International Journal of Geo-Information

MDPI

Article

Geospatial Queries on Data Collection Using a Common Provenance Model

Guillem Closa [1,*], Joan Masó [1], Núria Julià [1] and Xavier Pons [2]

[1] Grumets Research Group, CREAF, Edifici C, Universitat Autònoma de Barcelona, 08193 Bellaterra, Catalonia, Spain; joan.maso@uab.cat (J.M.); n.julia@creaf.uab.cat (N.J.)
[2] Grumets Research Group, Dep de Geografia, Edifici B, Universitat Autònoma de Barcelona, 08193 Bellaterra, Catalonia, Spain; Xavier.Pons@uab.cat
* Correspondence: guillem.closa@uab.cat

Abstract: Lineage information is the part of the metadata that describes "what", "when", "who", "how", and "where" geospatial data were generated. If it is well-presented and queryable, lineage becomes very useful information for inferring data quality, tracing error sources and increasing trust in geospatial information. In addition, if the lineage of a collection of datasets can be related and presented together, datasets, process chains, and methodologies can be compared. This paper proposes extending process step lineage descriptions into four explicit levels of abstraction (process run, tool, algorithm and functionality). Including functionalities and algorithm descriptions as a part of lineage provides high-level information that is independent from the details of the software used. Therefore, it is possible to transform lineage metadata that is initially documenting specific processing steps into a reusable workflow that describes a set of operations as a processing chain. This paper presents a system that provides lineage information as a service in a distributed environment. The system is complemented by an integrated provenance web application that is capable of visualizing and querying a provenance graph that is composed by the lineage of a collection of datasets. The International Organization for Standardization (ISO) 19115 standards family with World Wide Web Consortium (W3C) provenance initiative (W3C PROV) were combined in order to integrate provenance of a collection of datasets. To represent lineage elements, the ISO 19115-2 lineage class names were chosen, because they express the names of the geospatial objects that are involved more precisely. The relationship naming conventions of W3C PROV are used to represent relationships among these elements. The elements and relationships are presented in a queryable graph.

Keywords: provenance; lineage; graph; data queries; metadata

Citation: Closa, G.; Masó, J.; Julià, N.; Pons, X. Geospatial Queries on Data Collection Using a Common Provenance Model. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 139. https://doi.org/10.3390/ijgi10030139

Academic Editors: Wolfgang Kainz and Mohsen Kalantari

Received: 9 January 2021
Accepted: 1 March 2021
Published: 5 March 2021

1. Introduction

According to [1], over two-thirds of Earth and Environment works cannot be reproduced, due to (1) the lack of methodology or code, (2) access limitations to raw data, or (3) incomplete metadata documentation. This so called "usability gap" [2] can be resolved by a solid data model for provenance information, which includes a mechanism for inferring common processing chains form it. This can be supported by new tools that improve the user understanding of the data production process [3]. Some recommendations for increasing the level of transparency and for capturing the "Whole Tale" of the computational environments are presented in [4]. In the geospatial world, the Committee on Earth Observation Satellites (CEOS) [5] Analysis Ready Data (ARD) prepared with minimum processing requirements and metadata [6]. This facilitates the use and interoperability of Remote Sensing (RS) products and it aims to reduce the usability gap. One of the requirements in the ARD product family specification is to clearly state the processes applied to the data. However, ARD data may not be applicable under certain circumstances, because case studies occasionally generate slightly and sometimes clearly different products for apparently identical implementations of common algorithms [7,8]. An example of this

is processing methods that favor a particular condition (e.g., mountain areas or Mediterranean climate). Therefore, it is necessary to have precise information regarding how an ARD product was created and, thus, be able to define its a priori limitations. Therefore, information at the different abstraction levels of geoprocessing services will help to distinguish and discriminate geoprocessing tools [9].

In this paradigm, geospatial lineage, information regarding the origins of geospatial data products has been indicated to be a fundamental issue in spatial information [10]. Tracking back the production workflow, scientists can assess the usability in terms of data quality, which is conditioned by steps that are more sensitive to uncertainties and error propagation [11,12]. Moreover, when lineage information is complete and indicates actual data sources and process code, it can be used for data replication (reproducibility purposes) and workflow reuse (with other inputs). Summarizing, lineage information helps to overcome the knowledge gap between data providers and data consumers who want to reuse these models, data, or algorithms in different contexts, regions, or purposes.

We could assume that an accurate statement would be enough for documenting lineage; however, a well-known internal structure is necessary for extracting the maximum benefit from it. The usefulness of lineage increases when an interoperable metadata model is used, which makes it possible to exchange and share lineage in a distributed information environment [13]. The Provenance Working Group of the World Wide Web Consortium (W3C) established a model to represent domain independent provenance over the web. The W3C PROV (PROV from now on) defines provenance as information regarding the entities, activities, and people involved in producing a piece of data or thing [14]. In the geospatial domain, the term lineage has been used to define the provenance of Geographic Information System (GIS) products [15]. The Spatial Data Transfer Standard (SDTS) [16] of the Federal Geographic Data Committee (FGDC) defined a lineage model, and the International Organization for Standardization (ISO) included a lineage model first in ISO 19115:2003 [17] and later in ISO 19115-1:2014 [18] and ISO 19115-2 [19]. Although the term lineage is preferred by geospatial standards, several works also use provenance as a synonym [20,21]. In this paper, a slight differentiation between them is used: the term lineage is the history of a single dataset, while provenance refers to the integrated history of one or more datasets.

According to [22], the initial version of the ISO model offered an unstructured narrative of the history of the spatial resource and, therefore, it is unsuitable for automation purposes. To cover this gap, Refs. [23,24] propose adapting the PROV model to the requirements of the geospatial community. Other authors, such as [10,13], went further and semantically enriched the PROV structure with geospatial particularities. However, the recently edited version of the ISO metadata standard [19] has been substantially improved in structure and it is now able to better represent the process chain of a production line [25].

The selected provenance visualization approach is another key factor in enhancing the understanding of the data production. In complex environments, scientists rely on visualization tools to help them understand large amounts of data that are generated from experiments [26]. Visualization tools are essential in the phases of discovery and inspection of data and process chains [27]. Provenance can have a complex structure with multiple relationships and dependencies. This can overwhelm users exploring the different process steps that lead to a dataset. Given the linked nature of provenance information, Ref. [28] suggests a graph approach that effectively summarizes the process chain.

Some GIS and RS tools provide users with a functionality to store lineage information. However, despite the potential of the recorded lineage information, systems rarely provide query capabilities that go beyond basic metadata visualization. Therefore, there is a need for interactive systems and tools able to visualize, query, or mine provenance information [29]. However, given the multiple relationships and dependencies between different datasets that provenance information can describe, designing these tools is a challenging task.

This contribution tackles this issue, presenting a system that provides query capabilities for a graph representing the provenance of a collection of datasets or federated

metadata services. Our graphs go beyond the typical lineage graphs that are sources or process chain oriented. Instead, the system can show the tools used, executions carried out, outputs generated, and agents that are involved in the collection of datasets. Four different levels of geoprocessing abstraction are proposed in order to be able to include all the heterogeneity and variety of provenance information in a single graph, namely the execution, tool, algorithm, and functionality levels. The preliminary results have been implemented and tested in a web map browser.

The rest of this paper is organized, as follows: in Section 2, we present the chosen provenance representation model, paying special attention to the different levels of abstraction of the geoprocessing tools; in Section 3, the potential for provenance query is described; in Section 4, a query provenance system is presented; in Section 5, we describe a use case to show the implementation of the system and the visualization tool. Section 5 also provides a discussion that is based on a use case that exemplifies the usefulness of our proposal. In Section 6, we identify future work and, finally, a summary of the conclusions is presented in Section 7.

