Pioneering Easy-to-Use Forestry Data with Forest Explorer

Guillermo Vega-Gorgojo [a,b,*], José M. Giménez-García [a], Cristóbal Ordóñez [c,d], and Felipe Bravo [c,d]

[a] *Group of Intelligent and Cooperative Systems, Universidad de Valladolid, Spain*
E-mails: guiveg@tel.uva.es, jm.gimenez.garcia@gsic.uva.es
[b] *Departamento de Teoría de la Señal y Comunicaciones e Ingeniería Telemática, E.T.S. Ingenieros de Telecomunicación, Universidad de Valladolid, Valladolid, Spain*
[c] *Sustainable Forest Management Research Institute, iuFOR, Universidad de Valladolid & INIA, Spain*
E-mails: a_cristo@pvs.uva.es, fbravo@pvs.uva.es
[d] *Departamento de Producción Vegetal y Recursos Forestales, E.T.S. Ingenierías Agrarias, Universidad de Valladolid, Palencia, Spain*

Abstract. Forest Explorer is a web tool that can be used to easily browse the contents of the Cross-Forest dataset, a resource containing the forestry inventory and land cover map from Spain. The tool can run in any device with a modern browser without requiring any further installation. The user interface completely hides the complexity of RDF, OWL or SPARQL from the user. An interactive map is provided, allowing users to navigate to the area of interest and presenting forestry data with different levels of detail according to the zoom level. Forest Explorer offers different filter controls and is localized to English and Spanish. All the data is retrieved from a SPARQL endpoint containing the latest versions of the Spanish forest inventory and land cover map. Forest Explorer uses a cache and smart geographic querying to limit data exchanges with the endpoint. A live version of the tool is freely available for everybody that wants to try it. Since December 2019, more than 1,800 users have employed Forest Explorer and it has appeared 12 times in the Spanish media. Future work includes the integration of new datasets from other countries and supporting advanced forest management capabilities.

Keywords: forestry, geospatial Linked Data, data access, map visualizations, user interfaces

1. Introduction

Forest science and forestry rely on the use of large-scale datasets that cover lengthy periods of time [19]. This is because trees are long-lived organisms that require continuous monitoring to obtain sound and accurate information. Similarly, large-scale monitoring systems are used to capture the complexity and structure of forests in their full extend. Permanent and extensive data recording systems, particularly land cover maps and forest inventories [27], are needed to implement sound sustainable forest management [4, 21, 22] to ensure a constant flow of ecosystem services such as habitat conservation and raw materials (timber, resin, cork. . .).

Due to long-term scale of forestry actions, the private sector has no incentives to conduct this type of data collection and curation. As a result, the public sector is the main responsible for monitoring forests and providing this information to society. Such information is consumed for different purposes by diverse end users, including forest stakeholders (governments at different levels, environmental NGOs and other lobby groups), operational foresters, data and environmental journalists, interested citizens, and start-up promoters.

*Corresponding author. E-mail: guiveg@tel.uva.es.

However, exploiting forest inventories and land cover maps is a non-trivial task that requires both domain expertise and technical skills. Forestry datasets are typically isolated, described with disparate data schemas, and using unfamiliar (sometimes even proprietary) formats. Given these limitations, Linked Open Data and Semantic Web technologies can help to facilitate the integration and accessibility of forestry data. Towards this goal, the EU project Cross-Forest[1] publishes as Linked Open Data a set of national forestry data sources from Spain – the Cross-Forest dataset from now on. This resource has been released through the creation of a suite of ontologies to represent forestry data, the transformation of their forest inventories and land cover datasets, and the connection among them and with relevant external datasets.

The Cross-Forest dataset is a complex resource that integrates spatial features corresponding to plots, trees, and land cover patches. Target users from the forestry domain need to explore and analyze this dataset in order to fulfill their goals. This is especially challenging as this user group is not fluent in Semantic Web languages. While there are some Linked Data browsers for lay users [7, 8, 13], most of them have limited support for geospatial data. Moreover, visualization tools of geospatial Linked Data are very scarce and typically require knowledge of SPARQL, e.g. Sextant [16].

In this paper we present Forest Explorer, a web tool for easily accessing the contents of the Cross-Forest dataset. Forest Explorer is designed for non-Semantic Web experts, so the user interface is based on an interactive map that completely hides the manipulation of RDF data and exchange of SPARQL queries behind the scenes. The rest of the paper is organized as follows: Section 2 presents the functional requirements of Forest Explorer. Section 3 describes the design and implementation of the tool. In Section 4 we give evidence of the impact of Forest Explorer. The paper ends with a discussion and future work lines in Section 5.

2. Requirements

The Spanish forest inventory [2] is a continuous dataset that is updated every 10 years. A plot is an homogeneous and small area of the territory that constitutes the sampling unit. Forest technicians survey plots to gather tree data (location, species, diameter, and height). The current version of this inventory accrues 1.4M trees, 91.9K plots, and 4.3M positions. The Spanish land cover map[2] contains patches of terrain with similar characteristics, described using polygons over the territory. Patch data includes soil use and dominant tree species. The current version of this dataset accrues 680.2K patches. Spanish provinces are employed to aggregate data in both cases.

The Cross-Forest dataset integrates the latest versions of the Spanish forest inventory and land cover map in Linked Data format. Provinces, patches, plots and trees are modeled as spatial features with a geometry and a set of measures extracted from the corresponding source. The details of the main features of each dataset are shown in Table 1. Their positions are represented using a simple ontology that indicates the Coordinate Reference System and the coordinates of the position. This ontology makes safe reuse [6] of relevant geographical ontologies, including GeoSPARQL [18], the W3C Basic Geo Vocabulary [5], and the ISA Programme Location Core Vocabulary [17]. The transformation of the land cover map into RDF includes the creation of a new layer of patches in a lower resolution in order to make efficient the drawing of polygons on top of a Web map (see Best Practice 4 in [26]). Both datasets have been connected with relevant external sources, such as DBpedia [14], Wikipedia,[3] WikiData [10], The Plant List,[4] or the NCBI taxonomy data [11]. The Cross-Forest dataset is currently published in a SPARQL endpoint[5] and openly available to anyone.

