

Earthquake Analytics

Marco Calzana

Abstract

For many years observatories have been collecting earthquakes information. More recently, the world wide web has become the cheapest way to publish this data, so inexpensive that the information about more than 250 thousands earthquakes has been released from the Italian INGV Earthquake Observatory between 2006 and 2017.

Unfortunately, the majority of this information is available just as raw data. Thus, any form of analytics on this data is hardly possible without dedicated analysis tools that are not publicly available, for example to visualize the evolution of foreshocks and aftershocks in the case of big earthquakes.

The goal of this project is to develop an analytics platform to visualize and interact with data stemming from the INGV Observatory. This entails modeling and processing the INGV data stream and creating an interactive website which allows users to visualize all the collected data and interact with them. We will illustrate two use cases of our project that show how Earthquake Analytics can be used to support effective analytics on the earthquake data published by INGV.

Advisor

Prof. Dr. Michele Lanza

Assistant

Dr. Andrea Mocci

Advisor's approval (Prof. Dr. Michele Lanza):

Date:

Contents

1	Introduction	2
1.1	Motivation	2
1.2	Physics Of Earthquakes	3
2	State of the Art	4
3	Project requirements and analysis	5
3.1	Information Collection	5
3.2	Earthquake Data Visualization	5
3.3	Show an Interactive Map of Italy and Render all the Collected Earthquakes	7
4	Earthquake Analytics Project Design	7
4.1	Technologies and Libraries	8
4.2	Earthquakes Retrieve and Data Modeling	9
4.3	Server Design	11
4.4	Client Visualization Techniques	11
5	Earthquake Analytics Functionalities and User Interface	14
6	Use Cases	16
7	Conclusions and possible future work	18

1 Introduction

Italy is a seismic active country which has used many resources to study earthquakes, producing a solid network of stations that collaborate to record all the territory earthquakes. Only a small fraction of that information is currently visible by the public, since all the earthquakes related web applications show the strongest or recent earthquakes. Although the past and small earthquakes are incredibly important because it has been recognized that in seismically active regions the determinations of the hypocenters of small earthquakes help delineate where the active faults lie. Earthquake Analytics fills this gap allowing a user to visualize past and small events too. We collected more than 250'000 earthquakes happened in Italy between 1985 and 2017. The information is retrieved from INGV (Italian National Institute of Geology and Vulcanology) services where, every day, up to hundreds of new events are published and then synchronize by Earthquake Analytics.

1.1 Motivation

The project focuses on earthquakes because they are unpredictable events in nature that cause extreme damages every year. Indeed earthquakes cause huge damages in seismic active countries as Italy, where earthquakes cost more than 3 billion euros every year and provoke, on average, casualties every four, though damage and casualties are too often caused by constructions too much vulnerable to ground shakes. Is indeed true that many of the annually earthquakes damages could be prevent by restructuring with anti-seismic technique private building (especially houses) near active faults.

Earthquake Analytics has been developed as a web application that provides an interactive visualization to reach more users as possible and inform them on the current and past seismic situation of Italy, to increase both knowledge and awareness about earthquakes.

Then the application also presents more scientific features for researchers to help understand the active seismic region, visualizing past events and allowing a quick retrieval of data for their tasks.

Visualize Minor earthquakes

Even if scientists know how earthquakes forms and (in many cases) where they can happen, it is a challenge to try to understand when a fault will release its energy, because we cannot directly measure all the aspects of an earthquake.

To understand better when, where, and how earthquakes will happen, scientists need to find new techniques to analyze the current set of data, including the huge set of small earthquakes (magnitude lower than 3). For example within years of data recording some correlation has been found:

Every major earthquake is followed by many *aftershocks* (earthquakes with a lower magnitude) near the same origins: these earthquakes are used to understand better the major event, even if it is not clear for how long aftershocks can happen after a big earthquake.

Less understood are the *foreshock*, small earthquakes that happen before a major one. Scientist have tried to predict upcoming earthquakes using foreshocks, achieving some isolated success, but this minor earthquakes seem to have different behavior with respect to the geographic zone, and more importantly most earthquakes lack obvious foreshock patterns.

Earthquake Analytics concentrates on any magnitudes because it wants to offer a visual tool to quickly understand the situation in a region, including small earthquakes that for scientist are as important as big earthquakes. We hope that it can be useful to find correlations between earthquakes in a fast and easy way.

Focus on One Country

Earthquake Analytics reports only earthquakes registered by the INGV and the events are restricted to a region around Italy. We decided to focus on one country for 3 reasons:

- Earthquakes measures change with respect to the geographical area, for example the intensity scale is adapted to the local infrastructure and terrain, being the result of years of data analysis.
- A seismic active country can count thousands of earthquakes every year, while approximately one million of earthquakes happen in the same period in the world. Not only it would be a challenge to visualize them all (even a year of events) but it would be technically challenging to keep all the related data.

- Furthermore, international data services keep only major earthquakes, while small and medium events must be retrieve from national services. Unfortunately, there is not a standard between those services and worst, near state borders, two national services could disagree on the origin or magnitude of the same earthquake because the information is not shared.

Earthquake Analytics has been thought to be able to easily introduce new data services and visualize any earthquake from any country. But, for the reason above, including a new data service require specific knowledge and time.

Focus on Italy

Italy has been chosen for many reasons, the most important are:

First Italy is an earthquake active country: Every year thousands of earthquakes are registered, a hundred of which have a magnitude of five or above.

INGV offers public publishing of information about earthquakes happen since 1985, with a lot of data for events happen after then 2007 and a total of more than 250 thousands earthquakes available. Moreover INGV station network is fast to register earthquakes, which are always manually revisioned in a few hours, and they are active in sharing and standardizing information between geophysical national institutes.

Finally, there are few online services that offer visualization of Italian earthquakes, each one showing a limited number of recent or strongest earthquakes, with the biggest international services some times ignoring European countries.

1.2 Physics Of Earthquakes

An earthquake is a shaking movement of the ground provoked by the slippage or rupture of a fault inside earth crust or upper mantle (up to 800 km deep in the ground).

Along a fault two plates push in different directions. The result is a slow grown of potential energy in the rocks nearby, that eventually will be released in a few seconds as heat and earthquake waves in a point called hypocenter. It is estimated that at most ten percent of an earthquake total energy is transform in earthquake waves. This percentage is more than enough to provoke massive damage nearby the biggest earthquakes epicenter (the point on the surface directly above the hypocenter).

2 State of the Art

There many examples of web-site of national institutions where recent and local earthquakes are shown on an interactive map. For example, **INGV web page**¹ has a 2D map that shows all the recent earthquakes happened in Italy where magnitude and depth are visually representing the earthquake with different shapes and colors.

This visualization (that represent a standard in the sector) is limited to the recent earthquakes with magnitude above 2. It is not possible to change the date interval or the informations visually represented (for instance it is not possible to color the point according to other measures). Moreover, the effects of an earthquake (i.e, its intensity area) is visually represented as a stand-alone static picture.

