



QQuake, a QGIS Plugin for Loading Seismological Data From Web Services

Mario Locati^{1*†}, Roberto Vallone^{2†}, Matteo Ghetta^{3†} and Nyall Dawson^{4†}

¹Istituto Nazionale di Geofisica e Vulcanologia (INGV), Sezione di Milano, Milan, Italy, ²Istituto Nazionale di Geofisica e Vulcanologia (INGV), Sezione di Roma 1, Rome, Italy, ³Faunalia Srl, Pontedera, Italy, ⁴North Road Consulting, Sunshine Coast, QLD, Australia

OPEN ACCESS

Edited by:

Lei Wang,
Institute of Remote Sensing and Digital
Earth (CAS), China

Reviewed by:

Weiguo Han,
University Corporation for
Atmospheric Research (UCAR),
United States
Yuchu Qin,
Chinese Academy of Sciences, China

*Correspondence:

Mario Locati
mario.locati@ingv.it

†ORCID:

Mario Locati
orcid.org/0000-0003-2185-3267
Roberto Vallone
orcid.org/0000-0003-1208-9412
Matteo Ghetta
orcid.org/0000-0002-5321-4497
Nyall Dawson
orcid.org/0000-0001-9812-7584

Specialty section:

This article was submitted to
Environmental Informatics and Remote
Sensing,
a section of the journal
Frontiers in Earth Science

Received: 06 October 2020

Accepted: 07 January 2021

Published: 26 February 2021

Citation:

Locati M, Vallone R, Ghetta M and
Dawson N (2021) QQuake, a QGIS
Plugin for Loading Seismological Data
From Web Services.
Front. Earth Sci. 9:614663.
doi: 10.3389/feart.2021.614663

An increasing number of web services providing convenient access to seismological data have become available in recent years. A huge effort at multiple levels was required to achieve this goal and the seismological community was engaged in the standardization of both data formats and web services. Although access to seismological data is much easier than in the past, users encounter problems because of the large number of web services, and due to the complexity of the discipline-specific data encodings. In addition, instead of adopting cross-disciplinary standards such as those by the Open Geospatial Consortium (OGC), most seismological web services created their own standards, primarily those by the International Federation of Digital Seismograph Networks (FDSN). This article introduces “QQuake,” a plugin for QGIS—the Open Source Geographic Information System—that aims at making access to seismological data easier. The plugin is based on an Open Source code available on GitHub, and it is designed in a modular and customizable way, allowing users to easily include new web services.

Keywords: QGIS, FDSN, webservices, seismology, earthquakes, seismicity, OGC

INTRODUCTION

Since natural phenomena do not follow administrative boundaries, seismology has had a long-standing tradition of open data sharing. International cooperation among seismologists includes the International Association of Seismology and Physics of the Earth’s Interior (IASPEI) that was established in 1922 (Schweitzer and Lay 2019), and the ESC (European Seismological Commission) that was established in 1951. These associations promoted data encoding formats for sharing seismological data and knowledge among scientists that closely followed the evolution of communication technologies, especially after the advent of the Internet and the World Wide Web in the early 1990s.

Some of the most widely adopted standards (Dost et al., 2012) for sharing earthquake parameters were the IMS (International Monitoring System) published by IDC (International Data Center) in 1999, superseded by ISF (IASPEI Seismic bulletin Format) published in 2001 (see <http://www.isc.ac.uk/standards/>), and more recently superseded by QuakeML (Schorlemmer et al., 2004) which is XML-based. Another example of a seismological standard for sharing equally spaced time-series data is SEED (Standard for the Exchange of Earthquake Data), which was officially adopted by the Federation of Digital Seismograph Networks (FDSN; Suarez et al., 2008) in 1987, and since 1992 in a stripped-down version called miniSEED. In 2012 FDSN adopted StationXML as the encoding format for describing the equipment deployed at a site. In 2013 FDSN selected QuakeML and StationXML as

the data encoding formats respectively for web services dedicated to earthquake parameters (fdsnws-event), and for web services for seismic station information (fdsnws-station).

In recent years, there was a surge in the number of web services that lowered the complexity of accessing seismological data by using the above-mentioned formats. However, these formats create barriers preventing smooth and easy access to data, particularly for students or researchers that do not have programming skills or who do not come from a seismological background. In fact, field-specific data encodings or web services require time-consuming re-processing or adaptations before being able to load any data in general purpose software such as spreadsheets or GIS.

