Ref. Ares(2022)1493930 - 01/03/2022

UNION CIVIL PROTECTION MECHANISM

Directorate General for European Civil Protection and Humanitarian Aid Operations

PREVENTION AND PREPAREDNESS PROJECTS IN CIVIL PROTECTION AND MARINE POLLUTION

EVE

European
Volcano
Early
Warning
System

Grant Agreement Number	826292
Proposal Title / Acronym	EUROPEAN VOLCANO EARLY WARNING SYSTEM (EVE)

D15-DATABASE ARCHITECTURE AND DATA COMPILATION

General issues

We live in an age where data as well as the information generated from it is growing at an overwhelming rate. No less rapidly, the number and heterogeneity of sources that generate them is growing, and the technologies responsible for supporting their life cycle are born, updated, and die. This scenario has caused the inability to govern this vast ocean of data - and therefore to be able to extract information from it - to become an endemic evil of the 21st century.

The lack of agreed protocols for: extracting, organizing, transforming and storing data, make the current scenario of Volcanology a clear paradigm of this problem. In this context, it is difficult to take advantage of cutting-edge technologies to create tools that are able to extract information automatically.

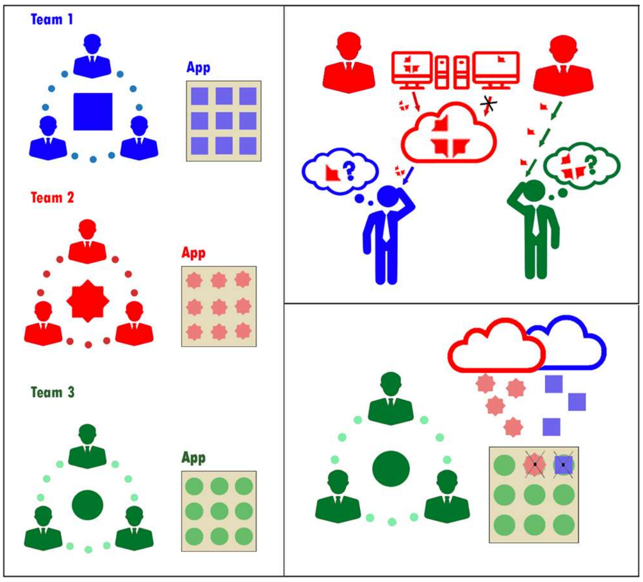
On the other hand, the lack of defined content structure implies that many valuable resources, in terms of knowledge, are at best forgotten on a hard drive, whether due to human or technological barriers.

It is therefore necessary to create a series of standard protocols and tools to address this issue, which affects both pre-existing resources and those that have not yet been generated.

Database Architecture

When working on a scientific product in the area of volcanology, the team generates a series of files that in many cases depend on third-party tools for their interpretation. Generally, once the experiment is completed, these documents are stored following - at best - certain protocols. The problem in part is that these are not agreed between the different teams. In addition, they tend to evolve and do not create tools to adapt old work to change.

In the long run, what usually prevails are the reports resulting from the experiment. The inability to reproduce them - due to the obsolescence of the tools and formats used, as well as the lack of consensus on where and how they should be stored -generates a bottleneck when it comes to making the most of it.



[Figure 1 | a]. Three teams perform similar assessments, but use different formats.

[Figure 1 | b]. Team 2 did not follow a specific protocol for storing data associated with a particular experiment. Team 1 accesses team 2's cloud to get them, Team 3, on the other hand, has requested them directly from a member of Team 2. Team 1 and Team 2 are unable to reproduce the experiment for lack of data.

[Figure 1 | c]. Team 3 accesses Team 1 and Team 2 clouds, but can't use data on them due to compatibility issues

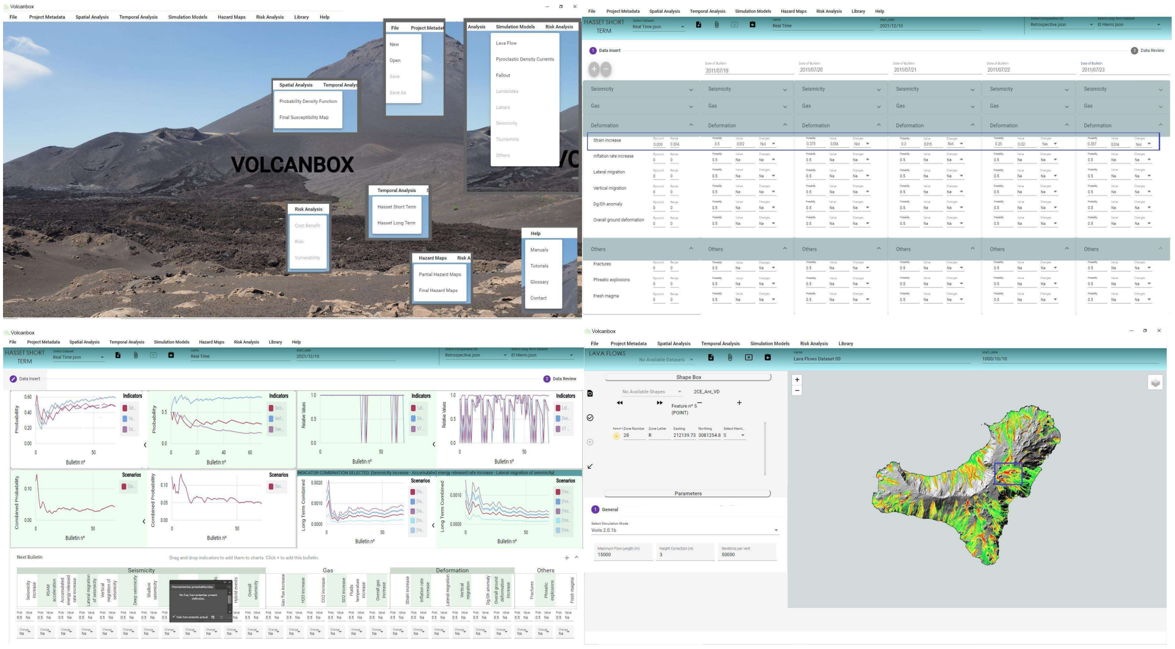
On the other hand, creating protocols that standardize these processes involves defining protocols and structures that adapt to the data and content you are working with. In such a heterogeneous context, modelling all content using a single scheme is by no means trivial, which is why it seems like a good strategy to take a multilingual persistence-based approach.

This philosophy, instead of adapting all the information to fit it into a single technology or structure, prefers to divide the system into a set of subsystems - each of them chosen taking into account both the type of data to be hosted, as expected of them - that they communicate with each other. This communication is essential, as it allows you to resolve requests as if they were a single system.

By adopting this approach, if, for example, in the future we need to store content that is difficult to fit into our system, we could add an extra piece to it - tailor-made to carry this responsibility - without having to make major modifications to the system.

Given this problem, an infrastructure has been designed, which will be partially implemented during the development of the EVE project. It consists of two distinct parts:

- A tool called *Volcanbox* that will offer:



[Figure 2] Volcanbox Desktop Application

  - GIS toolkit capable of:
    - Extract georeferenced data from encoded files in most formats on the market.
    - Import, modify, or generate user-friendly vector files by typing coordinates or clicking on specific points on the map.
    - Perform - in a manner that is transparent to the user - the operations necessary to be able to cross-encode data on more than 11,000 compatible reference systems.
  - *Long and Short Term* statistical analysis.
  - Elaboration of *Probability Density Functions* with different methods of estimating bandwidth.

    - Elaboration of *Susceptibility Maps* from a weighted set of probability density functions

    - Suite of sections for the assessment of the following volcanic hazards:

        - Lava Flows.
        - Pyroclastic Density Currents
        - Fallout
        - Landslides
        - Lahars
        - Seismicity
        - Tsunamis
        - Others

    - Partial and total hazard maps.

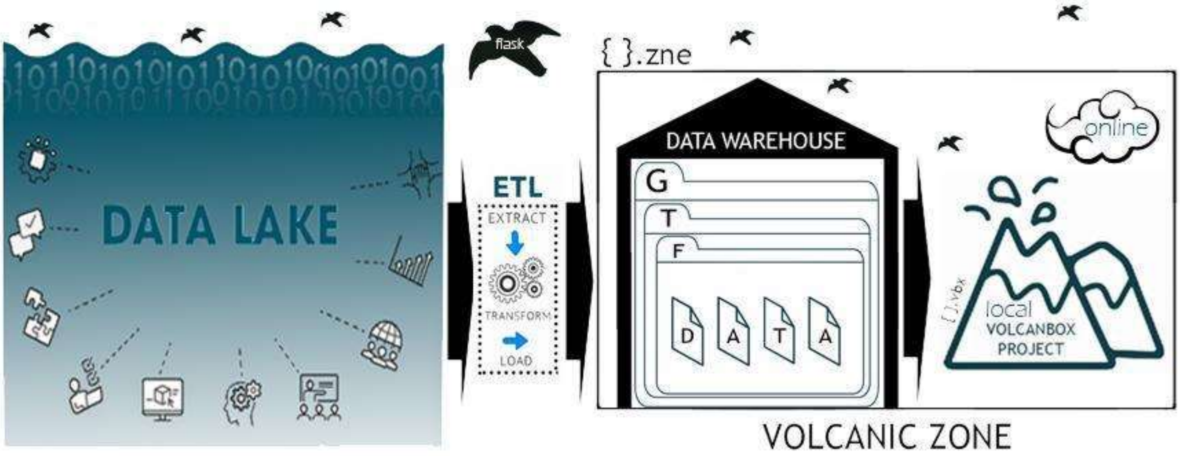
    - Library that acts as a database, to organize and offer information, in the same application, that is, by "GIS" tools of third parties.

    - Tools for viewing and extracting information from data uploaded to the application.

    - Online functions to publish information and facilitate collaboration between teams.

    - Generation of new data from the Crossover of different experiments in order to search for new information.

- A data storage system focused on polyglot persistence. This will be structured in *Volcanic Zones* that will contain the following elements:



[Figure 3] Volcanbox Platform Data Architecture

- **Data Lake** : A storage repository that contains raw acquired data. Each item is assigned a unique identifier and a set of tags. These allow you to relate them to each other, without the need for a closed data structure and relationships.

- **Flask Microservices** : A set of services capable of storing, managing, and delivering data stored on the system.

- **ETL** : are the acronyms for Extraction, Transformation and Loading, refer to a set of techniques, tools and technologies that aim to extract data from various sources and transform them to be able to load them into other systems.

- **Volcanic Zone:** A set of subsystems that resolve data relating to a given geographical area. Includes:

  - **Data Warehouse** : A series of guidelines and good practices for storing data extracted from the Data Lake component. Their purpose is to provide a standard structure that facilitates their subsequent recovery, as well as ensuring compatibility with *Volcanbox*.