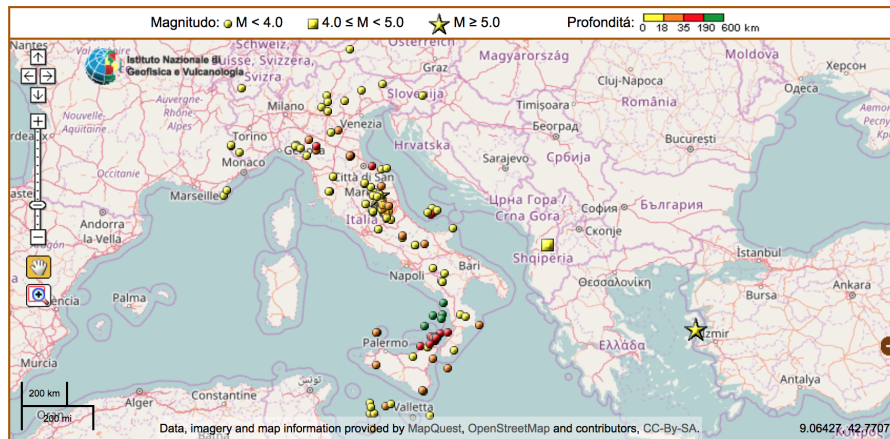


Figure 1. INGV earthquake map

The INGV map quickly shows the main information related to each earthquake, but it is limited because it does not allow to change the represented data. For example, the color of the point can only underline the depth of the earthquake.

To find different earthquake visualizations we looked for private projects, and we found a Japanese project called **Japan Quake Map**². This project, as the INGV website, visualizes earthquakes on an interactive map using colored points, but it also represents the geographical area they impacted using a translucent circle around the point. It also possible to select a date and sequentially play all the earthquakes until now changing the speed of the player.

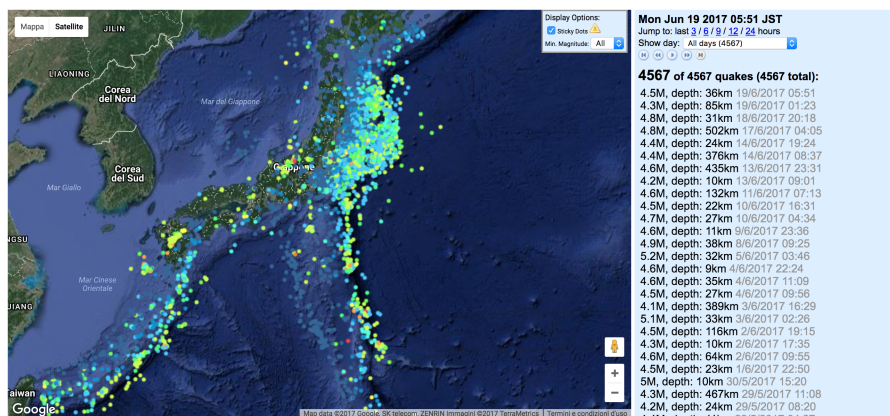


Figure 2. Japan Quake Map web application

Japan Quake Map communicates more informations than usual and it has nice functionalities, but it has two problems: the application does not allow to modify the represented information and, more importantly, the user interface is quite old and difficult to use such that the experience is not good as the quality of the data reported.

Searching for a better design we find a beautiful application called **Ninepointfive**³. This web application show earthquakes as elevated point (according to the depth) around the globe allowing to select the desire magnitude and time

¹INGV web page: <http://terremoti.ingv.it/>

²Japan Quake Map: <http://www.japanquakemap.com/>

³Ninepointfive web application: <http://www.ninepointfive.org/>

interval. It also possible to visualize the impact of the earthquake (with nice animations) and read historical information on the earthquake selected.



Figure 3. ninepointfive web application

The application gives a nice and global view on the major earthquakes, even if it lacks local earthquakes and the search is limited by a fixed time window. It remains an example of design and effective visualization of earthquake. We got inspired from the listed applications to produce an application simple to read as INGV, full of details as Quake Map, and beautiful to navigate as Ninepointfive.

3 Project requirements and analysis

There are three high level requirements for this project, they are: Information Collection, Earthquake Data Visualization and Show an Interactive Map of Italy and Render all the Collected Earthquakes.

3.1 Information Collection

The first requirement is to collect information about Italian earthquakes and keep the data updated. It was not difficult to find all the basic information since INGV publishes information for each new earthquake and it keeps an historical database of all recorded earthquakes since 2005, plus they republished all the medium and high events (i.e, with magnitude greater than 3) after 1985. The INGV earthquake model is a complex file with redundant and too much specific information, at least if read by a person not in earthquake science. It took some research before it was possible to figure out which data was relevant to be visualized (more details see section 4.2).

3.2 Earthquake Data Visualization

The second requirement is to represent the collected earthquake data visually. An earthquake can be visualized looking to its event or its effects:

An **earthquake event** is defined by the magnitude value at its hypocenter, physically representing the energy released at the hypocenter during the seconds or minutes of ground shaking. This physical attributes could be easily represented as a point in space and time: An earthquake event needs five variables to be defined, the position or origin (hypocenter express as 3D coordinate), the size (magnitude or energy realized), and the time of the event. Thus we can define an earthquake event as a tuple:

$$earthquake_event = (3D_coordinate, magnitude, time)$$

The main challenge is to visually represent all the five variables in a two-dimensional space, because in such environment we can define 4 variables for a point: a 2-d coordinate to position the earthquake representation on a map, a radius value that define the size of the point and a color to differentiate it.

Our solution is to split the hypocenter position coordinate in a 2d point, as longitude and latitude expressed in degrees, and a distance point as depth expressed in kilometres. Then we choose the magnitude value to define the size of the point to offer consistency in the visualization, since both time and depth are not bounded measures which would create unreadable visualizations. Finally, we need to represent both time and depth but we can use only the color of the point. We choose to let the user what to visualize as the color of a point, i.e, either by depth, time or magnitude. The final formula is:

$$earthquake_event_representation = (2D_coordinate, point_size, color)$$

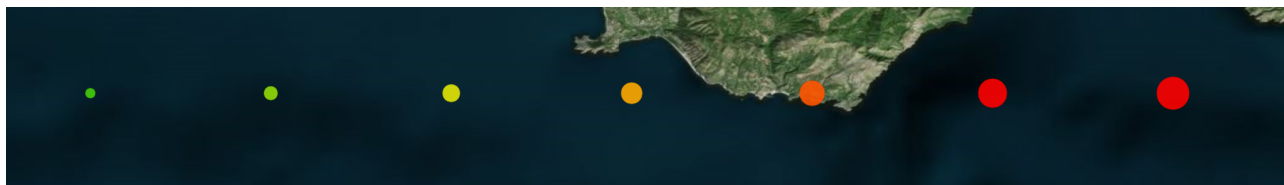


Figure 4. Scale of earthquakes point from magnitude 1 to magnitude 7 and colored by depth

An **earthquake effect** is defined as all the consequences that an earthquake event determines. This includes not only the collapse of structures, casualties, and ground lifting but also many secondary damages. To describe all those effects Geologists use the Modified Mercalli Scale, but this measure is a generic estimation of the maximal damages produced by an earthquake.

Earthquake effects are caused mainly by superficial seismic waves which provoke a destructive, vertical, and horizontal ground motion. It is extremely difficult to predict how superficial seismic waves will propagate because they depend more from local ground rock conformations than from the magnitude and depth of an earthquake event. Furthermore, superficial waves are usually not registered by seismographs because they travel short distances and, even if recorded, it is not possible to extract the wave route and its previous destruction intensity.

To collect precise intensity information, geologists installed many accelerometers on the Italian territory. An accelerometer registers the maximum ground acceleration (or PGA) that occurred during earthquake shaking at a location. This value can be mapped directly to a specific intensity value. Once all the data are collected, geologists use specific information of the territory (as rocks map) in order to interpolate those points and create a dense grid. Therefore, the effects of an earthquakes could be expressed as a matrix where each entry represent the intensity (PGA) registered or computed for a small area (usually one entry represents $1km^2$). This grid matrix can be easily represented in a 2D environment: each value of the matrix is used to color the correspondent area. The resulting colored area shows the earthquakes effect for each location close to the epicenter. Moreover, the collection of significant intensity entries (usually an area with register intensity over V) will show the geographical area that the earthquake effected.

