7
1.1. Motivación	7
1.2. Objetivos	8
1.3. Tecnologías utilizadas	9
1.3.1. SQL Server y SQL Server Management Studio	9
1.3.2. Oracle SQL Data Modeler	9
1.3.3. Python y pyodbc	10
1.3.4. JavaServer Pages y JDBC Driver	10
1.3.5. Tomcat	10
1.3.6. Eclipse	11
1.3.7. Visual Studio Code	11
1.4. Estructura del documento	11
2. Detección de Meteoros	13
2.1. La Sociedad Malagueña de Astronomía	14
2.2. La Red de Detección de Bólidos y Meteoros de la UMA y la SMA	14
2.3. La red BOOTES	15
2.4. ¿Cómo se detecta un meteoro? De la fecha del evento a la conclusión del astrónomo	15
2.5. Bases de datos de meteoros	17
2.5.1. Bases de datos operativas	17
2.5.2. Beneficios del uso de una Base de Datos de la UMA/SMA para el de análisis y estudio de lluvias de meteoros	19
3. Definición de una Base de Datos para informes de meteoros	21
3.1. Diseño de una Base de Datos para la Detección de Meteoros	21
3.2. Modelo conceptual	21
3.2.1. Lluvias	22
3.2.2. Secciones	22

3.2.3.	Observatorios	22
3.2.4.	Meteoros	22
3.2.5.	Informe Z	23
3.2.6.	Puntos ZWO	23
3.2.7.	Ecuación paramétrica	23
3.2.8.	Trayectoria medida	24
3.2.9.	Trayectoria por regresión	24
3.2.10.	Elementos Orbitales	24
3.2.11.	Informe Radiante	24
3.2.12.	Lluvia Activa InfRad	25
3.2.13.	Trayectoria estimada	25
3.2.14.	Velocidades angular	25
3.2.15.	Informe Fotometría	25
3.2.16.	Estrellas usadas para regresión	26
3.2.17.	Datos meteoro fotometría	26
3.2.18.	Puntos del ajuste	26
3.3.	Modelo lógico	26
4.	Extraccion de datos y Carga de la Base de Datos	31
4.1.	Proceso de carga asíncrona de datos	31
4.1.1.	ETL Observatorios y Calendarios	33
4.1.2.	ETL Informes	37
4.1.3.	Informes Radiantes	40
4.1.4.	Informes Fotometría	41
4.2.	Comprobación de los datos	41
4.3.	Visualización	41
5.	Ciclos de Desarrollo	45
5.1.	Ciclo 1: Analisis de los informes y Carga inicial de datos	46
5.2.	Ciclo 2. Primera implementacion de la Base de DATos y carga de informes . . .	48
5.3.	Ciclo 3: Completando la Base de datos y Desarrollo del acceso por la Web . . .	52
5.4.	Ciclo 4: Despliegue de la Web en un servidor y mejoras	55

5.5. Ciclo 5: Publicación de la página de acceso a la Base de Datos	57
6. Conclusiones	59
6.1. Conclusiones	59
6.2. Líneas Futuras	60
Apéndice A. Manual de uso	67
A.1. Guía de despliegue de la Base de datos	67
A.2. Guía de configuración de los scripts	68
A.3. Guía de uso de los scripts	69

1

Introducción

1.1. Motivación

En Astronomía se estudian bolas de fuego, bólidos, y meteoros [Madiedo Gil(2017)], fenómeno que se produce cuando cuerpos de más o menos masa, en ocasiones procedentes de cometas o asteroides, entran en la atmósfera terrestre a gran velocidad y se desintegran por fricción, lo que produce el rastro luminoso que llamamos meteoro o estrella fugaz [SEA(2022)]. Estas lluvias permiten a los astrónomos realizar estudios que permiten vincularlos a lluvias de estrellas. Para ello, se llevan a cabo cálculos complejos que generan indicadores que se analizan para obtener conclusiones sobre cada fenómeno astronómico. Actualmente, la Universidad de Málaga y la Sociedad Malagueña de Astronomía [SMA(1975)], con el profesor Alberto Castellón Serrano [Castellón(2023)], dispone de un conjunto de epigramas informáticos que realizan esos cálculos complejos [Castellón and Castro-Tirado(2019)] a partir de imágenes y vídeos tomados en varias estaciones que integran la Red de la Universidad de Málaga y de la Sociedad Malagueña de Astronomía. Los resultados se almacenan en informes en formato de texto plano. Actualmente se disponen de gran cantidad de ficheros que se han generado a lo largo de varios años. Su contenido ha ido evolucionado según las necesidades de los astrónomos y estudiosos del tema. Eso hace que:

1. Se dispongan de más de 22.000 informes en formato txt, que tienen gran valor para los investigadores tanto para el estudio de eventos pasados como para predecir o entender eventos presentes y futuros.
2. Los ficheros tienen un formato similar pero no exactamente igual, de forma que es necesario un proceso de limpieza de datos, reestructuración y posiblemente algún mecanismo de inferencia de datos, para unificar su contenido cara a su posterior tratamiento. Ello es debido a que el programa que procesa la lluvia de meteoros ha ido evolucionando a lo

largo del tiempo y el campo de estudio también, de forma que actualmente los informes antiguos están – a veces – incompletos.

3. Los informes están diseminados en múltiples ficheros, organizados por fechas. Es necesario un mecanismo automatizado para localizarlos y posteriormente tratarlos.
4. Actualmente, hay muchos observatorios tomando fotos para detectar lluvias de meteoros. Esto supone que se genere una gran cantidad de información para cada meteoro que hace compleja la comparación y búsqueda de un informe específico, si éstos están en formato txt en ficheros separados.
5. Además de los informes están los observatorios y las lluvias de meteoros que hay que almacenar, gestionar y relacionar con los informes.

En este contexto, este TFG aborda un problema real para una institución real sobre organización y gestión de datos. Para acometer la tarea de mejorar el tratamiento de la información de estos astrónomos, se plantea el desarrollo de un sistema automatizado de lectura y compilación de los informes, y su almacenamiento datos en un gestor de bases de datos, con el fin de poder agilizar y facilitar el acceso a los datos específicos que describen las lluvias de meteoros.

1.2. Objetivos

El objetivo principal de este TFG es organizar y automatizar el proceso de selección, limpieza e inferencia de datos sobre meteoros (cálculo de nuevos datos a partir de los existentes para completar informes pasados), y su carga en una base de datos que habrá que diseñar, crear e implantar desde cero.

Este objetivo puede dividirse en tres subobjetivos principales:

1. Estudiar, analizar y clasificar los informes, según la complejidad de los datos y según el tipo de informe, sencillo (de un único observatorio) o complejo (de dos o de más) y según su temporalidad (fecha y datos que se calculaban en cada fecha).
2. Diseñar una base de datos que recoja toda la información en tablas, organizándola, estableciendo relaciones entre estas y normalizando los datos almacenados

3. Definir procesos automatizados de limpieza, inferencia de datos para cada grupo de informes. Se plantea la posibilidad, si se considera necesario, de incorporar mecanismos para la inferencia de los datos, o – si no fuera posible – el procesamiento de fotos de meteoros adicionales para calcular los datos necesarios que están ausentes en algún informe.
4. Realizar de forma eficiente la carga de datos de todos los informes en la Base de Datos y los mecanismos para automatizar esa carga cada vez que se genera un informe
5. Abordar la instalación de todos los elementos que permitan el acceso público a estos datos mediante una web.

1.3. Tecnologías utilizadas

Para la base de datos se usará: SQL Server [Microsoft(2022)], SQL Server Management Studio [Microsoft(2023)], Oracle SQL Data Modeler [ORACLE(2009)]. Para los scripts se usará: Python [van Rossum(1980)], pyodbc [Kleehammer(2023)] y Visual Studio Code [Microsoft(2015)]. Para la visualización de los datos en la web: JavaServer Pages [Microsystems(1999)], JDBC Driver [Microsoft(2014)], Eclipse [Canadá(2001)] y Apache Tomcat [Foundation(1999)].

1.3.1. SQL Server y SQL Server Management Studio

SQL Server es un sistema de gestión de bases de datos relacional que utiliza SQL como lenguaje principal para manipular los datos. Junto con este, se usará SQL Server Management Studio. Este es un entorno integrado usado para gestionar infraestructuras SQL. En este proyecto, este software nos proporciona herramientas para facilitar la configuración y la administración de la base de datos SQL Server. Con este software, podemos consultar de una forma más sencilla los datos subidos a la base de datos, por lo cual, nos sirve para validar los datos subidos a la misma. También, proporciona asistencia a la hora de crear consultas SQL y detectar errores en la misma, por tanto, nos servirá como apoyo para detectar fallos en las consultas realizadas por los programas realizados.

1.3.2. Oracle SQL Data Modeler

Oracle SQL Data Modeler es una herramienta que nos permite diseñar y modelar bases de datos. Esta, nos permite crear y documentar modelados de bases de datos con una interfaz

gráfica intuitiva. Esta herramienta se ha usado para el diseño del modelado tanto conceptual como lógico de la base de datos.

1.3.3. Python y pyodbc

Python es un lenguaje de programación de alto nivel multiparadigma ya que nos permite tanto programación imperativa como programación orientada a objetos. Este lenguaje destaca por su sintaxis sencilla que nos permite generar un código fácil de leer y comprender. A parte de esto, se ha decidido usar este lenguaje por su amplia biblioteca estandar ya que esta nos proporciona una gran cantidad de funciones que facilitará mucho el desarrollo del proyecto. En específico, el módulo pyodbc [Kleehammer(2023)] se trata de una biblioteca de código abierto que nos permite acceder a las bases de datos mediante el estándar ODBC (Open Database Connectivity). Gracias a esto, podremos conectar todos los scripts a la base de datos de cara a extraer y añadir datos a esta gracias a la posibilidad de enviar consultas SQL. Cabe destacar que es compatible con diferentes sistemas operativos, de esta forma nos aseguramos de que funcione independientemente del sistema en el que se use.

1.3.4. JavaServer Pages y JDBC Driver

Como requeríamos de la creación de una web dinámica para este proyecto TFG, se ha decidido el uso de JSP (JavaServer Pages). Esta tecnología nos permite desarrollar estas web dinámicas permitiendo mezclar código Java y HTML. De esta forma, a partir de consultas Java, podremos extraer datos de la base de datos y mostrarla en la web HTML. Gracias a la inclusión del código Java, se usará el controlador JDBC (Java Database Connectivity), controlador de bases de datos desarrollado por Microsoft, para la conexión de la web con la base de datos. Ya que trabajaremos con una base de datos SQL Server, específicamente se usará Microsoft JDBC Driver for SQL Server.

1.3.5. Tomcat

Apache Tomcat es un servidor web de código abierto usado principalmente como entorno de ejecución para aplicaciones web Java. Al implementar las especificaciones de JavaServer Pages, nos servirá como soporte para la ejecución del mismo. Ya que este es compatible con

una gran variedad de sistemas, se eligió como opción principal para alojar la aplicación web creada.

1.3.6. Eclipse

Eclipse es un IDE (Entorno de Desarrollo Integrado) multiplataforma usado principalmente para el desarrollo de aplicaciones Java. Ya que este nos permite un edito de código con funciones de autocompletado y navegación rápida por el código entre otras funciones, se decidió su uso como IDE para el desarrollo de la web dinámica JSP.

1.3.7. Visual Studio Code

Visual Studio Code es un editor de código fuente de código abierto y multiplataforma que, al contar con una interfaz de usuario limpia, permite una edición de código enfocada. Gracias al gran número de extensiones que tiene, podemos ampliar la funcionalidad el mismo a lenguajes específicos. En este caso se usará para el desarrollo de todo el código Python.

1.4. Estructura del documento

La memoria está organizada en los siguientes apartados:

En el primer apartado, Introducción, se detallan la motivación, los objetivos y las tecnologías utilizadas

En el segundo apartado, Detección de Meteoros, se introduce a los meteoros y las lluvias de estrellas, se describe la Sociedad Malagueña de Astronomía, la red de Detección de Meteoros y bólidos [UMA/SMA(2017)], el proceso de detección y extracción de datos de cada meteoro y las bases de datos actuales encargadas de almacenar datos referentes a la detección de meteoros

En el tercer apartado, Definición de una Base de Datos sobre meteoros, se detallará la base de datos a usar en el proyecto, así como su modelado conceptual conjunto con una descripción de cada una de las tablas aparentes en este y las relaciones presentes entre ellas. Finalmente, se mostrará un modelado lógico sobre la base de datos.

En el cuarto apartado, Exploación y Uso de la Base de Datos, se detallarán todos los procesos ETL usados para la carga de datos en la base de datos, con una explicación en profundidad de cada uno de estos. Después, el proceso de acceso a los datos y las pruebas que se han he-

cho para comprobar que los datos están bien. Finalmente, se explicará la web creada para la visualización de estos datos.

En el quinto apartado, Ciclos de Desarrollo, se detallará la metodología de desarrollo usada en el proyecto, así como todos los ciclos presentes en la metodología y una desarrollo de lo realizado en cada uno de estos.

Como sexto y último apartado, Conclusiones, se detallarán las conclusiones finales del proyecto así como las líneas futuras posibles.

2

DetECCIÓN DE METEOROS

En el campo de la Astronomía, un *meteoro* es un fenómeno luminoso en el cielo nocturno causado por la entrada de un objeto en la atmósfera. Cuando el meteorioide entra en contacto con la atmósfera produce colisiones de alta velocidad de la partícula con los átomos y moléculas atmosféricas. Durante este proceso, la capa mas externa se funde y posteriormente se desintegra. Es este fenómeno en conjunto con los gases atmosféricos, lo que emite la luz en el cielo nocturno. Dentro del conjunto de estos eventos llamados meteoros, existen los llamados *bólidos*. Estos son los meteoros cuyo brillo es superior a una magnitud absoluta de -3 [Ocaña-González(2011)]

Una *lluvia de meteoros* ocurre cuando muchos meteoroides se detectan procedentes de puntos cercanos al mismo tiempo. Estas lluvias ocurren debido a que cuando un cometa se acerca al sol, desprende una gran cantidad de residuos a lo largo de su trayectoria. Cuando la Tierra entra en contacto con la trayectoria que seguía ese cometa, choca contra los residuos dejados y se producen estas lluvias de meteoros. Las lluvias de meteoros se nombran conforme a la constelación de la que parecen provenir. Por ejemplo la lluvia de las Perseidas, la cual ocurre en agosto, parece provenir de la constelación de Perseo [NASA(2018)].