We identified several user groups interested in the resulting integrated dataset: forest managers, operational foresters, data journalists, and citizens. All of them are not knowledgeable about Semantic Web technologies, so they need a tool that shows the data while hiding the complexities of its representation. Specifically, we identified the following requirements:

[1] https://crossforest.eu/

[2] https://www.miteco.gob.es/es/biodiversidad/servicios/banco-datos-naturaleza/informacion-disponible/mfe50.aspx

[3] https://www.wikipedia.org/

[4] http://www.theplantlist.org/

[5] https://crossforest.eu/sparql/

Table 1
Feature types and feature data of the Cross-Forest dataset.

Feature type	Feature data
Province	Number of trees
	Basal area (m^2)
	Volume with bark (m^3)
Patch	Province
	Polygon
	Area (m^2)
	Soil use
	Canopy cover
	Tree species
Plot	Province
	Coordinates
	Number of trees / ha
	Basal area (m^2/ha)
	Volume with bark (m^3/ha)
Tree	Plot
	Coordinates
	Height (m)
	Diameter (mm)
	Tree species

R1 Portable. The tool should run in different devices ranging from desktop computers to mobile phones. Moreover, the installation process should be as simple as possible to facilitate the adoption of the tool.

R2 Effective hiding of RDF/OWL/SPARQL. The end user does not need to know about RDF, OWL, SPARQL, or any other Semantic Web language. Instead, the user interface has to offer appropriate visualizations that present the information in an appropriate way to the users' needs.

R3 Interactive map. Since lay users have embraced map applications [28] and the target dataset is about spatial data, an interactive map seems a suitable visualization for this case. Typical map navigation controls like panning or zooming can be added to easily explore the zone of interest.

R4 Adaptable to different zoom levels. The tool should serve to explore large or small areas. In the former case, lower resolution of geometries should be served. When zooming in to a small area, a high level of detail of geometries is appropriate and tree markers should be drawn.

R5 Filtering capabilities. The user needs to control the information at display. In particular, they should be able to set filters of tree species and land uses, as well as controlling which elements of the view to show, e.g. choosing between province or landscape visualizations.

R6 Multilingual. The tool should be localized to English and Spanish.

3. Design and implementation

In this section we present the tool devised to easily access the contents of the Cross-Forest dataset. The proposed tool is Forest Explorer and satisfies the requirements described in Section 2. The following subsections dive deep into the design and implementation of Forest Explorer.

3.1. Logical architecture

The logical architecture of Forest Explorer is depicted in Figure 1. The *Map generator* is in charge of displaying the view for the end user. This component employs a base map obtained from the *Map server* and listens to the requests made from the different *Feature managers* for showing markers, polygons, popups, or tooltips on top of the map. More specifically, the *Province manager* gathers province geometries and forest inventory data aggregated by province and prepares a suitable display request to the *Map generator*. *Patch, Plot*, and *Tree managers* work in a similar way, obtaining first the information about the corresponding features (see Table 1) located in the geographical bounds of the current map view, and then sending display requests to the *Map generator*.

In order to comply with requirement R4, the different *Feature managers* are activated depending on the zoom level. In case of large areas, the user can choose between province or patch views (R5). In the former case, the *Province manager* takes control and requests the presentation of inventory data aggregated by provinces. In the latter case, the *Patch manager* is activated and requests the lowest resolution layer available of the land cover map. With intermediate zoom levels, the *Patch* and *Plot managers* are both enabled to present the plots on top of a land cover map of the visible area.

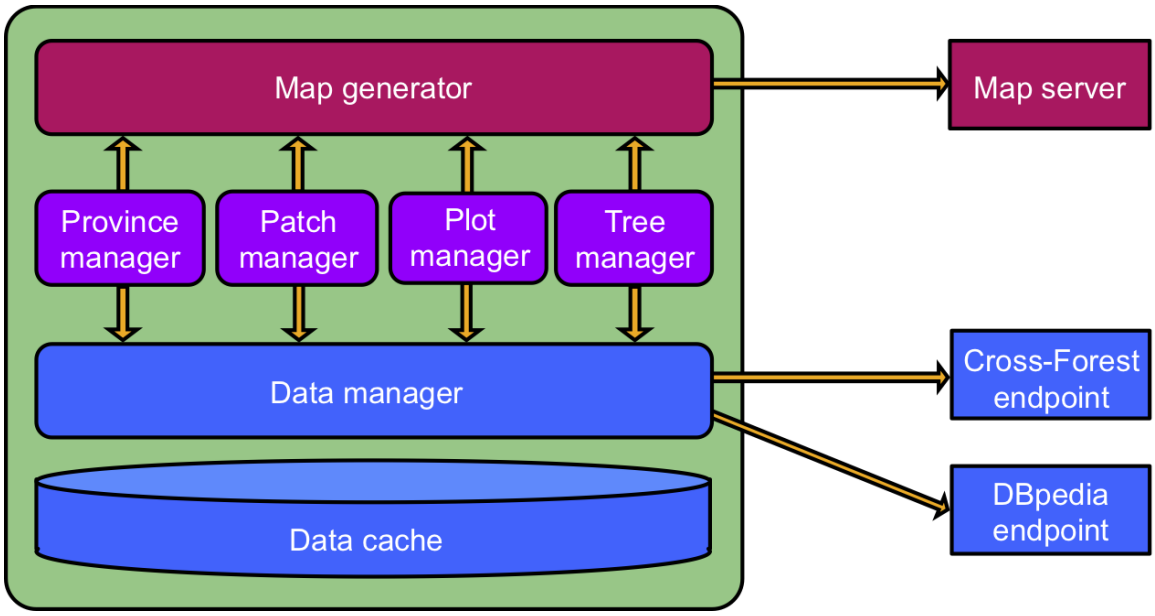

Figure 1. Logical architecture of Forest Explorer.

With high zoom levels, the *Patch* and *Tree managers* are activated – the highest resolution layer available of the land cover map is employed in this case, while the *Tree manager* requests the presentation of tree markers to the *Map generator* based on the inventory data for the target area. Note that this design is compliant with Best Practice 4 in [26], recommending to provide multiple resolution versions of features at different zoom levels.