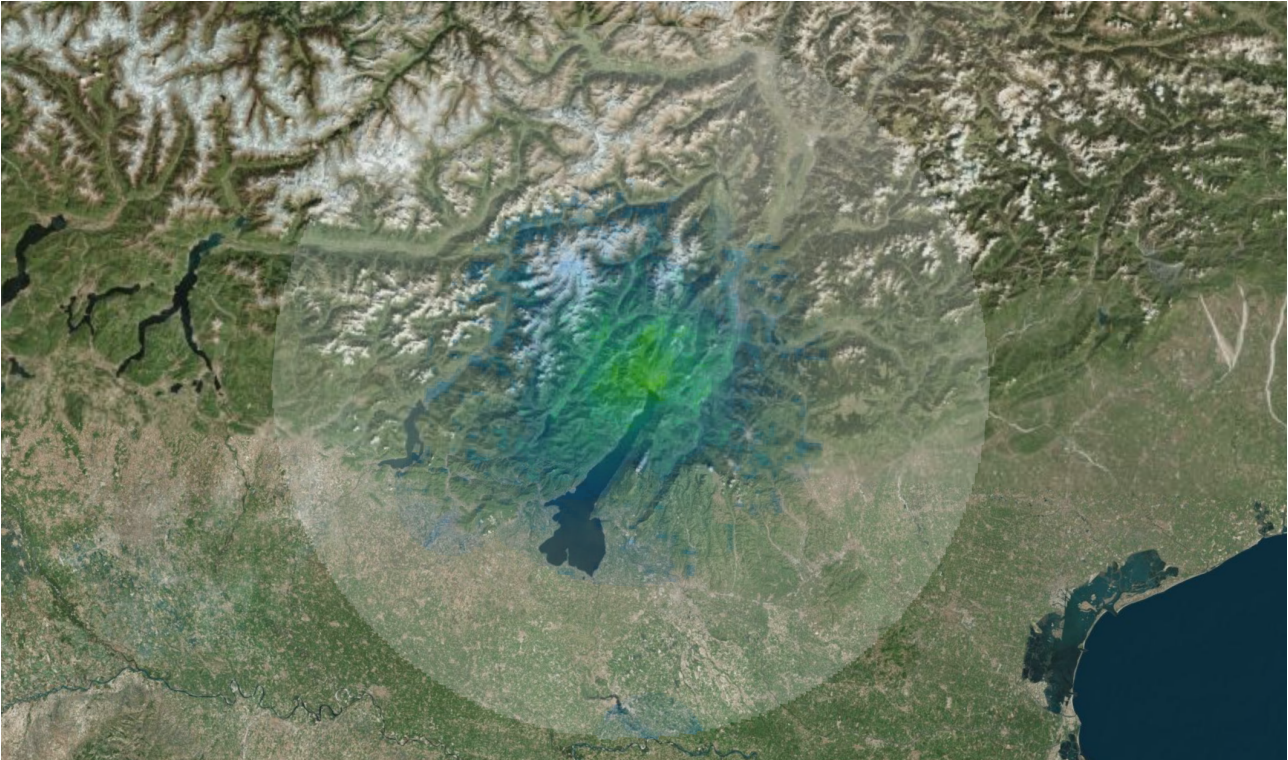


Figure 5. Earthquakes intensity visualization. A redder color means higher intensity.

3.3 Show an Interactive Map of Italy and Render all the Collected Earthquakes

This two requirements are explained together because we looked for a web solution able to display a map in the browser and handle the mapping between screen coordinate and map coordinate.

We tried to use classic solutions as Google Maps or Open Street Maps services to render a map of Italy, the problem with those solutions was the impossibility to render many objects: Google maps cannot handle too many points displayed and there are not advanced graphics settings.

It was clear we need more control on the map layer to be able to render more earthquakes. We tried to use json-maps files to represent Italy, but these files are rendered using the CPU and trying to display more than 2 thousand circles causes the browser to crash. We then thought to group close earthquakes in one point and show them only with an appropriate zoom, but the resulting visual effects was not a clear visualization because:

- Until points are grouped, single point information cannot be visible before zooming on a region.
- Visualize the single information require a lot of zooming, since many earthquakes happen near each other (foreshock and aftershock).
- Earthquakes happen along faults, therefore grouping earthquakes results in a unappealing and incorrect distribution of points.

A nicer solution would be visualize the most important earthquakes first and show the less significant events only when the user zooms on a region. In this representation there would be always the same number and quality of information on screen. We moved in this direction and found that only a solution that uses web-GL to render the map and the points could handle the desire effect especially with a huge number of objects in the screen.

We finally found the perfect library in Cesium.js, a JavaScript based library that offers a globe visualization and many built-in graphics shapes, all rendered using webGL (more details in section 4.1).

4 Earthquake Analytics Project Design

We will first discuss the languages and technologies used to create the server and client of Earthquake Analytics. Then we will concentrate separately on the server services (API services, controller, etc...), the database modeling, and finally the client and its design.

4.1 Technologies and Libraries

To build Earthquake Analytics we decide to use JavaScript for the client-side because many useful or essential libraries (as CesiumJS) are javascript compatible and, for similar reason, we use Java in the server side since we decide to use Spring framework and Spring-Boot library as base for our server.

Earthquake Analytics is a web application that uses many technologies and libraries, the below picture shows the high level architecture of the application including all the essential external resources.