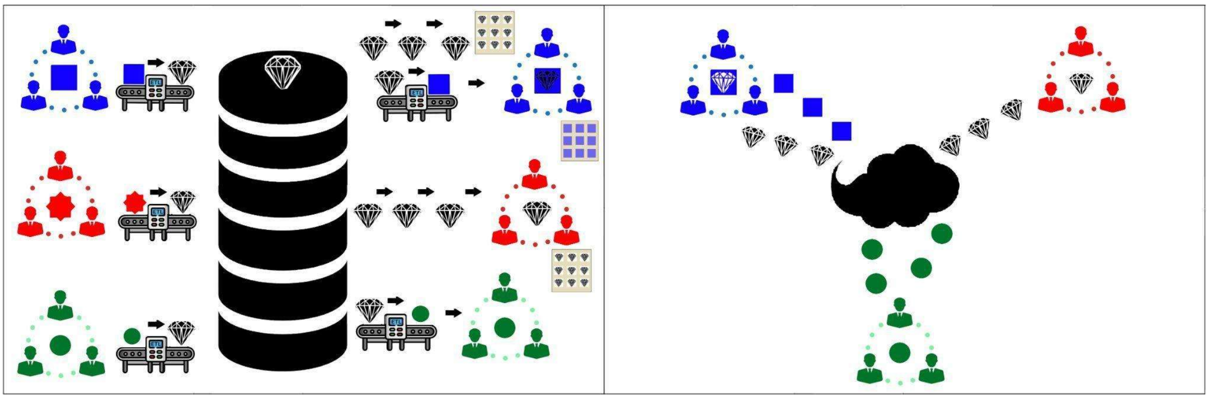
  - **Local Volcanbox Project** : A directory and file structure resulting from a Volcanic Risk Assessment generated using the *Volcanbox* application, as well as all input data that has been used to perform it.

  - **Volcanbox Online Project** : Database architecture designed to serve the online features of the *Volcanbox* application.

The purpose is that this whole set of subsystems can work as a whole in a way that is transparent to the user. To accomplish this task, an ecosystem of microservices implemented through the *Python Flask Library* is proposed.

- ❖ This structure will allow, among others:

  - ➢ Preserve and recover the sources in their original format and contents.
  - ➢ Standardize the way data is organized, maximizing its usefulness when it is exchanged between computers.
  - ➢ Compatible data from different sources so that it can be cross-referenced.
  - ➢ Generate new data from existing data.
  - ➢ Easily reproduce any previous experiments.
  - ➢ Automate early warning delivery using the VEWS Volcano Early Warning System platform.
  - ➢ Prepare the terrain for the application of machine learning algorithms.



[Figure 4 | a]. Teams send data for storage, and *ETL*s transform data into suggested formats. The red team has adopted the *Volcanbox* platform, so it can use the data without the *ETL*s having to transform it. In the blue team some members have decided to adapt the proposed formats, otherwise there are members who have not yet made the leap; however, thanks to ETL transformations all members can receive the data in the desired format. The green team still uses its usual formats, thanks to the *ETL*s they can now get the data that the blue and red teams have saved.

[Figure 4 | b] The complexity of the platform is transparent to the user, they see the system as a black box that accepts the desired formats.

** VEWS is a platform that through a set of web tools aims to facilitate interaction and cooperation between scientists and Civil Protection Agencies to anticipate volcanic disasters in a timely manner. The Volcanbox application will be able to connect to the VEWS to generate or update new alarms and include in them all the results considered appropriate.*

Data Lake

One of the challenges associated with the problem that this methodology seeks to resolve is to ensure the availability of all relevant content in terms of volcanic risk assessment. The disparity - in terms of format, structure, nature, purpose, etc. - make it very difficult to store them using a closed scheme. This is a very common issue in the so-called *big data* environments, for this reason the concept of *Data Lake* -very present in these environments- has been taken as inspiration.

A *Data Lake* is a large set of raw data, which does not yet have a definite purpose - unlike for example a *Data Warehouse* where data has already been structured, filtered and processed for a specific purpose.

[Figure 5] Data Lake

When content is in Data Lake, it can be normalized and enriched. This may include metadata extraction, format conversion, augmentation, entity extraction, crosslinking, aggregation, denormalization, or indexing.

This type of implementation has been chosen so that users can include as much heterogeneous content as possible, otherwise it will also allow the creation of a database to, in the near future, combine and process this data using mass data techniques, and so on to be able to carry out searches and analyses that would otherwise have been impossible.

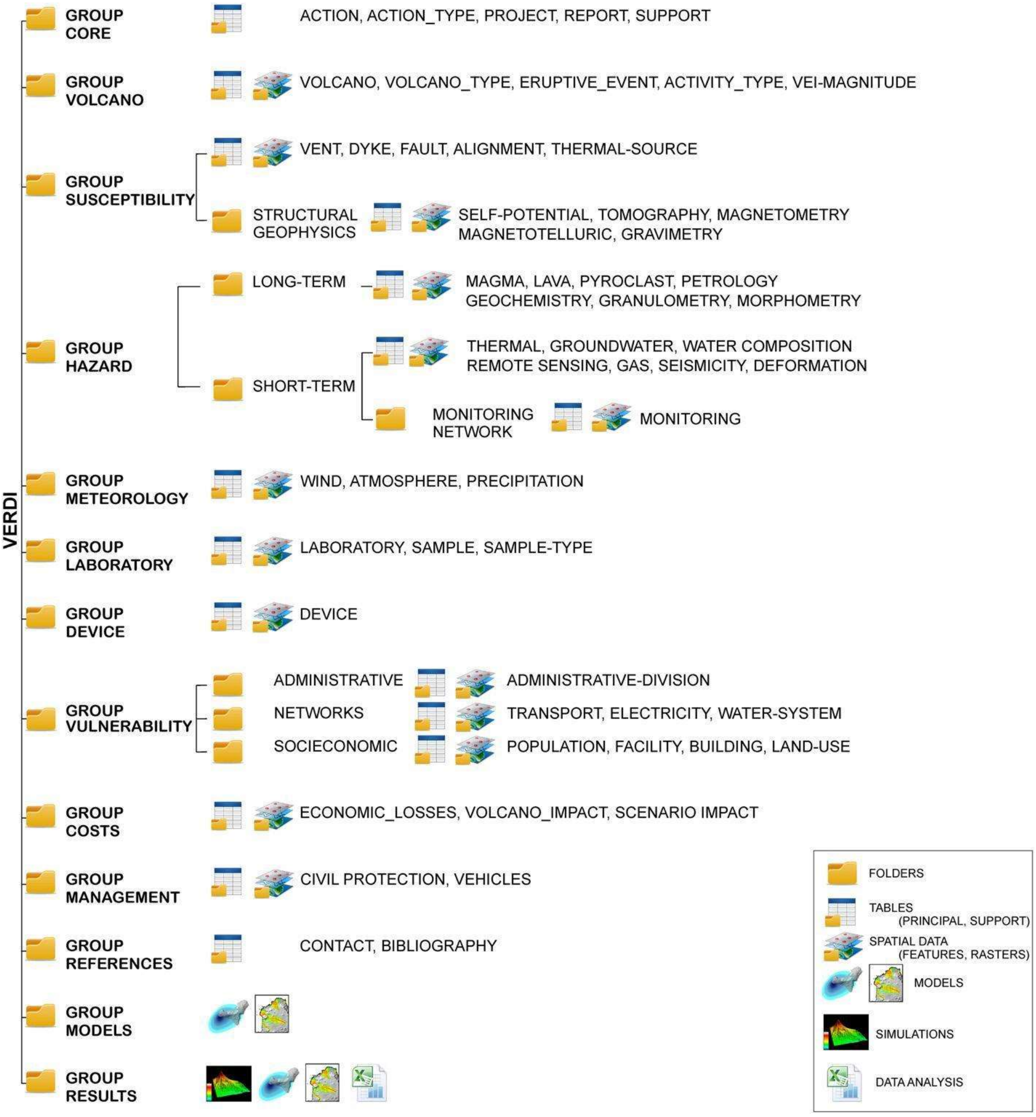
Label system

As mentioned above, the *Data Lake* component does not have a defined structure. But how does *Data Lake* then be able to provide us with information when we want to retrieve it?

Unlike other systems where data is stored following certain formats and / or a certain hierarchy of directories, files, tables, relationship tables, etc. our *Data Lake* component assigns each item a unique identifier, and a series of tags.

These tags can be either a manual assignment - that is, made by the same user or automated - that is, made by the same system using, for example, artificial intelligence techniques. In fact, in the latter case, as *the Data Lake* grows, the system will be able to learn and discover new similarities between the different stored content, and thus enrich the data by assigning new tags automatically. For example, it may be the case that a user stores content without being clear about the correct tag and that the system itself finds one or more suitable ones. It is worth noting that for time limitations, only the first option has been implemented, however, the system is fully compatible with the second.

In order to label the contents, we follow the work carried out in *[Bartolini et al. 2014]* * which proposes a database structure called *VERDI* aiming at data storage for the assessment of volcanic hazard and risk.


VERDI
GROUP CORE
ACTION, ACTION_TYPE, PROJECT, REPORT, SUPPORT
GROUP VOLCANO
VOLCANO, VOLCANO_TYPE, ERUPTIVE_EVENT, ACTIVITY_TYPE, VEI-MAGNITUDE
GROUP SUSCEPTIBILITY
VENT, DYKE, FAULT, ALIGNMENT, THERMAL-SOURCE
STRUCTURAL GEOPHYSICS
SELF-POTENTIAL, TOMOGRAPHY, MAGNETOMETRY
MAGNETOTELLURIC, GRAVIMETRY
GROUP HAZARD
LONG-TERM
MAGMA, LAVA, PYROCLAST, PETROLOGY
GEOCHEMISTRY, GRANULOMETRY, MORPHOMETRY
SHORT-TERM
THERMAL, GROUNDWATER, WATER COMPOSITION
REMOTE SENSING, GAS, SEISMICITY, DEFORMATION
MONITORING NETWORK
MONITORING
GROUP METEOROLOGY
WIND, ATMOSPHERE, PRECIPITATION
GROUP LABORATORY
LABORATORY, SAMPLE, SAMPLE-TYPE
GROUP DEVICE
DEVICE
GROUP VULNERABILITY
ADMINISTRATIVE
ADMINISTRATIVE-DIVISION
NETWORKS
TRANSPORT, ELECTRICITY, WATER-SYSTEM
SOCIECONOMIC
POPULATION, FACILITY, BUILDING, LAND-USE
GROUP COSTS
ECONOMIC_LOSSES, VOLCANO_IMPACT, SCENARIO IMPACT
GROUP MANAGEMENT
CIVIL PROTECTION, VEHICLES
GROUP REFERENCES
CONTACT, BIBLIOGRAPHY
GROUP MODELS
GROUP RESULTS
FOLDERS
TABLES (PRINCIPAL, SUPPORT)
SPATIAL DATA (FEATURES, RASTERS)
MODELS
SIMULATIONS
DATA ANALYSIS

[Figure 6] Verdi Database Structure Groups

** Despite taking advantage of the work done in the Verdi data architecture, the concept of a label should not be confused with a specific storage structure. On the contrary, the only thing that is advised to the user is that, if he uses, for example, the tag, shape, also use the top tags, this way you can filter the information as if it existed in a real hierarchy - in terms of implementation - without it really being that way.*

One detail to keep in mind is that, by its nature, this component is not designed to provide real-time data. This is where the concept of polyglot persistence makes sense, so to solve requests in real time, we already have other more suitable components in our system.