The *Data manager* handles all the data requests from the *Feature managers*. This component is able to communicate via SPARQL and is configured to use the Cross-Forest and DBpedia endpoints. As described in Section 2, the Cross-Forest dataset contains a Linked Open Data version of the Spanish forest inventory and land cover map. DBpedia is employed as a source of tree species information, providing images and multilingual descriptions (R6). Upon receipt of a request, the *Data manager* first checks if the result is already available in the *Data cache*. In case of a miss, the *Data manager* sends one or more SPARQL queries to the endpoints. Section 3.2 gives further details of the functioning of the *Data manager*.

3.2. Data gathering

As depicted in Section 3.1, the *Data manager* is in charge of all data gathering operations in Forest Explorer. The design of this component is challenging due to a number of reasons: (1) requests look quite varied, referring to different types of features (provinces, patches, plots, and trees) and all their associated data; (2) the size of the Cross-Forest dataset is not small – 4.3GB corresponding to 73.7M triples; and (3) requests can be very numerous since any change in the interactive map will trigger data requests by one or more *Feature managers*.

Fortunately, the different request types can be abstracted to the identification of features localized in a specific area and then obtaining their corresponding feature data. The *Data manager* uses SPARQL template queries for retrieving the features localized in the map view. Since plots and trees have points as geometries, a suitable query basically has to check which points are included in a bounding box. Listing 1 shows the template used for plots in which `{{latnorth}}`, `{{latsouth}}`, `{{lngwest}}`, `{{lngeast}}` are placeholders for the map view bounds.

Listing 1: SPARQL query template for retrieving plots in the map view.

```
SELECT ?plot ?lat ?lng
WHERE {
    ?plot a finv:Plot ;
        pos:hasPosition ?pos .
```

```
    ?pos pos:hasCoordinateReferenceSystem crs:4326 ;
        axis:1 ?lat ;
        axis:2 ?lng .
    FILTER (?lat > {{latsouth}}) .
    FILTER (?lat < {{latnorth}}) .
    FILTER (?lng > {{lngwest}}) .
    FILTER (?lng < {{lngeast}}) .
}
```

As patches and provinces have polygons as geometries, queries have to be adapted to find the polygons intersecting with the map view. The template used for patches is included in Listing 2. Note that we use the bounding box of the polygon (included for all polygons in the dataset) to simplify the detection of patches in the map view. In order to comply with requirement R4, the template is parametrized to select a specific `{{layer}}` and to define a minimum threshold for the area of polygons, i.e. `{{minarea}}`. The *Data manager* chooses the most appropriate layer based on the zoom level of the map view, while it sets the minimum area to 10 pixels in the user device.[6]

Listing 2: SPARQL query template for retrieving patches in the map view.

```
SELECT ?patch ?poly ?west ?east ?north ?south ?area
WHERE {
    ?patch a map:Patch ;
        pos:hasPolygon ?poly .
    ?poly epsg:hasLeftBound ?west ;
        epsg:hasRightBound ?east ;
        epsg:hasUpperBound ?north ;
        epsg:hasLowerBound ?south ;
        pos:hasAreaInSquareMeters ?area ;
        pos:isInLayer {{layer}} .
    FILTER (?south < {{latnorth}}) .
    FILTER (?north > {{latsouth}}) .
    FILTER (?west < {{lngeast}}) .
    FILTER (?east > {{lngwest}}) .
    FILTER (?area > {{minarea}}) .
}
```

Once the *Data manager* obtains the set of features in the area of interest, it retrieves the corresponding data in a next step. Listing 3 shows the query template used for this purpose; it is extremely generic and easily adaptable to each feature type. For example, obtaining the height of a collection of trees just requires setting their IRIs and the height property IRI defined in the CrossForest ontology. As a result, gathering feature data is just a matter of selecting the set of properties to extract for each feature type (see Table 1). Class membership is required in some cases – for example to obtain tree species – so we have included another generic template for retrieving the classes of a set of individuals.

Listing 3: SPARQL query template for obtaining the values of a property for a set of individuals.

```
SELECT DISTINCT ?iri ?value
WHERE {
    ?iri {{propiri}} ?value .
    VALUES ?iri { {{iris}} }
}
```

The *Data manager* stores all retrieved data in a *Cache* so as to reduce exchanges with the endpoint. This is quite effective with feature data, since the template query in Listing 3 can be easily parametrized to avoid requests of feature data already available in the *Cache*. With respect to feature locations, a similar approach consists on caching the results for every map view bounds request, i.e. queries built with Listing 1 for plots, Listing 2 for patches, and so on. However, this solution is too naive as the cache will only get a hit in case of a request with exactly the same map boundaries as a previous one. Instead, the *Data manager* keeps track of the map regions with localized features and exploits this information to restrict queries to unknown areas. Figure 2 illustrates the idea:

(a) At the beginning the *Data manager* has no information of any part of the map.

(b) Upon receiving a feature location request for region $R0$, it asks the endpoint about the features in such area – the *Data manager* stores the results in the *Cache* and updates its tracked area to the bounding box of $R0$.

(c) A request about region $R1$ is decomposed into subregions $R1'$ and $R1''$ – $R1'$ is the intersection of $R1$ and the tracked area, so the included features can be obtained from the *Cache*. Since $R1''$ is unknown to the *Data manager*, it has to query the endpoint to retrieve the features in $R1''$. Then, the *Data manager* updates its tracked area to $R0 \cup R1$.

[6] The assumption here is that patches of less than 10 pixels are barely visible so it is better to not waste time and computation resources with them. The area of a pixel changes with the zoom level, so a patch discarded for being too small can be later displayed in case of zooming in.