Estos meteoros y bólidos son captados por observatorios y a partir de las imágenes o vídeos de éstos se generan tipos de informes, en los que se profundizará más adelante. Todos estos informes actualmente se almacenan en formato texto. Debido a esto, al ser informes con una extensión considerablemente larga hace que el trabajo con ellos se vuelva muy tedioso.

El uso de una Base de Datos no sólo ayudará a recuperar los informes sino que permitirá relacionar datos, representarlos gráficamente, explorar y comparar datos así como generar conclusiones avanzadas a partir de toda la información almacenada.

2.1. La Sociedad Malagueña de Astronomía

La *Sociedad Malagueña de Astronomía* ([SMA(1975)]) (en adelante SMA) se fundó en el año 1975 y se encuentra entre una de las agrupaciones astronómicas más antiguas y activas de España. Esta abarca campos tales como observación, astrosenderismo, astrobicolage o la historia, aunque sus líneas predominantes son la divulgación de la astronomía, la contaminación lumínica y los meteoritos.

En sus primeros años, la SMA se enfocó principalmente en reuniones regulares para la discusión de temas astronómicos y realizar observaciones nocturnas con telescopios. Tiempo después, la organización amplió su alcance para organizar eventos, charlas, exposiciones y actividades.

La SMA tiene un papel importante en la conservación del patrimonio astronómico de la región, donde se incluyen las exposiciones sobre la historia de la astronomía en la ciudad y la restauración de un antiguo observatorio en la Universidad de Málaga.

En resumen, la SMA se dedica a acercar la Ciencia y la Astronomía al público general y promover la investigación y la observación astronómica.

2.2. La Red de Detección de Bólidos y Meteoros de la UMA y la SMA

La *Red de Detección de Bólidos y Meteoros de la UMA y la SMA* [UMA/SMA(2017)] es un sistema automatizado de cámaras de alta sensibilidad que se utilizan para detectar y registrar la entrada de meteoros y bólidos en la atmósfera terrestre.

“Contamos con una quincena de cámaras que vigilan el cielo 24 horas al día”[Castellón(2020)]. Actualmente constan veintiuna estaciones en funcionamiento y otras siete en proceso de instalación, además de utilizar las cámaras de la *Red BOOTES* (IAA/CSIC).

La información recopilada por las cámaras presentes en Red de Detección de Bólidos y Meteoros de la UMA y la SMA es utilizada para estudiar y comprender mejor los meteoros y bólidos en la atmósfera, así como para rastrear las posibles caídas de estos meteoritos. Esta información se recopila a raíz de informes que son generados por el personal de la SMA. Estos informes generados se almacenan en un sistema de ficheros local que se fundamenta en una carpeta raíz correspondiente al año y dentro de esta carpeta correspondientes a la fecha que, dentro de estas encontraremos carpetas para cada hora donde se generaron uno o

más informes.

2.3. La red BOOTES

La *Red BOOTES* (Observatorio de estallidos y Sistema de exploración de fuentes esporádicas ópticas) [IAA-CSIC(1998)] es la primera red de telescopios robóticos con estaciones en los cinco continentes del mundo. Su principal objetivo es la observación de forma autónoma de lo que se conoce como fuentes transitorias, objetos astrofísicos que no emiten luz de forma constante, sino que emiten luz de forma breve, intensa y repentina [Tejada(2023)]. Las cámaras de todo el cielo que se incorporan en el equipamiento de cada uno de los observatorios de esta red trabajan en conjunto con la Red de Detección de Bólidos y Meteoros de la UMA y la SMA.

2.4. ¿Cómo se detecta un meteoro? De la fecha del evento a la conclusión del astrónomo

Cada vez que un bólido o meteoro es captado por alguna de las cámaras de la Red de Detección de Bólidos y Meteoros de la UMA y la SMA, el sistema registra los datos del evento, como la ubicación, velocidad, dirección y duración del mismo. A raíz de estos datos, se generan diferentes tipos de informes dependiendo de si el evento ha sido captado por una o más estaciones.

Estos datos se almacenan en diferentes tipos de informes dependiendo de diferentes situaciones. Entre los tipos de informes encontramos los siguientes:

1. **Informe-Z:** Informe que se genera cuando un mismo bólido o meteoro es captado por dos o más observatorios. Por cada par de estaciones que lo captan, se generarán dos informes distintos. Estos informes contienen mucha más información que los siguientes por lo que, son los usados para sacar conclusiones y comprobar datos de los mismos.
2. **Informe-Radiante:** Informe que se genera cuando un bólido o meteoro es captado únicamente por un observatorio. Dentro de estos encontramos dos tipos, si no se ha asociado el meteoro a ninguna de las lluvias cercanas activas a esa fecha, se generará un informe muy escueto. En cambio, si este meteoro se ha asociado a alguna de las lluvias,

se generará un informe más extenso. Estos informes, al contener mucha menos información que los anteriores, suelen usarse para calcular regiones desde las que debería haberse visto en otros observatorios, comprobar a mano en estos y generar Informes-Z en caso de que sea visible en alguno.

3. **Informe-Fotometría:** Informe que se genera únicamente cuando existe una foto del meteorito. Estos informes, a partir de todos los puntos extraídos de la trayectoria en esa imagen, son usados para calcular una recta de regresión a partir de la cual se infiere el brillo del bólido y la masa invertida en producir luz.

Todos estos informes se generan y almacenan en archivos de texto plano. Una vez generados por el sistema y almacenados, son los astrónomos encargados de consultar la gran cantidad de datos presentes en cada uno de estos y generar las conclusiones necesarias.

A groso modo, todo el proceso desde la captura del evento hasta el análisis de los datos en los informes, se compone de los siguientes pasos:

1. **Captura de fotogramas:** Cuando un observatorio o más de uno consiguen captar un meteorito, se almacenan todos los fotogramas donde este es visible para poder realizar cálculos sobre su dirección, altura... etc. Si este evento ha sido captado por más de un observatorio, podremos inferir su trayectoria atmosférica [[Castellón and Castro-Tirado\(2019\)](#)].
2. **Cálculo de indicadores:** El cálculo de indicadores diferencia si el meteorito ha sido registrado por una o más estaciones, si este se ha registrado por una única estación “se calcula la circunferencia máxima resultado de la proyección sobre la esfera celeste de su traza y las distancias angulares mínimas entre ella y los distintos radiantes activos a la fecha” [[Castellón\(2015\)](#)] a modo de obtener información acerca de si este meteorito está asociado a una lluvia activa o no. Si este ha sido captado por más de una estación, se generan datos a raíz de un programa que adapta los métodos descritos en [[Dubyago\(1961\)](#)] y [[Lovell\(1954\)](#)].
3. **Generación de informes:** Dados los datos generados por los fotogramas, se pueden generar varios informes. Los informes más completos, “Informes Z” se generarán cuando el evento ha sido captado por más de una estación. En cambio, los “Informes Radiantes” se

generarán cuando el evento ha sido captado únicamente por una estación y sirven para tratar de asociar estos meteoros con lluvias activas a la fecha.

4. **Obtención de conclusiones:** Conjunto a estos datos generados y almacenados en informes en conjunto con los fotogramas de cada evento, los astrónomos se encargan de obtener conclusiones.

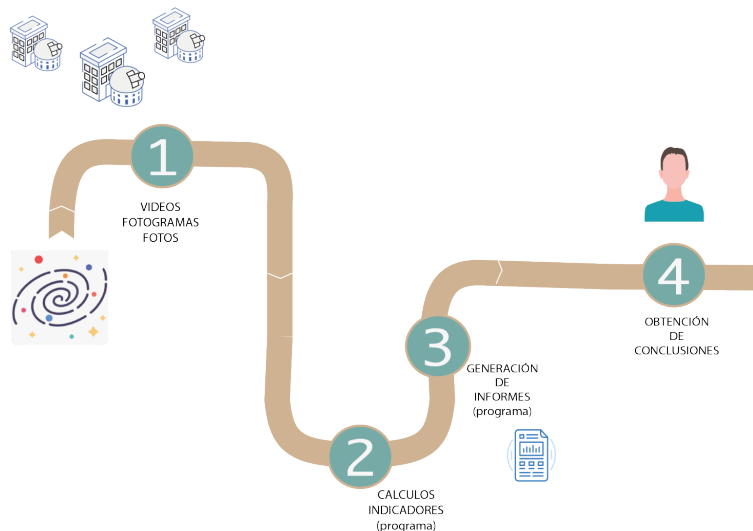


Figura 1: Proceso de detección de Meteoros. Bóolidos y Lluvias de meteoros

2.5. Bases de datos de meteoros

El tema que se plantea en este TFG responde a una necesidad recurrente de los astrónomos, matemáticos y físicos que trabajan la detección de meteoros: gestionar correctamente la cantidad ingente de datos que generan los observatorios.

En esta sección se va a hacer un breve repaso a las bases de datos que ya existen para observar sus características y considerarlas en la que aquí se va a desarrollar. El otro objetivo es hacer el desarrollo considerando la posibilidad de conectar o compartir datos con otras instituciones para estudios e investigaciones científicas.

2.5.1. Bases de datos operativas

Algunas de las Bases de Datos que tienen otras sociedades e instituciones internacionales se resumen en el cuadro 1 De la información encontrada se obtienen las siguientes conclusio-

nes sobre la estructura y características de esas Bases de Datos:

INSTITUCIÓN	CONTENIDO	DIRECCION WEB
NASA	Caídas de meteoritos	https://data.nasa.gov/Space-Science/Meteorite-Landings/gh4g-9sfh
LPI	Avistamientos de meteoros	https://www.lpi.usra.edu/meteor/metbull.php
IMO	Avistamientos de meteoros	https://www.imo.net/members/imo_vmdb
IAU	Lluvias de meteoros y avistamientos de meteoros	https://www.astro.sk/iaumdcDB//home/catalog/photo
FRIPPON	Avistamientos de meteoros	https://fireball.fripon.org/list_pipeline.php
SONACO	Software y avistamientos de sus miembros	https://sonotaco.jp/doc/SNM/

Cuadro 1: Bases de datos de diferentes instituciones internacionales

A continuación se resumen las características de las bases de datos mencionadas:

- NASA La National Aeronautics and Space Administration dispone de una Base de Datos que contiene información sobre los aterrizajes de meteoritos conocidos. Contiene datos de aproximadamente cuarenta y cinco mil setecientos aterrizajes de meteoritos en los que, para cada uno podemos encontrar nombre, id, tipo de meteoro, masa y datos relacionados a la fecha y localización del impacto.
- LPI Lunar and Planetary Institute tiene una base de datos de acceso público con más de setenta mil registros de meteoritos. Para cada registro podemos encontrar nombre, abreviatura, si ha caído o la posibilidad de ello, año, lugar, tipo, masa y en que boletín se ha publicado.
- IMO La International Meteor Organization tiene una base de datos visual de acceso público con alrededor de cuatro millones de meteoros obtenidos con técnicas de visualización estandarizadas. A través de una herramienta presente en su web podemos ver las observaciones de cada año y quién las hizo. Siendo miembro de la organización es posible descargar esta base de datos
- IAU La International Astronomical Union tiene un data center donde se almacenan las lluvias de meteoros y datos almacenados de los meteoros observados por fotografía, videos y radares. En cuanto a las lluvias, se distinguen entre lluvias establecidas y no establecidas. En cuanto a los meteoros poseen una interfaz para consultar datos en la

que podemos seleccionar una gran diversidad de bases de datos y más de 50 parámetros distintos para cada meteoro.

- FRIPON La Fireball Recovery and InterPlanetary Observation Network posee una base de datos donde se almacenan propiedades orbitales y físicas de los meteoros observados por su red. El catálogo de datos ofrece datos orbitales con sus errores, tamaño, masa, entalpía, velocidad, posición, altura, radiante entre otros.
- SONOTACO SonotaCo ofrece una gran cantidad de software usado para registrar datos de visualizaciones de meteoros tales como UFOCapture. También tiene un data set donde almacena las detecciones que se han producido por parte de sus miembros junto con un txt con datos para cada observación. En este podemos encontrar algunos datos referentes a la observación.

2.5.2. Beneficios del uso de una Base de Datos de la UMA/SMA para el de análisis y estudio de lluvias de meteoros

Usando una base de datos como nuevo almacenamiento de los datos procedentes de estas observaciones facilitaríamos el trabajo con estos ya que encontraríamos la información mucho más ordenada que con el previo sistema de ficheros. De esta forma, podríamos realizar consultas complejas a los datos tales como filtrar los informes por cierto valor, que actualmente sería imposible hacer ya que necesitaríamos acceder manualmente a los informes para comprobar este dato. Estos datos se estandarizarían por lo que será posible la compartición de la misma sin la necesidad de enviar informes por parte de la persona que los generó. Al ubicar toda esta información en una base de datos, será posible mostrarla públicamente en una web y crear visualizaciones de estas a través de gráficos de una forma más sencilla.

3

Definición de una Base de Datos para informes de meteoros

3.1. Diseño de una Base de Datos para la Detección de Meteoros

Como se ha mencionado anteriormente, la necesidad de la creación de una base de datos surge de la ingente cantidad de datos presentes en los informes generados por la SMA. Estos datos se encuentran actualmente en archivos de texto plano almacenados en un sistema de ficheros, lo que dificulta enormemente realizar cualquier consulta a estos datos donde queramos filtrar por cualquiera de los datos presentes en estos. Esto también supone una pérdida de almacenamiento ya que no existe forma de comprobar que no existan datos repetidos en directorios distintos. La organización de estos datos en una base de datos relacional nos supondría nuevas formas de filtrar informes como por ejemplo, consultar todos los informes generados por una estación. Ya que en el sistema de ficheros, los informes se guardan en directorios que indican su año, fecha y hora, filtrar los informes conforme a lo mencionado resultaría en un proceso manual que requiere una gran cantidad de tiempo. Todos estos problemas actuales en conjunto con las ventajas que supondría el almacenamiento de estos informes en una base de datos relacional, han concluido en la creación de la misma.