On the other hand, sources can often contain information that is considered confidential, fortunately, the data lake is equipped with tools to ensure control of access to information.

Obtaining Contents

Regarding the process of obtaining the contents in order to include it in our *Data Lake*, we can also make a simile with the *Silo* concept of massive data environments.

In our case, *Silos* can be the hard disks of research groups, databases such as WOVOdat, web pages and databases of volcanological laboratories (e.g. Volcanological Observatory of Piton de la Fournaise, INGV Osservatorio Etneo) or the website of the National Geographic Institute (IGN).

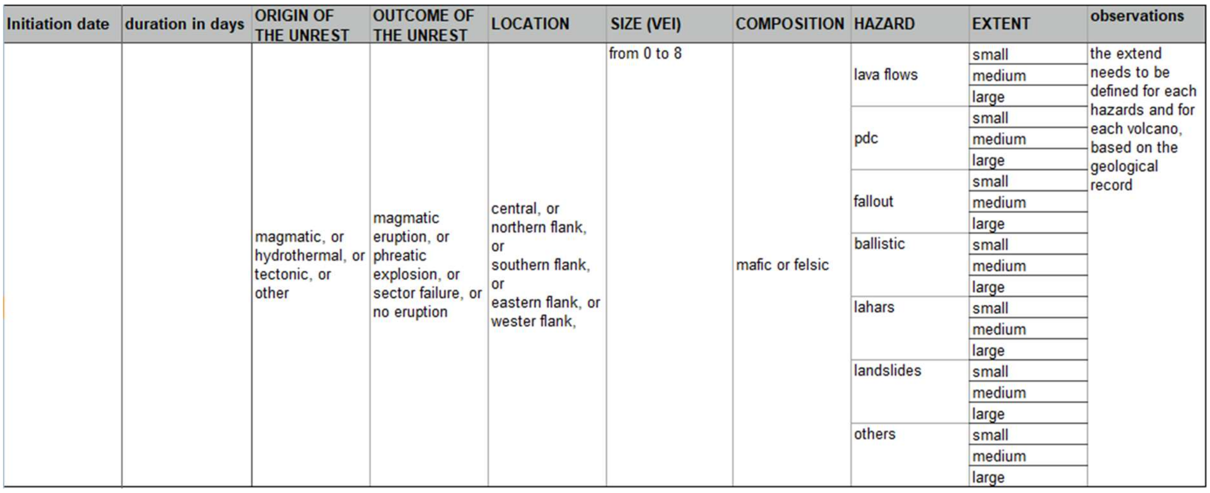
Data Silos occur when there is no centralized system to store all the data on a computer. A *Silo*, therefore, makes it more difficult to discover new data, as each is controlled by an independent department, with different policies and even technologies.

You may feel that *Data Silos* are needed to allow more flexibility for computers, iterate faster, and adjust policies to needs in a simple way. However, from a global point of view, it is very difficult to extract value from the data and discover new ideas.

One of the main reasons for adopting a Data Lake is usually to avoid Data Silos, which often occur due to rapid and uncontrolled growth.

As for the **EVE** project, the task of compiling data has had to be mostly manual and has required the collaboration of the different partners of the project. Each workgroup has its own databases, and its own storage systems, but the vast majority of these do not have an extraction system for external users, which makes it difficult to extract the data without their collaboration. In order to obtain the necessary data to be able to perform the *Long Term and Short term analysis*, the partners were provided with two templates in *'.ods'* format in order to gather the data of the volcanoes and eruptions selected for study.

The template that aims to collect the eruptive history of a given volcano is what we have called *Long Term Template* and is as follows:



Initiation date	duration in days	ORIGIN OF THE UNREST	OUTCOME OF THE UNREST	LOCATION	SIZE (VEI)	COMPOSITION	HAZARD	EXTENT	observations
		magmatic, or hydrothermal, or tectonic, or other	magmatic eruption, or phreatic explosion, or sector failure, or no eruption	central, or northern flank, or southern flank, or eastern flank, or wester flank,	from 0 to 8	mafic or felsic	lava flows	small	the extend needs to be defined for each hazards and for each volcano, based on the geological record
								medium	
								large	
							pdc	small	
								medium	
								large	
							fallout	small	
								medium	
								large	
							ballistic	small	
								medium	
								large	
							lahars	small	
								medium	
								large	
							landslides	small	
								medium	
								large	
							others	small	
								medium	
								large	

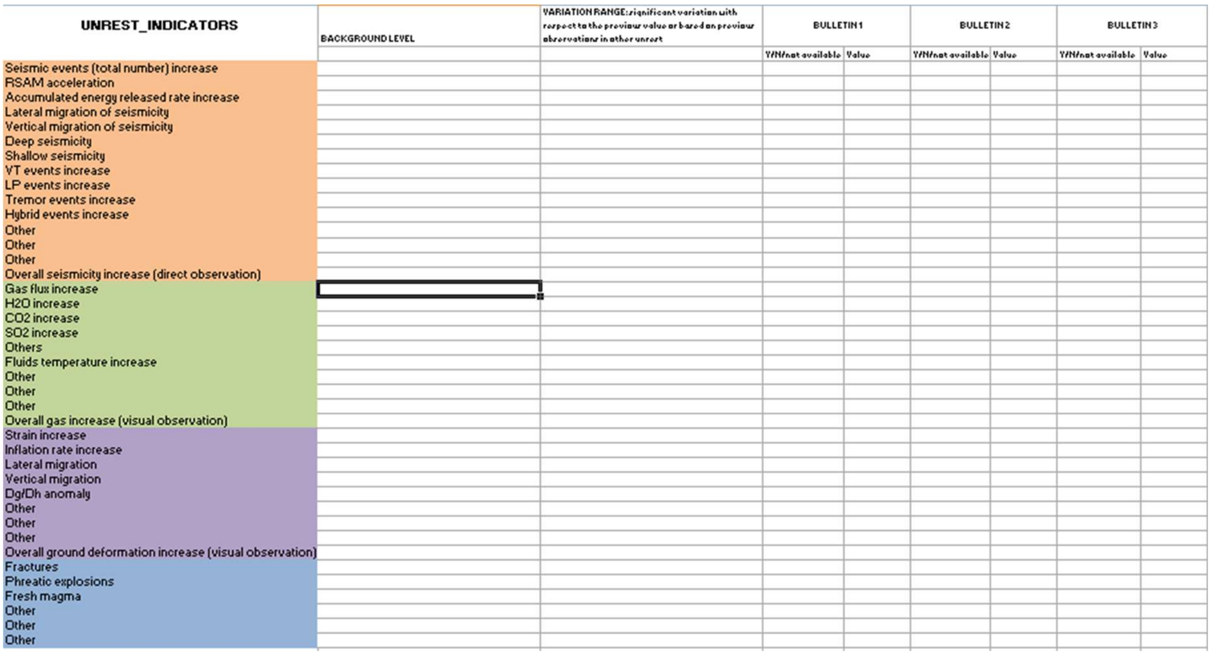
[Figure 7] Long Term Original Template

In this table, the information of the different identified episodes of unrest of a certain volcano has been collected. Each row in the table corresponds to an episode of unrest, and for each of these episodes the following data has been entered in the different columns:

- Initiation date
- Duration in days
- Origin of the unrest
  - Magmatic, hydrothermal, tectonic or other
- Outcome of the unrest
  - Magmatic eruption, phreatic explosion, sector failure, or no eruption
- Location
  - Central, northern flank, southern flank, eastern flank or western flank
- Size (VEI)
  - Form 0 to 8
- Composition
  - Mafic or felsic
- Hazard
  - Lava flows
  - PDC
  - Fallout
  - Ballistic
  - Lahars
  - Landslides
  - Others
- Extent
  - Small, medium or large

With this information entered, each unrest episode will be characterized, and with the whole set of unrest episodes, we will have collected the eruptive history of the volcano, through the time-lapse selected. These data are necessary to be able to make calculations of eruption probability and the most probable scenarios by using the tools developed in this project.

In order to obtain the data needed to perform the Short Term analysis of selected eruptions, the template we have called *Short Term Template* was developed and is as follows:



UNREST_INDICATORS	BACKGROUND LEVEL	VARIATION RANGE: significant variation with respect to the previous value or based on previous observations in other unrest	BULLETIN 1		BULLETIN 2		BULLETIN 3	
			Y/N/not available	Value	Y/N/not available	Value	Y/N/not available	Value
Seismic events (total number) increase								
RSAM acceleration								
Accumulated energy released rate increase								
Lateral migration of seismicity								
Vertical migration of seismicity								
Deep seismicity								
Shallow seismicity								
VT events increase								
LP events increase								
Tremor events increase								
Hybrid events increase								
Other								
Other								
Other								
Overall seismicity increase (direct observation)								
Gas flux increase								
H2O increase								
CO2 increase								
SO2 increase								
Others								
Fluids temperature increase								
Other								
Other								
Other								
Overall gas increase (visual observation)								
Strain increase								
Inflation rate increase								
Lateral migration								
Vertical migration								
Dg/Dh anomaly								
Other								
Other								
Other								
Overall ground deformation increase (visual observation)								
Fractures								
Phreatic explosions								
Fresh magma								
Other								
Other								
Other								

[Figure 8] Short Term Original Template

The aim of this template was to collect the data obtained through the monitoring networks of the different volcanological observatories and of different episodes of unrest. The most widely used unrest indicators used by the experts were chosen and which include seismic, gas, deformation and other observations such as the presence of fractures, groundwater explosions or the presence of fresh magma. The different parameters are in the rows of the table. In the first column, we must enter the "Background level" of the parameter whose data we are entering. This value will mark the moment when, in the case of having higher parameter values, the volcano will have entered a state of unrest. In the second column we will introduce the “Variation range”, a value from which we will consider that there has been a change with respect to the previous bulletin. The ideal scenario is that these values are introduced by experts from different volcanological observatories. The next column is that of the bulletin. Here we will enter the date, and the value of the parameter that has undergone a change in the "value" column. In the event that we do not have absolute values but relative ones, in the “Y / N / Not available” column, we will include “Y” in the event that the parameter has changed with respect to the previous bulletin, “N” in the case that has not changed, and "Not available" when we do not have information.