On the contrary, web services standards developed and promoted by the Open Geospatial Consortium (OGC; <https://www.ogc.org/>) are not tight to a specific field of Science and are meant to offer a solid interoperability framework for sharing geospatial data. OGC standards became increasingly important particularly for European Union Member States that are requested to follow the Directives of the Infrastructure for Spatial Information in the European Community (INSPIRE) published in 2007. INSPIRE set guidelines for a seamless sharing of geographical data in the EU, and recommended OGC services for implementing such guidelines. OGC services, and their related data encoding formats such as the Geography Markup Language (GML), are not widely adopted in the seismological community, thus preventing an easy integration of existing services in non-seismologically oriented software. Reasons for such a lack of support of OGC standards for seismological data may include: 1) the existence of specific seismological standards for circulating data, 2) a large part of seismological data (i.e., waveforms) are not geographical, and a 3) the lack of a strong interaction with other fields of Science that does not encourage the changes in the existing workflows.

The situation has been slightly changing in recent years with the progressive adoption of the Open Science paradigm (Nielsen, 2011) which is based on convenient and quick solutions for sharing data using modern communication technologies. Web services play a key role in this scenario, as they provide direct access to updated and well-formatted data that can be easily integrated in most research workflows.

In Europe, activities related to EPOS ERIC (<https://www.epos-eu.org/>), the European Research Infrastructure Consortium dealing with the broader discipline of solid Earth science, are very important for the not-so-open seismological community. Multiple scientific communities interact in this environment, all with the goal of integrating heterogeneous data in a common platform (Bailo et al., 2016; Bailo et al., 2018; Bailo et al., 2020). Harmonization of data encodings and web services is an essential activity, and every community involved is challenged to adhere to common guidelines.

An additional environment forcing the seismological community to heavily interact with other ones, are research projects focusing on multi-hazard analysis and services. An example is ARISTOTLE (All Risk Integrated System Toward the Holistic Early-Warning; <http://aristotle.ingv.it/>), that begun as a research project and developed into a formal and well-

structured scientific advice service provided to the ERCC (EC Emergency Response Coordination Service; <https://ercportal.jrc.ec.europa.eu/>). Again, in this framework the seismological community is forced to interact and adopt solutions interoperable with those developed in other communities, and to combine its results with those in the fields of severe weather, flooding, forest fires, tsunamis and volcanoes (Michelini et al., 2017; Michelini et al., 2018).

“QQuake,” presented in this article, is a technical solution aimed at lowering the barriers to accessing seismological data for anybody operating a GIS (Geographic information system). It is meant to fill the gap between existing seismological web services and seismologists or potential users from other disciplines that might be interested in integrating seismological data in their research. The idea is to deliver a ready-to-use and intuitive tool that could be easily extended, and with the long-term goal of expanding the user base of seismological data.

KEY FUNCTIONALITIES

We selected QGIS as the target application, because it has a large base of users, it is open source and well documented, and has a vibrant community of developers. In addition, QGIS adopts Python as its scripting language that is widely adopted in the seismological community and it allows developing extensions that well integrate with external tools. It is out of the scope of the present article to illustrate how to deal with seismological data using tools provided by QGIS and its huge variety of plugins.

The original idea while drafting the plugin was to develop a modular plugin able to interact with any service compliant to a standard, instead of developing a plugin tied to a particular web service. Along the way, we went a step further, and we opted for supporting multiple service standards. To make life easier for the user, we pre-configured the plugin with a series of web services so that users have immediate access to them.

In the first round of development, we had to focus on a limited number of web service standards, and we selected those that are supported by the seismological data providers involved in the EPOS ERIC, namely:

- fdsnws-event, a RESTful web service providing earthquake parameters (i.e., time of the event, location and magnitude). It may output a basic set of parameters encoded as plain text, or a complete set of parameters encoded in QuakeML v1.2 (Schorlemmer et al., 2004; Schorlemmer et al., 2011);
- The macroseismic intensity data web service developed by the European Archive of Historical Earthquake Data (AHEAD; Albini et al., 2013; Locati et al., 2014; Rovida and Locati 2015). This service is a slightly modified version of fdsnws-event, it use the same query parameters but it encodes the output using QuakeML 2.0 (Euchner and Kästli, 2014; Euchner et al., 2016; Kästli and Euchner 2018) and its macroseismic package (Locati, 2014);
- fdsnws-station, a RESTful service with metadata describing data collected by seismological instruments. The service