3.2. Modelo conceptual

Como modelo conceptual se ha creado un modelo entidad-relación (Figura 2), usado para representar datos y relaciones entre ellos. Este modelo se ha decidido conforme a los datos extraídos de los informes y la organización que presentan en estos. Como tabla principal desde la que relacionar todo, tendremos la tabla “Meteoros”. A continuación, se detallarán todas las

tablas presentes en este modelo.

3.2.1. Lluvias

En esta tabla se almacenarán los datos de las lluvias de otros los años, para poder relacionar estas con los informes. Como identificadores únicos (UIDs) se usarán el Identificador, que es nombre abreviado que se le da a la lluvia y el año. De esta forma, se podrán almacenar todas las lluvias de cada año. A parte de los UIDs, se almacenará para cada lluvia. Se usan las lluvias oficialmente reconocidas como tales por la International Meteor Organization (IMO).

3.2.2. Secciones

Dentro de cada lluvia, existen secciones de las mismas, a las cuales se les atribuye una fecha, una ascensión recta y una declinación del radiante. En esta, se establece una relación uno a muchos desde la tabla “Lluvia”, de modo que los UIDs de la tabla serán el identificador y el año de la lluvia en conjunto con la fecha de la sección.

3.2.3. Observatorios

En esta tabla se almacenarán los datos de los observatorios presentes en la Red de Detección de Bóolidos y Meteoros de la UMA Y la SMA. Como UID se utilizará el identificador numérico del observatorio. No es un valor creado para la base de datos, este identificador ya existe tanto en los archivos donde aparecen los datos de los observatorios como en los informes para hacer referencia a los observatorios que lo generaron.

3.2.4. Meteoros

Es la tabla principal de la base de datos. En esta se almacenarán los meteoros que se detecten, de este modo se podrán relacionar fácilmente varios informes distintos sobre el mismo meteoro. Como UID se usará un identificador que se generara automáticamente al añadir cada meteoro. En la tabla se almacenarán la fecha y la hora del meteoro. Ya que esta tabla está relacionada con los tres tipos de informes pero no necesariamente para cada meteoro disponemos de los tres tipos de informe, el origen de estas relaciones será opcional.

3.2.5. Informe Z

Es la tabla que hace referencia a los informes mas grandes de la base de datos, los “Informes Z”. Como UID se usará un identificador que se generará automáticamente. Se hará esto debido a que muchos informes carecen de ciertos datos y algunas excepciones carecen de la gran mayoría de los datos del informe. Es debido a esto que los únicos valores obligatorios de la tabla son el identificador, la fecha y la hora. Ya que estos informes son generados por dos observatorios distintos, se crearán dos relaciones uno a muchos desde la tabla “Observatorio”. Ya que en el informe aparece un registro de todas las lluvias activas cuando se observó el meteoro, esto nos crea una relación muchos a muchos con la tabla “Lluvia”. Como tenemos un dato para cada una de las lluvias activas, la “Distancia mínima entre radianes y trayectoria”, crearemos una tabla intermedia “Lluvia activa”. En esta tabla se almacenarán los datos que aparecen en el informe y que no necesitan una tabla propia, para solventar esos casos se crearán tablas adicionales y se relacionarán con esta.

3.2.6. Puntos ZWO

En esta tabla se almacenarán las coordenadas referentes a los Puntos ZWO (Coordenadas en el chip del centroide del trazo dejado por el meteoro) aparentes en el informe. Ya que en cada informe tenemos muchos puntos ZWO, necesitamos agruparlos en una tabla propia. Esta tabla tiene una relación uno a muchos desde la tabla “Informe Z” para poder asociarlos con el informe del que provienen. Como UIDs usaremos X e Y de las coordenadas junto con el Identificador del informe proveniente de la relación.

3.2.7. Ecuación paramétrica

En esta tabla se almacenarán los datos de la ecuación paramétrica de la trayectoria del Informe Z. De cara a evitar que los coeficientes de la ecuación se repitan en la ecuación de diferentes informes, se usará como UID un identificador creado para cada ecuación que entra en la base de datos. A parte de los coeficientes de la ecuación, se guardan los valores de inicio y fin de la trayectoria para cada observatorio del informe.

3.2.8. Trayectoria medida

En esta tabla se almacenan los datos de la trayectoria medida del informe. Ya que la trayectoria contiene varios puntos distintos, se almacenará en una tabla propia. Como proceden de un informe Z, se establece una relación uno a muchos desde la tabla "Informe Z". Como UUIDs se usará la hora y el identificador del informe proveniente de la relación.

3.2.9. Trayectoria por regresión

Bastante similar a la tabla anterior, en esta se almacenarán los datos de la trayectoria calculada por regresión en el informe. Como proviene de un informe Z, se establece una relación uno a muchos desde la tabla "Informe Z". Como UUIDs se usará el valor t referente al tiempo y el identificador del informe proveniente de la relación.

3.2.10. Elementos Orbitales

En esta tabla se almacenará el apartado de elementos orbitales del informe, ya que estos pueden haber sido generados con la velocidad inicial (v_i) o con la velocidad media (v_m) y que algunos informes los generan de ambas formas, estos necesitan ser almacenados en una tabla propia. Como UUIDs se usará el valor "Calculados con", que contendrá los valores referentes a la velocidad inicial o velocidad media, y el identificador del informe proveniente de una relación uno a muchos desde la tabla "Informe Z". Ya que para todos estos datos se almacena el máximo, mínimo, valor y error, en cada dato almacenaremos estos datos separados por un espacio, de modo que quedará (valor máximo mínimo error). Se almacenará de este modo para evitar la creación masiva de estas tablas.

3.2.11. Informe Radiante

Tabla que hace referencia al segundo tipo de informe en la base de datos, los informes radiantes. En este caso volvemos a usar un identificador único generado por la base de datos para cada informe, que se usará como UUID, ya que de nuevo, gran parte de los datos pueden no aparecer en el informe. En este caso solo estos informes se generan asociados únicamente a un observatorio, por lo que establecemos una relación uno a muchos desde observatorio. Es el mismo caso desde meteoro, ya que cada informe se relaciona con un meteoro. En esta tabla

se almacenarán los datos que aparecen en el informe a excepción de los que necesiten tablas propias.

3.2.12. Lluvia Activa InfRad

Esta tabla se usa como intermedia para la relación muchos a muchos entre “Informe Radiante” y “Lluvia”. A diferencia de la tabla previa usada para lo mismo en el anterior tipo de informe, esta almacena mas valores. Ya que es una tabla intermedia para una relación muchos a muchos, los UUIDs serán en conjuntos lo de la tabla “Informe Radiante” con las de “Lluvia”.

3.2.13. Trayectoria estimada

Tabla usada para almacenar los datos de la trayectoria estimada que aparece en los informes radiante. Ya que estos datos tienen formato tabla con varias filas en el informe, es necesario usar una tabla propia para ellos. Como son datos provenientes de un informe radiante, establecemos una relación uno a muchos desde la tabla “Informe Radiante”. Los UUIDs de la tabla serán el dato “Lon Inicio” de la tabla en conjunto con el Identificador del informe proveniente de la relación.

3.2.14. Velocidades angular

Tabla usada para almacenar los datos provenientes de las velocidades angulares de los informes radiantes. Estos datos aparecen en el informe en formato tabla, por lo que para almacenarlos, necesitamos una tabla propia para ellos. Como provienen de los informes radiantes, establecemos una relación uno a muchos desde la tabla “Informe Radiante”. Los UUIDs de la tabla serán el valor “id” el identificador de el informe proveniente de la relación.

3.2.15. Informe Fotometría

Entrando con la última de las tablas mas importantes de la base de datos tenemos la tabla “Informe Fotometría” que hace referencia al último tipo de informes, que tienen el mismo nombre que la tabla. Al igual que las anteriores, esta se relaciona con los meteoros con una relación uno a muchos desde la tabla “Meteoro” ya que cada informe está asociado con un meteoro. A diferencia de los informes anteriores, este no está relacionado con observatorios

ni lluvias en la base de datos. En esta tabla usaremos como UID un identificador generado por la base de datos para cada informe nuevo que entre en ella. Al igual que en los otros dos tipos de informe, almacenamos en la tabla los datos que no necesitan una propia.

3.2.16. Estrellas usadas para regresión

Ya que se trata de un conjunto de datos de los informes que aparece en formato tabla, necesitamos una tabla propia para almacenarlos. Al provenir de los informes radiantes, establecemos una relación uno a muchos desde la tabla “Informe Radiante”. Como UID de la tabla usaremos un identificador propio generado para esta tabla.

3.2.17. Datos meteoro fotometría

Tabla usada para almacenar el apartado de los datos del meteoro en los informes fotometría. Ya que se trata de un apartado de estos informes, establecemos una relación uno a uno desde la tabla “Informe fotometría”. Al tratarse de datos que solo aparecen una vez en el informe, usaremos como único UID el identificador del informe proveniente de la relación. Estos datos se han agrupado en una tabla externa para facilitar su consulta.

3.2.18. Puntos del ajuste

Última tabla propia necesaria para almacenar datos provenientes de los informes fotometría. En este caso se trata de datos que aparecen en formato tabla con una longitud muy extensa. Al provenir de los informes fotometría, establecemos una relación uno a muchos desde la tabla propia de ellos. Como UIDs usaremos el valor t y el identificador del informe proveniente de la relación.

3.3. Modelo lógico

Con el modelo conceptual dado y lo explicado sobre cada tabla previamente, podemos construir un modelo lógico (Figura 3), en este caso un modelo relacional, que nos servirá para diseñar la estructura de la base de datos y dar una vista abstracta de los datos. Pese a que por su amplio tamaño, no termina de apreciarse en la imagen, en este modelo se aclara el tipo

de valor que almacena cada dato en las tablas así como las claves ajenas y relaciones de cada tabla.

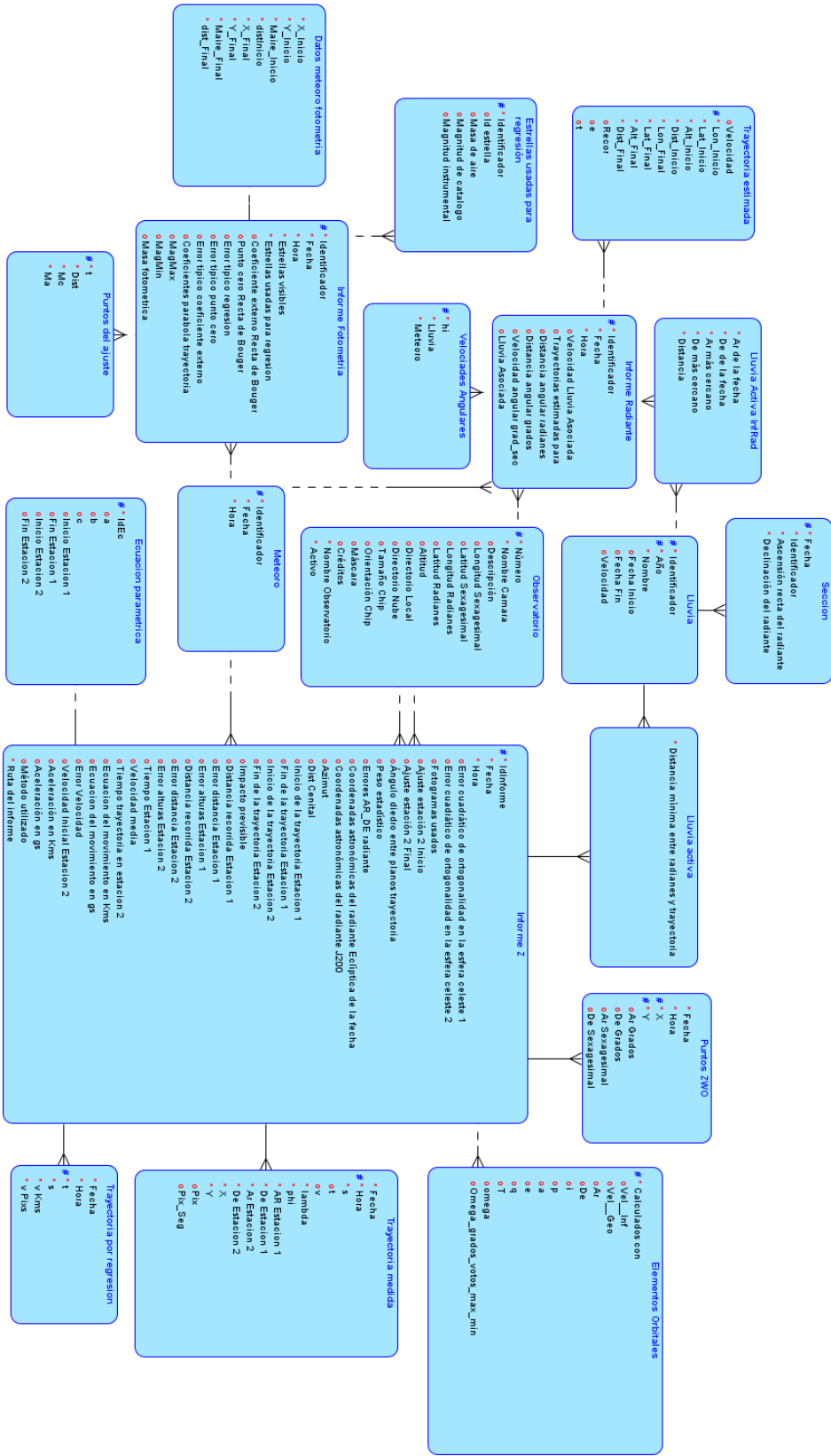


Figura 2: Modelo conceptual de la base de datos

4

Extracción de datos y Carga de la Base de Datos

Los informes generados por la SMA, ya mencionados anteriormente, nos proporcionan una cantidad inmensa de datos pero en un formato de difícil lectura, ya que estos se basan en archivos de texto plano con una larga extensión, complicando la extracción e identificación de estos a los astrónomos.