The data obtained have been extended by searching to:

- WOVOdat database (https://wovodat.org/)
- Global Volcanism Program, Smithsonian Institution web page (https://volcano.si.edu/)
- Catalogues and bulletins published on the websites of the various observatories, the main ones consulted were:

    - National Geographic Institute (IGN) (https://www.ign.es/web/ign/portal/vlc-area-volcanologia)
    - Volcanological Observatory of Piton de la Fournaise (https://www.ipgp.fr/fr/dernieres-actualites/344)
    - National Institute of Geophysics and Volcanology (INGV) Etnean Observatory. Catania Section (https://www.ct.ingv.it/index.php)

- Publications in scientific journals
- Master's thesis and Doctoral Thesis

Unrest monitoring data has been compiled for the following volcanoes: Asama, Aso, Bezymianny, Chichón, Colima, Dabbahu, Etna, Galeras, St Helens, El Hierro, Fagradalsfjall, Kilauea, Mauna Loa, Merapi, La Palma, Pinatubo, Pitón de la Fournaise, Popocatepetl, Redoubt, Sakurajima, Stromboli, Tenerife, Tungurahua and Unzen.

Unfortunately, although in order to carry out the objectives of this proposal it is not desirable to have to adopt this mostly artisanal methodology, ours is a paradigmatic case. However, in order to find solutions to this problem, this first approach was strictly necessary, as we needed to know in depth the characteristics of the domain in which we are working.

This issue highlights the need for a proposal like ours and, more specifically, the development of tools to automate the maximum number of processes surrounding the data life cycle. These tools should be comfortable, secure, and accessible to all types of users involved, and should be minimally intrusive and most compatible with pre-existing systems. This is where flask microservices come into play.

Microservices Flask

Python Flask microservices, among others, can assume the responsibility of “translator” between technologies, offering a unique method and language of consultation to communicate with the different subsystems. Imagine for example that we have different subsystems each with a completely different query language, a Microservice is able to link a particular query and create a new one adapted to the needs of a subsystem. In this way you can offer the user the feeling of being working on a single system source.

As we will see later, there are cases where the data requires intermediate processes to extraction and storage. In these cases, the power of *Python* can be harnessed to carry them out. In fact, processes can communicate with each other, providing a gateway to the adoption of external tools in case using *Python Scripts* is not the best option. Thus, we can create a whole ecosystem of specialized Microservices with the aim of obtaining maximum efficiency.

It will therefore be necessary to adopt and elaborate tools for extracting, transforming and loading information.

ETL

The tools of information extraction, transformation and loading are very important in architectures composed of different subsystems -as is the case of ours-, as they have the responsibility to act as a link between the different technologies that are involved.

The ultimate goal is to make these tasks automated and linked to microservices that handle requests, but as we will see later, we are currently a long way from that goal.

If we think about our system, as we have described it, there is clearly a flow, and with each advance, the data goes from being potentially unstructured to having a more defined structure. Specifically, and in terms of data storage, we have the following phases:

Data Lake: No hierarchy required, any format is accepted.

Data Warehouse: It follows a hierarchy of directories and formats. Formats can be quite different for the same type of data.

Local Project: It follows a hierarchy of directories and formats. Formats are always the same for the same type of data.

Online Project: The data is indexed following a closed table and relationship scheme.

It is at the midpoint between these phases that *ETL*s make sense.

Data transformation

In the case of *Data Lake*, in order to preserve the original contents, only a loading and extraction process is carried out, so it only depends on microservices that are able to obtain the target content and store it with the corresponding tags. Otherwise, in the case of the *Data Warehouse*, once extracted, the necessary transformations must be carried out to follow its standards. *Volcanbox*, on the other hand, is prepared to carry out all the necessary transformations automatically, when the formats you receive follow the specifications of the Data Warehouse.

To describe these processes, we will take as an example the case of Short and Long term analysis. The data collected from the template shared with partners and enriched by bibliographic data had to be stored into a spreadsheet that has the .ods format and meets the requirements of *Volcanbox*.

Admittedly, some transformations could have been avoided if the final *Volcanbox* compatible template had been available, but it was not yet defined. In future data requests, the new template will be sent in order to avoid this transformation.

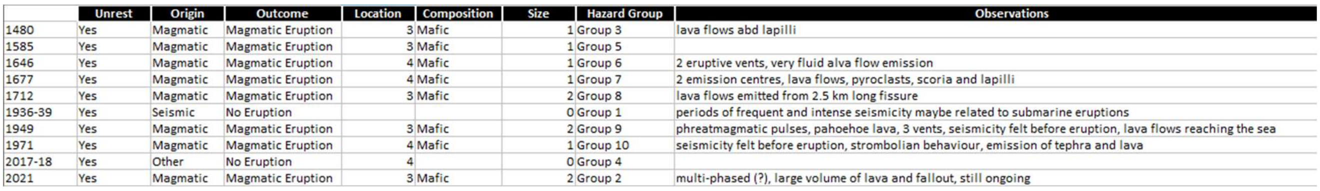
The following is an example for each case:

Hasset Long Term

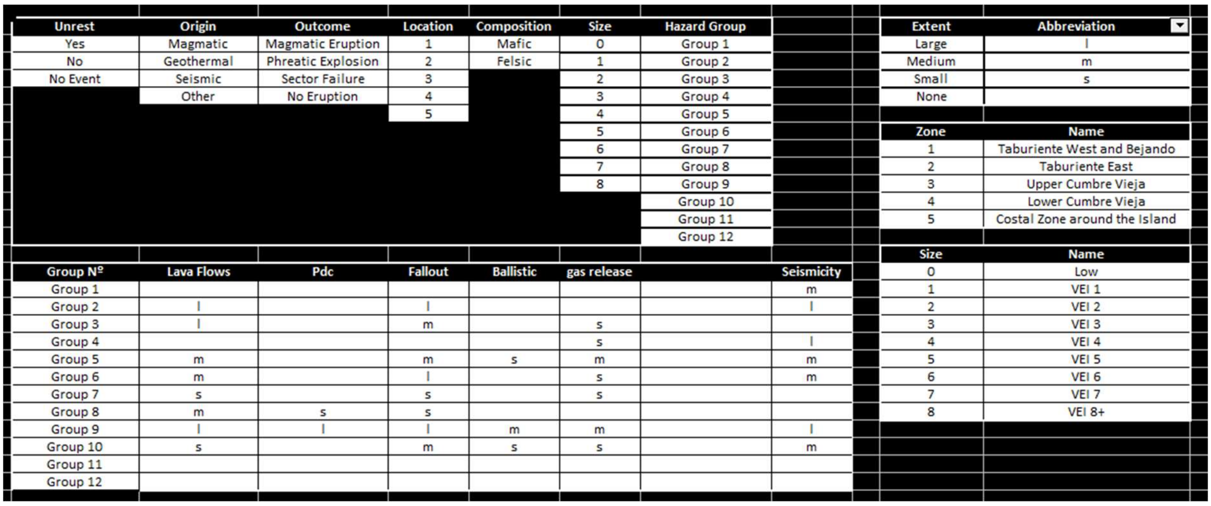
The data collected from the Long Term template must be partially transformed and stored in a spreadsheet. The transformations that must be performed in order to be compatible with *Volcanbox* are as follows:

- Location column. Up to a maximum of 5 areas listed from 1 to 5 will be defined, based on the information collected in the *Long Term Template*. The corresponding number will be entered in the "location" field.
- Hazard Group column. Up to 12 Hazard Groups will be defined, listed from 1 to 12. Each of the groups consists of a combination of hazard and extent. The corresponding number will be entered in the "Hazard Group" column.

The following is an example of the spreadsheet corresponding to La Palma transformed to be uploaded in the *Hasset Long term*:



	Unrest	Origin	Outcome	Location	Composition	Size	Hazard Group	Observations
1480	Yes	Magmatic	Magmatic Eruption	3	Mafic	1	Group 3	lava flows abd lapilli
1585	Yes	Magmatic	Magmatic Eruption	3	Mafic	1	Group 5	
1646	Yes	Magmatic	Magmatic Eruption	4	Mafic	1	Group 6	2 eruptive vents, very fluid alva flow emission
1677	Yes	Magmatic	Magmatic Eruption	4	Mafic	1	Group 7	2 emission centres, lava flows, pyroclasts, scoria and lapilli
1712	Yes	Magmatic	Magmatic Eruption	3	Mafic	2	Group 8	lava flows emitted from 2.5 km long fissure
1936-39	Yes	Seismic	No Eruption			0	Group 1	periods of frequent and intense seismicity maybe related to submarine eruptions
1949	Yes	Magmatic	Magmatic Eruption	3	Mafic	2	Group 9	phreatmagmatic pulses, pahoehoe lava, 3 vents, seismicity felt before eruption, lava flows reaching the sea
1971	Yes	Magmatic	Magmatic Eruption	4	Mafic	1	Group 10	seismicity felt before eruption, strombolian behaviour, emission of tephra and lava
2017-18	Yes	Other	No Eruption	4		0	Group 4	
2021	Yes	Magmatic	Magmatic Eruption	3	Mafic	2	Group 2	multi-phased (?), large volume of lava and fallout, still ongoing



Unrest	Origin	Outcome	Location	Composition	Size	Hazard Group
Yes	Magmatic	Magmatic Eruption	1	Mafic	0	Group 1
No	Geothermal	Phreatic Explosion	2	Felsic	1	Group 2
No Event	Seismic	Sector Failure	3		2	Group 3
	Other	No Eruption	4		3	Group 4
			5		4	Group 5
					5	Group 6
					6	Group 7
					7	Group 8
					8	Group 9
						Group 10
						Group 11
						Group 12

Extent	Abbreviation
Large	l
Medium	m
Small	s
None	

Zone	Name
1	Taburiente West and Bejando
2	Taburiente East
3	Upper Cumbre Vieja
4	Lower Cumbre Vieja
5	Costal Zone around the Island

Size	Name
0	Low
1	VEI 1
2	VEI 2
3	VEI 3
4	VEI 4
5	VEI 5
6	VEI 6
7	VEI 7
8	VEI 8+

Group Nº	Lava Flows	Pdc	Fallout	Ballistic	gas release	Seismicity
Group 1						m
Group 2	l		l			l
Group 3	l		m		s	
Group 4					s	l
Group 5	m		m	s	m	m
Group 6	m		l		s	m
Group 7	s		s		s	
Group 8	m	s	s			
Group 9	l	l	l	m	m	l
Group 10	s		m	s	s	m
Group 11						
Group 12						