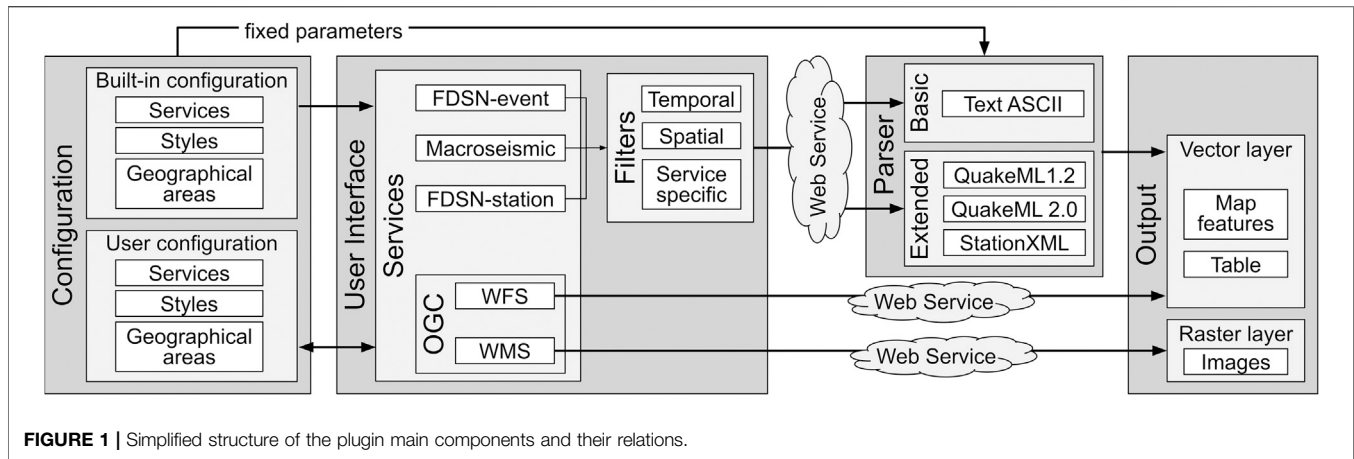


FIGURE 1 | Simplified structure of the plugin main components and their relations.

may output a simplified plain text or a more complex StationXML (<https://github.com/FDSN/StationXML>);

- Open Geospatial Consortium (OGC) Web Feature Service (WFS; <https://www.ogc.org/standards/wfs>), the non-disciplinary-specific standard serving geographical features encoded with the Geography Markup Language (GML; <https://www.ogc.org/standards/gml>). GML is an OGC standard itself, an XML for encoding various geometry types (point, lines, polygons) and additional information such as the coordinate reference system, dynamic features, and temporal objects;
- OGC Web Map Service (WMS; <https://www.ogc.org/standards/wms>), the non-disciplinary-specific standard providing access to georeferenced map images.

The plugin has a modular structure, its key components are (**Figure 1**):

- The configuration files which contain the list of services, the definition of their capabilities, and the default values. In addition, they contain the mapping of XML elements included in QuakeML and StationXML to their corresponding tabular data to be associated to the QGIS output layer.
- The user interface, that guides the user to the selection of the service, the filtering of the data, and the customization of the final output.
- The data parser, which compile the query call for the web service, send it, and retrieves the output, then apply a data conversion if required; in case an OGC web service is selected, the data parsing is skipped, as these web services are natively supported by QGIS.
- The generator of the output layer, a standard QGIS layer composed by a data table compiled taking into consideration both the user and the built-in settings, and by the geographical features that will be plotted on the map using the symbol set specified in the configuration file.

The overall design of the Graphical User Interface (GUI; **Figure 2**) of the plugin focuses on simplicity, as the main

target user is not an expert seismologist and does not necessarily know the various aspects of the different web services and their output data. In order to make the plugin useful to expert users too, we added advanced optional features, such as a more rich and complex output, or the possibility to search data using identifiers. Finally, we added the possibility to bypass the use of web services for allowing users to load QuakeML files directly, as they become increasingly available on the web (e.g., Italian Seismic Bulletin, <http://terremoti.ingv.it/en/bsi>).

STRUCTURE AND WORKFLOW

The QQuake plugin is written in Python, using the PyQGIS library for all spatial operations and the additional PyQt5 library for all the graphical user components.

The workflow can be divided into three areas: filling the GUI with the constraint parameters characterising each web service, sending a remote request containing these constraint parameters and finally parsing the returned resulting document and converting it to a standard QGIS map layer.

The configuration of each service is encapsulated in a comprehensive “json” file that is parsed dynamically by the plugin and used to fill the GUI with the options and filter parameters supported by the selected web service. The framework of QQuake has been designed for easy extensibility and user configurability: all parameters for querying a web service are editable, and a user may conveniently add and configure new web services using the graphical user interface.

All filtering parameters specified by the user are included in the equivalent web request. The core part of the plugin is a “data parser” that converts on the fly the output from the selected web service into a representation of the data suitable for display, analysis and storage in QGIS.