2. Provenance Model

This paper makes the most of the legacy of the lineage model in the ISO 19115 family standard and the W3C PROV provenance model to propose an evolved model that relates collections of datasets in a network, while using provenance as a basis and considering a set of levels of abstraction for process steps.

2.1. Levels of Abstraction of Process Steps

The definition and capture of different levels of granularity of geospatial provenance data have motivated some works, such as [30]. A comparison between ISO and PROV describes the provenance at different levels of granularity of geospatial data (feature types, features, attribute types, attributes) and proposes a description of the different levels of granularity with PROV [24]. In this paper, different levels of abstraction of the process steps: process run, processing tool, algorithm, and functionality are proposed (see Figure 1). The definitions of these concepts are as follows (going from more concrete to more abstract):

- Process run (process step): an individual execution of a processing tool with a specific set of parameters. It is a single GIS execution. Represented by LE_ProcessStep in ISO and as an Activity in PROV.
- Processing tool (executable or web service): a specific version of an implementation of an algorithm in a piece of software that can obviously be executed several times with different sources and parameters. This is what we can find in the GitHub, buy from a software vendor, use in a web processing service, etc. Represented by LE_Processing in ISO and as an Entity in PROV.
- Algorithm (model): a set of mathematical and logical steps that allow for transforming some inputs into some outputs. It can be implemented in software in different ways and programming languages. This is what a scientific paper usually describes. Represented by LE_Algorithm in ISO and as an Entity in PROV.
- Functionality (operation): an operation that transforms data into other data with spatial problem-solving orientation. This is a black box that can be implemented with different algorithms, potentially giving slightly different results. This is what a GIS and RS textbooks describe. It does not exist in ISO and is represented as an Entity in PROV.

These levels of abstraction are related, as follows (see Figure 1): the process runs as a single execution and executes a processing tool. The processing tool implements an algorithm. Finally, the algorithm gives a functionality.

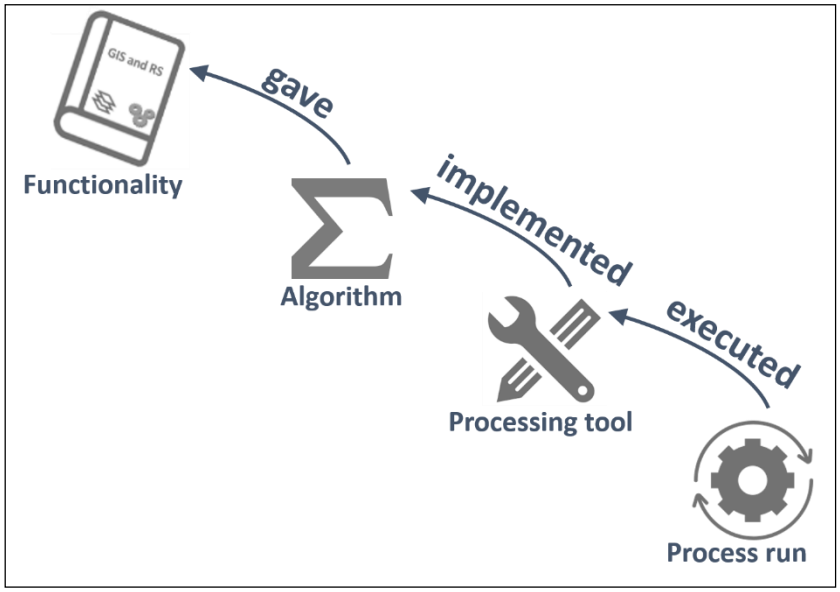

Figure 1. Representation of the four levels of process step abstraction.

The inclusion of functionality and algorithm descriptions as a part of provenance provides high-level provenance information that is independent from the software or web processing tool used. This makes it possible to take a provenance graph that is initially documenting specific processing tools and then abstract it into a higher-level diagram that describes the aim of the processing chain. This idea goes beyond pure reproducibility by providing reasoning and the intentions that are behind each process step. Exploiting this approach makes it possible to:

- Represent together in a single provenance representation the origin of different datasets.
- Formalize provenance queries at different levels of abstraction. For instance, (from more abstract to more specific):
    - What functionalities are used more frequently in my organization?
    - What is the best algorithm that I can use for a quality test of my final products?
    - How are my results affected by a specific processing tool version that has a bug?
- Translate a lineage description that is executed with one software vendor into another software product and reproduce the results.

The description of all functionalities used in a processing chain depicts the task that the workflow was designed for. In the geospatial domain, a task describes all of the actions that require human input or the knowledge about context and it is usually composed by functions [31,32] (e.g., watershed delineation or a polluting industry buffer zone delimitation). In any case, tasks are not bound to specific tools. Back in 1998, Ref. [31] demonstrated that it is feasible to translate flow charts that are based on a universal GIS functionality into specific GIS software flow charts when functionalities can be mapped to GIS operations (tools used). However, although several classifications of the principles of GIS functionalities have been formulated [33,34], semantic descriptions are still ambiguous or incomplete [35]. Nevertheless, the main GIS and RS software products perform a common core of functionalities (tools with the same problem-solving intentionality). Table 1 shows a subset of the common GIS and RS functionalities and the name of the implementation in ArcGIS [36], MiraMon [37], GRASS [38], and SNAP [39].

Table 1. Some Geographic Information System (GIS) and Remote Sensing (RS) functionalities with the different names in the ArcGIS, MiraMon, GRASS, and SNAP software.

GIS functionality	ArcGIS tool	MiraMon tool	GRASS tool
Geometric union	Union	CombiCapa	v.overlay(or)
Extraction	Clip	Retalla	v.overlay(and)
Proximity	Buffer	BufDist	v.buffer
Distance	Distance	BufDist	r.distance
Surface interpolation	Interpolation	InterPNT	r.resamp.interp
Slope	Slope	Pendent	r.slope.aspect
Aspect	Aspect	Pendent	r.slope.aspect
Shade	Hillshade	Illum	r.relief
Visibility	Viewshed	Visible	r.viewshed
Cell statistics	Cell statistics	EstRas	r.univar
Field statistics	Field statistics	EstCamp	v.vect.stats
Joining	Join	GestBD	v.db.join
Merging	Merge	GestBD	v.patch
Aggregation	Dissolve	Ciclar	v.disolve
Feature selection	Select by features	VecSelect	v.extract
RS functionality	**SNAP**	**MiraMon tool**	**GRASS tool**
Georeferencing	Orthorectification	CorrGeom	i.ortho.photo
Radiometric correction	Sen2Cor	CorRad	i.atcorr

2.2. Linking Geospatial Dataset Collections Through the Process History

One possible approach to present to present lineage is to build a system that is based on the ISO 19115 family standard [25]. More specifically, the interactive metadata visualization tool used (GeMM) [37] provides a graphical interface that shows lineage information in a hierarchical tree form. A tree represents the lineage of one geospatial dataset. While the lineage model in the ISO metadata standards focuses on the final product instances and their sources and process steps, the PROV Data Model (PROV-DM) focuses on the relationship between *agents*, *entities* and *activities*:

- PROV:*used* → Relates activities (PROV:Activity) with entities (PROV:Entity).
- PROV:*wasAssociatedwith* → Relates activities (PROV:Activity) with agents (PROV:Agent).
- PROV:*wasGeneratedBy* → Relates entities (PROV:Entity) with activities (PROV:Activity).
- PROV:*wasAttributedTo* → Relates entities (PROV:Entity) with agents (PROV:Agent).
- PROV:*wasInformedBy* (These PROV core relationships are not used in this proposal) → Relates activities (PROV:Activity) activities (PROV:Activity).
- PROV:*wasDerivedFrom* → Relates entities (PROV:Entity) with entities (PROV:Entity).
- PROV:*actedOnBehalfOf* (These PROV core relationships are not used in this proposal) → Relates agents (PROV:Agent) with agents (PROV:Agent).