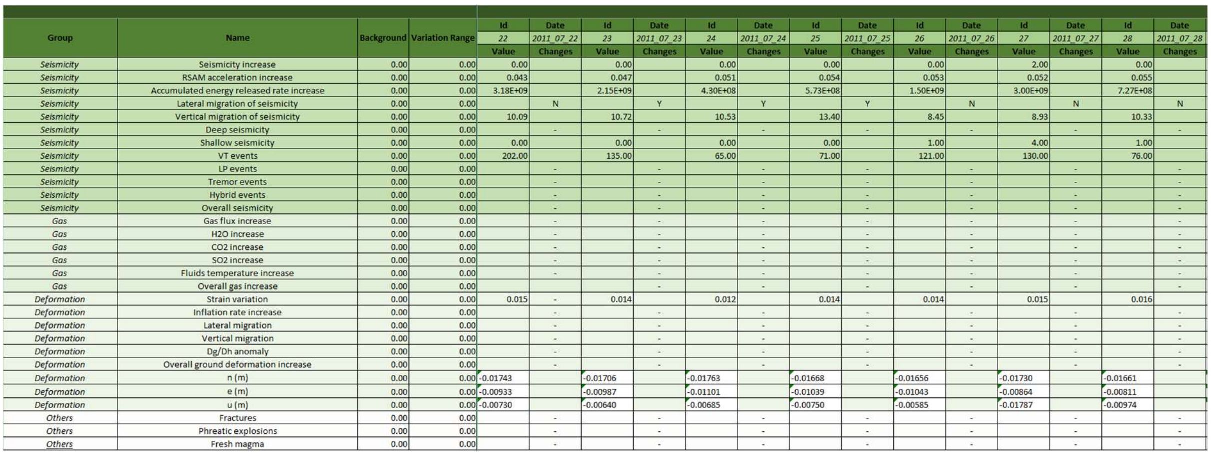
[Figure 9] Long Term new Template

Hasset Short Term

As for unrest data, compiled from the Short Term Template and expanded with data from catalogues and bulletins published by various volcanological observatories (among others), they have also been transformed and stored in spreadsheets in .ods format. The transformations that have had to be carried out in order to meet the requirements of the *Hasset Short Term* are as follows:

- Date must be in YYYY_MM_DD format
- The "Id" column should include the bulletin number
- Numbers with decimals must be 00.00 (English format)

The following is an example from the El Hierro transformed spreadsheet for inclusion to the *Hasset Short Term*:



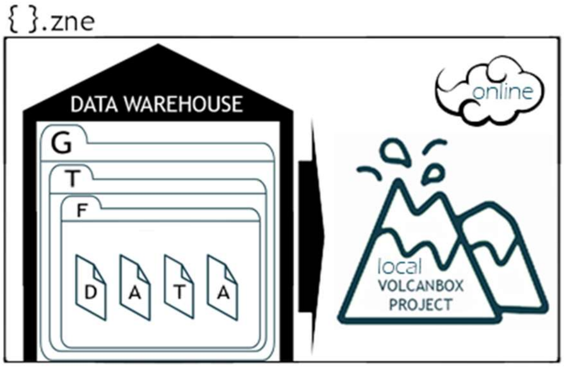
Group	Name	Background	Variation Range	Id	Date	Id	Date	Id	Date	Id	Date	Id	Date	Id	Date	Id	Date
				22	2011_07_22	23	2011_07_23	24	2011_07_24	25	2011_07_25	26	2011_07_26	27	2011_07_27	28	2011_07_28
				Value	Changes	Value	Changes	Value	Changes	Value	Changes	Value	Changes	Value	Changes	Value	Changes
Seismicity	Seismicity increase	0.00	0.00	0.00		0.00		0.00		0.00		0.00		2.00		0.00	
Seismicity	RSAM acceleration increase	0.00	0.00	0.043		0.047		0.051		0.054		0.053		0.052		0.055	
Seismicity	Accumulated energy released rate increase	0.00	0.00	3.18E+09		2.15E+09		4.30E+08		5.73E+08		1.50E+09		3.00E+09		7.27E+08	
Seismicity	Lateral migration of seismicity	0.00	0.00		N		Y		Y		Y		N		N		N
Seismicity	Vertical migration of seismicity	0.00	0.00	10.09		10.72		10.53		13.40		8.45		8.93		10.33	
Seismicity	Deep seismicity	0.00	0.00		-		-		-		-		-		-		-
Seismicity	Shallow seismicity	0.00	0.00	0.00		0.00		0.00		0.00		1.00		4.00		1.00	
Seismicity	VT events	0.00	0.00	202.00		135.00		65.00		71.00		121.00		130.00		76.00	
Seismicity	LP events	0.00	0.00		-		-		-		-		-		-		-
Seismicity	Tremor events	0.00	0.00		-		-		-		-		-		-		-
Seismicity	Hybrid events	0.00	0.00		-		-		-		-		-		-		-
Seismicity	Overall seismicity	0.00	0.00		-		-		-		-		-		-		-
Gas	Gas flux increase	0.00	0.00		-		-		-		-		-		-		-
Gas	H2O increase	0.00	0.00		-		-		-		-		-		-		-
Gas	CO2 increase	0.00	0.00		-		-		-		-		-		-		-
Gas	SO2 increase	0.00	0.00		-		-		-		-		-		-		-
Gas	Fluids temperature increase	0.00	0.00		-		-		-		-		-		-		-
Gas	Overall gas increase	0.00	0.00				-		-		-		-		-		-
Deformation	Strain variation	0.00	0.00	0.015	-	0.014		0.012		0.014		0.014		0.015		0.016	
Deformation	Inflation rate increase	0.00	0.00		-		-		-		-		-		-		-
Deformation	Lateral migration	0.00	0.00		-		-		-		-		-		-		-
Deformation	Vertical migration	0.00	0.00		-		-		-		-		-		-		-
Deformation	Dg/Dh anomaly	0.00	0.00		-		-		-		-		-		-		-
Deformation	Overall ground deformation increase	0.00	0.00		-		-		-		-		-		-		-
Deformation	n (m)	0.00	0.00	-0.01743		-0.01706		-0.01763		-0.01668		-0.01656		-0.01730		-0.01661	
Deformation	e (m)	0.00	0.00	-0.00933		-0.00987		-0.01101		-0.01039		-0.01043		-0.00864		-0.00811	
Deformation	u (m)	0.00	0.00	-0.00730		-0.00640		-0.00685		-0.00750		-0.00585		-0.01787		-0.00974	
Others	Fractures	0.00	0.00		-		-		-		-		-		-		-
Others	Phreatic explosions	0.00	0.00		-		-		-		-		-		-		-
Others	Fresh magma	0.00	0.00		-		-		-		-		-		-		-

[Figure 10] Short Term new Template

It should be said that this and other processes can be carried out with the help of tools such as *Hevo Data, Pentaho kettle, GeoKettle, Python scripts* etc.

Volcanic Zone

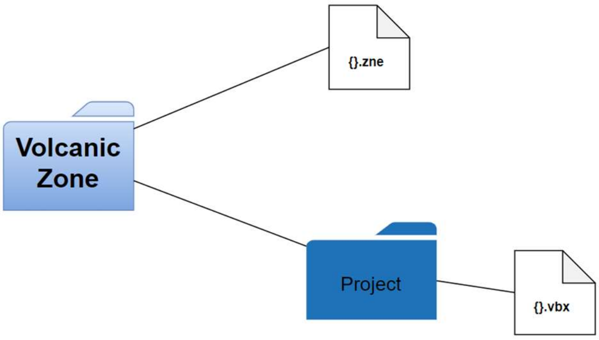
A volcanic area refers to the area covered by certain data in terms of georeferencing.



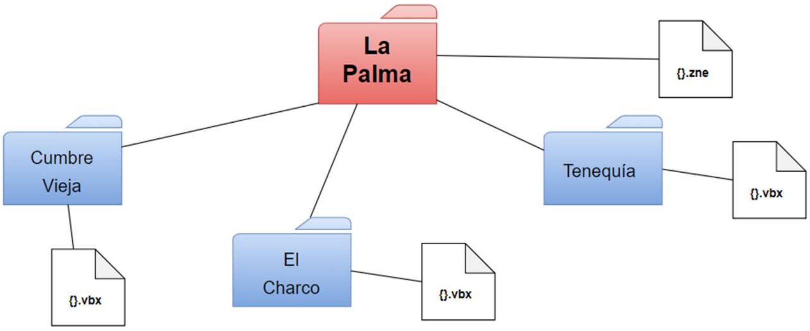
[Figure 11] Volcanic Zone

For each *Volcanic Zone,* there will be a *Data Warehouse* that will contain all the data referring to its geographical extension -Basically, in terms of implementation, it is actually a large distributed warehouse that contains all the warehouses of all the volcanic zones, but we have thought that giving this vision would help the user to work more comfortably and focus on the case of study in question-.

On the other hand, when you want to study a subzone - which may contain data from one or more volcanoes - a new *Volcanbox Project* will be created.



[Figure 12] Volcanic Zone Folder Structure Abstraction

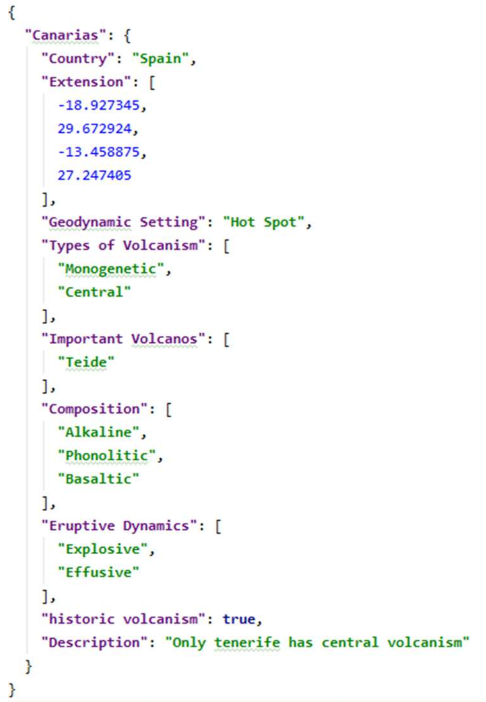


[Figure 13] Volcanic Zone Folder Structure Example

For example, *La Palma* is a *Volcanic Zone* that in geographical terms contains the following volcanoes: El Charco, Cumbre Vieja and Teneguía. The *Data Warehouse* of La Palma will therefore contain all the data needed to create a *Volcanbox Project* for each of the volcanoes. These projects will be stored within the same volcanic area to which the volcano belongs. It may also be the case that it contains projects for two or more volcanoes.

On the other hand, all zones contain a metadata file in *Json* format with '*.zne'* extension which, similar to *Data Lake* tags, allows you to index the information to retrieve it by applying different filters. The file contains the following fields:

- **Name:** Name of the Volcanic Zone.
- **Country:** Country
- **Extension:** Geographical coordinates that refer to the total Volcanic Zone.
- **Geodynamic Setting:** Tectonic regime that characterises the Volcanic Zone -for example, subduction, ridge, oceanic hotspot, etc.-
- **Types of Volcanism:** Describes whether they are central volcanoes or monogenetic fields.
- **Important Volcanoes:** List of most representative volcanoes in the area.
- **Composition:** Main chemical composition of magma.
- **Eruptive dynamics:** Briefly describe the main types of eruptions.
- **Historical volcanism:** Existence or not of historical volcanism with eruption or not.
- **Description:** Field where the user can enter extra information or which does not fit in the other fields.