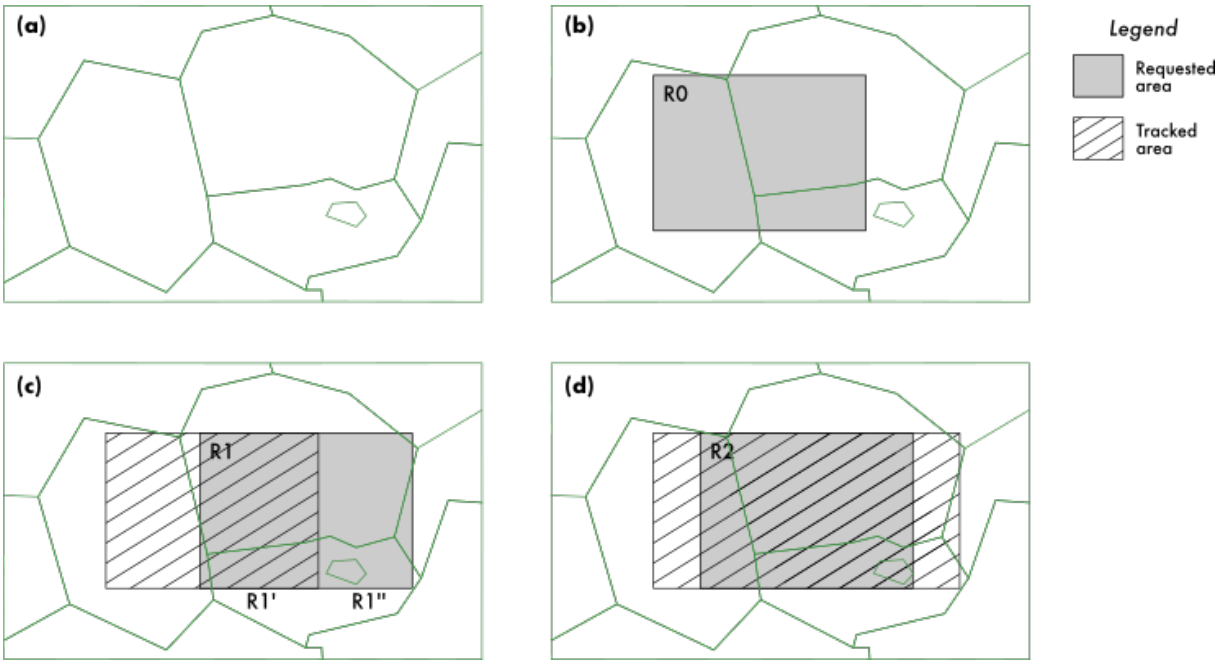

Figure 2. The *Data manager* limits query exchanges with the endpoint by keeping track of the map regions with localized features. The map is initially unknown to the *Data manager* (a), so a feature location request for region $R0$ (b) requires querying the endpoint – the *Data manager* stores the results in the *Cache* and updates its tracked area to $R0$. In (c), a request about region $R1$ is decomposed into subregions $R1'$ and $R1''$ – the features in $R1'$ are obtained from the *Cache* and the endpoint is queried to retrieve the features in $R1''$. The tracked area now includes $R0 \cup R1$. As region $R2$ is contained in the tracked area, the feature location request in (d) is answered with the *Cache*.

(d) A request about region $R2$ does not require further queries to the endpoint, as $R2$ is contained in the tracked area.

3.3. User interface

In order to comply with requirement R3, the user interface of Forest Explorer exposes an interactive map as its main component. Figure 3 shows some snapshots that illustrate the design of the user interface. Since the tool has to work with a diversity of devices (R1), the map is set in fullscreen mode to easily adapt to different screen sizes. Similar to other map applications, panning and zooming are naturally supported for both point-and-click and touch interfaces. In this regard, zoom buttons are included in the lower-right corner; the extra button with a person icon is used to navigate to the user location – this latter functionality is especially useful when employing Forest Explorer with a mobile device in a field trip.

As the user navigates with the map, a *Feature manager* takes control by obtaining feature information from the *Data manager* and then sending display requests to the *Map generator* (see Section 3.1). For example, the *Province manager* controls the view of Figure 3(a); the *Patch manager* is activated in Figure 3(b); the *Patch* and the *Plot managers* collaborate to build the view in Figure 3(c); and the *Tree manager* is active in Figure 3(d).

Beyond map navigation, the user interface needs to include controls to allow the user to adjust the information to display (R5). For this purpose, Forest Explorer includes a form in the upper-left part – see Figure 3(a). This form can take a significant part of the screen in mobile devices, so it can be collapsed by pushing the '-' button – the collapsed form is shown in Figure 3(b)(c)(d).

A key characteristic of the dataset is the tree species of land cover map and forestry inventory. Thus, the form includes a 'Filter species' button that can be pushed to easily browse the taxonomy of tree species (obtained from the Cross-Forest ontology) and then select one or more filters. The form in Figure 3(a) includes two species filters, *Pinus sylvestris* in indigo color and *Pinus pinaster*