Several authors, such as [10,40], have proven that it is possible to map both models. When considering that ISO datasets sources results and executable are PROV entities, the ISO process steps are PROV activities and ISO responsible parties (persons or institutions) are PROV agents, the set of PROV relationships between agents, entities, and activities (Figure 2) are immediately applicable to ISO model, as seen in Table 2.

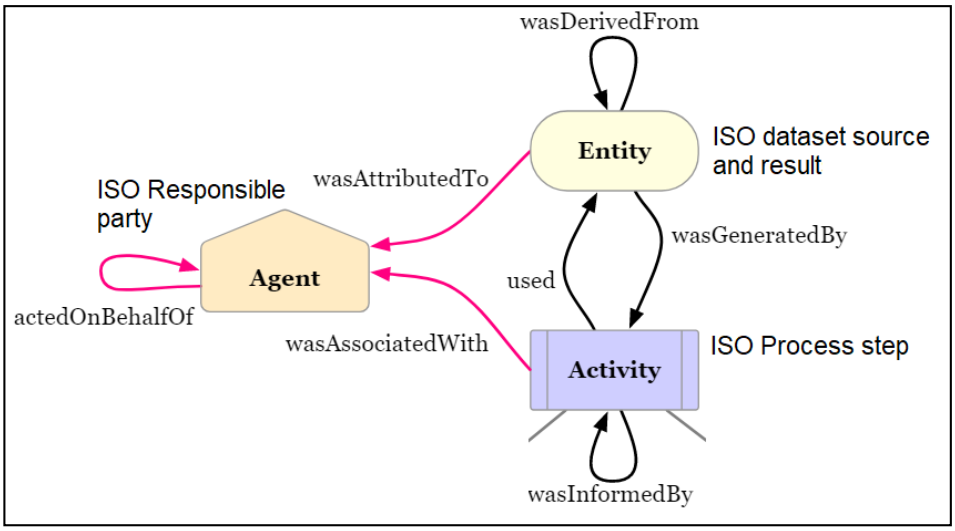

Figure 2. The three Starting Point classes in PROV mapped to International Organization for Standardization (ISO) classes for lineage. PROV properties that relate them emerge (from the PROV ontology modified).

Table 2. W3C PROV relationships and the ISO classes that they connect.

W3C PROV Relationships	ISO 19115 Classes Connected
Used	LI_ProcessStep/source → LI_Source (or LE_Source)
Was Associated with	LI_ProcessStep/processor → CI_Responsability
Was Generated By	LI_ProcessStep/output → LE_Source
Was Attributed To	LE_ProcessStep/processingInformation/LE_Processing/softwareReference/CI_Citation/citedResponsibleParty→CI_Responsability
Was Derived From: gave	LE_ProcessStep/processingInformation/LE_Processing/procedureDescription →CharacterString (representing the Functionality; no existing in ISO)
Was Derived From: implemented	LE_ProcessStep/processingInformation/LE_Processing/algorithm → LE_Algorithm
Use: executed	LE_ProcessStep/processingInformation → LE_Processing

2.3. W3C PROV for Representing the Process Abstraction Levels

In addition to the presented PROV core type relationships, which are high-level descriptions, there is a mechanism to 'open up' these descriptions to a lower level specification. Therefore, three subtypes of PROV-DM core relationships were introduced to relate the four levels of processing abstraction that are described in Section 2.1:

- *executed* → The subtype *used:executed*, is introduced to relate a LE_ProcessStep (PROV:Activity) with its LE_Processing tool (PROV:Entity). A LE_ProcessSet *executed* a LE_Processing tool once.
- *implemented* → The subtype *wasDerivedFrom:implemented* is used to relate the LE_Processing tool (PROV:Entity) with an LE_Algorithm (PROV:Entity). A LE_Processing tool *implemented* an LE_Algorithm.
- *gave* → The subtype *wasDerivedFrom:gave* is used to relate an LE_Algorithm (PROV:Entity) with a Functionality (PROV:Entity). An LE_Algorithm *gave* a Functionality.

2.3.1. Combining W3C PROV and ISO19115 to Represent Provenance

In this paper, a composed solution to encode the provenance of a collection of datasets: combining ISO with PROV is chosen. To present each element, we chose the ISO 19115-2 lineage class names (LI_Lineage), because they express the names of the geospatial objects that are involved more precisely, and we mapped them to PROV core types. The relationship naming conventions of PROV were used to represent relationships among agents (CI_Responsability), actions (LE_ProcessingSteps), and entities (all other classes). Figure 3 presents the result of this combination.

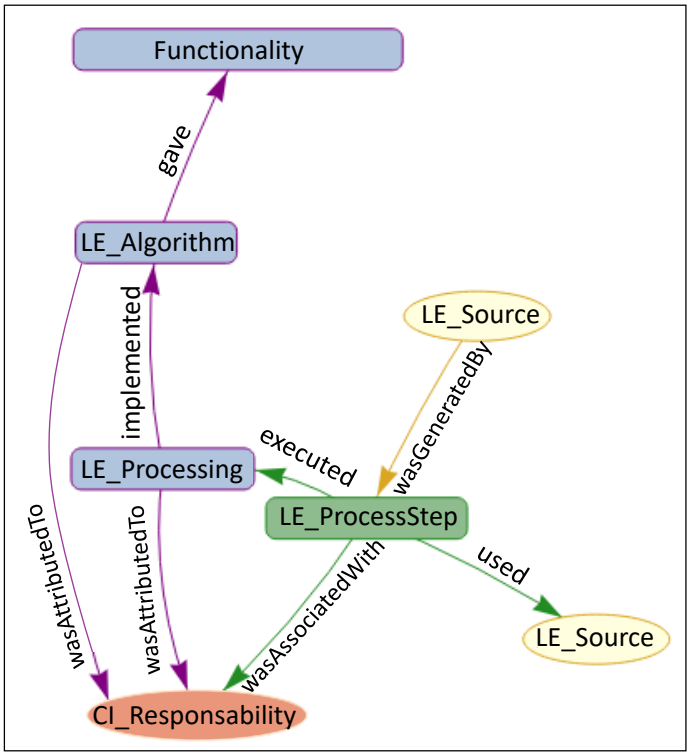

Figure 3. How the use of PROV relates the lineage ISO elements.

By automatically transforming ISO lineage metadata into PROV, we can merge ISO diagrams for a single product and then describe the provenance of many products in a single graph. Global identifiers for sources and processing tools that are supported by the ISO MD_Identifier class [41] make it possible to coalesce repeated objects and integrate remaining objects in a provenance graph connected by PROV relationships (see Figure 4).

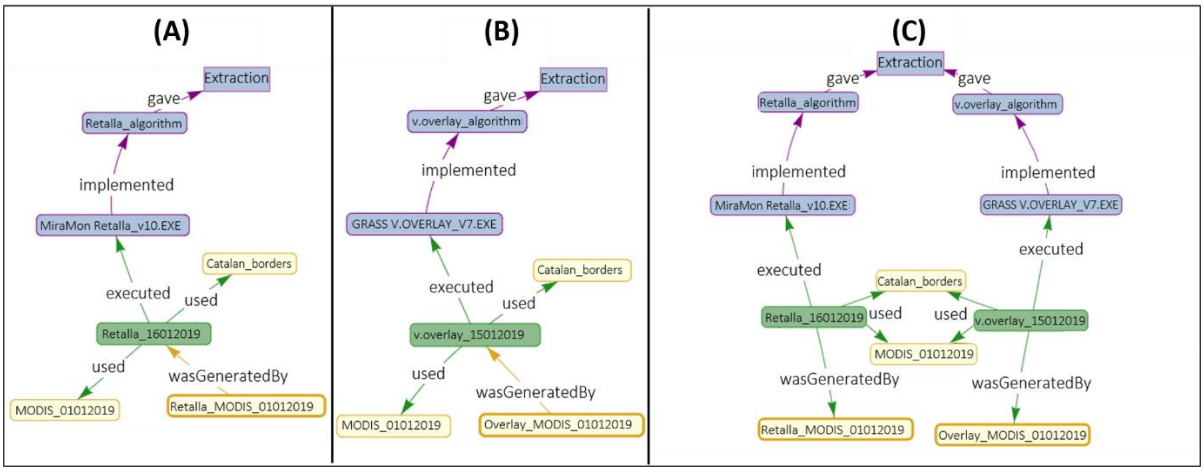

Figure 4. Graph (**A**,**B**) represent individual executions with the same functionality carried out with MiraMon and GRASS, respectively. (**C**) represents provenance of the two executions merged into one.