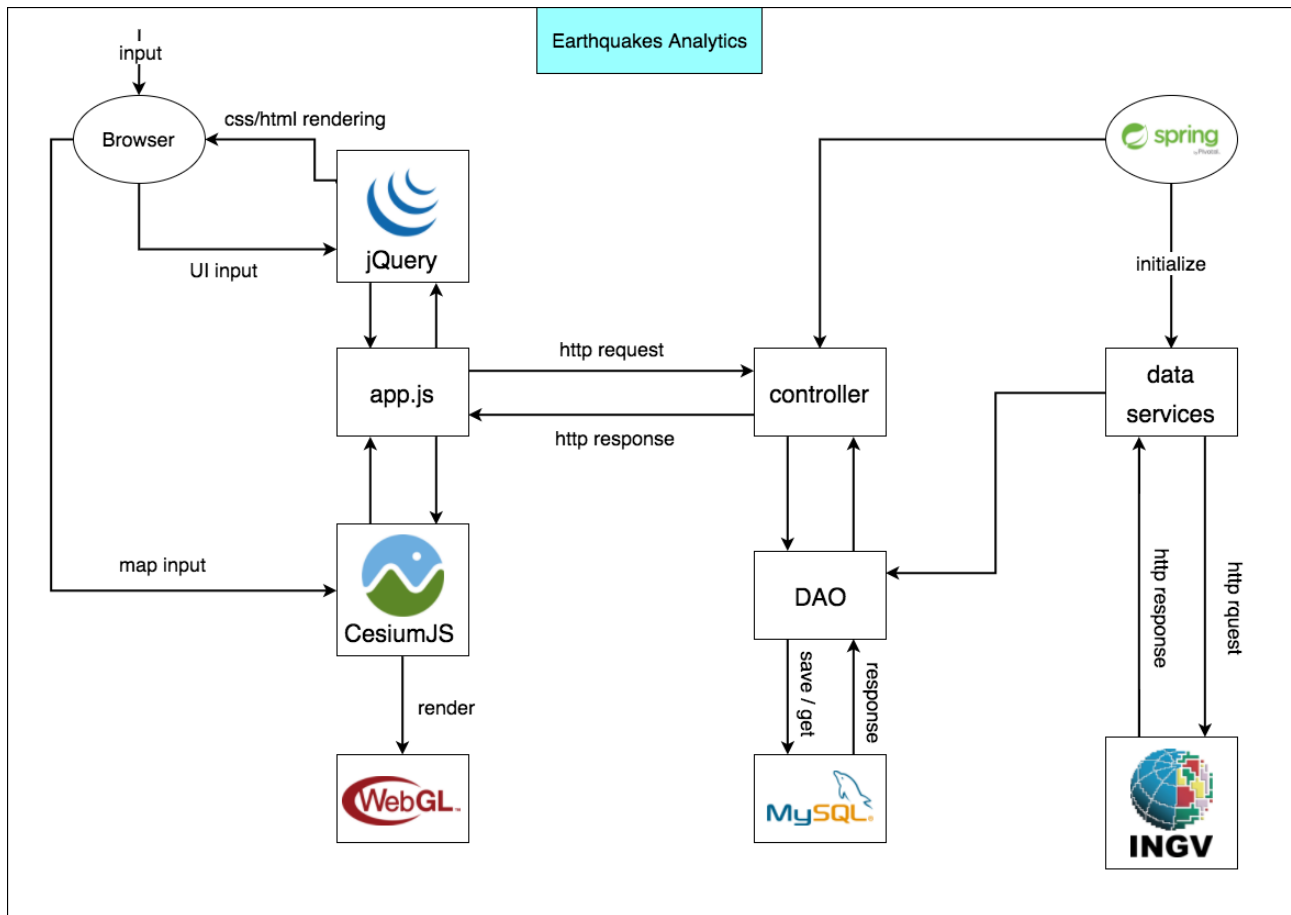


Figure 6. High level design of Earthquake Analytics with the most important Technologies and external libraries (INGV logo stands for the external API calls to INGV services)

Spring and Spring-Boot

Spring is an open-source application framework and inversion of control container for the Java platform. It provides infrastructure support for developing Java applications.

We choose Spring Framework because it provides great support to develop an application server: Spring can eliminate the creation of the singleton and factory classes supporting bean creation. A useful feature that manages all the application utility classes as service, DAO, or controller classes. Moreover, Spring has a well-designed web MVC framework.

On the other hand Spring framework can be complex to use because it is difficult to configure (especially if compared with other solutions as Node.js) since it has been designed for extreme flexibility and big java application. We decide to use Spring-boot instead, that is a Spring's convention-over-configuration solution:

- Spring-Boot auto configure many external libraries.
- It is easy and quick to configure a Maven based project with Spring-Boot.
- It automatically configures Spring whenever possible without losing flexibility.
- As opposed to Spring, it is possible to create and handle beans using annotations (without defining them in XML files), it makes the code easier to write and clearer to read.

MySql

MySql is a relational database management system. It is based on the Structure Query Language (SQL), which is used for adding, removing, and modifying information in the database. It is not the newest solution or the more versatile, but we choose MySql for its robustness and high performance because Earthquake Analytics need to quickly search and retrieve information from big unbounded tables which grow every day.

We also use JDBC, that is a standard for Java Database Access, to communicate between the database and the application server. We also use Hibernate, a JPA (Object Relational Mapping) solution, to define the database structure directly from the Java models and for simple queries. Even if Hibernate use JDBC to access the database, we use distinctly both the solution because JDBC can be a much faster solution when the application need to retrieve large and personalized set of data.

CesiumJs

Cesium is an open-source JavaScript library for world-class 3D globes and maps. Given a map provider it automatically handles the globe representation of Earth and the Non-Euclidean geometry calculations required to place correctly the map layer over the globe. Cesium is mostly used to create scientific visualizations and it provides many built in graphics shapes, graphics optimizations and utility functions such that is possible to let Cesium handle all the WebGL rendering and still obtaining a good level of personalization and control over the scene.

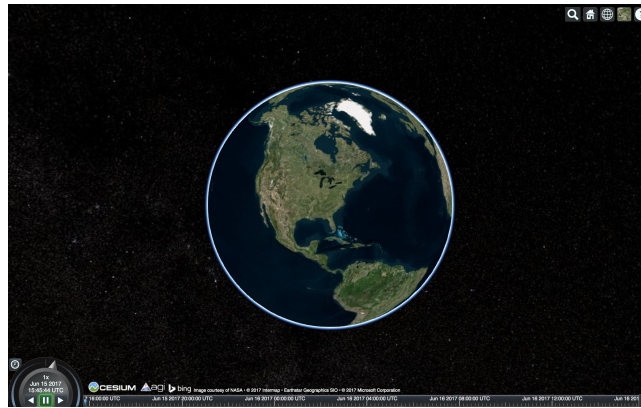


Figure 7. Cesium default view and interface

The problems with cesiumJs is that is more a project than a product: Cesium is not bug free, and some operations are not optimized. We use CesiumJs anyway because: it is a promising product with an active community, the advantages are much more than the problems and it remains the best web solution to build geographic scientific visualization.

jQuery

jQuery is a JavaScript library design to simplify the client html scripting. We choose jQuery for its productive syntax and useful built in functions. jQuery is used to build Earthquake Analytics UI together with the HTML CSS web framework Bootstrap.

4.2 Earthquakes Retrieve and Data Modeling

Earthquake Analytics takes data about earthquakes asking to INGV a list of earthquakes that match a fixed time interval. Unfortunately INGV publishes only relevant earthquakes on Twitter, not providing any internal subscription service for applications. Also the information about an earthquake origin and magnitude changes over time because INGV publishes the data as soon as available, but when new stations send their information about an event to INGV all the measures of the earthquake are recomputed and updated. Therefore Earthquake Analytics asks the latest earthquakes every fixed amount of time, usually it requests the earthquakes of the latest 48 hours to ensure that every earthquake has the correct information.

Earthquake data changes in the first hour after the event because origin measures strongly depends on the number of seismic stations that record the event and share it. For example, the epicenter of an earthquake is computed using a triangulation technique.

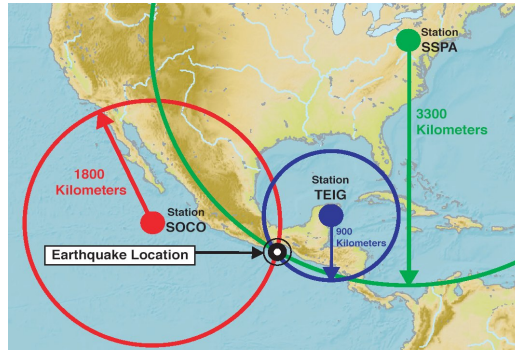


Figure 8. An example of collaboration between stations: each station knows the distance from the station to the earthquake, but the direction is computed by triangulating the collected distances.

Furthermore, the measurements can change because INGV manually verify every earthquake after a few hours. Those two particular details are shared across any information channel of earthquakes. Therefore, we model our database knowing that all the informations that describe the earthquake could change once saved but the event remains the same. So we need to record the identifier of the events given by the INGV network to be able to asks for the same earthquake later and we record the identifier for magnitude and origin to quickly check if an earthquake has been updated. The following UML digram show how the earthquakes are saved (this schema respect also the java model of an earthquake since the tables are defined using JPA).