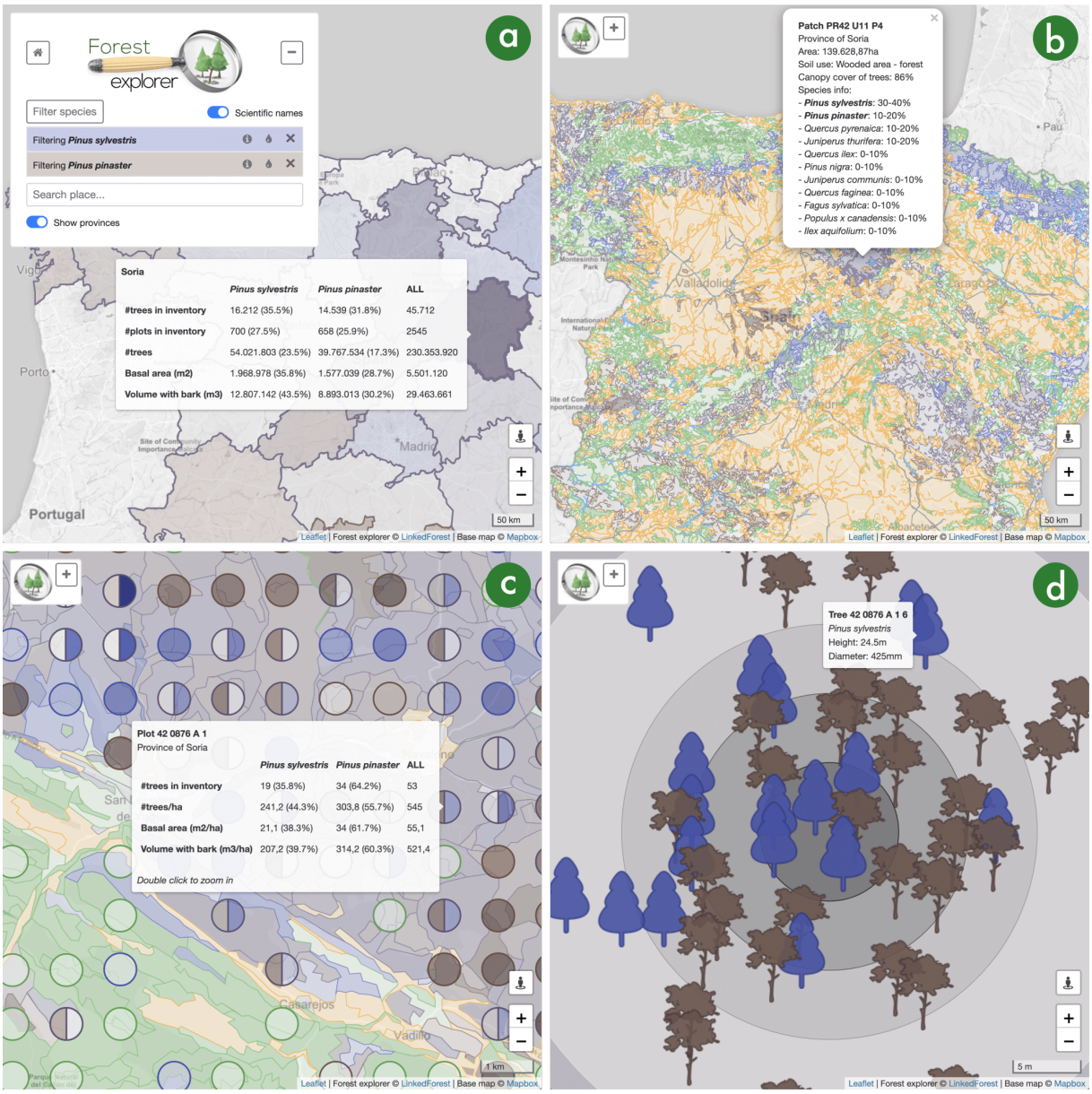

Figure 3. Snapshots of the user interface of Forest Explorer. (a) View of the Spanish provinces in a large area (see the map scale in the lower-right corner) with the form in the upper-left part expanded and showing two species filters, *Pinus sylvestris* in indigo color and *Pinus pinaster* in brown; inventory tree data for the province of Soria is displayed in a tooltip. (b) View of the patches in a large area of Spain with the form in the upper-left part collapsed; patches are plotted in different colors depending on their use (farms in orange, water in blue, artificial in grey, and forests in green); forest patches containing a filtered species use the color filter (indigo and brown in this running example); a pop-up shows the data of a forest patch in the province of Soria. (c) View of a small forest area (see the map scale in the lower-right corner) in the province of Soria; plots are displayed as circles on top of the patches; plots and patches employ the same color code as before; a tooltip shows inventory data for a plot. (d) View of a tiny small forest area corresponding to the center of a plot in Soria; tree markers are shown in their actual positions using taxa-dependent icons and corresponding filter colors; a tooltip shows the species, height, and diameter of a specific tree.

in brown. Each filter includes buttons for removal ('x' icon), color change ('tint' icon), and more info ('info' icon) – the latter one displays a popup with additional information about a tree species such as

an image and a localized description obtained from DBpedia. Species filters have a significant impact in the map view: the color code of species filters is applied to the different features, while species data is also displayed in the different tooltips and popovers – see the snapshots in Figure 3.

In addition, the form includes a textbox for searching places (obtained from the GeoNames dataset[7]) that sets the view of the map to the location of the selected place. Unsurprisingly, the 'Scientific names' switch allows the user to choose between scientific and vulgar names of tree species. Other controls are dependent on the zoom level: the 'Show provinces' switch is displayed with low zoom levels so as to choose between the province – Figure 3(a) – and patch – Figure 3(b) – visualizations. With intermediate zoom levels, the user can hide or show the plots in the map – this applies to Figure 3(c), although the form is collapsed in this snapshot. Color saturation for plots can also be adjusted with a range input. Finally, it is possible to highlight patches by land use, e.g. to find plantation forests, dehesas, thickets, and so on.

3.4. Implementation

Forest Explorer is coded in JavaScript to facilitate its deployment as a web application. As a result, the tool can be used in any device with a modern web browser.[8]

The code is organized in several files that reflect the logical architecture in Figure 1. The implementation effort has been considerably reduced by the integration of a number of JavaScript libraries. We use Leaflet[9] for the interactive map, thus fulfilling the role of the *Map generator* component (see Section 3.1). This library offers almost all the mapping functionality needed for Forest Explorer: vector layers for plotting features, markers, popups, tooltips, map controls, and interaction capabilities. We use two additional Leaflet plug-ins: Leaflet.Locate[10] to geolocate the user, and Leaflet.Circle-sector[11] for drawing pie-shaped plots when filtering multiple tree species – see Figure 3(c). We use the custom Mapbox Light map[12] as a background for displaying forestry data on top – this is the *Map server* component in Section 3.1.

The form of the user interface is built with Bootstrap[13] to simplify front-end development across different browsers and devices. We use jQuery[14] for event handling and manipulating the Document Object Model [30]. We also employ the utility functions of Underscore[15] for handling collections. We use Mustache[16] for templating SPARQL queries (see Section 3.2). Lastly, Google Analytics[17] is included to keep track of the usage of Forest Explorer.

As for the *Data manager* component, all the queries are included in a separate file using templating parameters as necessary, e.g. Listings 1, 2, and 3. We also use a mapping file that assigns keys to ontology IRIs; the *Data manager* only references the keys and this level of indirection decouples the implementation from the Cross-Forest ontology as a result.

Since the tool needs to be localized to English and Spanish (requirement R6), the Cross-Forest dataset is localized to these languages and all the labels employed in the user interface are included in a multilingual strings file. When the tool starts, it gets the browser language preferences in order to select the session language that is then applied to every text shown.