Finally, the plugin can load and apply specific styles for plotting symbols of the resultant map layers. This further simplifies the use of the plugin, as the generated maps are automatically preloaded with the appropriate thematic styling. These predefined symbol sets are applied in case the selected web

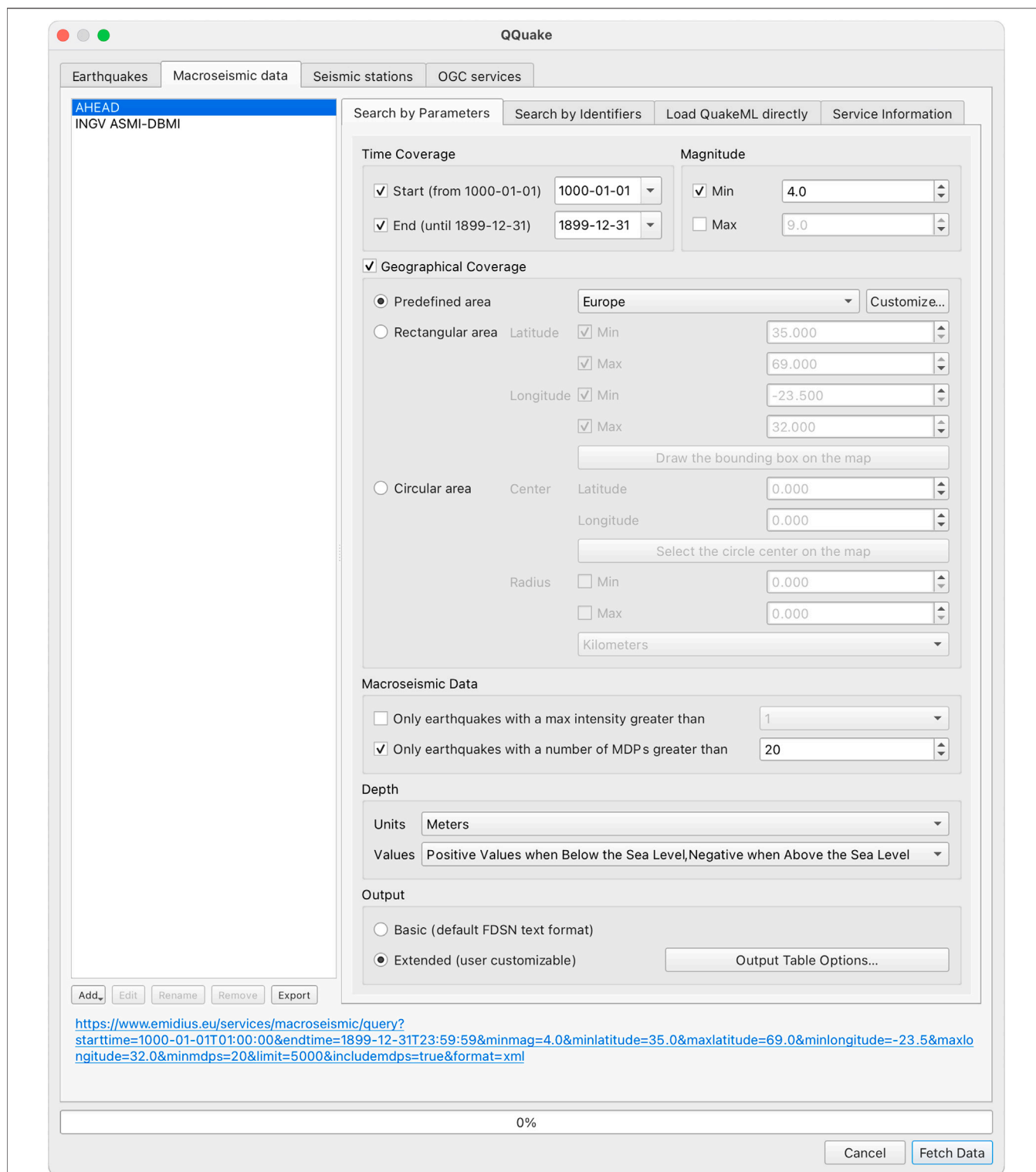


FIGURE 2 | Screenshot of the plugin GUI. Example section “Macroseismic data” with the data source “AHEAD” selected, and all query parameters that the user may apply.

service does not deliver any style information, such as `fdsnws-event`, `fdsnws-station`, `macroseismic` or `OGC WFS`. Built-in styles cannot satisfy all possible needs as different users create maps with different goals, showing multiple types of different data at the same time, using a different colour scheme, and may use different backgrounds, from single-colour polygons, to complex topographic maps or colourful satellite photos. In addition, seismology does not use a standard for mapping each type of data, neither for the shape of the symbols, nor for their colours. Earthquake epicentres for instance are usually represented with a variety of shapes, colour-coded and sizes, whereas seismic stations are commonly represented using triangles, colour-coded for identifying specific aspects such as the network they belong to. QQuake plot macroseismic intensity data with the symbol set created in the framework of the European Archive of Historical Earthquake Data (electronic supplement to Locati et al., 2014; Musson and Cecić, 2012).