Para poder facilitar la identificación, uso y compartición de los datos de estos informes, se precisa de un sistema capaz de extraer y normalizar estos datos así como cargarlos a una Base de Datos previamente creada donde se relacionen todos estos datos. Como método de visualización de estos datos extraídos y cargados se necesita de una web encargada de mostrar y filtrar toda esta gran cantidad de datos presentes en los informes.

4.1. Proceso de carga asíncrona de datos

Un proceso ETL (de *Extract, Transform, Load*) es un conjunto de herramientas que permiten la extracción, transformación y carga de datos para su posterior análisis y uso en un sistema de información.

Un ETL consta de tres procesos principales: Extracción, Transformación y Carga (Figura 4).

- **Extracción:** En esta etapa del proceso, se extraen los datos desde una o más fuentes de datos tales como bases de datos, archivos planos, servicios web, entre otros. El objetivo principal de esta etapa es extraer datos importantes de las fuentes dadas para el almacenamiento o análisis posterior.
- **Transformación:** Una vez tenemos los datos extraídos de la etapa anterior, pasamos a un proceso de transformación para limpiar y reorganizar los datos en un formato apto para

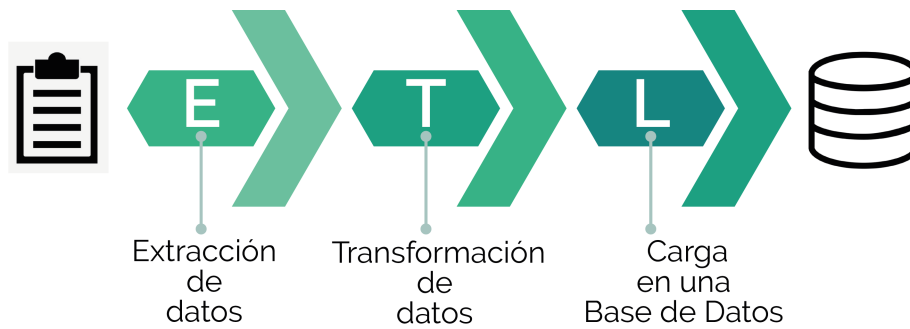


Figura 4: ETL: Extracción de datos de los informes para cargarlos en la Base de Datos (imagen adaptada de freepik)

la etapa posterior.

- Carga:** Una vez que han terminado las etapas anteriores y tenemos los datos extraídos y transformados a un formato apto para su carga, procedemos a la etapa de carga en el destino final. Los datos se insertan en las tablas insertando datos nuevos o en su defecto, actualizando los datos existentes.

Ya que los informes presentes en este proyecto presentan un formato distinto y también a la hora de presentar diferentes datos en cada uno de ellos, se precisará la creación de diversos procesos ETL. Primeramente, uno encargado de cargar las lluvias y observatorios a la Base de Datos. Después, tres distintos que se encarguen de cargar cada uno de los informes independientemente. Para este proceso se precisa de un módulo capaz de conectarse y trabajar directamente con SQL Server. En este caso se usará pyodbc [Kleehammer(2023)], módulo que se descarga desde pip y se carga en Python con “import pyodbc”. Una vez importado, se establece la conexión con la base de datos como se muestra (Figura 5).

Para este proyecto se han creado tres procesos ETL independientes. Uno encargado de procesar las lluvias y los observatorios ya que requerimos estos datos antes de procesar los informes y otros tres scripts donde cada uno se encarga de procesar cada tipo de informe.

Ya que gran parte de estos scripts son distintos para cada informe, se detallará el funcionamiento de cada uno de ellos respectivamente así como sus diferentes etapas.

```
#####Conexion a la BD#####
server = 'server'
database = 'database'
username = 'usuario'
password = 'password'

try:
    cnxn = pyodbc.connect('DRIVER={ODBC Driver 17 for SQL Server};SERVER='+server+';
DATABASE='+database+';UID='+username+';PWD='+ password+';Connection Timeout =15')
    cursor = cnxn.cursor()
#####Conexion a la BD#####
```

Figura 5: Conexión del Script con la Base de Datos

4.1.1. ETL Observatorios y Calendarios

Debido a que los informes están asociados con lluvias y con los observatorios desde los que se han observado el meteoro para el que se genera el informe, necesitaremos los datos de estas lluvias y observatorios antes de poder almacenarlos.

Ya que estos son los dos tipos de datos que necesitamos de inicio cargados en la base de datos, se han agrupado en el mismo script. De esta forma, el script se encargará de preguntar primeramente por el directorio donde se almacenan los archivos referentes a los observatorios y posteriormente, lo mismo para los calendarios (Figura 6).

```
directorio = input("Directorio donde se encuentran los archivos de los
observatorios (No escribir ruta si no se quieren cargar): ")
if directorio != "":
    cargarObservatorio(directorio)

directorio = input("Directorio donde se encuentran los calendarios (No
escribir ruta si no se quieren cargar): ")
if directorio != "":
    with os.scandir(directorio) as itr:
        for entrada in itr:
            if entrada.name[:3] == "cal":
                print("Procesando: " + entrada.name)
                cargarLluvia(entrada.name,directorio)
```

Figura 6: Entrada de datos en el script

Empezando por los observatorios, los datos se extraen de tres archivos de texto plano distinto: “observatorios-astrometria”, “observatorios-meteoros” y “descripcion-observatorios”.

Estos tres observatorios tienen formatos distintos así que se extraerán por orden de cada uno de ellos y finalmente, se juntarán todos para la transformación y carga de estos.

El primero de ellos, “observatorios-astrometría”, contiene los datos referentes al nombre del observatorio, la longitud y latitud en sexagesimal, el tamaño del chip y la orientación. Para poder relacionarlo con el resto de archivos, se establece que el número del observatorio es el número de línea del archivo.

Inta	+37	+06	+14.66	-006	-44	-02.51	+4096	+4096
Mayora	+36	+45	+33.17	-004	-02	-27.51	+4096	+4096
Blemheim	-41	-29	-36.27	+173	+50	+20.72	-4096	+4096
Yunan	+26	+41	+43.00	+100	+01	+47.00	-4096	+4096
Pinillo	+36	+36	+34.47	-004	-31	-11.11	-0765	+0510
ZW0	+36	+57	+09.50	-004	-32	-42.18	+0640	+0480
Sanjorge	+36	+36	+34.47	-004	-31	-11.11	-1530	+1020
ZW02	+36	+57	+09.50	-004	-32	-42.18	-1280	-0960
SanFran	+37	+52	+08.38	-006	-15	-19.15	+1280	+0960
Torcal	+36	+57	+09.50	-004	-32	-42.18	-0640	+0480

Figura 7: Datos en observatorios-astrometría

El archivo “observatorios-meteoros” contiene el número del observatorio, la latitud y longitud (positiva hacia el oeste) en radianes, la altitud en metros y un campo a cero que no se usa. Para poder relacionar estos datos con los anteriores, se usará el número del observatorio. A tener en cuenta que no para todos los observatorios tenemos este tipo de datos, como se puede comprobar en la imagen (Figura 8)

```
0:0.644945204258:0.079326085746:1209:0
1:0.647588233:0.117531001568:50:0
2:0.641569316387:0.070528320968:74:0
5:0.638957622:0.078884567:0
6:0.644945204258:0.079326085746:1209:0
7:0.638957622:0.078884567:0
9:0.660938620567:0.109175920486:390:0
```

Figura 8: Datos en observatorios-meteoros

Por último, el archivo “descripción-observatorios” contiene los datos referentes al número de la cámara, nombre de la cámara, descripción, directorio local, directorio nube, máscara, créditos, nombre del observatorio y un campo indicando si el observatorio está activo o no.

Una vez más, para poder relacionar estos datos con los anteriores, se usará el número del mismo. En este caso, aparecerán siempre todos los observatorios pero no siempre aparecerán todos los datos.

```
1:Inta:CASANDRA-1 antigua a 4096x4096:Inta:Inta:-1:BOOTES1/IAA/CSIC/INTA:BOOTES-1:0
2:Mayora:CASANDRA-2 a 4096x4096:Mayora:Mayora:mascara-Mayora-20211215-3.jpg:BOOTES2/IAA/CSIC/UMA:BOOTES-2:1
3:Blemheim:CASANDRA-3 antigua a 4096x4096:-1:-1:-1:-1:BOOTES-3:0
4:Yunan:CASANDRA-4 a 4096x4096:Yunan:-1:-1:BOOTES4/IAA/CSIC:BOOTES-4:1
5:Pinillo a 765x510:Pinillo:Pinillo:Pinillo:-1:-1:El Pinillo:1
6:ZWO:ZWO120MM Torcal a 640x480:ZWO:ZWO:mascara-ZWO-Torcal.jpg:UMA/SMA/OAT:El Torcal:1
7:Sanjorge:Antigua SBIG SanFrancisco a 1530x1020:SanFrancisco:SanFrancisco:-1:-1:Dehesa San Francisco:1
```

Figura 9: Datos en descripción-observatorios

Para la extracción de estos datos, dividiremos en líneas los archivos y para cada línea dividiremos los datos de forma que cada vez que se encuentre bien sea un espacio para el primero de los archivos o dos puntos para los otros dos archivos, se guardará en una lista el dato (Figura 10). De esta forma, conseguimos extraer todos los datos de estos archivos.

```
with open(ruta + "/" + "descripcion-observatorios.csv", encoding="utf-8") as fname:
    for lineas in fname:
        obsDescripcion.append(lineas.strip('\n'))

for frase in obsDescripcion:
    datosDescripcion.append(frase.split(":"))
```

Figura 10: Extracción de datos de descripción-observatorios

Una vez extraídos los datos, en cuanto a transformaciones necesitamos especificar que los datos presentes en “descripcion-observatorios” aparezcan con valor -1 cuando se quiere indicar que no existe ese dato. De esta forma, cambiamos los datos de este informe cuando aparecen en -1 para que en su lugar en la base de datos y la web aparezca el dato “No tiene”.

Extraídos los datos de los informes, para el proceso de carga preparamos una sentencia SQL que se encargue de comprobar si existe ese observatorio en la base de datos y, en el caso de existir, actualizarlo. En caso contrario, añadirlo como nuevo (Figura 11).

Para el procesado de los calendarios, tenemos los archivos que almacenan el calendario de lluvias de cada año, con nombre calXXXX donde XXXX representa el año. En estos archivos, encontramos primeramente el listado de lluvias donde cada lluvia está separada por un salto de línea y esta aparecen con los datos respectivos a la fecha de inicio, fecha de fin, identificador, nombre, línea donde empiezan sus secciones y la velocidad (Figura 12). Posterior a las

```

exists="IF EXISTS (SELECT * FROM Observatorio WHERE Número = "+str(numero)+"")
update="\n\tUPDATE Observatorio"
set="\n\tSET Nombre_Camara = '"+nombreCamara+"', Descripción = '"+descripcion+'",
where="\n\tWHERE Número = "+str(numero)
insert = "\nELSE\n\tINSERT INTO Observatorio (Número, Nombre_Camara, Descripción,
sentencia = exists+update+set+where+insert
cursor.execute(sentencia)
cursor.commit()

```

Figura 11: Carga de datos de los observatorios

lluvias, tenemos todas las secciones agrupadas de la misma forma que estos primeros datos. Para detectar estas secciones, se aprovechará que previo a cada conjunto de secciones, aparece el identificador de la lluvia a la que pertenecen. Para estas secciones encontramos la fecha, la ascensión de la recta y la declinación.

```

*:01:01:09:15:ANT1:Antihelio:44:30
*:01:01:01:12:QUA1:Cuadrántidas:96:41
*:01:01:02:04:DLM1:Leomenóridas de diciembre:100:64
*:01:10:01:22:GUM:Gamma-Úrsidas-minóridas:109:31
*:01:31:02:20:ACE:Alfa-centáuridas:114:56
*:02:25:03:28:GNO:Gamma-nórmidas:120:56
*:04:14:04:30:LYR:Lyridas:128:49
*:04:15:04:28:PPU:Pi-púpidas:133:18
*:04:19:05:28:ETA:Eta-acuáridas:137:66
*:05:03:05:14:ELY:Eta-lyridas:146:43
*:05:14:06:24:ARI:Ariétidas diurnas:150:38
*:06:22:07:02:JBO:Boótidas de junio:153:18

```

Figura 12: Calendario e lluvias

Para la extracción de datos, al igual que para los observatorios, agruparemos en líneas el archivo y separaremos los datos cada vez que se encuentren dos puntos. Ya que estos datos no necesitan transformación, el proceso de carga se hará similar al de los observatorios. Se comprobará si existe para actualizarlo y en caso de no existir, se añadirá.

Una vez añadidas las lluvias, para procesar las secciones se sigue un proceso totalmente similar al anterior. Una vez detectado el nombre de la lluvia al que pertenece un conjunto de secciones, se extraen todas las líneas hasta el siguiente nombre de lluvia. Ya que los datos vuelven a aparecer separados por dos puntos, el proceso de extracción será totalmente similar al de las lluvias. Para la carga de estos, también se usará un proceso similar.

4.1.2. ETL Informes

Pese a que existen tres tipos de informes distintos a procesar en el proyecto, la mayoría de procesos son similares en los tres ETL. Es decir, para todas las líneas con datos sueltos, el procesamiento es prácticamente idéntico pero adaptado a esa línea. Para todos los datos que aparecen en formato tabla, el procesamiento es idéntico pero adaptado a cada tabla. Por esto, se explicarán los procesos usando como ejemplo los Informes Z y después se contarán las peculiaridades de los demás procesos ETL para cada informe.