The source code of Forest Explorer is available on GitHub.[18] We have also set up a live version of the tool[19] for anybody who wants to use it. All in all, Forest Explorer can be used with different devices (R1), the user interface hides the intricacies of Semantic Web languages from the user (R2), an interactive map (R3) is employed to present the data adapted to different zoom levels (R4), with a number of filtering capabilities (R5), and localized to English and Spanish (R6), as elaborated above.

[7] http://www.geonames.org/

[8] We have tested Forest Explorer with the latest versions of Mozilla Firefox and Google Chrome in a variety of mobile phones, tablets, and desktop computers.

[9] https://leafletjs.com

[10] https://github.com/domoritz/leaflet-locatecontrol

[11] https://github.com/kluizeberg/Leaflet.Circle-sector

[12] https://www.mapbox.com/maps/light-dark/

[13] https://getbootstrap.com

[14] https://jquery.com/

[15] https://underscorejs.org/

[16] https://mustache.github.io/

[17] https://marketingplatform.google.com/about/analytics/

[18] https://github.com/Cross-Forest/forestexplorer

[19] https://forestexplorer.gsic.uva.es

4. Impact

We submitted a preliminary prototype of Forest Explorer to *Desafío Aporta 2019*,[20] an open data challenge on agrofood, forestry and rural areas that was sponsored by the Spanish Ministry of Economy. This challenge received 40 proposals and ours was shortlisted to a final panel, although we did not get an award. The communication board of Universidad de Valladolid prepared a press release about Forest Explorer reaching the final round of *Desafío Aporta 2019* in 20/12/2019 (media release MR1 in Table 2). At that time the live site of Forest Explorer was openly available and we shared the website link with our contacts. Spanish local media published several articles about the tool – MR2 to MR10 in Table 2. In addition, two newspapers published long interviews with us covering Forest explorer (MR11 and MR12).

In January 2020 we used our social networks to announce the release of Forest Explorer. Specifically, we sent 8 tweets in Twitter that received over 6,500 impressions, more than 100 likes, and about 30 retweets. We also prepared three short posts in Facebook and one more in Twitter, receiving over 1,000 and 375 impressions, respectively.

Beyond media coverage and social networks, we used Google Analytics to assess the uptake of Forest Explorer. Tracked data was obtained from the live site in April 2020. About usage, more than 1,800 users have accessed the tool and have carried out over 3,100 sessions with an average session time of 4 minutes and 10 seconds. 91% of the traffic comes from Spain and the preferred language was Spanish in 86% of the sessions – this is not surprising, as the dataset only covers Spanish forests. Google Analytics gives also information about the employed devices: 42% were Windows computers, 41% Android mobiles or tablets, 12% iOS devices, 4% Mac computers, and 1% Linux computers.

5. Discussion

Since the earlier years of the Semantic Web, the challenge of exploring RDF data was evident [12]. Preliminary projects in this area were targeted to either Semantic Web experts or technology enthusiasts willing to put in the time to learn. More recent works [8] have stressed the importance of supporting both lay users and domain experts, e.g forest managers. Such users may not have any knowledge of SPARQL, OWL, or RDF and require appropriate tools to work with Linked Open Data.

There are some Linked Data browsers, visual query builders, and exploration tools that do not require expert knowledge of Semantic Web technologies [7, 8, 13]. A recent example is RDF Surveyor [29], a lightweight exploration tool targeted to lay users that is part of our previous work. Unfortunately, most of the exploration tools available have limited support for geospatial Linked Data if any, e.g. the visualization of an entity in RDF Surveyor includes a geo widget with a marker if a point location annotated with the Basic Geo Vocabulary[21] is found.

Some systems have been designed to support the visualization of geospatial Linked Data. Again, many proposals are targeted to Semantic Web experts: GeoYASGUI [1] is a SPARQL editor that natively supports GeoSPARQL and provides a result set visualizer; Sgvizler [24] is a JavaScript library that can produce different charts – including maps – with the results of SPARQL queries; and Sextant [16] is an advanced visualization tool that can combine spatial data from several endpoints and represent the temporal dimension, although it still requires knowledge of SPARQL in order to use it. There are seldom visualizers of geospatial Linked Data for non-Semantic Web experts: LinkedGeoData browser [25] is a dedicated visualization tool for OpenStreetMap; and Map4RDF [9] is a browsing tool of geospatial RDF datasets that uses a faceted interface to control the information to display.

In the forestry domain, we can find several initiatives at national or regional level focused on the delivery of raw data, but not using Linked Data and Semantic Web technologies. A remarkable example at pan-European level is the EFISCEN database portal [23] that offers forestry datasets from 32 countries, but using disparate schemata and data formats. National data portals are common, although the typical case is downloadable raw data in proprietary format (e.g. Spanish for-

[20] https://datos.gob.es/es/desafios-aporta/desafio-aporta-2019

[21] http://www.w3.org/2003/01/geo/

Table 2
Spanish media releases covering Forest Explorer.