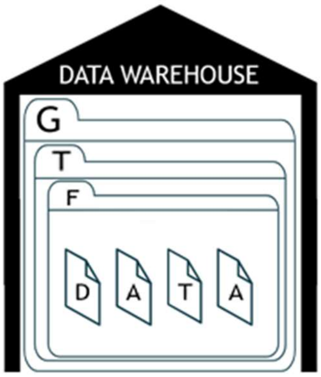


```
{
  "Canarias": {
    "Country": "Spain",
    "Extension": [
      -18.927345,
      29.672924,
      -13.458875,
      27.247405
    ],
    "Geodynamic Setting": "Hot Spot",
    "Types of Volcanism": [
      "Monogenetic",
      "Central"
    ],
    "Important Volcanos": [
      "Teide"
    ],
    "Composition": [
      "Alkaline",
      "Phonolitic",
      "Basaltic"
    ],
    "Eruptive Dynamics": [
      "Explosive",
      "Effusive"
    ],
    "historic volcanism": true,
    "Description": "Only tenerife has central volcanism"
  }
}
```

[Figure 14] Volcanic Zone Metadata File Content

Data Warehouse

Following the architecture design, we have been inspired by the Data Warehouses of Big Data environments to create the next piece of our system. It is common to see articles where the virtues of a *Data Lake* are confronted with those of a *Data Warehouse*. However, when it comes to polyglot architectures like ours, the two subsystems can coexist and add value to the system.

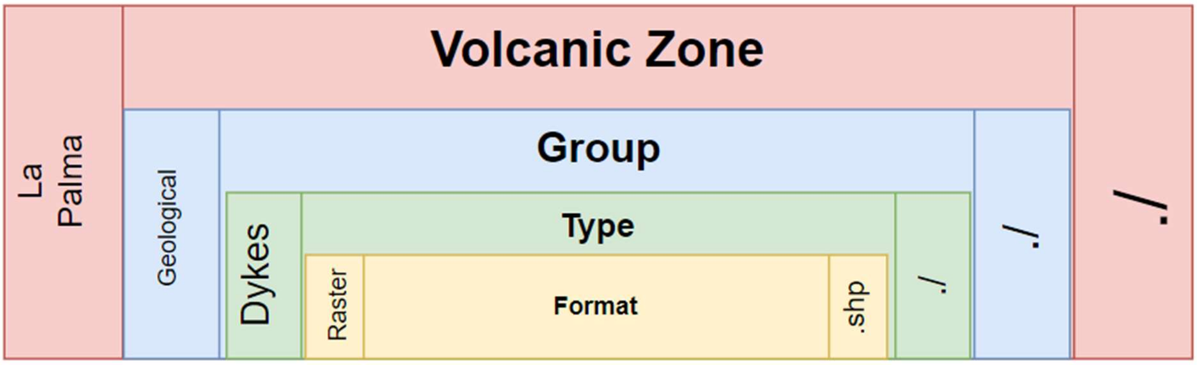


[Figure 15] Data Warehouse

In our case it is a very simple *Data Warehouse* where only data that has gone through validation processes is stored where the necessary transformations have been applied to be compatible with *Volcanbox*. These transformation processes only apply if necessary. *Volcanbox* Application accepts several geodata formats, and any format accepted can be stored into the Data Warehouse.

This philosophy is followed to try to make minimal changes to the original data and to be able to recover it, because otherwise only the unvalidated version of *Data Lake*, or the modified version of the *Volcanbox Project*, could be recovered, probably containing modifications or errors.

For example, in the case of *Spatial Analysis*, in order to be able to retrieve the contents stored in the *Data Warehouse* of a volcanic zone, the following directory hierarchy has been defined:



[Figure 16] Data Warehouse Structure

- **Group:** Set of information of the same type, for example, Geological, Geographical, Volcanological, Infrastructure,
    - **Type:** Refers to the type of structural element - Example: 'wind', dyke ',' fault ', etc.-
        - **Format:** Content type, for example, 'Document', 'raster', 'shape', 'chart','spreadsheet', etc.

Volcanbox

Given a volcanic area, a project is a directory that contains all the structured data needed to reproduce a risk assessment using the *Volcanbox* application, as well as its results.

The application is divided into sections - such as *Short Term Analysis, Probability Density Function, Lava Flows,* etc. - and these are grouped according to the type of analysis being performed. In order to use these sections, the user must first create a new project and choose or create a *Volcanic Zone* where to place it. The input and output data of the different experiments that the user carries out will be distributed in *Datasets* that the user will be able to store within the Local *Project*.

If you want to use sections with GIS functionality, the user will have to select a main digital elevation model, this action must be carried out at the time of creation of the project. Important information will be extracted from this model, such as: the geographical extension covered by the project, the height for each point of it, the resolution, the geographical reference system that will be used as a basis, etc.

This way, when a user wants to retrieve an experiment, all he has to do is open the target project, select the *Dataset* referring to it, and then continue working at the point where he left it. In case this section includes GIS sections, its *Datasets* will be associated with one more layer.

The goal is for the datasets to end up enriching both the *Data Lake* and the *Data Warehouse*. For example, imagine that a user obtains certain results using the *Volcanbox* application, and once validated, decides to enter them in *Data Lake*. Depending on the tags you use - or as determined by the system - you will be able to relate them to pre-existing data for: Carrying out comparisons, new analysis with Big Data techniques, generating new data sets, etc.

The application contains a section called *Library*, where the user can consult, upload or download, for each volcanic zone, all existing projects, as well as their datasets. In the current version of the application, you can only work with projects located on the computer where it is running, but it is planned to offer a network connection to work with remote libraries.

If we were to make a simile once again with the world of Big Data, the *Volcanbox* application would be a data mart.

To be able to create a project using the Volcanbox application, the user must first have created or imported at least one *Volcanic Zone* - choosing one is a *sine qua non condition* for creating a new project. The projects - as well as the *Volcanic Zones* - also contain a *'.vbx'* metadata file where a *Volcanic Zone* field is added. This field contains a replica of the entire contents of the *'.zne'* file in the Volcanic Zone that was selected when you created the project. This process is carried out to allow users to import projects even though they do not have information regarding this volcanic zone. This is information that, despite being replicated, is not very important in terms of disk space.

The software, then, when an import is carried out, checks if this *Volcanic Zone* exists. If not, ask the user if they want to create this zone from the imported metadata, or prefer to specify it manually. If so, in case there is a difference, ask the user if you want to update the local zone- based on the metadata of the imported project - or instead keep the existing ones.

[Figure 17] Local Volcanbox Project

Since the application is designed to load and store data both locally and remotely - in future updates - two very different approaches have been designed.

Local Project

As for the local version, one of the initial requirements of the application is that the results generated are searchable by the most used external applications on the market - regardless of the hardware or operating system in which they run - therefore, they need to be saved in formats that are compatible with them. To meet this requirement, the following formats have been chosen as the main pillars for storing information:

- **GeoTiff** : Its main advantages are its suitability for a wide range of applications and its independence from computer architecture, operating system and graphics hardware.

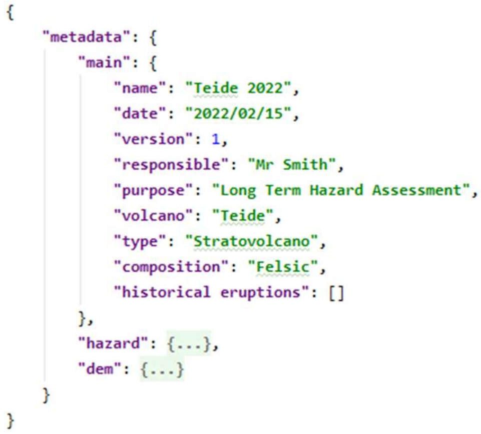
- **JSON** : It is a format, in plain text, this fact makes it suitable and secure for transfer between platforms and operating systems that do not easily share more complex types of documents. It is lightweight and its syntax and structure can be easily interpreted by applications that do not yet know what type of data they will receive.

- **Shape** : Its simple structure allows you to spatially describe vector features: points, lines and polygons, which represent, for example, winds, fissures, dykes, etc. Each element can have attributes that describe it.

All volcanic areas, as well as projects in this version, are saved in the *Volcanbox Library* folder. This folder is structured in folders referring to Volcanic Zones and these are structured in folders referring to *Projects*. All folders related to *Volcanic Zones* contain a *Json* format file with a *'.zne'* extension at the root of their directory, as well as those referring to projects with a *'.vbx'* file. These contain your metadata and are essential for its execution.

- **Name**: Name of the project.
- **Date**: Date the project was created.
- **Version**: Project version.
- **Responsible**: Responsible for the project.
- **Purpose**: Description of the project objectives.
- **Volcano**: Name of the volcano.
- **Type**: Volcanic building type - for example: shield volcano, stratovolcano, caldera, dome, scoria cone, maar, tuff ring, tuff cone, fissure.
- **Historical eruptions**: Dates of representative historical eruptions.
- **Hazard**: Contains for each danger to be studied, the conventions referring to the range of values that correspond to a long, medium, or small extension.
- **Dem**: Contains metadata related to the main elevation model of the project.

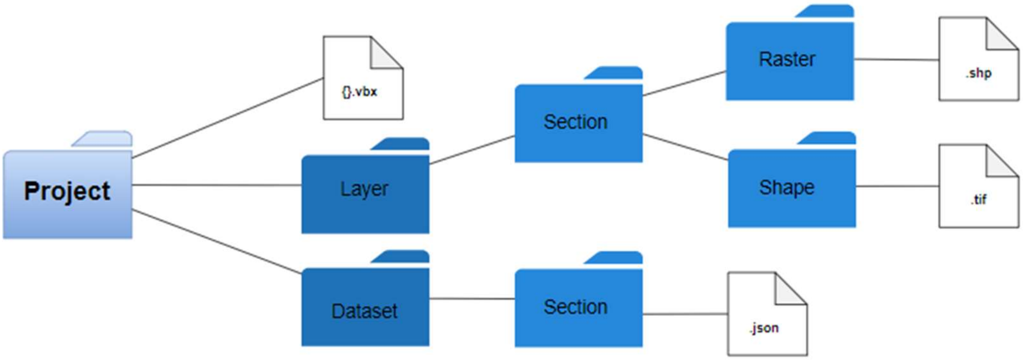
- **Volcanic Zone**: Replica of the content of the '*.zne*' file of the zone to which the project belongs.