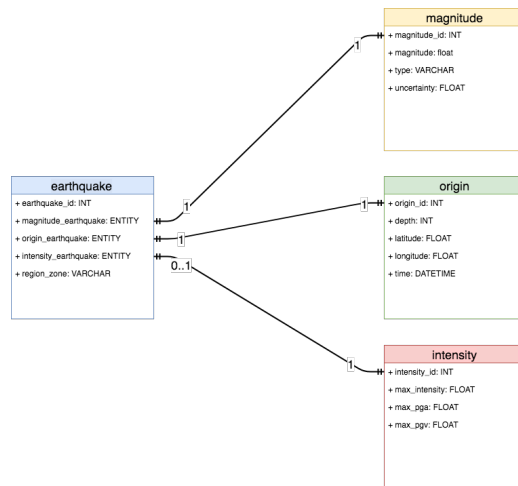


Figure 9. The ER digram of Earthquake Analytics database

Earthquake table contains the unchangeable informations about the earthquake and the references to its measures.

1. **earthquake_id** contains the identifier for an earthquake and it is generated by the INGV station network.
2. **magnitude_earthquake** contains the id of the magnitude entity of the earthquake; this field cannot be null.
3. **origin_earthquake** contains the id of the origin entry of the earthquake; this field cannot be null.
4. **intensity_earthquake** contains the id of the intensity entity of the earthquake, this field can be null because the information on the intensity of Italian earthquakes are available only for significant earthquakes happened after 2008.

Magnitude table contains the data about the magnitude of the event. This data is computed by INGV using the Origin of an earthquake, but the magnitude can also change according to in place measurements near the epicenter.

1. **magnitude_id** contains the identifier of the magnitude generated by INGV.
2. **magnitude** is the float value of the magnitude.
3. **type** is a string and it is the scale used to compute the magnitude (or letter 'M' if the scale is not provided).

Origin table contains data about the time and place of an event.

1. **origin_id** is the identifier of the origin generated by INGV.
2. **depth** is expressed in meters and it is used to define the hypocenter of the earthquake.
3. **time** contains the date and time of the earthquake.

Intensity table contains information about the maximum instrumental Intensity computed for the earthquake. The maximum value for the intensity is taken from a dense grid of values where each entry represent the intensity computed by INGV for a square kilometer.

1. **intensity_id** is, as before, the identifier for the intensity generated by INGV.
2. **max_intensity** contains the float number of the maximum registered intensity.
3. **max_pga** is the maximum peak ground acceleration recorded during an earthquake. In general PGA is equal to the amplitude of the largest absolute acceleration recorded on an accelerogram at a site during a particular earthquake.

We also use the file system to **store the matrix grid** that define the intensity values around an epicenter. Each entry of the grid is an array composed by PGA, latitude, and longitude values.

We try to use the database to store the grid, but each grid has 60 thousand entry, for a total of more than 120 millions entry for the current earthquakes. Using the file system we can create a file per Intensity and link the file naming it with the unique id of the earthquake. This technique allows faster access to the grid with respect to creating a one to many relation using the database.

4.3 Server Design

Earthquake Analytics server provides three main services: the controllers, the earthquake services and the DAO. After a client query request, **the controller** uses the DAO layer to retrieve the desired earthquake information and (in case of a large request) it compresses the response Json data. The controller service is simple but independent because we use Spring to initialize and handle the Controllers.

The second main component is the **earthquake services**. It provides the interface to communicate with any earthquake API service and parse the informations. Any method of the earthquake services requires two objects as parameter: a object that extend Parser and a object that extend Query (both are interfaces defined inside Earthquake Analytics). Then earthquakes services use the Query object to construct the link query for the desire service and when the response arrive it parses the data using the provided parser. To add a new earthquake provider, we just need to define a new Query and Parser for that service. For example, adding the SED (Swiss Seismic Service) to Earthquake Analytics requires to write only two objects: a SEDQuery class which generate the correct link for the API of the target service and a SEDParser class which translate the SED service response to Earthquake Analytics objects.

The last main component is the **DAO** where the method to get, save and update the earthquakes are implemented. We mainly use Hibernate to handle this part, but for large requests (for example more than 50 thousand earthquakes) we use JDBC to parse the data faster.

4.4 Client Visualization Techniques

The client is divided between the operations related to the earthquake visualization on map and the UI design. The main client file is app.js where first the Cesium library is initialized, then, when the document is loaded, app.js call many services as the controller (as the draw services and the UI services). The section below will discuss the two man visualizations, point and area visualization, while the User Interface will be discussed in the next chapter.

Fixed Size Point Visualization

We need to render a lot of fixed size circles in order to show the position and the magnitude of the collected earthquakes. We first tried to use a ellipse geometry, a 2D entity of Cesium that supports many features, but using only the

earthquakes happened in 2016 (34'000 points) the browser crashed. After many optimizations and failures it was clear that we need to use simpler objects, therefore we decided to represent the earthquakes using **point primitive**, a light cesium object with incredible performance (and a lot of limitations).



Figure 10. On the left an Ellipse Geometry, on the right a Point Primitive.

Figure 10 shows how primitive point and ellipse geometry are visually equivalent when used to represent fixed point, but the benefit of using point primitive is that its possible to render even more than one million points using a low level GPU, but there are many drawbacks. The main problems of using the point primitive, and how we manage to solve them, are in the following:

- A Point Primitive is a graphical object positioned in a 3D scene, meaning that, **a point primitive does not really have a z value**. It is rendered on top of every other geometry in the scene and the interaction with other objects is not trivial.

Earthquake Analytics needs to select a point when clicked, but, using Cesium labels, the result is an unappealing effects (z-index flickering). We tried to use HTML-CSS to show a pointer on top of the scene, but moving and resizing the pointer for any user interaction with the map was too expensive and complex. Finally, we found a nicer solution: when the user clicks on a point, the point border is increased (according to the point size) and colored with the complement color of the inside of the point. The resulting effects emphasize the point without changing its color which represents a measure (magnitude, depth or date of the earthquake).

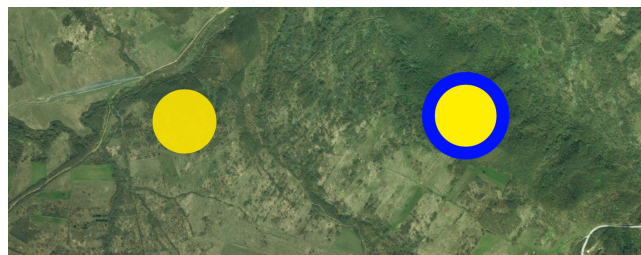


Figure 11. Difference between an unselected point (left) and a selected point.

- **Increasing the pixel size (diameter) of the point reduces the performance.** Consider that the GPU needs to color 78 pixels in each frame if a point has a diameter of 10 pixels, while a point with diameter of 30 pixels requires to color 706 pixels in each frame.

Earthquake analytics already uses a small pixel size to represent the earthquakes: A point can be between 5px (magnitude 0) and 35px (magnitude above 8). We need a small pixel size because many regions of Italy are dense of points (see Figure 13), but it can be difficult to select the smaller earthquakes because points have fixed size that does not change zooming. We change it resizing all points that are near the camera, such that a high level of zoom results in increasing the size of all the points on screen (see Figure 11). The performance is not affected, because a this level of zoom only a fraction of all the earthquakes is rendered with a big pixel size, and the proportion between circles is not changed since all the visible points are changed together.

- **A point primitive is rendered above another point according to its position inside the collection of points.** This means that the geographical position and the height of the point are not taken in account. Adding the earthquakes sorted by magnitude to the Cesium collection avoids that many small earthquakes

cover the most important ones, but when the user selects a point it could be covered by many others, making useless the selection.