La principal complicación en estos informes, a parte de que todos los datos del informe presentan un formato distinto a lo largo del informe, es que algunas veces estos informes paran de generarse en alguna línea del informe y a partir de ese momento, no aparecen más datos en el informe. Para poder controlar esto, almacenaremos todas las líneas del informe en una lista y se hará que el script pare el proceso de extracción desde el momento en que el informe haya parado de generarse, controlaremos en todo momento en que línea nos encontramos y que la siguiente a leer no exceda el tamaño del informe (Figura 13).

```
# Apuntamos la línea por la que vamos, para evitar problemas
actual = 0

# Sacamos el tamaño del informe para saber hasta donde avanzar (por precaución)
tamInforme = len(lineasarchivo)
```

Figura 13: Control de líneas en el Script

Para la extracción de datos en estos informes, a excepción de los que aparecen en formato tabla, se creará una variable correspondiente al dato y el proceso será el siguiente: Se comprueba que la línea a procesar no excede el tamaño del archivo, se extraen los datos de la línea y se aumenta el contador de la línea donde nos encontramos. Si no existen estos datos en el informe, se guarda en la variable un indicador de que no existe el dato (Figura 14).

En cuanto a la extracción para los datos que se presentan en formato tabla en estos informes, como por ejemplo los Puntos ZWO (Figura 15), son extraídos procesando en bucle todas las filas de esta tabla. Los datos de cada fila se almacenarán en un conjunto y este se añadirá a una lista. De esta forma, almacenaremos en una misma lista todos los datos de la tabla. En el caso de estos Puntos ZWO, se procesa como se observa en la figura 16. Para cada fila de

```

if actual < tamInforme:
    # Sacamos los siguientes datos
    anguloDiedro = lineasarchivo[actual][40:]
    pesoEstadistico = lineasarchivo[actual+1][18:]
    # Actualizamos actual
    actual = actual + 3
else:
    anguloDiedro = "NULL"
    pesoEstadistico = "NULL"

```

Figura 14: Ejemplo de Extracción de datos en los Informes Z

la tabla, se separan los datos a partir del carácter veinticinco separándolos en distintos datos usando como separador el espacio entre ellos. Se separan a partir del carácter veinticinco ya que aquí es donde termina el dato referente a la fecha y la hora. Se ignora este dato puesto que la fecha ya la tenemos y la hora se añade al final de cada conjunto de datos posteriormente.

```

Ajuste de puntos ZWO.
Date-Obs X/Y Ar/De (grados) Ar/De (sexagesimal)
2022-11-10T22:00:15.5802 1451.471 468.263 271.21523 42.44326 18:04:51.65 +42:26:35.72
2022-11-10T22:00:15.6802 1453.958 469.197 270.78840 42.41389 18:03:09.22 +42:24:50.00
2022-11-10T22:00:15.7802 1456.402 470.126 270.36734 42.38334 18:01:28.16 +42:23:00.02
2022-11-10T22:00:15.8802 1459.058 471.146 269.90811 42.34823 17:59:37.95 +42:20:53.62
2022-11-10T22:00:15.9802 1461.381 472.049 269.50504 42.31586 17:58:01.21 +42:18:57.11
2022-11-10T22:00:16.0802 1464.235 473.172 269.00759 42.27393 17:56:01.82 +42:16:26.15
2022-11-10T22:00:16.1802 1466.384 474.028 268.63181 42.24079 17:54:31.63 +42:14:26.84
2022-11-10T22:00:16.2802 1468.989 475.076 268.17445 42.19875 17:52:41.87 +42:11:55.50
2022-11-10T22:00:16.3802 1471.622 476.150 267.71012 42.15415 17:50:50.43 +42:09:14.94
2022-11-10T22:00:16.4802 1473.999 477.130 267.28942 42.11207 17:49:09.46 +42:06:43.45
2022-11-10T22:00:16.5802 1476.592 478.214 266.82828 42.06411 17:47:18.79 +42:03:50.80

```

Figura 15: Puntos ZWO en los informes Z

Respecto al proceso de transformación en estos informes, se ha precisado de la creación de identificadores para añadir los datos a las tablas y de la eliminación de espaciados que aparecen en ciertos informes, pero no en todos. De esta forma se ha controlado para algunos de los datos presentes si el dato extraído contenía espaciados de más y eliminarlos. También, agrupaciones de datos que han necesitado reorganizarse para mayor comodidad en la base de datos, tal como es el caso de los elementos orbitales (Figura 17). En este caso, los datos aparecen primeramente mostrando el máximo y el mínimo. Seguidamente en el informe, vuelven a aparecer mostrando el valor y el error. Para mayor comodidad al analizar estos datos, se han agrupado en un mismo

```

ajustesZWO = []
if actual + 3 < tamInforme:
    # Sacamos los ajustes ZWO
    actual = actual + 3
    while lineasarchivo[actual] != "" :
        if lineasarchivo[actual][25] != " ":
            ajuste = lineasarchivo[actual][25:].split(' ') # Añadimos X/Y y Ar/De del ajuste
        else:
            ajuste = lineasarchivo[actual][26:].split(' ') # Añadimos X/Y y Ar/De del ajuste
        ajuste.append(lineasarchivo[actual][11:][:2]) # Añadimos la hora
        ajuste.append(lineasarchivo[actual][14:][:2]) # Añadimos el minuto
        ajuste.append(lineasarchivo[actual][17:][:7]) # Añadimos el segundo
        ajustesZWO.append(ajuste)
        actual = actual+1

    # Actualizamos actual
    actual = actual + 1

```

Figura 16: Extracción de los Puntos ZWO

conjunto, de modo que para cada elemento orbital tendremos: máximo, mínimo, valor, error.

```

----- ELEMENTOS ORBITALES -----
                Máximo / Mínimo
Vel. inf. (Km/s): 14.6013317769306 14.6009149082723
Vel. Geo. (Km/s): 9.25052900399905 8.63959737164338
Radiante geocéntrico (grados):
Ar:                331.341801 330.726993
De:                22.811129 22.596396
i:                 7.52882529609992 7.12024640144561
p:                 89.2850162499958 87.001633728486
a:                 2.40532250787377 2.18060093335671
e:                 0.59341200742871 0.551043005086821
q:                 0.979053812635776 0.977913869154886
T:                 19.6709019295664 19.160699021453
Omega:             228.491883184829 228.491883184829
omega:             194.834336162185 194.460263039634

Elemento / Valor / Error
Vi: 14.51 0.190765013388941
Vg: 8.95 0.305465816177835
ar: 331.03 0.307403999999963

```

Figura 17: Elementos Orbitales en un Informe Z

Para el proceso de carga de los datos, primeramente se comprobará que no existe un informe de ese tipo en esa misma fecha, hora, minuto, segundo y décima de segundo. En caso de no existir, se le asignará un nuevo identificador siendo este el siguiente identificador al último informe añadido a la base de datos. Una vez comprobado esto, en el caso de ser un informe nuevo, se creará una sentencia SQL para cada una de las tablas a rellenar en dicho

informe. Ya que existen diversas relaciones entre ellas, primero se ejecutarán las sentencias que no precisen de datos presentes en el informe introducidos previamente a la base de datos.

Cabe destacar que en todos los informes estará presente el código que se encarga de comprobar si el meteoro para el que se generó el informe a procesar, ya existe en la base de datos. En caso de existir, asignarle el identificador del mismo a dicho informe y si no existe, crear un nuevo meteoro en la base de datos con los datos de este informe. Puesto que este requisito, el asignar meteoros a los informes y comprobar si con una diferencia de dos segundos, existe otro similar, se solicitó en una fase avanzada del proyecto, se detallará en los ciclos de desarrollo (Capítulo 5).

4.1.3. Informes Radiantes

En específico, para los informes radiantes, necesitamos comprobar los tres tipos de informes que existen de este estilo. Estos tres tipos son: no asociados a una lluvia, asociados a una lluvia con tiempo desconocido, asociado a una lluvia con tiempo conocido. Empezando con los no asociados a una lluvia, podemos detectarlos ya que en el mismo informe existe una línea que nos indica si ha sido asociado a una lluvia o no. Una vez detectado que estamos procesando este tipo de informe, pararemos de leer el informe, y lo subiremos a la base de datos. Esto se debe a que la línea donde se indica que no se ha asociado a una lluvia es la última línea de estos informes (Figura 18).

```
if lineasarchivo[actual][:2] == "No":
    lluviaAsociada = "Ninguna"

cursor.execute("SELECT COUNT(*) FROM Informe_Radiante")
for i in cursor:
    idInf = i[0] + 1
insert = "INSERT INTO Informe_Radiante (Identificador, Fecha, Hora, Velocidad_Lluvia_Asociada,
cursor.execute(insert)
cursor.commit()
```

Figura 18: Elementos Orbitales en un Informe Z

Para detectar los otros dos tipos de informes nos serviremos de esta misma línea ya que se indicará si se ha asociado a una lluvia con cierta velocidad y si se ha asociado con un rango de alturas. Ya que dependiendo del tipo, cambian los datos restantes en el informe, estos se marcarán como “null” en el tipo donde dichos datos no aparezcan. Los procesos de

transformación y de carga serán similares a los explicados previamente.

4.1.4. Informes Fotometría

Ya que estos informes no tienen particularidades más allá de diferencias a la hora de especificar cada dato, se usarán procesos similares a los anteriores para los datos sueltos y los datos que aparecen en formato tabla. También serán similares las transformaciones y carga posteriores.

4.2. Comprobación de los datos

Para facilitar el acceso a los datos cargados en la base de datos se ha hecho uso del programa SQL Server Management Studio [Microsoft(2023)]. Este programa nos permite acceder, modificar y configurar a las bases de datos SQL Server mediante una interfaz, por lo que la consulta de tablas y datos cargados en ellas se facilita. Para cada informe, como primer método de análisis de los datos, se seleccionaron al azar informes de todo el conjunto ofrecido por el personal de la SMA y se almacenaron en un directorio específico. Indicando al script que se carguen únicamente estos informes seleccionados, facilitamos la comprobación. Una vez se comprueba que para estos informes, se cargan los datos correctamente, se aplica el script a todo el conjunto de informes restante y, cada vez que el script falla en un informe específico, se localiza este y se procesa por separado. De esta forma se detectan y solucionan fácilmente todos los fallos encontrados.

4.3. Visualización

Como método de visualización de los datos procesados en la base de datos, se decidió la creación de una web dinámica donde poder acceder a los datos de una forma sencilla y directa. Esta página web se desarrollará con Java Server Pages ya que nos permite crear una web con las características indicadas, capaz de consultar datos a una base de datos y mostrarla en una web HTML.

Puesto que la base de datos usa como tabla desde la cual relacionar todo el contenido, la tabla meteoros, se decidió que esta sería la principal desde la cual se desarrollaría la web. Primeramente se mostrarán todos los meteoros y, al pulsar en la hora de cada uno, redireccionará

a otra página donde poder consultar los informes generados para ese meteoro. Ya que la cantidad de meteoros es muy alta, se ha creado un filtro para poder buscar de forma cómoda lo requerido. Este filtro nos permitirá buscar entre una fecha de inicio y fin y qué tipo de informes existen relacionados a este meteoro (Figura 19).

En la web se ha creado un header donde aparece el logo de la SMA a la izquierda, que nos redireccionará a su página web en el caso de pulsarlo. A la derecha de este encontramos un menú con tres botones: Meteoros, Observatorios, Lluvias (Figura 19). El primer botón nos redirecciona a la web donde se muestran los meteoros, mencionada previamente. Los botones Observatorios y Lluvias nos llevan a una página específica cada uno donde se nos muestran los datos correspondientes al nombre del botón.



Fecha	Hora
2021-06-02	02:12:37
2021-06-04	02:28:39
2021-06-07	20:34:50

Figura 19: Encabezado de la web

Volviendo con la página de los meteoros, si pulsamos en la hora de alguno de estos, nos llevará a otra página donde saldrán listados los diferentes informes generados para dicho meteoro, ordenados por tipo (Figura 20). Para cada uno de los informes presentes aquí, se muestran ciertos datos del informe. Para consultar el resto de datos de cada informe, se pulsará en el identificador del mismo.



<u>Informe de dos estaciones</u>				
ID	Fecha	Hora	Observatorio 1	Observatorio 2
1078	2022-10-14	18:46:47	17	54

<u>Informes de una estación</u>								
ID	Fecha	Hora	Observatorio	Lluvia Asociada	Trayectorias estimadas para	Distancia angular (Radianes)	Distancia angular (Grados)	Velocidad angular (Grados/s)
660	2022-10-14	18:46:47	17	Ninguna	No medido	null	null	459

<u>Informe de fotometría</u>				
ID	Fecha	Hora	Estrellas visibles	Estrellas usadas en regresión
17	2022-10-14	18:46:47	37	18

Figura 20: Informes en la web

Aquí, se puede pulsar en el identificador del informe requerido para consultar el resto de datos del mismo. No se enseñan todos los datos en la página anterior debido al extenso número de datos de cada uno, esto provocaría que la web quedara muy extensa a nivel vertical y por tanto, poco práctica. Para cada dato presente en las tablas de estos informes, se puede consultar una descripción del dato en cuestión. Cabe destacar que tanto qué datos mostrar en cada apartado y las descripciones han sido corregidas y aprobadas por el personal de la SMA.

5

Ciclos de Desarrollo

Este proyecto se ha desarrollado siguiendo una metodología ágil, similar a Scrum, en varios ciclos que han sido probados e valuados por los tutores. Se desarrollaron varias versiones de la Base de Datos y del programa Web que permite visualizar su contenido, en cinco ciclos (Figure 21).