ID	Media release
MR1	*Una herramienta web desarrollada por miembros de la Universidad de Valladolid que facilita la exploración del inventario forestal, finalista del Desafío Aporta 2019* 20/12/2019 – Gabinete de Comunicación de la Universidad de Valladolid https://comunicacion.uva.es/es_ES/detalle/Una-herramienta-web-desarrollada-por-miembros-de-la-Universidad-de-Valladolid-que-facilita-la-exploracion-del-inventario-forestal-finalista-del-Desafio-Aporta-2019/
MR2	*Crean un explorador forestal para hacer seguimiento de incendios y plagas* 24/12/2019 – Agencia EFE https://www.efe.com/efe/castillayleon/sociedad/crean-un-explorador-forestal-para-hacer-seguimiento-de-incendios-y-plagas/50000473-4138848
MR3	*Crean un explorador forestal para hacer seguimiento de incendios y plagas* 24/12/2019 – Voz Populi https://www.vozpopuli.com/actualidad/explorador-forestal-incendios-plagas_0_1312369090.html
MR4	*Crean un explorador forestal para hacer seguimiento de incendios y plagas* 24/12/2019 – Libre Información https://www.libreinformacion.com/article/Crean-un-explorador-forestal-para-hacer-seguimiento-de-incendios-y-plagas
MR5	*Crean una aplicación para hacer seguimiento de incendios y plagas en España* 24/12/2019 – España Diario https://espanadiario.es/meteo/crean-aplicacion-seguimiento-incendios-plagas-espana
MR6	*Crean un explorador forestal para hacer seguimiento de incendios y plagas* 24/12/2019 – COPE https://www.cope.es/actualidad/tecnologia/noticias/crean-explorador-forestal-para-hacer-seguimiento-incendios-plagas-20191224_581307
MR7	*Crean un explorador forestal para hacer seguimiento de incendios y plagas* 24/12/2019 – eldiario.es https://www.eldiario.es/tecnologia/Crean-explorador-forestal-seguimiento-incendios_0_977452422.html
MR8	*Un explorador forestal seguirá la evolución de plagas y fuegos* 26/12/2019 – Diario Córdoba https://www.diariocordoba.com/noticias/agricultura-medio-ambiente/explorador-forestal-seguira-evolucion-plagas-fuegos_1344405.html
MR9	*Crean un explorador forestal para hacer un seguimiento de incendios y plagas* 26/12/2019 – El Obrero https://elobrero.es/ciencia/39245-crean-un-explorador-forestal-para-hacer-seguimiento-de-incendios-y-plagas.html
MR10	*Crean un explorador forestal para seguir incendios y plagas* 26/12/2019 – Diario Palentino https://www.diariopalentino.es/noticia/Z6F28CCB0-E002-253B-170722B909BEB907/201912/Crean-un-explorador-forestal-para-seguir-incendios-y-plagas
MR11	*Árboles a golpe de clic* 14/01/2020 – Diario de Valladolid. Innovadores http://www.diariodevalladolid.es/noticias/innovadores/arboles-golpe-clic_170743.html
MR12	*La 'App' que muestra los secretos del bosque* 18/01/2020 – El Economista. Revista de Agro http://l.e.eleconomista.es/rts/go2.aspx?h=260137&tp=i-H43-Dc-6T8-FyPmJ-1c-1gVE-1c-FyOR8-1NwpSj

est inventory[22]) or just aggregated results (e.g. the first edition of the Spanish forest inventory[23]).

Visualization of forestry data is commonly achieved through a Geographical Information System (GIS). For example, Global Forest Watch[24] is a web application that provides information about forest status and land use management at global scale. GISs do not rely on Linked Data technologies and use instead their own formats, typically standardized by the Open Geospatial Consortium.[25] As a result, geospatial data is syntactically interoperable, but the integration of external datasets into a GIS is time-consuming and complex [15].

Other forestry tools focus on the analysis of data. This is the case of BASIFOR[26] [3, 20], a flexible and powerful forestry analysis tool that can process data from the Spanish forest inventory. Initially designed as a tool for forestry research, BASIFOR is also used for management and planning purposes as it calculates timber stocks, density, forest structure, specific composition and other forestry variables in a geographical region defined by the user. BASIFOR holds strong capabilities in terms of data manipulation, but is not based on Semantic Web technologies and is thus tied to the schemata of the Spanish forest inventory.

To wrap up, Forest Explorer is a novel exploration tool of forestry Linked Open Data designed for non-Semantic Web experts. Users can interact with a map to navigate to the area of interest and to control the information to display. To limit data exchanges with the SPARQL endpoint, Forest explorer uses a cache and smart geographic querying. So far, we have attracted the attention of the forestry community, not very familiar with Semantic Web technologies. Our future work includes the integration of new data sources from other countries in the Cross-Forest dataset, beginning with Portugal. We also plan to introduce data analysis capabilities to support forest management and planning scenarios. In this way, Forest Explorer could provide similar functionalities as BASIFOR, but with strong visualizations and taking advantage of Linked Open Data for forestry data integration.

[22] http://www.miteco.gob.es/es/biodiversidad/servicios/banco-datos-naturaleza/informacion-disponible/ifn3.aspx

[23] http://www.miteco.gob.es/es/biodiversidad/servicios/banco-datos-naturaleza/informacion-disponible/primer_inventario_nacional_forestal.aspx

[24] http:://www.globalforestwatch.org

[25] https://www.ogc.org/

[26] http://www.basifor.es

Acknowledgements

This work has been partially funded by the European Commission through Cross-Forest (2017-EU-IA-0140) and Spanish Ministry of Science and Innovation through REFORM (PCIN-2017-027, ERANET SUMFOREST) projects.

References

[1] W. Beek, E. Folmer, L. Rietveld and J. Walker, GeoYASGUI: The GeoSPARQL query editor and result set visualizer, *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* **42** (2017), 39–42.

[2] F. Bravo, M. del Río and C. del Peso (eds), *El Inventario Forestal Nacional. Elemento clave para la gestión forestal sostenible*, Fundación General de la Universidad de Valladolid, 2002.

[3] F. Bravo, J.C. Rivas, J.A. Monreal-Núñez and C. Ordóñez, BASIFOR 2.0: Aplicación informática para el manejo de las bases de datos del inventario forestal nacional, *Cuadernos de la Sociedad Española de Ciencias Forestales* (2004), 243–247.

[4] F. Bravo, M. Fabrika, C. Ammer, S. Barreiro, K. Bielak, L. Coll, T. Fonseca, A. Kangur, M. Lof, K. Merganicova, M. Pach, H. Pretzsch, D. Stojanovic, L. Schuler, S. Peric, T. Rotzer, M. del Río, M. Dodan and A. Bravo-Oviedo, Modelling approaches for mixed forests dynamics prognosis. Research gaps and opportunities, *Forest Systems* **28**(1) (2019), ISSN 21715068. doi:10.5424/fs/2019281-14342.

[5] D. Brickley, Basic Geo (WGS84 lat/long) Vocabulary, Technical Report, World Wide Web Consortium, 2006, URL: https://www.w3.org/2003/01/geo/, last visited April 2020.