2.3.1.1. Relating Lineage Elements in Different Levels of Abstraction

Adding the four levels of abstraction that were introduced in Section 2.1 to this mapping makes representing provenance at a higher-level of abstraction possible. For instance, inputs and outputs can be directly related to the processing tool or even to the algorithm or functionality. To allow this, a set of different PROV relationships is introduced.

There are three possible scenarios in which it is necessary to use this set of different PROV relationships:

1. Relating outputs (LE_Source) with the different levels of process step abstraction (Figure 5A). The PROV core type relation *wasDerivedFrom* is used with LE_Processing, LE_Algorithm and Funcionatily. The reason is that, while LE_ProcessStep can be assimilated to a PROV:Activity (and use *wasGeneratedBy*), the others are assimilated to a PROV: Entity.
2. Relating inputs (LE_Source) with the different levels of abstraction (Figure 5B). The relationship *processed* is introduced, a subtyping of the PROV-DM core *wasDerivedFrom*. In this case, the relationship changes from connecting a PROV:Activity (LE:ProcessStep) with a PROV:entity (LE_Source), to connecting entities (LE_Processing, LE_Algorithm, Functionality) with entities (LE_Source).
3. Relating the different levels of abstraction between themselves (Figure 5C). In this case, as there in no change in the relationship type (it is always a PROV:Entity to PROV:Entity), we are free to use the lowest level (in terms of process abstraction) provenance element connector.

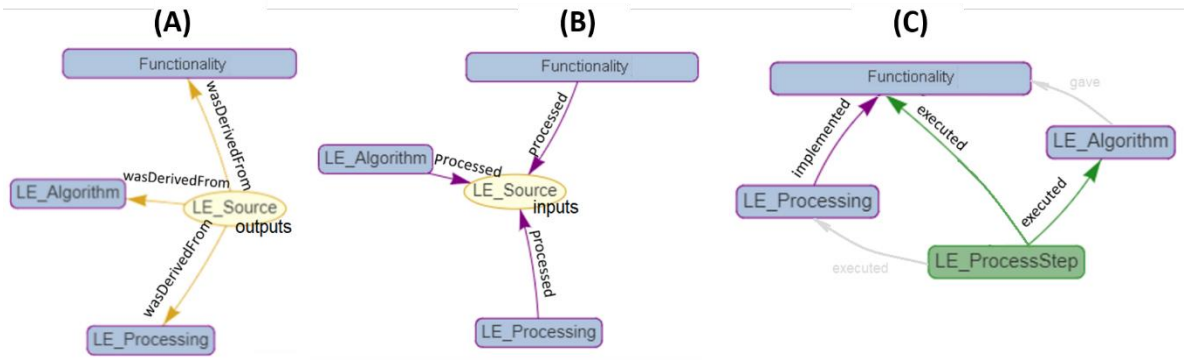

Figure 5. (**A**) connects entities (LE_Source—outputs) with entities (LE_Processing, LE_Algorithm, Functionality). (**B**) connects entities (LE_Processing, LE_Algorithm, Functionality) with entities (LE_Source—intputs). (**C**) connects the LE_ProcessStep, LE_Processing, LE_Algorithm, and Functionality.

Most of the times, these relationships are not necessary and they may not be shown. However, later, we will present a practical use case in which simplifications in the graph showing only higher levels of abstraction require that some of them are made explicit.

3. Queries Facilitated by the Provenance Data Model

The exploitation of provenance is deepened when queries are formulated. There are several examples in the literature where a consolidated and standardized data model and the associated interoperable vocabularies are the base for a query language that exploits the data that are expressed in the data model, Ref. [42] look at this relationship for spatiotemporal data; Ref. [43] show this relationship in the linked data; and, Ref. [44] recognizes the link between a model and query language for graphs. In our case, the separation of concepts and the introduction of the different levels of abstraction into the data model facilitate formulating formal queries that involve concepts and relationships. In addition, because the provenance model associates datasets, it allows queries to be formulated on a dataset collection.

Queries can be formulated over the different lineage elements. Table 3 provides forty-four general queries over lineage elements. These queries are only examples of the potential of what we can get by querying the provenance of a collection of datasets. Table 3 is a dual entry table that relates the different lineage elements between them, with the exception of the first row, which restricts queries to only one lineage element.

Depending on which aspect of provenance is queried, different benefits can emerge:

- *Information and transparency*: lineage allows for us to learn how datasets were developed.
- *Trust and authority* of the sources and tools used: the authority can help in determining liability.
- *Data quality*: the sources and processes involved can be used to estimate uncertainty and blunder propagation.
- *Documentation and reproducibility*: the documentation of the complete processing chain can help in the reproducibility, especially if provenance indicates the actual and exact datasets, parameters, and tools used.
- *License and accessibility*: related to the authorship rights of the sources used.

Table 3. Forty-four examples of queries about provenance.

	Process Run	**Processing Tool**	**Algorithm**	**Functionality**	**Agent**	**Source**	**Time**	**Output**
	Did any execution cover Africa?	Was version 3 of *InterPNT* used?	Was a *kriging* algorithm used?	Was a *reprojection* functionality used?	Did the user Bob have a role in the creation of this dataset?	Was a dataset called Rivers used?	Was something executed in 2013?	Was a rain dataset created?
Process run	**What was executed after Process step 5?**	Did Process step 5 use version 3 of *InterPNT*?	Was Process step 5 a *krigring* interpolation?	What was the purpose of Process *step 5*?	Was Process step 5 executed by Bob?	Did Process step 5 use a DEM of 2 m?	How long did Process step 5 last?	Which data were generated with Process step 5?
Processing tool		**Which tool is often used right after *InterPNT*?**	Does version 3 of *InterPNT* support an *IDW* interpolation?	Did *InterPNT* and *r.resamp.intep* implement equivalent functionalities?	Which interpolation tools were developed by a trusted software vendor?	Did version 3 of *InterPNT* use a GeoJSON format?	Is version 3 of *InterPNT* the last version available?	Which outputs were created with version 3 of *InterPNT*?
Algorithm			**Which different versions of the *buffer* algorithm are used?**	Did GRASS and MiraMon *buffer* tools use the same algorithm?	Was Bob the author of any of the algorithms used?	Is this algorithm suitable for categorical data?	When was this algorithm developed?	Which outputs were created using this algorithm?
Functionality				**Have all the corrections been made with the same software?**	Who did the radiometric corrections?	Which of the datasets used were reprojected?	Was something reprojected in 2015?	Which outputs were reprojected?
Agent					**Who used tools developed by Bob?**	Which of the sources used were produced by a public institution?	When did Bob make his first execution in this collection?	Which institution generated the resulting maps?
Source						**Which two sources were used together?**	Are all the sources from the same temporal interval as the output?	Was a rain intensity dataset needed to create a river flow dataset?
Time							**How long did it take to complete production?**	When was this output generated?
Output								**Was this output a revision of another output?**

4. Representation and the Query Provenance Tool

In this paper, we present the design of a new characteristic that is included in the Provenance Web system to provide support to an integrated provenance visualization and enable the potentialities of querying it. The client will need request linage information from the requested layers, depending on the user actions (see Figure 6, steps 0 and 1). Lineage is communicated from servers hosting the ISO metadata (as described in Section 4.1) to the client, which is capable of merging and presenting it in a provenance graph (see Figure 6, step 2 and Section 4.2). A web client will present the lineage in a window of a map browser. Provenance is presented in a graph that takes advantage of the data model, the common processes or sources, and the abstraction levels to create new connections (see Figure 6, step 3). On top of this, the window offers different ways to filter and query the graph (see Section 4.3). This allows for the user to control the amount of content in the graph and progressively increase the understanding of the graph itself and, with that, the understanding of the provenance information that it represents.