Se eligió esta metodología ya que no todos los requisitos estarían claros desde un comienzo y se necesitaría de diversas reuniones con el cliente. “La intención de Scrum es la de maximizar la realimentación sobre el desarrollo pudiendo corregir problemas y mitigar riesgos de forma temprana”. [Calderón and Valverde-Rebaza(2007)]

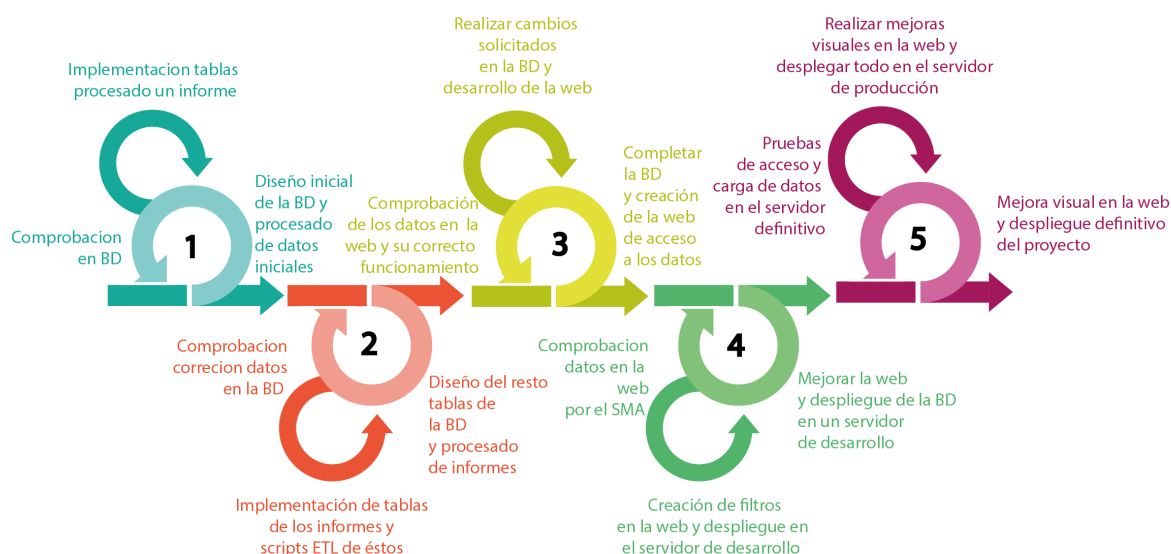


Figura 21: Ciclos de desarrollo del proyecto

Los ciclos se van a describir textualmente diciendo cual era su objetivo, qué elementos se consideraron para su diseño, luego qué se desarrollo en ese ciclo y los elementos que se probaron (y errores que aparecieron para ser resueltos en al fase siguiente). Gráficamente los

ciclos se representan con tres elementos: Diseño, Implementación y prueba (Figura 22),

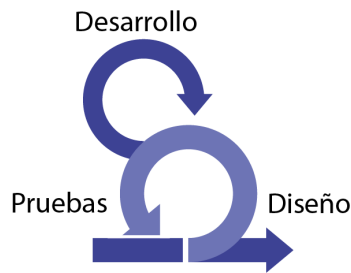


Figura 22: Esquema de un ciclo de desarrollo en este proyecto

5.1. Ciclo 1: Analisis de los informes y Carga inicial de datos

Los objetivos de este primer ciclo (Figura 23) fueron entender qué son las lluvias de meteoros registradas por la SMA, qué son los informes que se iban a manejar en este proyecto y entender los elementos relacionados con estos. Una vez entendido esto, preparar una primera versión de la base de datos y un script ETL que se encargue de procesar los datos necesarios antes de trabajar con los informes.

Con fin de poder comprender las lluvias de meteoros y qué son los informes, se realizaron diversas reuniones con el personal del SMA donde primeramente nos introdujeron a su trabajo. En la primera reunión que se realizó, nos enseñó su trabajo tanto a nivel visual como a nivel de informes. Una vez aclarado el contexto del trabajo, nos enseñó uno a uno cada uno de los tipos de informe que generan los observatorios pero sin llegar a profundizar demasiado en ellos, ya que esto se haría más tarde. Con esto se analizó que previamente a trabajar con los informes, se necesitarían procesar las lluvias y los observatorios en la base de datos.

Tras la primera reunión, el personal de la SMA envió por correo ejemplos de cada uno de los tres tipos de informes existentes a modo de poder analizarlos y entenderlos plenamente. Al estar acompañados en el correo con una breve explicación sobre qué era cada informe, facilitó la comprensión y análisis de estos. Los informes se organizan desde una carpeta raíz que es la carpeta correspondiente al año, dentro de cada una de estas aparece una carpeta por cada meteorito donde se indica la fecha pero sin especificar la hora. Una vez dentro de la carpeta referente a la fecha, encontraremos todos los Informes Radiantes referentes a esa fecha y una carpeta referente a la hora de cada meteorito detectado si existen Informes Z o

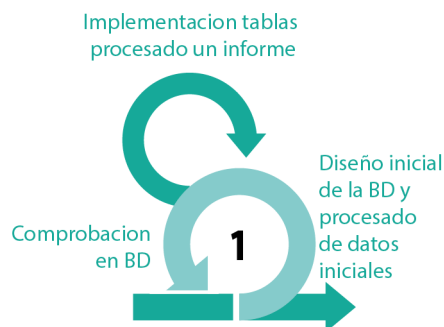


Figura 23: Esquema del primer ciclo

Informes Fotometría. Dentro de esta se encontrará una carpeta “Informe Fotometría” si existe un informe de este tipo y una carpeta para cada Informe Z generado en esa hora. Dentro de cada carpeta referente a un Informe Z, encontremos una carpeta con nombre “vm”, que contendrá un informe gemelo con los elementos orbitales generados con la velocidad media en caso de que este informe haya sido generado.

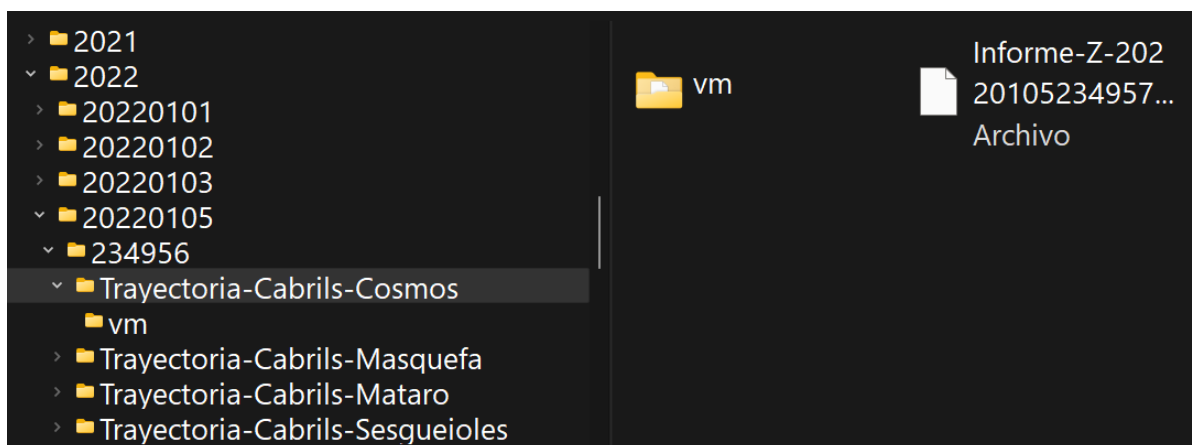


Figura 24: Árbol de directorios de los Informes Z

Tras esto, se realizó la siguiente reunión, última de este ciclo, donde una vez resueltas las primeras dudas que aparecieron al analizar estos ejemplos de informes enviados por él, nos explicaron la parte previa necesaria a trabajar con los informes, los observatorios y los calendarios. Empezando por los calendarios, nos explicó que estos son archivos de texto plano que se generan para cada año donde aparecen listadas todas las lluvias del año junto con su fecha de comienzo y finalización, su nombre, su identificador y su velocidad. Posteriormente en el archivo aparecen las secciones de cada lluvia, que son fechas destacables dentro de las

fechas de cada lluvia.

Respecto a los observatorios, los datos referentes a estos están repartidos en tres archivos de texto plano distintos, juntando y agrupando correctamente los datos que aparecen en estos tendremos los datos referentes a cada observatorio de la red.

Empezando con la creación de la base de datos, se añade a la base de datos una tabla para las lluvias, donde se añadirán los datos de las lluvias a excepción de lo que aparece en las secciones. Una vez finalizada esta tabla, se creó la tabla secciones que se relacionará con la tabla de las lluvias de forma que cada lluvia se relacione con muchas secciones. Finalizado esto, se creó la tabla de los observatorios, donde en una única tabla se añaden los datos de los tres archivos de los observatorios.

Con este apartado de la base de datos creado, se dió comienzo al desarrollo del script ETL que se encarga de procesar los datos de los calendarios de lluvias y los observatorios.

Se desarrolló primeramente el script donde únicamente se cargaría un calendario dado en la base de datos. Una vez desarrollado este script, se añadió la funcionalidad de leer los calendarios en un directorio dado, de forma que ya no haría falta procesar cada calendario uno a uno. Finalizado esto, se añadió el procesamiento de los observatorios al script, de forma que ahora el script solicitará el directorio donde se encuentran los archivos referentes a los observatorios y una vez procesados, se solicitará el directorio de los calendarios.

Finalizado el script, se realizaron pruebas para poder comprobar el correcto funcionamiento de este y de la base de datos. Una vez analizado el correcto funcionamiento de esto, se dió por finalizado el script “cargaInicial.py”.

5.2. Ciclo 2. Primera implementacion de la Base de DATos y carga de informes

Con los datos necesarios para el procesamiento de los informes presentes en la base de datos, se inició el ciclo (Figura 25) y estableció como objetivo entender los tipos de informes y sus variantes así como sus datos, creación de sus apartados en la base de datos y los scripts que se encargarán de procesarlos.

Para proceder con los objetivos, se establecerán una reunión con el personal de la SMA por cada uno de los tipos de informes a modo de informarnos completamente de estos antes



Figura 25: Esquema del segundo ciclo

de añadirlos al sistema, también se solicitarán informes para un periodo de tiempo en estas reuniones, a modo de poder realizar más pruebas. Realizada la reunión se procederá realizando el apartado en la base de datos para cada tipo de informe y realizando su correspondiente script.

La primera reunión que se realizó con el personal de la SMA fue para informarnos sobre el primer tipo de informes, los Informes Z. Estos informes se generan cuando un meteoro o bólido es detectado por más de un observatorio, de esta forma se generará un informe para cada par de estaciones presente en la observación. Estos informes cuentan con una cantidad masiva de datos, superando la mayoría las ciento cincuenta líneas de informe. Aunque varios apartados grandes como son los “Puntos ZWO”, “Trayectoria medida” y “Trayectoria” tienen formato tabla, facilitando su lectura, gran parte del informe son líneas con distintos formatos donde algunas contienen incluso varios datos (Figura 26).

```
2022-01-15T22:47:03.2171
Longitud/Latitud (geográficas) estación 55: 2.385500 41.519806
Longitud/Latitud (geográficas) estación 18: 0.744033 42.017628
Coordenadas rectangulares estación 55:
xi: -790.184738581818 eta: 4716.79228169877 zeta: 4205.82143061886
Coordenadas rectangulares estación 18:
xi: -649.78027061117 eta: 4701.55005716533 zeta: 4247.55149647418
Error cuadrático de ortogonalidad en la esfera celeste (en radianes): 1.35644170898327e-06
```

Figura 26: Parte inicial de un Informe Z

Una vez entendido el formato de estos informes, se apuntaron y agruparon los datos que aparecen en ellos y se procedió a la creación de tablas en la base de datos. Se decidió que la tabla principal sería “Informe Z”. Esta tabla tendrá una doble relación con los observatorios ya

que se detecta por dos estaciones a la vez. A parte, se crearon tablas propias a los datos que tienen formato tabla en el informe, mencionados anteriormente. También, se agruparon datos en tablas externas de cara a disminuir el tamaño de la tabla principal.

Creado el apartado de la base de datos, se procedió con la creación del script ETL que se encarga de procesar los Informes Z. Siendo el primer y más largo informe de los tres, fue el que, con diferencia, más tiempo requirió para la creación del script.

Tras la realización de pruebas, se ajustó el funcionamiento del script para detectar informes ya presentes en la base de datos. También, debido al amplio número de informes, existen casos específicos donde falta un dato, hay más espaciados o menos de lo normal o falta un conjunto de líneas. Por esto, se tuvo que ajustar de nuevo el funcionamiento del script para funcionar correctamente en estos casos y se finalizaría el desarrollo del script “CargaInformesZ.py”.

Tras el desarrollo de estos informes, se planificó otra reunión con el personal de la SMA para informarnos sobre los otros dos scripts presentes en el sistema, los “Informes Radiantes” y “Informes Fotometría”. Siendo estos mucho más concisos que los anteriores, no se requirió de una reunión específica para cada uno de ellos. Tras la reunión y, al igual que con los informes anteriores, solicitar un conjunto de informes de un intervalo de fechas para poder comprobar los fallos, se empezó con los “Informes Radiantes”. Estos informes a diferencia de los anteriores, están asociados a un único observatorio y además, encontramos tres tipos diferentes de estos mismos.

La principal diferencia en ellos es si el informe generado ha sido asociado a una lluvia (Figura 28) o si no ha sido asociado a ninguna (Figura 27). Los informes no asociados a una lluvia tienen una extensión menor que los asociados a una. Dentro del conjunto de los que son asociados a una, podemos encontrar dos tipos, los que han sido asociados con un tiempo conocido y otros en los que no se conoce el tiempo. La diferencia en estos será que en los informes donde conocemos el tiempo, aparecerá el dato “Recor.”(distancia recorrida). Si se desconoce el tiempo, en lugar de este dato aparecerán “e”(distancia recorrida estimada) y “t”(tiempo estimado).

Una vez comprendido esto, se actualizó la base de datos con las tablas referentes a estos informes. Ya que estos informes son generados para un solo observatorio. Se indicó que se estableciera una una relación con la tabla de los observatorios para indicarlo. Al igual que para los informes anteriores, se decidió crear una tabla principal, “Informes Radiantes” donde se almacenarán la mayoría de los datos. Los datos presentes en el informe con formato tabla se

```

Distancias mínimas (en grados) entre radiantes y trayectoria:
Lluvia| Ar/De a la fecha | Ar/de más cercano tray. | Distancia
=====
ANT1 | 123.00 | 18.60 | 79.591999 | 21.736074 | 40.738
QUA1 | 234.60 | 47.80 | 264.969598 | 50.060644 | 19.940
DLM1 | 181.00 | 20.60 | 267.030733 | 60.983998 | 70.177
GUM | 220.80 | 70.60 | 273.072944 | 74.353670 | 15.616

No asociado a ninguna lluvia.