[6] S. Corlosquet, R. Delbru, T. Clark, A. Polleres and S. Decker, Produce and Consume Linked Data with Drupal!, in: *Proceedings of the 10th International Semantic Web Conference (ISWC 2011)*, LNCS, Vol. 7031, Springer, Bonn, Germany, 2011, pp. 763–778.

[7] A.-S. Dadzie and E. Pietriga, Visualisation of Linked Data – Reprise, *Semantic Web* **8**(1) (2017), 1–21.

[8] A.-S. Dadzie and M. Rowe, Approaches to Visualising Linked Data: A Survey, *Semantic Web* **2**(2) (2011), 89–124.

[9] A. de León, F. Wisniewki, B. Villazón-Terrazas and O. Corcho, Map4rdf – Faceted browser for geospatial datasets, in: *Proceedings of the First International Workshop on Open Data (WOD-2012)*, Nantes, France, 2012.

[10] F. Erxleben, M. Günther, M. Krötzsch, J. Mendez and D. Vrandecic, Introducing Wikidata to the Linked Data Web, in: *Proceedings of the 13th International Semantic Web Conference (ISWC 2014)*, LNCS, Vol. 8796, Springer, Riva del Garda, Italy, 2014, pp. 50–65.

[11] S. Federhen, The NCBI taxonomy database, *Nucleic acids research* **40**(D1) (2012), 136–143.

[12] T. Heath, J. Domingue and P. Shabajee, User interaction and uptake challenges to successfully deploying Semantic Web technologies, in: *Proceedings of the 3rd International Semantic Web User Interaction Workshop (SWUI2006), co-located with the 5th International Semantic Web Conference*, Athens, GA, USA, 2006.

[13] J. Klímek, P. Škoda and M. Nečaský, Survey of tools for Linked Data consumption, *Semantic Web* **10**(4) (2019), 665–720.

[14] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P.N. Mendes, S. Hellmann, M. Morsey, P. Van Kleef, S. Auer et al., DBpedia – a large-scale, multilingual knowledge base extracted from Wikipedia, *Semantic Web* **6**(2) (2015a), 167–195.

[15] J. Lehmann, S. Athanasiou, A. Both, A. García-Rojas, G. Giannopoulos, D. Hladky, J.J. Le Grange, A.-C.N. Ngomo, M.A. Sherif, C. Stadler et al., Managing Geospatial Linked Data in the GeoKnow Project, in: *The Semantic Web in Earth and Space Science. Current Status and Future Directions*, T. Narock and P. Fox, eds, IOS Press, 2015b, pp. 51–77, Chap. 4.

[16] C. Nikolaou, K. Dogani, K. Bereta, G. Garbis, M. Karpathiotakis, K. Kyzirakos and M. Koubarakis, Sextant: Visualizing time-evolving linked geospatial data, *Journal of Web Semantics* **35** (2015), 35–52.

[17] A. Perego and M. Lutz, ISA Programme Location Core Vocabulary, Technical Report, World Wide Web Consortium, 2015, URL: https://www.w3.org/ns/locn/, last visited April 2020.

[18] M. Perry and J. Herring, OGC GeoSPARQL – A Geographic Query Language for RDF Data, OGC Implementation Standard, Open Geospatial Consortium, 2012, URL: http://www.opengis.net/doc/IS/geosparql/1.0, last visited April 2020.

[19] H. Pretzsch, *Forest Dynamics, Growth and Yield*, Springer, Cham, Switzerland, 2009.

[20] M.d. Río, J.C. Rivas, S. Condes, J. Martínez-Millán, G. Montero, I. Cañellas, A.C. Ordóñez, V. Pando, R. San-Martín and F. Bravo, BASIFOR: Aplicación informática para el manejo de bases de datos del Segundo Inventario Forestal Nacional, F. Bravo, M. del Río and C. del Peso, eds, Fundación General de la Universidad de Valladolid, 2002, pp. 181–191.

[21] R. Ruiz-Peinado, A. Bravo-Oviedo, E. López-Senespleda, F. Bravo and M. del Río, Forest management and carbon sequestration in the Mediterranean region: A review, *Forest Systems* **26**(2) (2017).

[22] R. Ruiz-Peinado, M. Heym, L. Drossler, P. Corona, S. Condes, F. Bravo, H. Pretzsch, A. Bravo-Oviedo and M. del Río, Data Platforms for Mixed Forest Research: Contributions from the EuMIXFOR Network, in: *Dynamics, Silviculture and Management of Mixed Forests*, A. Bravo-Oviedo, H. Pretzsch and M. del Río, eds, Springer, Cham, Switzerland, 2018, pp. 73–101.

[23] M.J. Schelhaas, S. Varis, A. Schuck and G.J. Nabuurs, EFISCEN Inventory Database, 2006, URL: http://www.efi.int/portal/virtual_library/databases/efiscen/, last visited April 2020.

[24] M.G. Skjæveland, Sgvizler: A javascript wrapper for easy visualization of SPARQL result sets, in: *Extended Semantic Web Conference*, Heraklion, Greece, 2012, pp. 361–365.

[25] C. Stadler, J. Lehmann, K. Höffner and S. Auer, LinkedGeoData: A core for a web of spatial open data, *Semantic Web* **3**(4) (2012), 333–354.

[26] J. Tandy, L. van den Brink and P. Barnaghi, Spatial data on the Web best practices, W3C Working Group Note, OGC 15-107, OGC & W3C, 2017, URL: https://www.w3.org/TR/sdw-bp/, last visited March 2020.

[27] E. Tomppo, T. Gschwantner, M. Lawrence and R.E. McRoberts (eds), *National forest inventories: Pathways for common reporting*, Springer, Cham, Switzerland, 2010.

[28] B. Veenendaal, M.A. Brovelli and S. Li, Review of web mapping: Eras, trends and directions, *ISPRS International Journal of Geo-Information* **6**(10) (2017), 317.

[29] G. Vega-Gorgojo, L. Slaughter, B.M. Von Zernichow, N. Nikolov and D. Roman, Linked Data Exploration With RDF Surveyor, *IEEE Access* **7** (2019), 172199–172213.

[30] WHATWG (Apple, Google, Mozilla, Microsoft), DOM, Living Standard, WHATWG, 2020, URL: https://dom.spec.whatwg.org/, last visited March 2020.