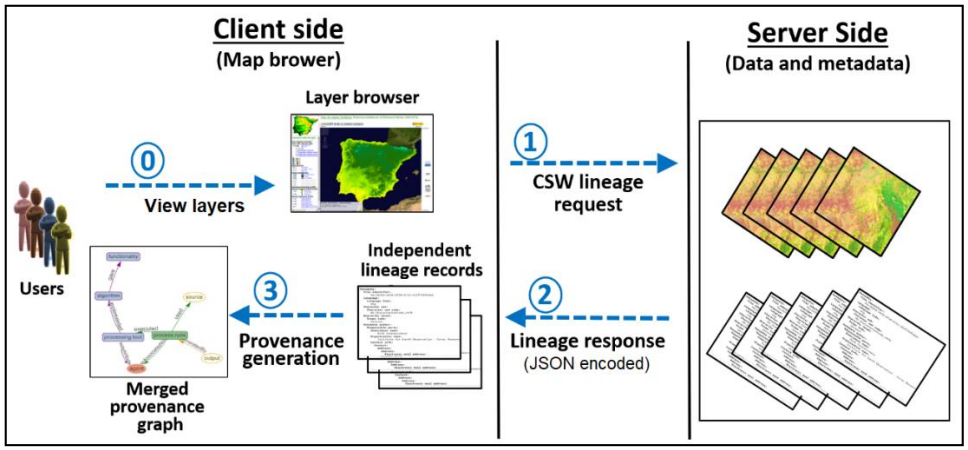

Figure 6. Provenance Web mapping system step by step. A Catalogue Service for the Web (CSW) lineage request (1) to the server allows the independent lineage of each requested dataset to be retrieved (2). Finally, the client generates the provenance graph (3).

4.1. Lineage Server

Lineage is part of the metadata, the natural solution for retrieving lineage in a standard way is to use the OGC Catalogue Service for the Web (CSW). The *GetRecordById* request retrieves the default representation of metadata records with this identifier. However, instead of getting the entire metadata record, we only wanted to retrieve the lineage information. Thus, we came up with a small extension of the CSW protocol that includes the ELEMENTSETNAME key that has "*lineage*" as a value. In addition, to facilitate the reading in the JavaScript client, the OUTPUTFORMAT key and for requesting the lineage in a JSON encoding was also included. The use of a JSON encoding is particularly convenient for a JavaScript client. A JSON file can be converted into a JavaScript data structure with only one sentence of code. There is currently no official JSON encoding for the ISO 19115, so we defined one using the draft rules that were proposed in the OGC Architecture DWG JSON best practice [45]. These extensions were implemented in the MiraMon Map Server. The MiraMon Server is a stand-alone CGI application that runs on the Windows operating systems in combination with a general-purpose web server (e.g., Internet Information Service, Apache, etc.).

Figure 7 shows a CSW *GetRecordById* operation that returns the lineage information in JSON encoding. Figure 8 presents a fragment of the JSON response.

```
www.ogc3.uab.cat/cgi-bin/mcsc/MiraMon.cgi?SERVICE=CSW&REQUEST=GetRecordById&OUTPUTSCHEMA=
http://www.isotc211.org/2005/gmd&ELEMENTSETNAME=lineage&id=MCSCv2Nivell2:EPSG:4326&OUTPUTFORMAT=application/json
```

Figure 7. A CSW GetRecordById request operation.

```
"lineage":{
   "statement": "The present raster is the result of the change of cartographic projection of the CLCM2 level 2 layer
from SR UTM-31N-ED50 to SR UTM-31N-ETRS89.",
   "processes":[
      {
         "processor":[
         {
            "role":"processor",
            "party":{
               "organisation": {"name":"CREAF"}
            }
         }
      ],
      "purpose":"Metadata edition of the CLCM2 level 2",
      "timeDate":"2020-06-19T19:29:38.069+02:00",
      "executable":{
         "reference": "c:/miramon/GeMM.exe",
         "compilationDate":"2020-06-19T19:29:38.069+02:00",
         "functionality": "Metadata management and edition "
      },
      "parameters": [
         {
            "id":"Param1",
            "name":"MCSCv2Nivell2_1",
            "direction":"in",
            "valueType":"source",
            …
```

Figure 8. Fragment of the CSW GetRecordById operation response in JSON.

4.2. Provenance Interface

The MiraMon Map Browser is a visualization, analysis, and download web application that runs in web browsers that were developed by the MiraMon team [46]. The map browser is coded in HTML5 and JavaScript. It is compatible with Open Geospatial Consortium web standard service protocols and APIs to communicate with web services to obtain the minimum subsets of the information that is necessary to create a fast and dynamic user interaction. The map browser can be configured to present an integrated view of several datasets that have something in common (geographic or thematic or both). These datasets might come from a single service or from several services of different institutions.

In the map browser, we wanted to represent lineage of one dataset or to combine lineage from more than one dataset in a single provenance diagram. Thus, a graph representation that was provided by the vis.js library was selected. A graph is defined as a set of nodes that have identifiers and a set of edges that connect nodes. In the vis.js library, nodes and edges are described as two interlinked arrays of JavaScript objects in an encoding that is very different from our JSON encoding of ISO19115, which is based on the concept of objects (e.g., LE_ProcessStep) that have other objects (e.g., LE_Source) as properties, recurrently. A JavaScript piece of code converts the JSON encoding of ISO 19115 into the JSON arrays that are required by vis.js [47]. In this conversion, a *process step* is represented as a blue box with a purple border, a *processing tool* as a dark green box, an *algorithm* as a purple box and a *functionality* as a green box. A *source* is represented as a yellow ellipsis and an *agent* as an orange circle. Edges use the color of their origin and have the PROV relationships as labels (see Section 2.3). Finally, the bright yellow ellipse is reserved for the *result* (see Figure 9, panel B2).

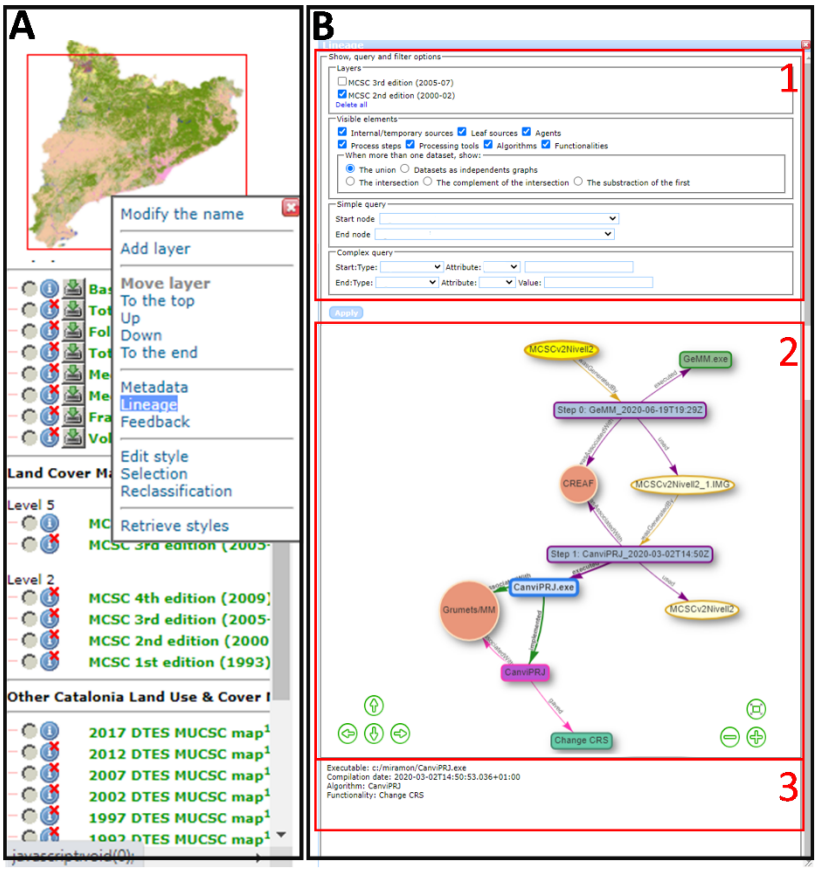

Figure 9. (**A**) Legend panel to click a layer and select Lineage. (**B**) Provenance window: 1—Visibility and query options panel (detailed in Figure 10). 2—Provenance graph panel. 3—Node attributes panel.