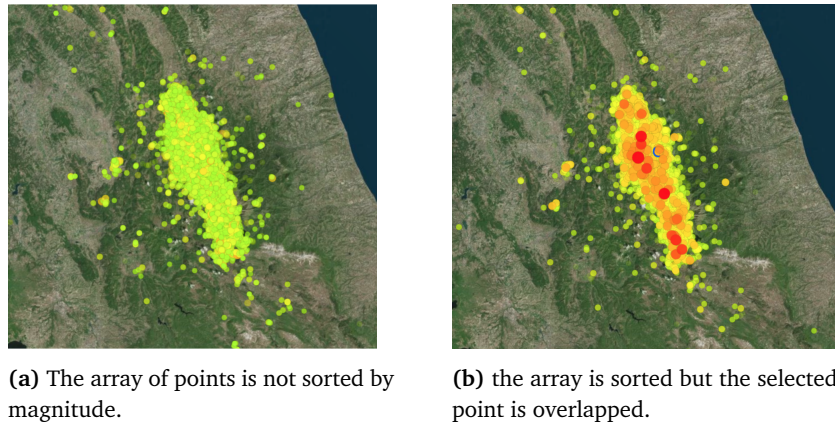


Figure 12. A visualization of a seismic active zone of Italy.

We avoid this situation hiding the selected point and copy all its properties on a default point, which is the last element of the collection of point primitives, therefore it is drawn over all the others circles. Now, when the user clicks on an earthquake, the point is hidden and visually replaced by its copy that has the bigger index in the collection.

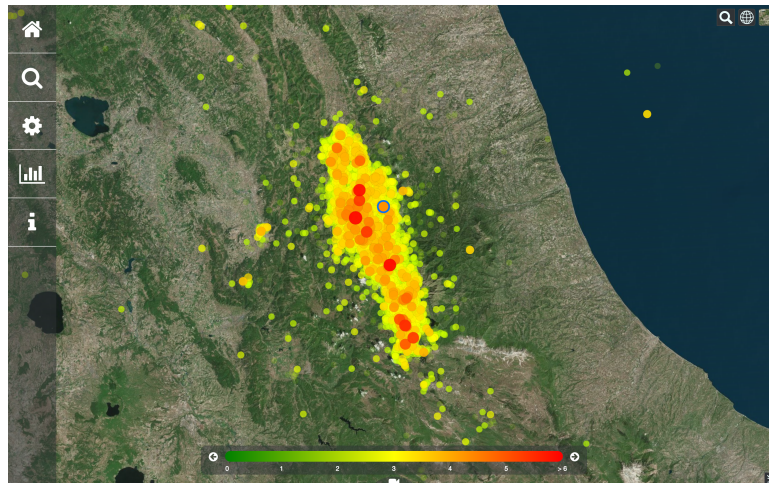


Figure 13. Final version of Earthquake Analytics: the selected point (blue border) is render over all the others points.

- **Primitive Points perform better if they are distributed geographically**, but the majority of earthquakes happen in a few places. We use many optimizations to reduce this problem.

The simplest ones were to change the OpenGL rendering settings to boost performance, for example we get double frame per second allowing only translucent objects to be rendered.

The major optimization is to show an earthquake according to its magnitude and the scene level of zoom. It works by creating different levels of buckets, each one represent a different show/hide condition, and assign each earthquake to a bucket according to its position inside the sorted array (sorted by magnitude). The earthquakes with a higher magnitude are always shown because their show condition is bigger than the smaller earthquake, which are shown only if the user zooms enough. This technique boosts performance and creates a less chaotic effect when the user wants to see many years of earthquakes events.

The final result allows Earthquake Analytics to render thousands of circles and run around 60 frames per second on a laptop with integrated graphics card (tested on Chrome web browser), or above 30 frames per second on a recent smartphone (after 2015).

Area Visualization

In section 4.2 we introduced the intensity grid map, a matrix that contains the intensity values of 60 squared kilometers (each entry map a 1km square) around the epicenter of an intense earthquake (magnitude above 3).

We start using Cesium ellipse geometry to render many colored circles around an epicenter to represent different intensity areas, but the intensity of an earthquake spreads according to the consistency of underground rocks. It is true that near the epicenter of an earthquake the shock is stronger, but nearby places can have different intensity values according to many conditions, such that using many concentric circles around the epicenter to represent different intensity zones result in a simplistic and (in many cases) wrong visualization.

We decide to use a more expensive, but visually correct, approach: When the user requests to see intensity area of an earthquake, Earthquake Analytics draws many squares of 1 squared kilometer around the epicenter coloring each square according to the Intensity value of the correspondence entry inside the grid map. Thus, we interpolate the color of near squares, so that the visualization shows precisely how superficial seismic waves spread around an epicenter and which intensity a places nearby had suffered. The resulting area intensity has a resolution of 1 square kilometer, that is enough to understand the geographical impact of an earthquake.

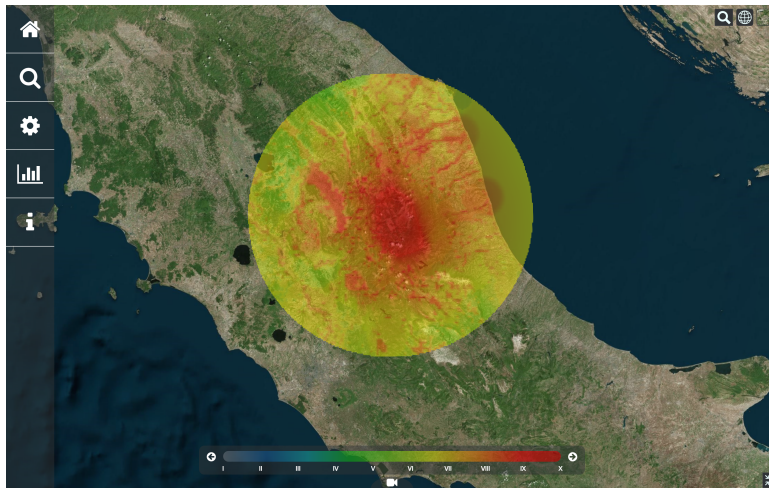


Figure 14. Intensity area of one of the recent strongest earthquakes (Norcia 30-10-2016)

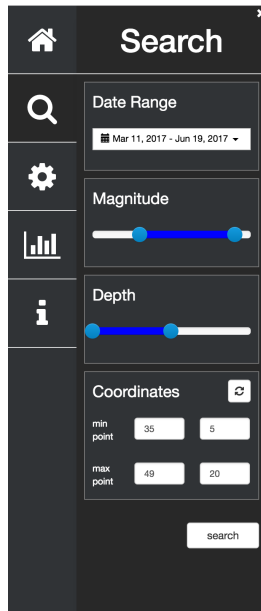
5 Earthquake Analytics Functionalities and User Interface

There are many optional functionalities in Earthquake Analytics and we are going to explain all of them grouping the following sections by the three main UI components of the application: bar menu, footer, and earthquake cards.

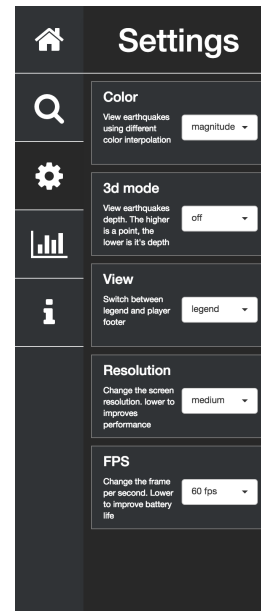
Bar Menu

The menu of Earthquake Analytics has 5 tabs:

1. The Home button resets the map camera to the initial view.
2. The Search button opens the search tab where is possible to search earthquakes selecting a range of time, magnitude, depth, and geographical coordinates.
3. The Settings button opens the settings tab, where many options are listed.
4. The Charts button opens the charts tab where information about the visualized earthquakes are listed.
5. The Info button opens the info tab which contains the credits.



(a) The search tab.



(b) The settings tab.

Figure 15. The pictures show two components of the menu of Earthquake Analytics.