Each type of web service required the development of a dedicated parsing module. For the `OGC WMS` and `WFS` web services, the development was trivial since they are natively supported by QGIS and the plugin only wraps existing functionalities to its GUI.

The support of FDSN web services was more challenging. These services are quite straightforward for what concerns the parameters that can be combined to compile the query string to be submitted to the service endpoint. Each type of FDSN service provides a specific set of query parameters depending on the data it provides. For example, a user may query an “`fdsnws-event`” service by providing a magnitude threshold, a type of magnitude and/or define a geographical area. An additional level of complexity is caused by the existence of two sets of parameters: a set of required parameters that must be supported by all FDSN services, and a set of optional parameters. The services maintainers are free to support or not each optional parameter. Therefore, each FDSN-compliant web service must provide the list of the optional supported parameters in a description file using the WADL (Web Application Description Language) standard. To make interaction with FDSN services even more complicated, each data provider may impose its own limit to the amount of data returned for any given request, causing error messages when queries generate too many results.

Parsing the output of an FDSN service is easy when the requested encoding format is a plain text, where each field is separated by a vertical bar character (“|”). On the contrary, getting data out of the XML output was more challenging because of the complexity and richness of the information it conveys. The plugin supports three XML based standards, `QuakeML 1.2`, `QuakeML 2.0`, and `StationXML`.

The parser for the XML-based `QuakeML 1.2` is used in conjunction with the `fdsnws-event` web services when users request the “Extended” output. By default, the output of the `fdsnws-event` web service provides the preferred parameters only via `QuakeML`, so that each earthquake is associated with one origin and one magnitude only. Compiling the resulting table row for the QGIS geographical layer with these parameters is straightforward, as one line corresponds to one earthquake only. Advanced users might want to retrieve all sets of parameters, and in this case the plugin outputs as many table

records for each earthquake as many origin/magnitudes pairs are in the `QuakeML` (Figure 3).

The parser dealing with web services for macroseismic intensity data uses most of the functionalities of the `fdsnws-event` parser. Additionally, this parser can manage the macroseismic section of `QuakeML` as described in the current revision of the XML schema (<https://quake.ethz.ch/quakeml/QuakeML2.0/Macroseismic>). `QuakeML v1.2` that is used for the `fdsnws-event` is already quite verbose, and `QuakeML 2.0` adds on top of that macroseismic information. Consequently, when dealing with macroseismic data, the plugin must retrieve and parse very long XML files causing a performance slowdown. Another difference between the macroseismic and the `fdsnws-event` parsers, is that the former outputs two distinct QGIS layers, one with the list of earthquake parameters and the associated geographical points representing the epicentres, and the other with the macroseismic data related to each earthquake.

The last parser deals with the `StationXML` standard provided by the `fdsnws-station` web service. At the present stage of development, the parser supports only a subset of the `StationXML` markup, specifically “network” and “station,” leaving out elements such as “channel” and “response”. The geographical symbols are plotted using the coordinates associated with each station.

As opposed to the text output provided by any of the FDSN services where the list of fields and related labels is fixed, when a web service output data using any of the supported XML-based data standard, the plugin allows the user to decide the fields to be included in the QGIS layer. In addition, the user can decide whether to export short field labels and stick to the 10 characters limit imposed by the widely used ESRI Shapefile format (<https://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>). If no compatibility with Shapefiles is required, the user may export long field labels that result in a more meaningful output. Other file formats supported by QGIS do not suffer from field length limitations (i.e., `GeoPackage`, `GeoJSON`, `GML`), therefore users may safely select long field labels. For an easier customization of the plugin, the mapping of XML tags into table fields and the list of corresponding short and long field labels are all stored in easy-to-read JSON files, one for each type of XML. Currently, there is a mapping file for `QuakeML v1.2`, one for `QuakeML v2.0` and its macroseismic package, and another one for `StationXML`.

BUILT-IN SERVICES

To offer a better usability experience and to allow non-expert seismologists to be productive right after the installation, the plugin comes with a series of pre-configured web services (Table 1). Everybody is free to suggest additional services compliant to one of the supported standards by submitting a request at <https://github.com/INGV/qquake>

Built-in services are stored in a configuration file called “`config.json`” with all settings and default values. The file is a simple and human readable JSON where web services are subdivided into sections, one for each service standard, and each service is described by a list of supported parameters. Metadata can be added too, for example a free-text title, the