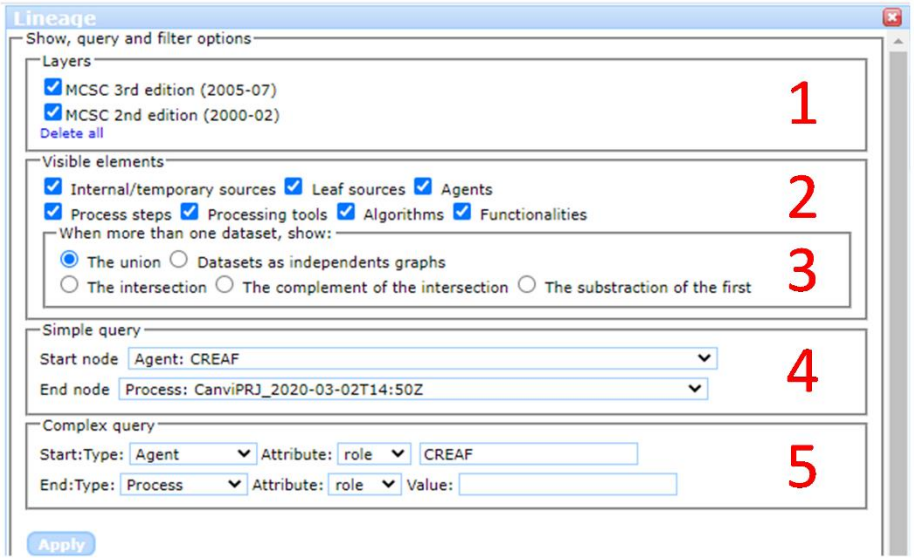

Figure 10. Visibility query and filter options panel: **1**—Layers panel. **2**—Lineage elements filtered/visible panel. **3**—Lineage fusion options. **4**—Simple queries panel. **5**—Complex queries panel.

Users start the process of visualizing integrated provenance by checking for the presence of lineage information of a layer in the legend (see Figure 9). Subsequently, all the *process steps*, *processing tools*, *algorithms*, *functionalities*, *agents*, and *sources* documented in the lineage of that dataset are displayed in a provenance window. The vis.js library calculates

the optimal node positions in the provenance window to avoid overlaps. Users can still move nodes around as required. In addition, JavaScript code handles the *onclick* events and shows more information regarding the node in a text area (see Figure 9, panel B3).

Once users have the provenance graph with a single dataset represented, there are several options to continue exploring provenance (see Figure 9, panel B1):

- Users can unselect some lineage element types to hide them in order to simplify the visualization (see Figure 10, panel 2):
  - *Agents* can be hidden without consequences to simplify visualization.
  - Leaf *sources* (sources that existed independently of the executed process step) can be removed from the view in order to make the process chain simpler.
  - Internal and often temporary *sources* (datasets that were produced during the process chain execution) can be removed from the view in order to enhance the understanding of the process chain.
  - *Process steps* can be removed, and *processing tools* take their place.
  - *Processing tools* can be removed, and they are replaced by *algorithms*.
  - *Algorithms* can be removed, and they are replaced by the *functionality* provided.
  - *Functionalities* can be hidden with no consequences.

In the last four points described, the provenance graph becomes more abstract and less dependent on the details of the software used. When this happens, the represented provenance uses the relationships introduced in Section 2.3.1.1.

- Users can select and incorporate another dataset. The "incoming" lineage elements are accumulated in the provenance window. This combined graph can be represented in different ways (see Figure 10, panel 3):
  - A new independent graph that is presented next to the previous one in the same window.
  - The union of all lineage elements in a provenance graph: the common elements are represented only once, allowing for users to see the full picture of the provenance, including provenance connections between two production processes, such as shared sources, tools, agents, etc.
  - The intersection between the two graphs: only the nodes that connect and are shared by both lineage graphs are presented. These elements are the ones that are most used.
  - The subtraction of the first graph: only elements of the first lineage that are not present on the other lineage are represented. This places the emphasis on what is different in the first layer from the second one.
  - The complement of the intersection: the elements that are not common in the two lineages are represented, placing the emphasis on the elements that are only used once.
- Users can right-click on the box of a *process steps* and request to group it with the previous step or with the next step. This creates a "virtual" process step that is the sequence of the previous two; in the same way as we create batch processes.
- Users can check the lineage statement by clicking with the right button on the resulting dataset (bright yellow ellipsis).

4.3. Provenance Query Tool

Queries regarding the provenance graph resulting from the datasets activated in the layers panel (see Figure 10, panel 1) can be formulated. The selection of lineage elements (see Figure 10, panel 2) described in Section 4.2 also applies to the query result. There are two types of queries:

- The simple queries are facilitated by a simple query interface that offers two lists with all the objects that are present in the graph (classified per type). An algorithm determines whether the start object is connected with the end object and selects them as well as all intermediate nodes that connect them. For instance, we would like to

check the activity of the *agent* "CREAF" regarding a specific "CanviPrj" processing tool (see Figure 10, panel 4).

- The complex query allows us to select two object types and their respective attribute values. This results in several start and end nodes being marked as selected. Not filling in the attribute value will result in selecting all of the nodes with the same attribute type as the start or end points. For instance, we would like to check the activities of the *agent* "CREAF" regarding all of the processing tools (see Figure 10, panel 5), whatever they may be. As in the previous case, the objects that match the query and all the objects that connect them are selected.

Once the provenance queries are solved, the provenance window can present the resulting provenance information in two different forms:

- A graph representing only the elements that were selected by the query. The result is simpler, but some relationships to other objects that are essential in understanding the graph might not be visible.
- A full graph with all elements, but with the selected elements emphasized. This option is more useful for graphs that contain a limited number of elements.

5. Use Case: Catalonia Land Use and Land Cover Map

In this use case, we want to examine, compare, and query the provenance of Catalan land use and cover maps. The Catalonia Land Cover Maps service (http://www.opengis.uab.cat/mcsc/, accessed on 9 January 2021) provides the first (1993), the second (2000), third (2005), and fourth (2009) editions of the Catalonia Land Cover Map (MCSC) (https://www.creaf.uab.cat/mcsc/, accessed on 9 January 2021), and the 1987, 1992, 1997, 2002, 2007, 2012, and 2017 editions of the Land Use and Cover Maps of Catalonia (MUCSC) [48] (see Figure 9).