Settings Tabs and 3D mode

The **settings tab** (figure 15b) implements many options. The first is the color mode, where a user can change the color of the earthquakes according to magnitude, depth, or date.

The 3D mode uses the depth of each earthquake hypocenter to elevate the points. Earthquakes with higher depth are shown above and the depth range dynamically change according to the maximum depth of the visualized earthquakes. This mode allows to quickly see how near earthquakes are related or how deep a fault is. The Resolution and Frame Per Second options allow to select different screen and render combinations choosing between a more detailed visualization or a better performance.

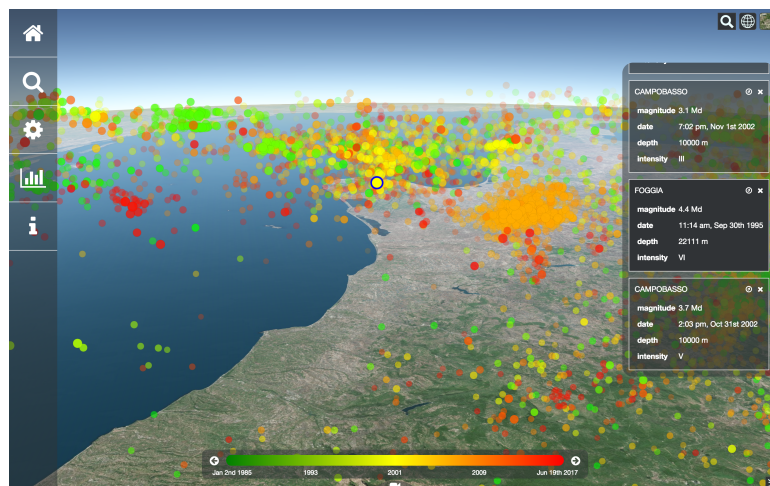


Figure 16. 3d mode active with earthquakes points colored by date

Footer Component and Player Mode

The footer component is a box above the page. The box can show two different informations: when the application starts the footer shows the magnitude color legend and it changes if the color mode is changed (see Figure 16). If the user clicks the camera button under the footer, the box switches with a player and the time line mode is activated.

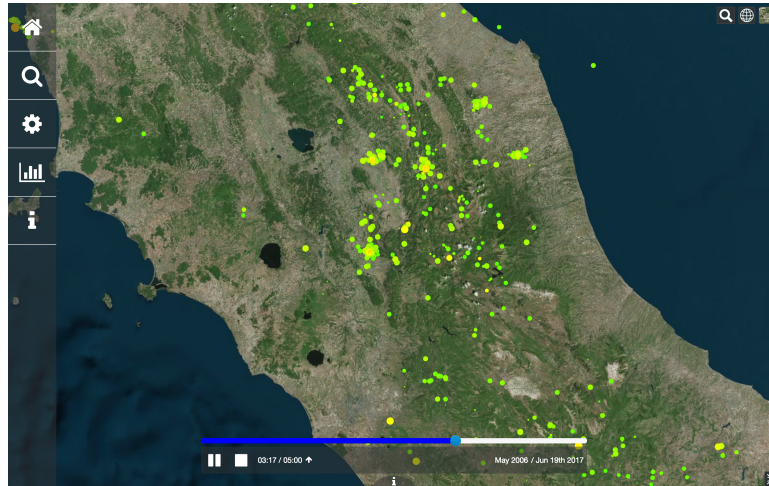


Figure 17. Time Line mode

The time line mode animates the selected earthquakes in chronological order. It is possible to dynamically change the total view time and move to a specific date using the player. The earthquake effects are computed in real time allowing the user to navigate the map or change the view settings (as color or 3D mode) while playing.

Earthquake Cards and Intensity Mode

Earthquake Cards is a UI component which appears on the right of the screen when an earthquake is selected. It is an infinite scrollable list where the information cards of previous selected earthquakes are stored. Each card shows information about the earthquake as magnitude, date, depth or Intensity.

Each card has a header with three buttons (see Figure 18), the “delete card” button, and “go to position” button are present on top of each card and, if clicked, they do what the name suggests. The intensity button is a circle present on top of a card only if the earthquake has a recorded intensity grid map. When clicked, the intensity mode is activated and the earthquake intensity area is rendered. Moreover the legend changes colors adapting to the new intensity view.

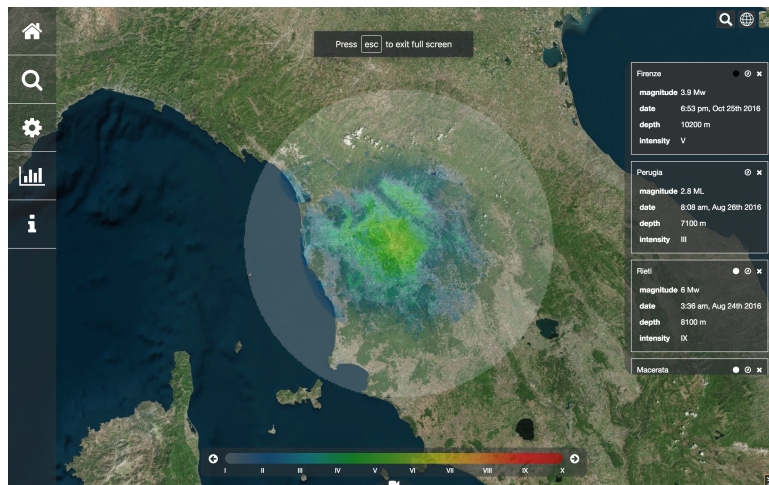


Figure 18. Intensity Mode

6 Use Cases

We now illustrate a possible use case with Earthquake Analytics. We start by looking to all the earthquakes happened between 1985 and 2017 focusing on the center of Italy, one of the most seismic regions of Italy.

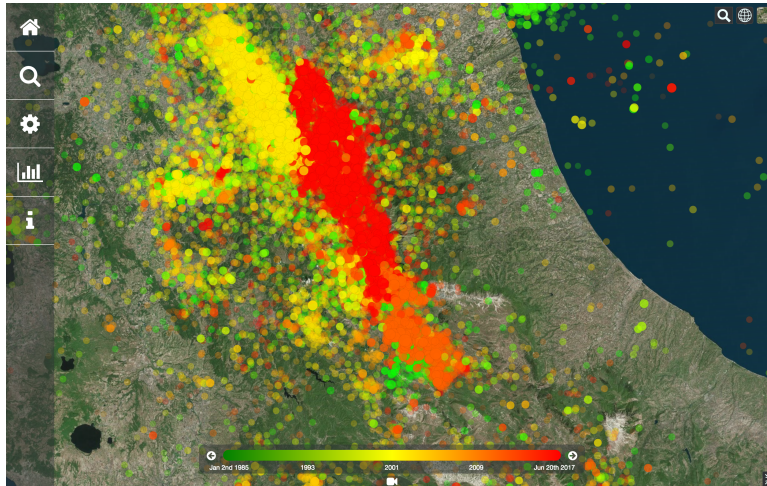


Figure 19. Earthquakes happen between 1985-2017 in the center of Italy, colored by date.