```

Figura 27: Informe Radiante no asociado a una lluvia

```

Distancias mínimas (en grados) entre radiantes y trayectoria:
Lluvia| Ar/De a la fecha | Ar/de más cercano tray. | Distancia
=====
ANT1 | 124.00 | 18.20 | 115.451680 | 24.161901 | 9.949
QUA1 | 235.20 | 47.60 | 235.795380 | 48.114369 | 0.651
DLM1 | 182.00 | 20.20 | 173.084131 | 57.227112 | 37.607
GUM | 221.60 | 70.20 | 209.230387 | 56.689523 | 14.532

Se ha asociado a las Cuadrántidas con velocidad (catálogo IMO) 41 Km/s.

Trayectorias estimadas para un tiempo de 0.400000095367432 segundos:
v | Inicio | Final | Recor.
  | Lon   | Lat   | Alt  | dist. | Lon   | Lat   | Alt  | dist. |
=====
38.0 | 00:49:56.78 | 42:04:17.04 | 16.65 | 18.51 | 00:56:15.54 | 42:06:57.61 | 28.05 | 33.53 | 15.20
41.0 | 00:50:21.88 | 42:04:32.27 | 17.91 | 19.98 | 00:57:10.40 | 42:07:25.39 | 30.21 | 36.17 | 16.40
44.0 | 00:50:46.97 | 42:04:47.50 | 19.16 | 21.44 | 00:58:05.27 | 42:07:53.15 | 32.37 | 38.82 | 17.60

```

Figura 28: Informe Radiante asociado a una lluvia

almacenarán en una tabla propia y se relacionará con esta principal. Con eso la base de datos está lista para poder almacenar los datos de estos informes.

Actualizada la base de datos, se procedió con el desarrollo del script ETL para el procesamiento de estos informes. Siendo este el segundo informe y teniendo estos una extensión más corta que los anteriores, el desarrollo fue ciertamente más rápido que los anteriores.

Tras crear este script, se sometió a pruebas con el conjunto de informes que teníamos actualmente. Una vez analizado el funcionamiento y comprobado que los datos extraídos eran los correctos en la base de datos, se dió por finalizado el desarrollo del script “CargaInformes-Rad.py”.

El último tipo de informes del sistema son los “Informes Fotometría”. Estos informes son los mas escasos ya que requieren de unas imágenes específicas para poder generarse y estas imágenes se captan un número limitado de veces. Tal es la escasez de estos informes, que a lo

largo del año 2022 solo se generaron 14 informes. En comparación a los anteriores informes, estos son extensos pero gran parte de la extensión son datos en formato tabla, por lo que la extracción de los estos es más directa que en los otros dos tipos de informe.

Una vez más, se añadieron a la base de datos las tablas referentes a estos informes. Siendo la tabla principal la “Informes Fotometría”, en este caso no tiene relación alguna con los observatorios, ya que estos informes no se asocian con ninguno. Al igual que para los otros tipos de informe, se crearán tablas para los datos que, en el informe, tienen formato de tabla y no pueden almacenarse en la tabla principal. Una vez añadidas estas tablas, la base de datos está lista para almacenar la información..

Con la base de datos terminada, comenzó el desarrollo del script ETL para estos informes. Ya que estos informes no presentan diferencia ninguna entre ellos a excepción de la extensión de las tablas, se finalizó rápidamente el script. Con esto, se procedió a las pruebas con los informes dados aunque, al haber tan pocos de estos informes, no se pudieron realizar pruebas exhaustivas. Comprobado el funcionamiento, se finalizó el desarrollo del script “CargaInformesFot.py”.

5.3. Ciclo 3: Completando la Base de datos y Desarrollo del acceso por la Web



Figura 29: Esquema del tercer ciclo

Como objetivos de este tercer ciclo (Figura 29) se establecieron comunicar y mostrar al personal del SMA los avances en el trabajo, tanto la base de datos como los scripts. Con estos finalizados, comenzar el desarrollo de la web desde donde se podrá acceder de forma pública a los datos de los informes cargados en la base de datos.

```

# Comprobamos si existe un meteoro en la base de datos a esta fecha y hora, en caso contrario, lo añadimos
cursor.execute("SELECT * FROM Meteoro")
insertar = True
for i in cursor:
    fechaBien = str(i[1])[8:][:2] + "-" + str(i[1])[5:][:2] + "-" + str(i[1])[4]
    if fecha == fechaBien:
        if hora[:5] == i[2][:5]:
            if float(i[2][6:]) < float(hora[6:]) and float(i[2][6:])+2 > float(hora[6:]):
                idM = i[0]
                insertar = False
            elif float(i[2][6:]) == float(hora[6:]):
                idM = i[0]
                insertar = False
            elif float(i[2][6:]) > float(hora[6:]) and float(i[2][6:])-2 < float(hora[6:]):
                idM = i[0]
                insertar = False

if insertar:
    cursor.execute("SELECT * FROM Meteoro")
    idM = 1
    for i in cursor:
        idM = i[0] + 1
    insert = "INSERT INTO Meteoro (Identificador, Fecha, Hora) values ('+str(idM)+'','"+fecha+"','"+hora+"'"
    cursor.execute(insert)
    cursor.commit()

```

Figura 30: Código utilizado para controlar los meteoros en la Base de Datos

Se organizó una reunión con el personal de la SMA para hablar sobre los avances en el proyecto. Una vez en ella, se nos comunicó la necesidad de añadir cambios en la base de datos. Estos cambios son añadir una tabla donde se almacenarán los meteoros captados por la red y todos los informes se relacionarán con estos. De este modo, podremos encontrar los informes relacionados a un meteoro en una fecha y hora dada. Ya que no todos los informes se generan exactamente en la misma fracción de segundo, se nos solicitó comprobar y ajustar esto también. De este modo si un informe ha sido generado como máximo dos segundos antes o dos segundos después de otro ya presente en la base de datos, se asociará con este meteoro.

Planteado esto, se añadió a la base de datos la tabla "Meteoro". También una relación uno a muchos con cada una de las tablas principales de los informes. Una vez la tabla de datos fue actualizada, se añadió para los tres scripts encargados de procesar los informes un mismo fragmento de código (Figura 30) encargado de comprobar si existe un meteoro acorde a la fecha y hora dada y asociar a ese informe con dicho meteoro. En caso de no existir dicho meteoro, añadirlo a la base de datos.

Con estos cambios realizados, se empezó a diseñar la web para la consulta de los datos. Con objetivo de poder consultar la base de datos y mostrarlos en la web, se decidió el uso de Java-Server Pages (JSP) [Microsystems(1999)] como tecnología a usar. Esta es una tecnología que

nos permite añadir contenido dinámico a las páginas HTML antes de enviarlas al navegador.

Esta web donde mostrar los datos se basará principalmente en poder seleccionar un meteoro de una lista dada, para seguidamente mostrar los informes que se generaron asociados a dicho meteoro. De estos informes se mostrarán los datos principales de cada uno, pudiendo pulsar en estos para acceder a otra web donde se muestre el resto de datos del informe. A parte, una web donde poder consultar los observatorios presentes en la red y una última donde consultar los calendarios de lluvias de la base de datos.

Como diseño de inicial de la web, se decidió usar los colores y un formato similar al que aparece en la página web de la SMA [SMA(1975)]. Decidido esto, se dio inicio al desarrollo de la web. Ya que primeramente se usará la web en local, no se necesitará de un servidor para poder comprobar los fallos en la web. Ya que estamos usando JSP, necesitaremos un servidor Tomcat donde poder compilar y ejecutar la aplicación. Tras decidir esto se desarrolló el diseño inicial de la web.



Figura 31: Diseño inicial de la Web

Tras comprobar el funcionamiento de la web y corregir los fallos presentes en ella, se dio por finalizado el diseño inicial de la misma. Como principal problema tenemos no poder filtrar los datos teniendo una lista tan inmensa de meteoros, estos fallos se arreglarán en el siguiente ciclo.

5.4. Ciclo 4: Despliegue de la Web en un servidor y mejoras

Con este primer diseño de la web, se decidió que previamente a realizar mejoras en ella, se realizaría un despliegue tanto de esta como de la base de datos para comprobar el funcionamiento de ambos en la nube a modo de arreglar y comprobar todo previo al lanzamiento definitivo de la misma (Figura 32). Como servidor auxiliar para estas pruebas se usó una máquina virtual con Debian donde usamos una VPN para trabajar con ella. Este servidor fue cedido por el laboratorio de IAIA con un servidor Tomcat preparado para el despliegue inmediato de la web. Para poder desplegar la base de datos en la web, se siguió un tutorial para el despliegue de SQL Server en Debian [Arvid(2023)].

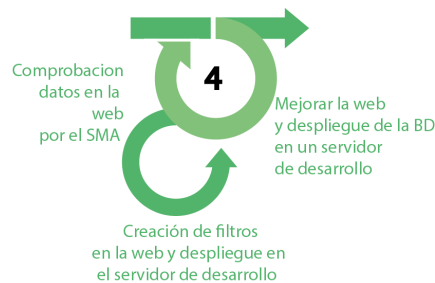


Figura 32: Esquema del cuarto ciclo

Una vez desplegado SQL Server y Tomcat en el servidor, se enviaron al servidor los archivos de la web y se montó la base de datos con el programa SQL Server Management Studio. Con ambas cosas desplegadas, se ajustaron los scripts ETL para cargar los datos en la base de datos del servidor en lugar de la base de datos local. Ya que solo requeríamos comprobar el correcto funcionamiento de la web en la nube, no se cargaron los informes, solo un conjunto lo suficientemente grande como para poder comprobar el funcionamiento. Con esto, se desplegó por primera vez la web en la nube (Figura 33).

Tras comprobar el funcionamiento de la web en un servidor, se decidió que se añadirían filtros para poder acceder de una forma más directa a la información buscada. Empezando por el listado de los calendarios, se le añadió un formulario a para poder filtrar por identificador de la lluvia, por año de la lluvia o por ambos.

En la consulta de datos de los informes se ajustó el tamaño de las tablas para que se ajustaran a un tamaño normal de la interfaz de la web. Estas tablas, al contener los datos de los



Figura 33: Primer lanzamiento de la Web en la nube

informes, excedían en gran medida el límite de la web, necesitando desplazar hacia la derecha para visualizar estos datos.

A modo de poder filtrar cómodamente los meteoros de la base de datos, se añadió un filtro conjunto donde poder indicar una fecha de inicio y fin del filtro en conjunto con el tipo de informe existente para ese meteoro. De esta forma se podrán filtrar los meteoros cómodamente para el cliente.



Figura 34: Filtros de los datos en la web

También se añadió en la web una descripción a cada dato de las tablas en modo de ventana

flotante al pasar el ratón sobre estos. De esta forma, se aclara la información presente en las tablas si no es suficiente con el nombre de la misma.

5.5. Ciclo 5: Publicación de la página de acceso a la Base de Datos

Una vez finalizado, se le notificó al personal de la SMA la publicación de la web provisional para que pudieran aprobar y corregir la correcta información de los informes, diseño de la web y descripción de los datos en las tablas. Estos, aclararon qué datos serán mostrados en la web y cuales no, ya que ciertos datos de los informes se usan únicamente para cálculos y no es necesario mostrarlos en la web (Figura 35). También, la cantidad de decimales a mostrar en los datos ya que ciertos de estos tienen una cantidad mucho más grande de la necesaria a la hora de mostrarlos. Al igual, se notificaron las modificaciones necesarias en las descripciones de los datos.



Figura 35: Esquema del quinto ciclo

Otra modificación que se solicitó fue respecto a los observatorios. Muchos de estos aparecen en los archivos pero no están activos a la fecha actual. Como solución a esto se acordó que el personal de la SMA modificaría el archivo “descripcion-observatorios” para añadir un campo que indique si el observatorio debe mostrarse o no en la web. Con esto acordado, se modificó el script de carga de los observatorios para poder extraer y almacenar este dato.

Una vez se realizaron estos cambios, tanto a nivel de script como de la web, se procedió a hacer una mejora a la visualización de los datos en la web. Los diseños y colores de la web se han elegido para resultar ciertamente similar al usado en la web de la SMA [SMA(1975)]



Seleccione una fecha de inicio:

Seleccione una fecha de fin:

Tipo de informe:

Fecha	Hora
2022-01-02	05:07:13
2022-01-02	06:05:59
2022-01-02	23:21:20
2022-01-02	04:30:59
2022-01-03	05:09:03
2022-01-03	23:08:19
2022-01-03	23:24:14

Figura 36: Diseño final de la web

6

Conclusiones

6.1. Conclusiones

En este Trabajo Fin de Grado se ha realizado en el diseño e implantación de una base de datos para la Sociedad Malagueña de Astronomía con información de meteoros recogidos con la Red de Detección de Meteoros y Bólidos. El aspecto más interesante del trabajo realizado es la oportunidad de trabajar en un problema real, datos reales y usuarios reales que, desde el primer momento, han manifestado sus necesidades e inquietudes acerca del proyecto.

Los resultados del proyecto son los siguientes:

- La Base de Datos: La disponibilidad de una base de datos donde almacenar la cantidad inmensa de datos tanto de los informes generados como de todos los observatorios en la red y calendarios de lluvias hace que el acceso a ella sea más cómodo y fácil a diferencia de almacenar todo en archivos de texto y en local. También, va a facilitar la creación de futuras aplicaciones con los datos almacenados en esta base de datos puesto que todo el proceso de extracción de datos de los informes ya está realizado.
- Almacenamiento de los Informes en la Base de Datos: El almacenamiento de todos los informes generados por la red nos facilitará el acceso a ellos al igual que la relación entre los mismos.
- Web de acceso a los datos: Disponer de una web donde poder acceder a los datos de estos informes de una forma ágil y ordenada acelerará la búsqueda y el acceso. Con esto, un trabajo de análisis de unos informes en una fecha específica se simplifica a una selección en pantalla, en lugar de una larga navegación por los directorios donde se almacenan los informes.