Even though the purpose of the two products is similar, their process chain generations and what they represent are quite different:

- The MCSCs were made by photointerpretation of aerial photographs and, sometimes, by incorporating elements of other cartography. The base materials for the photointerpretation were a set of orthophotos in natural color from the Institut Cartogràfic i Geològic de Catalunya (ICGC). The legends of the MCSC 2005 and 2009 editions have hierarchical levels of complexity (the simplest is level 1 and the most complex is level 5) [49].
- MUCSCs were generated using automatic classification of satellite imagery and auxiliary cartography. While the 1987, 1992, 1997, and 2002 maps were generated by the ICGC, the 2007, 2012, and 2017 editions were generated by the Geography Department of the Universitat Autònoma de Barcelona (UAB). In addition, the 1987, 1992, 1997, 2002, and 2012 maps were created using Landsat imagery (Landsat 5, Landsat 7, or Landsat 8, depending on the edition), and the 2017 map was based on Sentinel 2 imagery [50,51]. The software used has evolved over the years, with new methodologies and new versions of the same applications. Finally, the maps have been manually edited to fix some unavoidable errors of the automatic classification.

This scenario is a good example for validating the provenance visualization and queries techniques that were developed within the framework of MiraMon Map Browser.

5.1. Provenance Visualization Examples

Some examples of provenance visualization are shown based on the MCSC layers:

- **Example 1** (see Figure 11—Left): a provenance graph shows the agents that are involved in the generation of the MCSC version 1. The visibility, query, and filter options panel only has the layer MCSCv1Nivell2 selected and the agents as visible.
- **Example 2** (see Figure 11—Right): the provenance graph panel shows the processing tools and sources involved in the generation of MCSC version 2. Process steps have been abstracted into used processing tools. The visibility query and filter options

panel only has the layer MCSCv1Nivell2 selected and the internal sources, leaf sources, and processing tools as visible.

- **Example 3** (see Figure 12): the provenance graph panel shows a representation of the combination of the lineage of the MCSC versions 1 and 2. The shared lineage elements are detected and represented only once. The visibility, query, and filter options panel have both layers, MCSCv1Nivell2 and MCSCv2Nivell2 selected, and all of the lineage elements are selected.

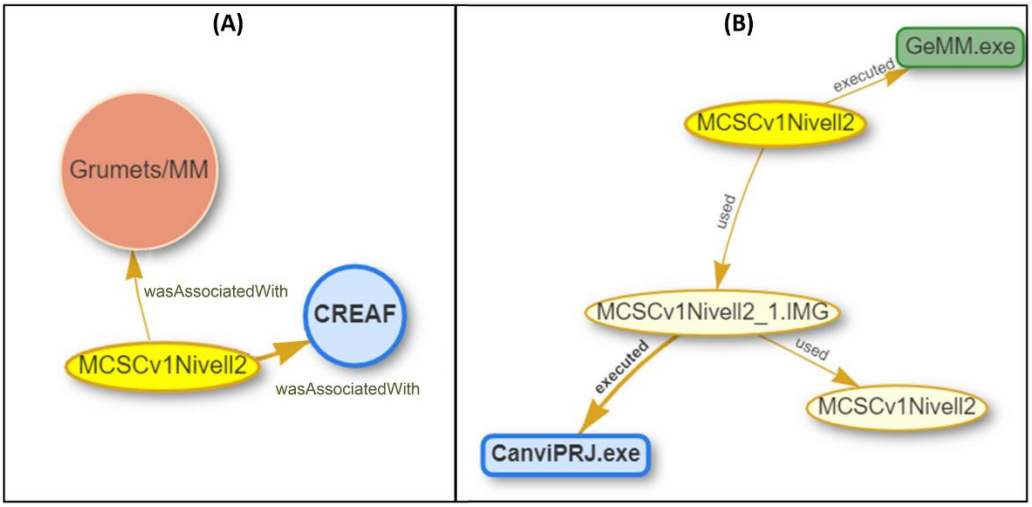

Figure 11. (**A**) agents involved in the generation of the MCSC version 1. (**B**) processing tools and sources involved in the MCSC version 2.

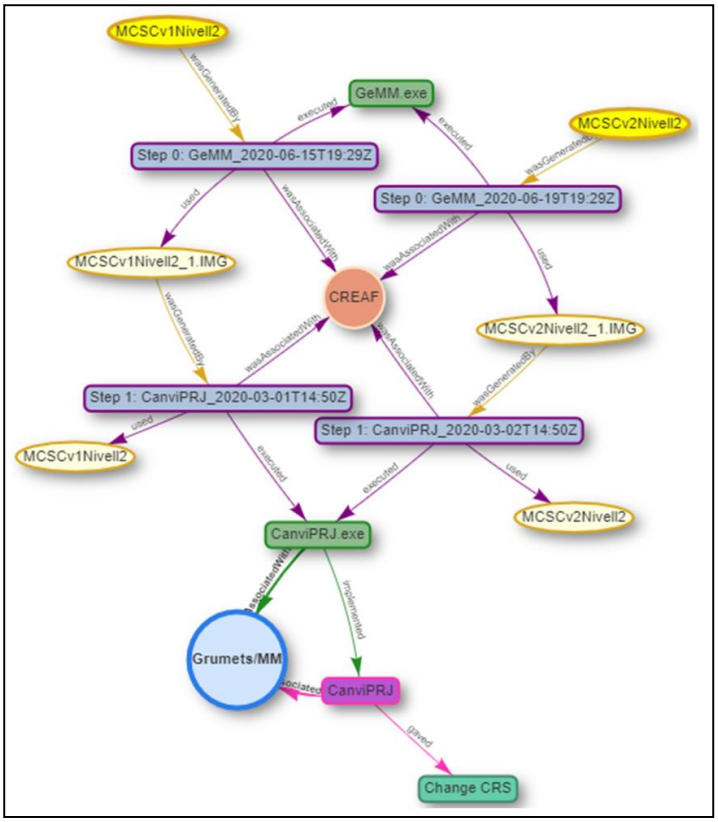

Figure 12. Provenance of the MCSC versions 1 and 2 merged together.

5.2. Provenance Query Examples

Table 4 shows forty-four possible queries that we can apply to the provenance of the land use and land cover map use case. The presented queries are only examples of the potential of querying provenance from a collection of datasets. In the dual entry table, queries that relate two lineage elements are shown, with the exception of the first row, which restricts queries to only one lineage element.

From the forty-four examples, we selected one example that corresponds to Q29 in Table 4 to show the results (see Figure 13):