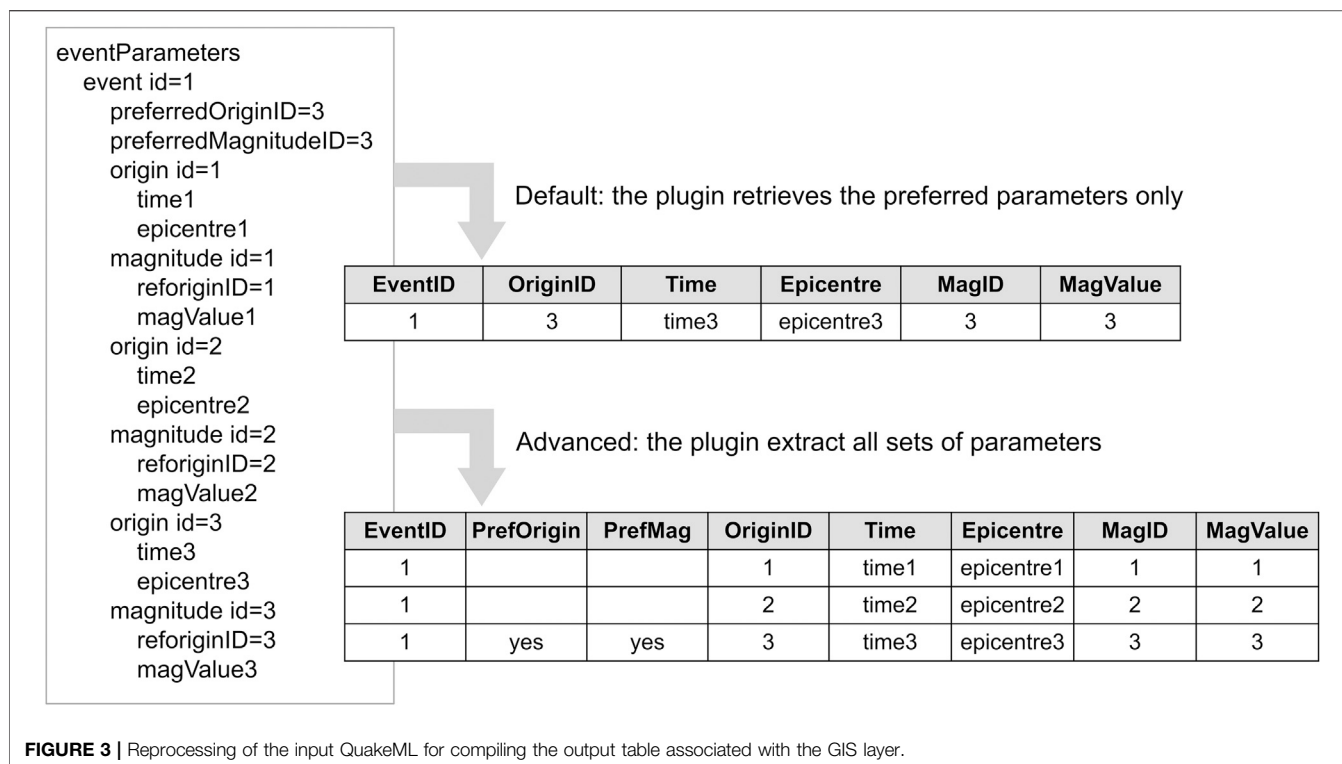


TABLE 1 | List of built-in services and the supported web service standards.

Title	Geog. Coverage	Temporal coverage	Web service standard
AHEAD, European Archive of Historical Earthquake Data, run by INGV (Albini et al. 2013; Locati et al. 2014; Rovida and Locati 2015)	Europe	1000–1899	Fdsnws-event, macroseismic, OGC WFS and WMS
ASMI, Italian Archive of Historical Earthquake Data, run by INGV (Rovida et al. 2017)	Italy	–461–2017	Fdsnws-event, macroseismic, OGC WFS and WMS
ISIDe, Italian Seismological Instrumental and Parametric Database, run by INGV (ISIDe Working Group 2007)	Italy	1985—today	Fdsnws-event
European-Mediterranean Seismological Centre, run by EMSC-CSEM	World	1998—today	Fdsnws-event
NEIC, National Earthquake Information Center, run by USGS	World	1600—today	Fdsnws-event
EIDA, European nodes of European Integrated Data Archive, run by Orfeus	World		Fdsnws-station
EDSF, European Database of Seismogenic Faults, run by INGV (Basili et al. 2013)	Europe		OGC WFS and WMS
DISS, Database of Individual Seismogenic Sources, run by INGV (DISS Working Group 2018)	Italy		OGC WFS and WMS

organization running the service, the license (if any) associated with the data, a list of scientific publications, and a general descriptive text.

DISCUSSION

QQuake fits the scenario depicted by the EPOS ERIC and its effort of creating a common platform for all European

organisations dealing with Earth sciences. The plugin represents an additional and practical integration tool for desktop GIS-oriented seismologists. It is designed to lower the barriers for accessing seismological data provided by multiple types of web services, independently from the organization managing them. A quick, friendly and pleasant GUI combined with a high level of user customization were integral parts of the design and are intended to satisfy both non-expert and advanced seismologists using QGIS.