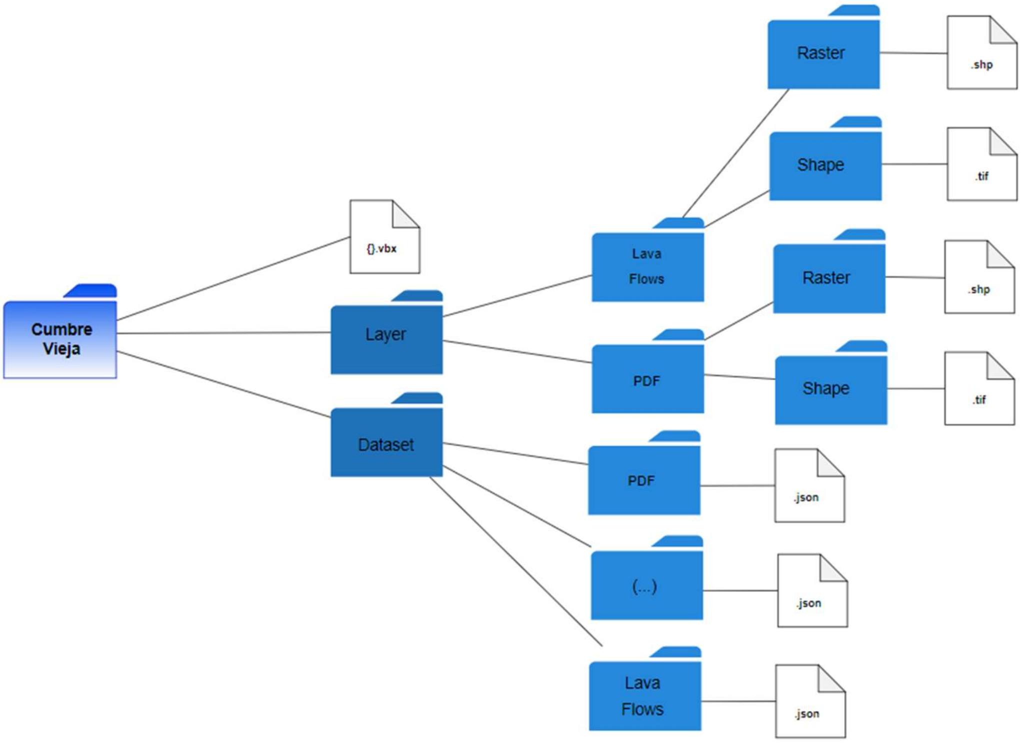
```
{
    "metadata": {
        "main": {
            "name": "Teide 2022",
            "date": "2022/02/15",
            "version": 1,
            "responsible": "Mr Smith",
            "purpose": "Long Term Hazard Assessment",
            "volcano": "Teide",
            "type": "Stratovolcano",
            "composition": "Felsic",
            "historical eruptions": []
        },
        "hazard": {...},
        "dem": {...}
    }
}
```

[Figure 18] Project Metadata File Content

Each *Project* will also contain two directories, one to store the *Datasets* in *Json* format and another to store the layers associated with them in *Geotiff* and Shape format.



[Figure 19] Project Folder Structure Abstraction



[Figure 20] Project Folder Structure Example

This directory structure is explained in detail below:

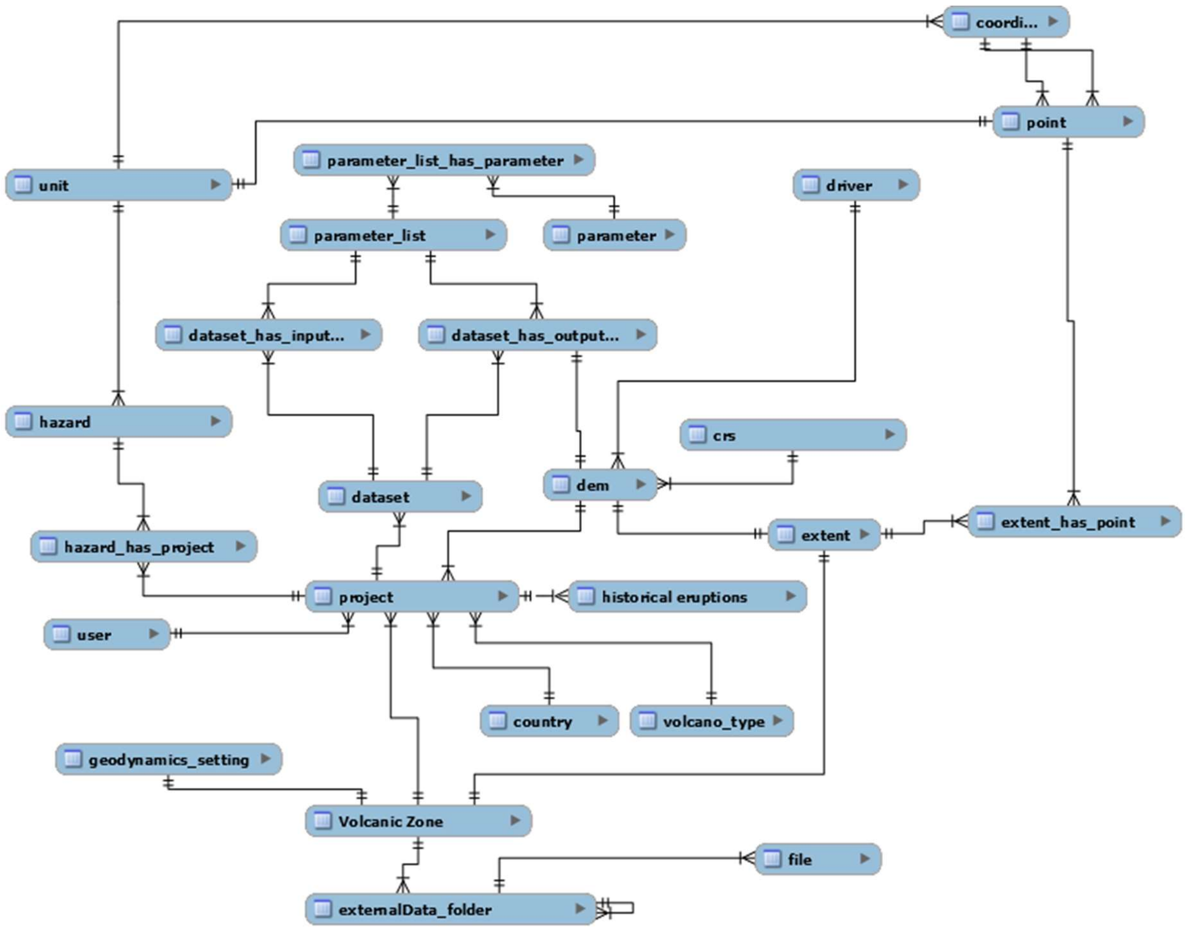
- **Volcanbox Library:**
    - **Volcanic Zone:**
        - **Project Name** : The root directory of the project is the only one that the user can name, but it is a requirement that if it exists, it is empty at the time of creating the project.
            - **Dataset** : Contains a directory for each available section
                - **Section** : For a given section, it contains all the *Datasets* belonging to the project in question.
            - **Layers** : Contains a directory called Main Dem, also contains a directory for each section.
                - **Main Dem** : This directory contains all the files related to the main digital elevation model of the project.
                - **Section** : Contains, for a given section, a directory for each type of georeferenced data file generated by the application.
                    - **Shape** : Contains the vector files of the project.
                    - **Raster** : Contains the maps generated with the application - elevation models, probability density maps, etc.

In terms of *Project* loading and storage, the online functionalities of the application are designed to facilitate collaboration between members of the same or different teams. On the other hand, in the medium term, the goal is to use *Volcanbox* to create a large database with which, thanks to standardization, computers can cross-check their data with those of other computers - provided they have of the appropriate permissions-, thus generating new content that creates a chain of feedback that exponentially increases the use of resources. For example, a user will be able to create a new project with their own information and enrich it to carry out operations such as:

- Compare the results obtained for the same *Volcanic Zone*.
- Compare the behaviour of two different *Volcanic Zones*, for example, to find patterns that offer knowledge.
- Enrich parameters for which not enough information is available.
- Real-time data provided.

On the other hand, despite being a desktop application, the prospect of generating software that can be easily translated into an online version has never been lost. This will allow taking advantage of the power of the current supercomputers to be able to work with models of high resolution, and to carry out operations that would not be possible in an average hardware.

In order for the application to be ready to support these features, the following conceptual scheme has been designed:



[Figure 21] Online Project dB Design

- **Volcanic Zone** :
  - Describes a particular volcanic area.

- **Project**
  - Describes the metadata described in the *'.vbx'* file. Either through their fields or through their relationships.

- **Section**
  - Describe the sections available in the app. *This table does not allow insertions by customers.*

- **Dataset**
  - For a certain section, it resolves the metadata needed to obtain all the data needed to run the experiment associated with it, as well as to retrieve the results.

- **Parameter**
  - Describes an input or output parameter of a dataset for a given section.

- **Hazard**

    - Describe the metadata of a type of hazard to study. It also solves some conventions, such as the range of values that correspond to a long, medium, or small extension.

- **Unit**
    - Describes a type of unit of measure.

- **Coordinates**
    - Describes the value of a coordinate.

- **Point**
    - Describe a location within the extent of the volcanic subzone under study. You can also describe the value for a couple of coordinates.
- **Driver**
    - Describes a geospatial data format compatible with *Volcanbox*.

- **User**
    - Describes a user's credentials. These are the email, first name, last name, and password

- **Geodynamics Setting**
    - Describes the possible tectonic regimes of a *Volcanic Zone*.

- **Country**
    - Describes the name of a country, and its reference code.

- **Volcano Type**
    - Describe the type of volcanic building.

- **Extent**
    - Describes a geographical extension for a layer or a Volcanic Zone.

- **Layer**
    - Describes a set of georeferenced data.

- **Crs**
    - Describes a coordinate reference system.

- **External Data Folder**
    - Describes the directories where the data that is structured within a given file is stored.

- **File** :
    - Describes the name, extension, format, and description of a file.

To simplify, all the tables referring to institutions, position, etc. have been omitted.
In the event that a point contains a value, it must be linked to a unit that will describe the unit of measure of the value of the point.

All points in a layer of a given project must be within the extent of a volcanic zone of a project.

The fact that we have chosen a relational scheme has been because in the end point of the chain where we are, where the data has been structured following certain protocols, we can already store them following a predefined scheme, because we also know that this will not change if the *Volcanbox* application does not, which would not escape our control. This allows us to take advantage of all the benefits of relational databases, without sacrificing more innovative features - which we will have to find in previous points of the gear, but which will be there.

Consultation and Visualization

Consultation

In order to provide the reader with an example of how the different contents of the *Volcanbox Platform* are structured, a service has been enabled that communicates with the different parts of the platform. The documentation related to this is interactive and can be consulted via the following link:

https://www.volcanbox.com:5000/api/

By accessing the aforementioned link, the user will be able to see a list of the different microservices that can be consulted and for each of these the end points to the consultation methods currently available. For each method a general description is shown, the type of method and a description of the input and output parameters, in addition, an interactive form is also included that allows you to enter the input parameters to make a request of execution.

At the time of writing, only server-level security protocols have been implemented, but not in service level. It is important to note that the purpose of this service is simply to allow users to download a sample of the content that has been generated. On the other hand, it can be useful to give an idea of how the whole system may or may not be transparent to the user depending on the needs, permissions, etc.