By coloring points by date, we can see that there are three major events: The 1997 earthquakes between Umbria and Marche (yellow dots), the 2009 earthquakes near L'Aquila (orange dots) and the 2016 earthquakes around Amatrice, Norcia and Visso (red dots). We can dynamically change the color of the points according to their magnitude and analyze better the situation underlining the major earthquakes:

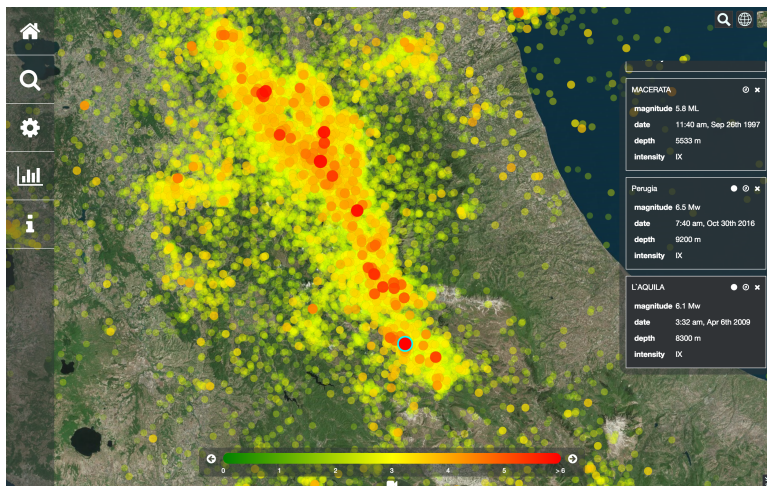


Figure 20. Earthquakes happen between 1985-2017 in the center of Italy, colored by magnitude.

The most important events are now visible and we can read information about each of them. We can also color each earthquake by depth and understand how deep the faults are:

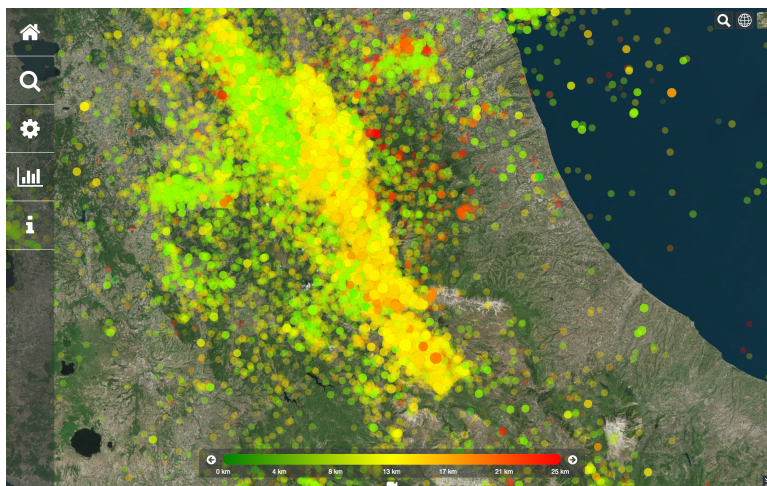


Figure 21. Earthquakes happen between 1985-2017 in the center of Italy, colored by depth.

The earthquakes of 1997 seem to happen near the surface, between 4 and 8 kilometers, while the earthquakes of 2009 and 2016 have roughly the same depths, i.e, between 8 and 13 kilometers. We decide to investigate better the 2009 and 2016 earthquakes by searching for a more precise date interval and using the time line mode.

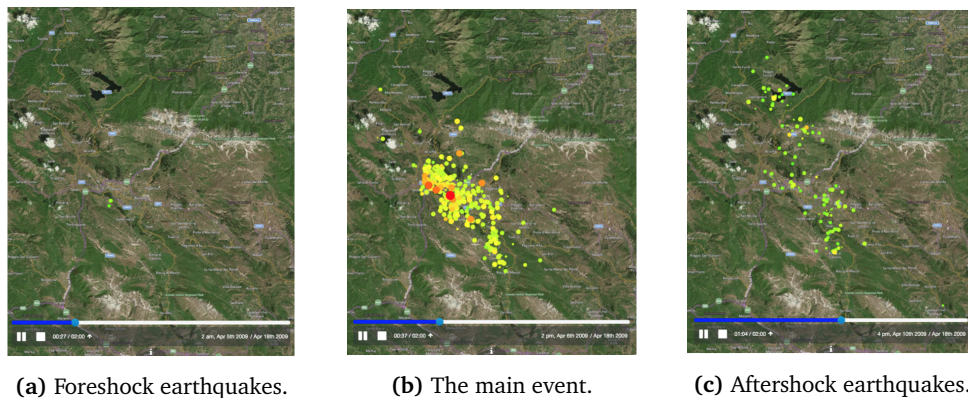


Figure 22. The moment before, during and after the Aquila earthquake of 6 April 2009. The pictures have been captured while the time line mode was playing

From Figure 22 we can see that before the 6th of April there are a few small foreshock earthquakes. After the main event the aftershock earthquakes keep happening for weeks, some of them with a magnitude above 4, but there is not any other major earthquake. We then focus on the interval between May 2016 and March 2017 and, using the data color mode, and we can see that the events of 2016 are formed by 3 major earthquakes spread between August 2016 and January 2017. By focus on a closer interval and by using the time line player, we can visualize the evolution of the three main earthquakes.

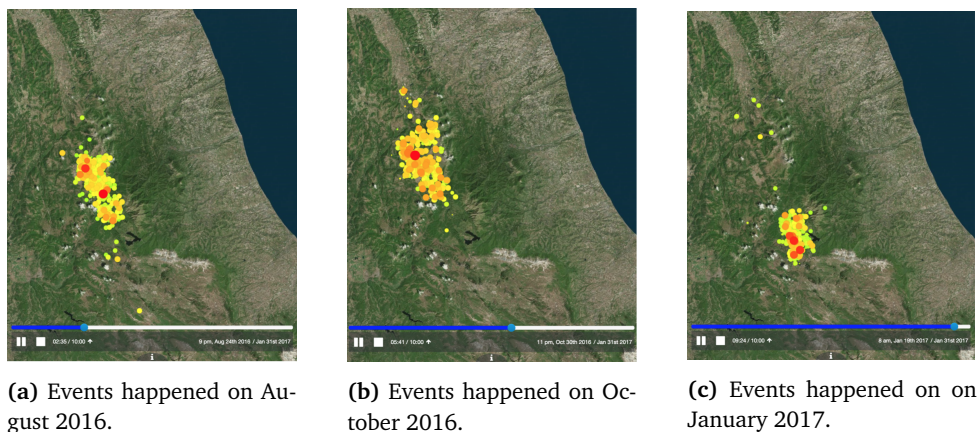


Figure 23. The three main events of the seismic activity stated on August 2016 between the country side of Rieti, Perugia and L'Aquila

7 Conclusions and possible future work

The project respected our initial requirements being able to collect, update and visualize all the Italian earthquakes published by INGV. We also extended our visualization by introducing a dedicated mode for each measure: time line mode to underline the date of the events, 3D mode to show the hypocenter of the earthquakes, and intensity mode to understand the geographical impact of an earthquake.

That sed, Earthquake Analytics is a project that can be improved in different directions:

- **Data Quantity:** Earthquake Analytics shows only Italian earthquakes, but only a few modifications are in principle required to show earthquakes from any desired country. The most expensive operations are to understand and parse a new earthquake service of information.
- **Data Quality:** there are many specific and useful earthquakes related data published. A nice improvement would be to study this specific data and try to visualize it in a interactive environment. One example could be a interactive visualization of the focal mechanism, which represents the direction of slip in an earthquake and the orientation of the fault on which it occurs.

- Earthquakes 3D-Visualization: The current 3D visualization show the depth and size of the faults, but, using this mode, it is difficult to distinguish the position of a single point. The 3D mode shows “clouds” of points because a point is render below or above another according to its position inside the array. We try to implement a ray-tracer to understand which element should be rendered over the others according to the camera view, but our attempt was too expensive to run with thousands of point on screen (especially in dense zones). We believe that solving this issue would bring new possibilities for future representations.