The version of QQuake described in this article refers to the first release of the plugin that was published in the official QGIS repository in December 2020. Its development lasted for about a year thanks to a close collaboration between seismologists working at INGV, a publicly funded research institute, and a private company with consultants who are QGIS core developers. The scientific expertise was used by software developers to create a tool with a solid core structure, combined with the respect of open-source development.

During the development, a limited number of users tested the plugin. We received multiple requests for enhancement at each milestone, and most of them were implemented in time for the first public release. There is always room for improvement and current plans contain, for example, the management and customization of multiple symbol styles, and the addition of data filtering for OGC web services using Contextual Query Language (CQL; <https://www.loc.gov/standards/sru/cql/>). As implementing these new extensions requires dedicated resources, any decisions will be taken based on the user's feedback.

The modular structure of the plugin should allow easy maintenance in the long run and should easily accommodate new modules for additional web service standards. In our intention, these characteristics should guarantee a long life to QQuake, and will make it the ideal basis for further extensions that will help its wide adoption. We expect that the Open Source nature of the plugin and the ease of configuring services will quickly lead to a wide community of contributors adding new service definitions to the plugin. The development is performed using the GitHub repository, where everybody may submit requests for integrating the existing functionalities, adding patches to fix existing bugs, or submitting issues that we will try to address. To facilitate the work of potential developers, we plan to extend the technical documentation about the plugin on a dedicated website (<https://www.emidius.eu/qquake>). Such a website will also be used to support users, and we plan to introduce a series of usage scenarios and examples over time.

REFERENCES

- Albini, P., Locati, M., Rovida, A., and Stucchi, M. (2013). European archive of historical Earthquake data (AHEAD). Milan, Italy: Istituto Nazionale di Geofisica e Vulcanologia (INGV). doi:10.6092/INGV.IT-AHEAD
- Bailo, D., Jeffery, K. G., Spinuso, A., and Fiameni, G. (2016). "Interoperability oriented architecture: the approach of EPOS for solid Earth e-infrastructure," in Proceedings of the 2015 IEEE 11th international conference on e-science, Munich, Germany, August 31– September 4, 2015 (IEEE), 529–534. doi:10.1109/eScience.2015.22
- Bailo, D., Paciello, R., Rabissoni, R., Sbarra, M., and Vinciarelli, V. (2018). *Integration of heterogeneous data, software and services in Solid Earth Sciences: the EPOS system design and roadmap for the building of Integrated Core Services*. Milan, Italy: Rapporti Tecnici, Istituto Nazionale di Geofisica e Vulcanologia (INGV), 22.
- Bailo, D., Paciello, R., Sbarra, M., Rabissoni, R., Vinciarelli, V., and Cocco, M. (2020). Perspectives on the implementation of FAIR principles in solid Earth research infrastructures. *Front. Earth Sci.* 8. doi:10.3389/feart.2020.00003
- Basilì, R., Kastelic, V., Demircioglu, M. B., Garcia Moreno, D., Nemser, E. S., Petricca, P., et al. (2013). *The European database of seismogenic faults (EDSF) compiled in the framework of the project SHARE*. doi:10.6092/INGV.IT-SHARE-EDSF

The evolution of QQuake will be closely linked to two factors: the evolution of the available methods for accessing data developed by the seismological community, and the evolution of QGIS, one of the most commonly used general-purpose Open Source software in the scientific community.

AUTHOR CONTRIBUTIONS

ML Supervised the design of the plugin, the integration of any webservice, extensively tested the resulting functionalities, and the overall content of the submitted article. RV Supervised the integration of Seismogenic Faults in the plugin and any Open Geospatial Consortium compliant web services. MG supervised the plugin Python development. ND was a Python developer.

FUNDING

This project has received funding from European Union's Horizon 2020 research and innovation programme under the Grant Agreement No.730900 and from the Presidency of the Council of Ministers Italian Civil Protection Department (DPC).

ACKNOWLEDGMENTS

We would like to thank all INGV participants to the Task "Virtual Access for Engineering Seismology" of the H2020 SERA project for their support to the development of the QQuake plugin: Andrea Rovida, Andrea Antonucci, Lucia Luzi, Maria D'Amico, Giovanni Lanzano, Roberto Basili and Gabriele Tarabusi. In addition, we thank Giorgio Maria De Santis and Valentino Lauciani for taking care of the public source code repository on GitHub.