The final points of interest currently implemented can be consulted through the section, *Volcanbox*, via the following link:

https://www.volcanbox.com:5000/api/volcanbox/

This works as a discovery and therefore returns all available query methods. Similarly, a direct discovery of the different subcomponents can now be made using the following links:

https://www.volcanbox.com:5000/api/volcanbox/lake/

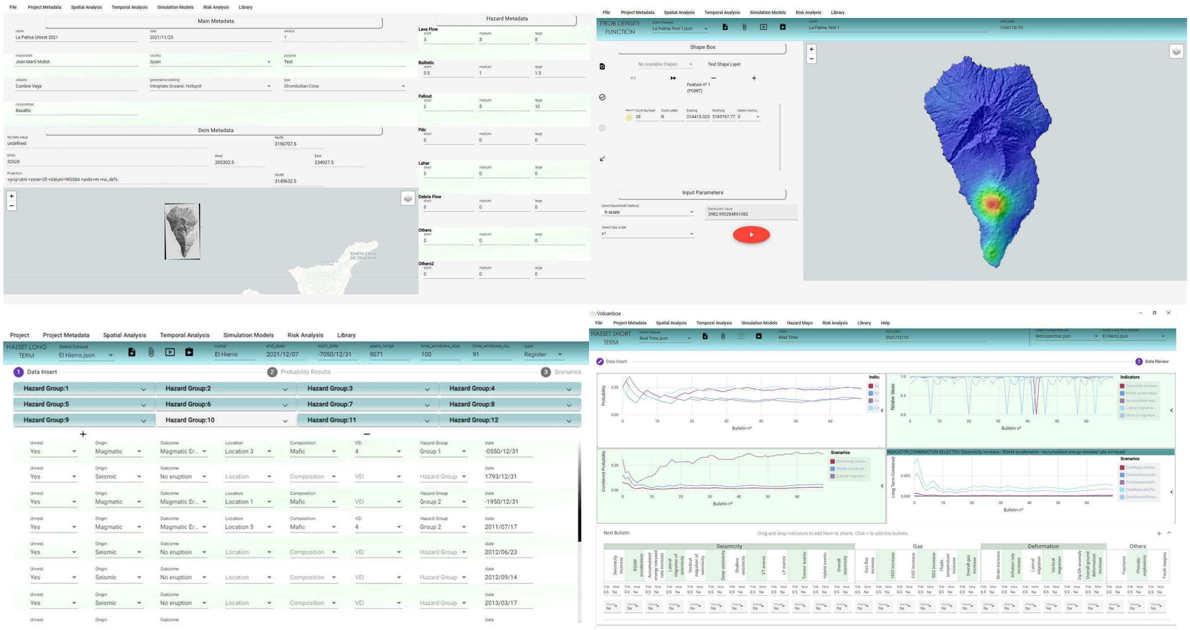
https://www.volcanbox.com:5000/api/volcanbox/warehouse/

https://www.volcanbox.com:5000/api/volcanbox/project/

For a better understanding of the methods and as an interactive tutorial, it is recommended to use the respective section of the documentation.

Visualization

In order to visualize the information, a great effort has been made so that the user can choose the application that he decides, regardless of, at what stage in terms of structuring it is. It is recommended, but from now on, to work with the *Volcanbox* application - the launch of its first release is imminent - to consult and perform new evaluations….



Main Metadata

name: La Palma Unrest 2021
date: 2021/11/25
version: 1
responsible: Joan Martí Molist
country: Spain
purpose: Test
volcano: Cumbre Vieja
geodynamicsSetting: Intraplate Oceanic Hotspot
type: Strombolian Cone
composition: Basaltic

Dem Metadata

No Data Value: undefined
North: 3196707.5
EPSG: 32628
West: 205302.5
East: 234927.5
Projection: +proj=utm +zone=28 +datum=WGS84 +units=m +no_defs
South: 3149632.5

Hazard Metadata

	short	medium	large
Lava Flow	3	5	8
Ballistic	0.5	1	1.5
Fallout	2	5	10
Pdc	0	0	0
Lahar	0	0	0
Debris Flow	0	0	0
Others	0	0	0
Others2	0	0	0

PROB DENSITY FUNCTION
Select Dataset: La Palma Test 1.json
name: La Palma Test 1
start_date: 1000/10/10

Shape Box

No Available Shapes
Test Shape Layer
Feature nº 1 (POINT)

Zone Number	Zone Letter	Easting	Northing	Select Hemis...
28	R	214415.325	3185767.77	S

Input Parameters

Select BandWidth Method: h score
Bandwidth Value: 3982.950284861082
Select Size Scale: x1

HASSET LONG TERM
Select Dataset: El Hierro.json
name: El Hierro
end_date: 2021/12/07
start_date: -7050/12/31
years_range: 9071
time_windows_size: 100
time_windows_nu...: 91
type: Register

1 Data Insert
2 Probability Results
3 Scenarios

Hazard Group:1
Hazard Group:2
Hazard Group:3
Hazard Group:4
Hazard Group:5
Hazard Group:6
Hazard Group:7
Hazard Group:8
Hazard Group:9
Hazard Group:10
Hazard Group:11
Hazard Group:12

Unrest	Origin	Outcome	Location	Composition	VEI	Hazard Group	date
Yes	Magmatic	Magmatic Er...	Location 3	Mafic	4	Group 1	-0550/12/31
Yes	Seismic	No eruption	Location	Composition	VEI	Hazard Group	1793/12/31
Yes	Magmatic	Magmatic Er...	Location 1	Mafic	VEI	Group 2	-1950/12/31
Yes	Magmatic	Magmatic Er...	Location 5	Mafic	4	Group 2	2011/07/17
Yes	Seismic	No eruption	Location	Composition	VEI	Hazard Group	2012/06/23
Yes	Seismic	No eruption	Location	Composition	VEI	Hazard Group	2012/09/14
Yes	Seismic	No eruption	Location	Composition	VEI	Hazard Group	2013/03/17

HASSET SHORT TERM
Select Dataset: Real Time.json
name: Real Time
start_date: 2021/12/10
Select Comparative DS: Retrospective.json
Select Long Term Dataset: El Hierro.json

Data Insert
2 Data Review

INDICATOR COMBINATION SELECTED: [Seismicity increase - RSAM acceleration - Accumulated energy released rate increase]

Next Bulletin

Drag and drop indicators to add them to charts. Click + to add this bulletin.

	Prob	Value	Change
Seismicity: Seismicity increase	0.5	Na	Na
Seismicity: RSAM acceleration	0.5	Na	Na
Seismicity: Accumulated energy released rate increase	0.5	Na	Na
Seismicity: Lateral migration of seismicity	0.5	Na	Na
Seismicity: Vertical migration of seismicity	0.5	Na	Na
Seismicity: Deep seismicity	0.5	Na	Na
Seismicity: Shallow seismicity	0.5	Na	Na
Seismicity: VT events	0.5	Na	Na
Seismicity: LP events	0.5	Na	Na
Seismicity: Tremor events	0.5	Na	Na
Seismicity: Hybrid events	0.5	Na	Na
Seismicity: Overall seismicity	0.5	Na	Na
Gas: Gas flux increase	0.5	Na	Na
Gas: H2O increase	0.5	Na	Na
Gas: CO2 increase	0.5	Na	Na
Gas: SO2 increase	0.5	Na	Na
Gas: Fluids temperature increase	0.5	Na	Na
Gas: Overall gas increase	0.5	Na	Na
Deformation: Strain increase	0.5	Na	Na
Deformation: Inflation rate increase	0.5	Na	Na
Deformation: Lateral migration	0.5	Na	Na
Deformation: Vertical migration	0.5	Na	Na
Deformation: Dg/Dh anomaly	0.5	Na	Na
Deformation: Overall ground deformation increase	0.5	Na	Na
Others: Fractures	0.5	Na	Na
Others: Phreatic explosions	0.5	Na	Na
Others: Fresh magma	0.5	Na	Na

[Figure 22] Example of Volcanbox Desktop App Visualizations

Future Work

Nowadays, when we talk about *Volcanbox*, we can do it as desktop software or as a platform to which it belongs.

The *Volcanbox Platform* encompasses all the solutions we have described and has been designed with a dual perspective, on one hand, it is itself a database and on the other hand it has a set of applications that help scientists predict when, how and where a future volcanic eruption may occur and when necessary to issue volcanic alerts (volcanic V*EWS*).

This platform has been designed to be implemented in different stages. Given that this is a project that will last over time, its scalability has been considered since its initial planning.

This platform will require continuous improvements and updates in order to become an efficient tool. Thanks to the **EVE** project, we have reached the first stage. However, the design covers a larger domain than the project itself in order to avoid that the content generated becomes part of the problem to be solved.

One conclusion reached is that *Data Lake* is a key piece in both centralized and standardized contents and making the most of it. We can say that this piece - by its nature - already existed, but it had not been given an entity or proposed a method to govern it. As part of this first stage, the work that has been carried out in terms of data obtaining is only a demo service, which allows you to consult the available tags and extract the content available for them. However, this is merely conceptual and lacks many features and content that will be implemented in future projects.

Concerning *ETL* tools, we have designed a set of templates, which have allowed us to structure the data extracted from the contents of the *Data Lake* - obtaining the desired results-. In this process, we have found the difficulties that can lead to errors, and we have made the appropriate changes to the data insert sections of the *Volcanbox Application*. Unfortunately, the part that connects *Data Lake* to the *Data Warehouse* still requires substantial user's manual effort, nevertheless, we are working to solve this problem.

The *Data Warehouse* is defined as a phase between open and close structure of data. It's simple, but that makes it user-friendly. It currently meets our needs, but as users begin to adopt the system, its complexity could increase. In this sense, we think that in order to make the *Data Warehouse* a good component for all users, good communication with them is important.

We have defined the formats and the data structure for the application Volcanbox. In addition, we have implemented its own *ETL and* Gis tools, the *Short and Long Term* analysis modules, the library and Susceptibility maps sections and partially the simulation models. Conversion tasks have also been performed to make it cross-platform -Mac, Linux, Windows-.

Although the application is in an advanced stage of development, the platform needs further work in terms of implementation. In particular, the efforts carried out in future projects must be aimed at automating the processes of content extraction, loading and transformation, as well as designing and implementing the necessary algorithms to take advantage of the possibilities of the different designed components. It is worth noting, however, that these tools must be created as knowledge of the nature of the data is gained. On the other hand, with regard to the *Volcanbox Application*, the version we are currently testing has not yet implemented the volcanic zones - we are working directly on a project scale - as this is a relatively new feature that generates some breaking changes. These changes will be added to the next beta test release. In addition, a partial part of the simulation models needs to be finished. The *Early Warning System* graphical interface of the application, has to be implemented. As well as a beta testing period to improve overall performance and stability.

The new funding approved will allow implementing all these functionalities, that given the temporary limitations of the project have not been implemented.