Como resultado de este TFG, se tiene una mejora sustancial en la forma de trabajar y de acceder a los datos en la organización (Figura 37). Gracias al almacenamiento de estos datos

en una base de datos, facilitaríamos la creación de aplicaciones con estos datos, ya que no requerirán de crear un proceso ETL antes de trabajar con los informes. A su vez, la creación de relaciones entre los datos en la base de datos y la publicación de estos contenidos en una web, facilitará al astrónomo el acceso a los datos de los informes para un meteoro concreto, de modo que no requerirá la búsqueda exhaustiva a través del previo sistema de ficheros.

También se destaca la mejora en la compartición de estos datos. Previamente solo tenían acceso a los datos de los informes los departamentos con acceso al sistema de ficheros y ahora se podrán compartir datos con otras instituciones a través de consultas en la base de datos o en su defecto, la consulta de los datos de la web pública.

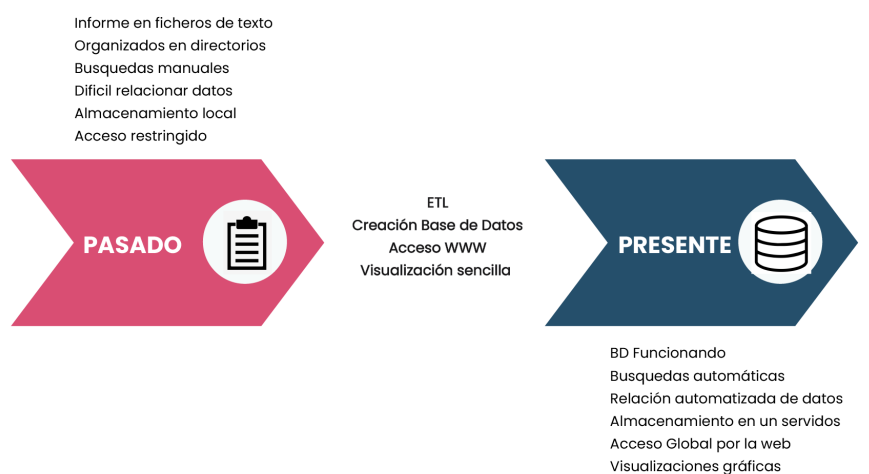


Figura 37: Comparación de la situación antes de la realización del TFG y la situación actual

En resumen, esto provocará una gran mejora en el almacenamiento de los datos así como en su manipulación y análisis. También, nos dará pie a la posibilidad de crear, de una forma más sencilla, nuevos proyectos asociados a estas observaciones.

6.2. Líneas Futuras

Como ya se ha comentado el resultado de este Trabajo Fin de Grado es un sistema real que va a ser utilizado por los miembros de la SMA. Este sistema puede ser el punto de partida para otros desarrollos que saquen provecho del esfuerzo de extraer y almacenar los informes. Estos trabajos futuros permitirán mejorar el acceso a la información a toda la comunidad científica, y permitirán avanzar un paso más en la automatización y gestión de la ingente cantidad de

datos que se recoge a diario (Figura 38) .



Figura 38: Pasado, Presente y posible Futuro para el sistema desarrollado

Algunos de ellos pueden ser los siguientes:

- Mejorar las opciones de acceso a los datos en la web. Puesto que actualmente solo se muestran datos mediante tablas y sin posibilidad de relacionarlos, se pueden desarrollar visualizaciones gráficas que nos permitan un acceso gráfico a los datos, y que muestren relaciones entre diferentes indicadores con metáforas visuales clarificadoras e ilustrativas.
- Trabajar en la estandarización de los datos presentes en el sistema para poder garantizar la integridad de estos, para posibilitar el intercambio con otras instituciones y para descartar informes cuyos datos no sean coherentes.
- Realizar un trabajo de generación de informes a raíz de los datos almacenados en la base de datos. De esta forma se podrían crear nuevos informes a partir de los generados por el personal de la SMA, se pueden hacer plurilingües, o se puede crear informes con datos restringidos para difundirlos públicamente.
- Conexión directa del generador de informes con la Base de Datos: Implementar un mecanismo en el generador de informes que, además de crear el fichero de texto, almacene todos sus datos en la base de datos. De la misma forma, incluir enlaces a las imágenes y a los vídeos en el servidor donde los telescopios los almacenan.

En conclusión, este proyecto TFG ha servido como oportunidad de trabajo en un problema real para el alumno y para generar una mejora consistente en el trabajo de la SMA sobre la

observación de meteoros y bólidos y la generación de informes a través de la *Red de Detección de Meteoros y Bólidos*. A raíz de este proyecto, se podrán compartir todos los datos extraídos de las observaciones cómodamente con otras instituciones que trabajen en el mismo contexto y todos estos datos se podrán mostrar de forma pública a partir de la web sumado a las imágenes y vídeos que ya se comparten en la web principal de la SMA. También, el desarrollo de este TFG permitirá la creación de diversos proyectos que servirán tanto para ampliar y completar lo creado como para una mejor manipulación de los datos extraídos gracias al almacenamiento en una base de datos para análisis y diseño de nuevos sistemas a partir de estos.

Referencias

- [Madedo Gil(2017)] J. Madiedo Gil, Meteoros y meteoritos, Marcombo, 2017.
- [SEA(2022)] SEA, Lluvia de estrellas, 2022. URL: <https://www.sea-astronomia.es/glosario/lluvia-de-estrellas>.
- [SMA(1975)] SMA, Sociedad malagueña de astronomía, 1975. URL: <https://www.astromalaga.es/>.
- [Castellón(2023)] A. Castellón, Red de detección de bólidos y meteoros de la uma y la sma, 2023. URL: <http://meteoros.astromalaga.es/author/albertocastellon/>.
- [Castellón and Castro-Tirado(2019)] A. Castellón, A. J. Castro-Tirado, AVOIDING MURPHY'S LAW ON DETECTING METEORS, Revista Mexicana de Astronomía y Astrofísica Serie de Conferencias 51 (2019) 124–126. URL: <https://doi.org/10.22201/ia.14052059p.2019.51.20>.
- [Microsoft(2022)] Microsoft, Sql server 2022, 2022. URL: <https://www.microsoft.com/es-es/sql-server/sql-server-2022>.
- [Microsoft(2023)] Microsoft, Sql server management studio, 2023. URL: <https://learn.microsoft.com/en-us/sql/ssms/sql-server-management-studio-ssms?view=sql-server-ver16>.
- [ORACLE(2009)] ORACLE, Sql developer data modeler, 2009. URL: <https://www.oracle.com/es/database/sqldeveloper/technologies/sql-data-modeler/>.
- [van Rossum(1980)] G. van Rossum, Python, 1980. URL: <https://www.python.org>.
- [Kleehammer(2023)] M. Kleehammer, pyodbc, 2023. URL: <https://pypi.org/project/pyodbc/>.
- [Microsoft(2015)] Microsoft, Visual studio code, 2015. URL: <https://code.visualstudio.com>.

[Microsystems(1999)] S. Microsystems, Javasever pages, 1999. URL: <https://www.oracle.com/java/technologies/jspt.html>.

[Microsoft(2014)] Microsoft, Microsoft jdbc driver for sql server, 2014. URL: <https://learn.microsoft.com/en-us/sql/connect/jdbc/microsoft-jdbc-driver-for-sql-server?view=sql-server-ver16>.

[Canadá(2001)] I. Canadá, Eclipse, 2001. URL: <https://www.eclipse.org>.

[Foundation(1999)] A. S. Foundation, Apache tomcat, 1999. URL: <https://tomcat.apache.org>.

[UMA/SMA(2017)] UMA/SMA, Red de detección de bólidos y meteoros de la uma y la sma, 2017. URL: <http://meteoros.astromalaga.es>.

[Ocaña-González(2011)] F. Ocaña-González, Optimización de técnicas de detección de bólidos y mF.eteoros, UCM. Trabajo Fin de Grado, 2011. URL: https://eprints.ucm.es/id/eprint/13292/1/TAD2011_Bolidos_Paco.pdf.

[NASA(2018)] NASA, ¿qué es una lluvia de meteoritos?, 2018. URL: <https://spaceplace.nasa.gov/meteor-shower/sp/>.

[Castellón(2020)] A. Castellón, La uma dispone de una red de detección de meteoros y bólidos junto con la sociedad malagueña de astronomía, 2020. URL: <https://www.uma.es/sala-de-prensa/noticias/la-uma-dispone-de-una-red-de-deteccion-de-meteoros-y-bolidos-junto-con-la-sociedad>

[IAA-CSIC(1998)] IAA-CSIC, Bootes, 1998. URL: <https://bootesnetwork.com/es/>.

[Tejada(2023)] P. Tejada, Bootes la red de telescopios robóticos española en la que participa la uma, 2023. URL: https://www.malagahoy.es/malaga/BOOTES-telescopios-roboticos-participa-UMA_0_1770124796.html.

[Castellón(2015)] A. Castellón, La red de seguimiento de bólidos y meteoritos de la sociedad malagueña de astronomía, 2015. URL: <http://hdl.handle.net/10630/9957>.

[Dubyago(1961)] A. Dubyago, The determination of orbits, The Macmillan Company, New York, 1961.

[Lovell(1954)] A. Lovell, Meteor Astronomy, Continuum, Oxford at the Clarendon Press, 1954.

[Calderón and Valverde-Rebaza(2007)] D. S. Calderón, A., J. Valverde-Rebaza, Metodologías ágiles, volume 37, Facultad de Ciencias Físicas y Matemáticas. Universidad Nacional de Trujillo, 2007. URL: https://www.academia.edu/23325522/Universidad_Nacional_de_Trujillo.

[Arvid(2023)] L. Arvid, Sql server en debian 11, 2023. URL: <https://www.howtoforge.com/how-to-install-and-use-microsoft-sql-server-on-debian-11/>.

Apéndice A

Manual de uso

A.1. Guía de despliegue de la Base de datos

La siguiente guía describirá los pasos a seguir para poder realizar correctamente una instalación de la base de datos tanto en local como en un servidor. Una vez terminada la instalación de esta, se aclararas los cambios a realizar en los scripts ETL para garantizar la conexión con la base de datos creada.

Previo a la explicación de los pasos para la instalación, será necesaria la instalación de los siguientes software:

- En el sistema donde vaya a intalarse la base de datos: SQL Server 2019.
- En el sistema donde vayan a ejecutarse los scripts: Python 3.10.

Una vez realizada la instalación del software previamente mencionado, se procederá con el despliegue de la base de datos.

- Creación de una nueva database en SQL Server (El nombre “MeteoDB” será el usado a lo largo de la guía pero no necesita ser este, puede usarse el que se prefiera):
 - CREATE DATABASE MeteoDB;
- Con la nueva database creada, se precisará de copiar todo el contenido del archivo ddl-basedatos.ddl en una nueva consulta asignada a la database recién creada. Para asignar la consulta a la nueva database se hará uso del siguiente comando al inicio de la misma:
 - USE MeteoDB;
- Una vez desplegada la base de datos, se requerirá de un usuario capaz de realizar INSERT, UPDATE, SELECT. Este usuario es el que se especificará en los scripts. Cabe destacar que se puede usar el usuario SA creado en la instalación, pero al tener mas permisos de los necesarios, no es recomendable por la seguridad de la base de datos. Para la creación

del usuario y asignación de permisos se realizarán los siguientes comandos (Tanto el nombre de usuario como contraseña pueden seleccionarse sin restricciones):

- `CREATE LOGIN Usuario WITH PASSWORD = 'password';`
- `USE MeteoDB;`
`CREATE USER Usuario FOR LOGIN Usuario;`
- `USE MeteoDB;`
`GRANT INSERT, DELETE, SELECT, UPDATE ON SCHEMA::dbo TO Usuario;`

Con esto realizado, la base de datos está lista para el correcto uso de la misma. Si esta ha sido instalada en un servidor, deberá garantizarse que el puerto 1433 está abierto, pues es el que será usado por SQL Server.

A.2. Guía de configuración de los scripts

Realizado el despliegue de la base de datos, pasamos a la instalación y preparación de los scripts.

- Para el correcto funcionamiento de los scripts, requerimos del módulo pyodbc [[Kleehammer\(2023\)](#)]. Para la instalación de este se recomienda seguir los pasos presentes en su correspondiente web.
- Una vez instalado este módulo que nos servirá para realizar la conexión con la base de datos, deberán realizarse los cambios en el apartado “Conexion a la BD.” presente en cada uno de los scripts:
 - **server** será “IPServidor;PORT=1433” donde IPServidor se remplazará por localhost en el caso de que queramos trabajar en local.
 - **database** será el nombre de la base de datos creada. En el caso del ejemplo, esta será “MeteoDB”.
 - **usuario** será el nombre del usuario creado para los scripts.
 - **password** será la contraseña asignada al usuario creado para los scripts.

Realizados estos ajustes, los scripts están listos para usarse.

A.3. Guía de uso de los scripts

Todos los scripts realizados tienen una ayuda que aparece por pantalla si estos se han ejecutado sin argumento alguno. “CargaInicial.py” necesariamente deberá ser ejecutado sin argumento y seguir la ayuda que aparecerá. A diferencia de este, “CargaInformesZ.py”, “CargaInformesRad.py”, “CargaInformesFot.py” pueden ser ejecutados pasando como argumento el directorio de los informes a cargar, en cuyo caso no aparecerá ayuda por pantalla y se ejecutará directamente. Todos estos scripts mostrarán por pantalla qué se está cargando en la base de datos en ese preciso momento a modo de guía visual de su funcionamiento. En caso de que estos se ejecuten pasando como argumento el directorio de los informes, no se mostrará por pantalla esta ayuda visual.



UNIVERSIDAD
DE MÁLAGA

| uma.es

E.T.S de Ingeniería Informática
Bulevar Louis Pasteur, 35
Campus de Teatinos
29071 Málaga

E.T.S. DE INGENIERÍA INFORMÁTICA