- DISS Working Group (2018). *Database of Individual Seismogenic Sources (DISS), Version 3.2.1: a compilation of potential sources for earthquakes larger than M 5.5 in Italy and surrounding areas*. Milan, Italy: Istituto Nazionale di Geofisica e Vulcanologia (INGV). doi:10.6092/INGV.IT-DISS3.2.1
- Dost, B., Zednik, J., Havskov, J., Willemann, R. J., and Bormann, P. (2012). "Seismic data formats, archival and exchange," in *New manual of seismological observatory practice 2 (NMSOP2)* Potsdam: Deutsches GeoForschungsZentrum GFZ. doi:10.2312/GFZ.NMSOP-2_CH10
- Euchner, F., Kästli, P., Heiniger, L., Saul, J., Schorlemmer, D., and Clinton, J. (2016). QuakeML 2.0: recent developments. *Geophys. Res. Abstr.* 18, EGU2016-13830. Available at: <https://meetingorganizer.copernicus.org/EGU2016/EGU2016-13830-1.pdf>.
- Euchner, F., and Kästli, P. (2014). "Towards quakeml 2.0: new packages, techniques, tools," in 2nd European conference on earthquake engineering and seismology, Istanbul, Turkey, 25–29 Aug 2014. Available at: http://www.eaee.org/Media/Default/2ECCES/2ecces_esc/2824.pdf [abstract]
- ISIDe Working Group (2007). *Italian seismological instrumental and parametric database (ISIDe)*. Milan, Italy: Istituto Nazionale di Geofisica e Vulcanologia (INGV). doi:10.13127/ISIDE
- Kästli, P., and Euchner, F. (2018). "QuakeML 2.0: data model on site characterization, presentation at the COSMOS Guidelines," in Workshop at the 36th General Assembly of the European Seismological Commission (36th ESC-GA), Valletta, Malta, September 6 2018 [abstract]

- Locati, M. (2014). "Describing non-instrumental seismological data in QuakeML: the need for a macroseismic extension," in 2nd European conference on earthquake engineering and seismology, Istanbul, Turkey, August 25–29, 2014. Available at: <http://hdl.handle.net/2122/11271>.
- Locati, M., Rovida, A., Albini, P., and Stucchi, M. (2014). The AHEAD portal: a gateway to European historical earthquake data. *Seismol. Res. Lett.* 85 (3), 727–734. doi:10.1785/0220130113
- Michelini, A., Wotawa, G., and Arnold-Arias, D. The ARISTOTLE Team (2017). ARISTOTLE (all Risk integrated system TOWards the hoListic early-warning). *Geophys. Res. Abstr.* 19, EGU2017–11355.
- Michelini, A., Wotawa, G., Arnold-Arias, D., Iley, G., and Forlenza, G. The ARISTOTLE Consortium (2018). The ARISTOTLE multi-hazard expert advice system for ERCC preparedness actions. *Geophys. Res. Abstr.* 20, EGU2018–9541.
- Musson, R. M. W., and Cécic, I. (2012). "Intensity and intensity scales," in *New manual of seismological observatory practice 2 (NMSOP2)* Potsdam: Deutsches GeoForschungsZentrum GFZ. doi:10.2312/GFZ.NMSOP-2_CH12
- Nielsen, M. (2011). *Reinventing discovery: the new era of networked science*. Princeton, NJ: Princeton University Press, 272.
- Rovida, A., and Locati, M. (2015). "Archive of historical earthquake data for the European-mediterranean area," in *Geotechnical, geological and earthquake engineering*. Berlin, Germany: Springer Science and Business Media LLC. doi:10.1007/978-3-319-16964-4_14
- Rovida, A., Locati, M., Antonucci, A., and Camassi, R. (2017). *Archivio storico macrosismico italiano (ASMI) (version 1.0)*. Istituto Nazionale di Geofisica e Vulcanologia (INGV). doi:10.13127/ASMI
- Schorlemmer, D., Euchner, F., Kästli, P., and Saul, J. (2011). QuakeML: status of the XML-based seismological data exchange format. *Ann. Geophys.* 54 (1), 59. doi:10.4401/ag-4874
- Schorlemmer, D., Wyss, A., Maraini, S., Wiemer, S., and Baer, M. (2004). QuakeML—an XML schema for seismology. *ORFEUS Newsletter*. 6 (2), October 2004.
- Schweitzer, J., and Lay, T. (2019). IASPEI: its origins and the promotion of global seismology. *His. of Geo- and Space Sci.* 10 (1), 173–180. doi:10.5194/hgss-10-173-2019
- Suarez, G., Van Eck, T., Giardini, D., Ahern, T., Butler, R., and Tsuboi, S. (2008). The international federation of digital Seismograph networks (FDSN): an integrated system of seismological observatories. *IEEE Sys. J.* 2 (3), 431–438. doi:10.1109/jsyst.2008.2003294

Conflict of Interest: Author MG was employed by the company Faunalia Srl, Pontedera, Italy. Author ND was employed by the company North Road Consulting, Sunshine Coast, Queensland, Australia.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2021 Locati, Vallone, Ghetta and Dawson. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.