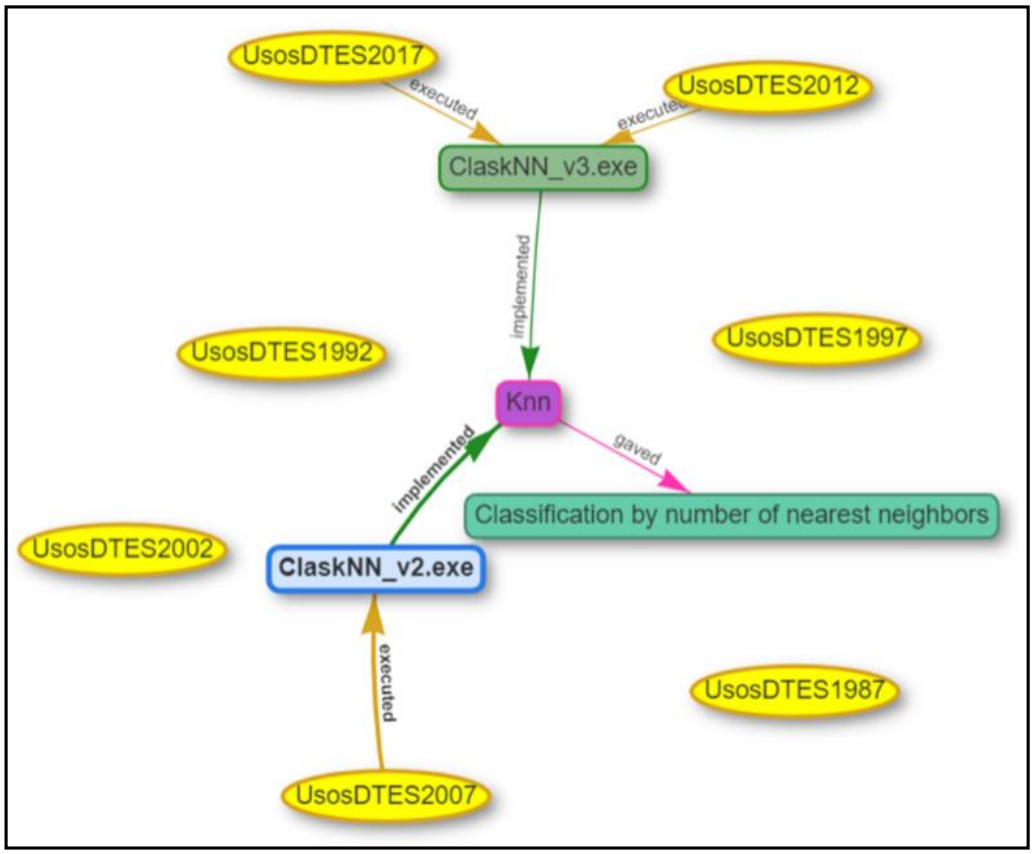

Figure 13. Query example and result: Over the complete MCSC dataset series, which versions have been generated using kNN algorithms? (Q29 in Table 4).

Over the complete MCSC dataset series, which versions have been generated using kNN (Classification by number of nearest neighbors) algorithms? The provenance graph shows the kNN algorithm, including the functionality, related to the different versions of tools that were implemented the kNN algorithm: ClassKnn_v2.exe and ClassKnn_v3.exe. These tools are related to the generated datasets. The isolated datasets (1987, 1992, 1997, and 2002) mean that the kNN algorithm has not been used in their lineage. The layers panel contains all MCSC versions; in the visible elements panel functionalities, algorithms and processing tools are selected, and, finally, a complex query is filled in to obtain only the target elements (those related to the kNN algorithm).

Table 4. Forty-four examples of queries that can be formulated on the Catalonia Land Cover Maps.

	Process Run	**Processing Tool**	**Algorithm**	**Functionality**	**Agent**	**Source**	**Time**	**Output**
(over a complete dataset)	Q1. Did any of the executions not use MiraMon software?	Q2. Was version 3 of Class*Knn* used?	Q3. Was the *Knn* algorithm used?	Q4. Was the *supervised classification* functionality used?	Q5. Which roles did CREAF do?	Q6. Was a dataset called Orto25m used?	Q7. Was something executed before 2013?	Q8. Was any output not owned by CREAF?
Process run (over MUCSC 2017 generation)	Q9. What was executed after ClassKnn v. 5?	Q10. Did process Step 5 use version 3 of Class*Knn?*	Q11. Was Process step 5 a *Knn classification*?	Q12. What is the functionality provided by Process step 5?	Q13. Which process steps were executed by Grumets?	Q14. Did Process step 5 use a DEM of 2m?	Q15. What was the last process step?	Q16. Which outputs were generated with Step 5 execution?
Processing tool (over MUCSC series generation)		Q17. Which tool was most often used right after the *ClassKnn* tool*?*	Q18. What algorithm implemented version 3 of *ClassKnn*?	Q19. Do the tools *ClassKnn* and *IsoMM* provide equivalent functionalities?	Q20. Were the tools used developed by trusted software vendors?	Q21. Did version 3 of *ClasskNN tool* need PIA (Pseudoinvariant areas) sources?	Q22. Was version 3 of *ClassKnn tool* the last version available?	Q23. Which versions of MUCSC used version 2 of the Class*Knn tool*?
Algorithm (over a complete dataset)			Q24. Which different versions of the *Knn* algorithm were used?	Q25. Did all the *Sup.classification* tools use the same algorithm?	Q26. Was CREAF the author and owner of any of the algorithms used?	Q27. Was the reclassification algorithm suitable for working with categorical data?	Q28. When was the current version of the *Sup.classification* developed?	Q29. Which versions of MUCSC were created using the Knn algorithm?
Functionality (over a complete dataset)				Q30. Were all radiometric corrections made with the same software?	Q31. Which institution performed the radiometric corrections?	Q32. Which of the datasets used were reclassified?	Q33. Were any of the datasets classified before 2015?	Q34. Which Land Cover Maps were photo-interpreted?
Agent (over a complete dataset)					Q35. Who used tools developed by CREAF?	Q36. Which of the sources used in this collection have open access licenses?	Q37. When did CREAF make their first execution in this collection?	Q38. Did CREAF create a MUCSC 2007dataset?
Source (over 2012 and 2017 MUCSC series generation)						Q39. Which sources were used in all processing chains?	Q40. Did the orthos use the same temporal interval as the MUCSC outputs?	Q41 Which orthos were used in the generation of MUCSC of 2012 and 2017?
Time (over 2017 MUCSC generation)							Q42. How long did the complete processing chain take to be completed?	Q43. When was the MUCSC map finalized?
Output (over 20017 MUCSC generation)								Q44. Was this LULC a revision of another MUCSC map?

6. Discussion

On the conceptual side, the levels of abstraction introduced for processes make it possible to transform a precise lineage graph into a more abstract workflow diagram or even into a list of functionalities that inform a GIS operator or student how to reproduce a dataset by chaining GIS tools. However, the main obstacle to apply this in a generic way is the lack of a single classification of the main GIS and RS functionalities and a well accepted ontology of semantic descriptions. The main GIS and RS software products could map their processing tools to this common set of functionalities if consensus could be achieved in the future.

On the technical side, in the solution that is presented here, the lineage part of the ISO 19115 metadata documents is sent to a JavaScript map browser, where (using the extended PROV ontologies) are merged in a provenance graph. The queries that are discussed in this paper are resolved directly in the client side, with no service intervention and the results presented to the user as subgraphs. There are other ways to address the same technical problem. Another alternative could be to persuade data producers to adopt the extended PROV ontology that was proposed in this paper and expose their lineage metadata in RDF representations connected to the semantic web. For example, the use of RDF was suggested by [52] for provenance metadata in the field of bioinformatics. By doing that, we could use the RDF based technologies for exploding the semantic web, such as a triple store database and a SPARQL query language, to solve the queries and even associate provenance to the SPARWL query responses themselves [53]. However, this will require the collaboration of the producers that will need to embrace the semantic web technologies. Currently, most of the producers are following the Spatial Data Infrastructures best practices and concentrating their efforts in implementing ISO TC211 standards family and Open Geospatial Consortium standard services, and it will be difficult for them to invest resources in other approaches in the near future.

Dependencies and relationships between elements are represented in a more natural way in a network graph than in a hierarchical tree form. However, and contrary to what we expect, a graph can be more difficult to follow than a tree representation, particularly in long process chains. The capability of present collections of datasets in a single view rapidly increases the complexity of the relations resulting in 2D representation that are too cluttered. In a 3D visualization, the user can navigate within the 3D scene to find the better perspective that reduces the number of line intersections helping to analyze data [54]. Although a provenance graph with full detail is more informative, filtering our unnecessary nodes in the graphs is a complementary strategy to simplify the diagram and make it understandable. However, where the graph fully deploys its potential is when queries are applied to it. Depending on which query is formulated, different benefits emerge (Some queries could provide more than one benefit. Only the most relevant benefit is presented in this classification):

- Information and transparency. These provide a better understanding and compare methodologies in Table 4 Q1, Q3, Q4, Q10, Q11, Q12, Q15, Q16, Q18, Q21, Q24 and Q25, Q27, Q28. Q32, Q33, Q39, Q42, and Q43.
- Trust and authority: agents and their responsibilities can be inferred based on the sources and tools used in Table 4 Q2, Q5, Q13, Q20, Q31, Q37, and Q38.
- Data quality can be deduced from the quality of the sources and precision of the processing tools used in Table 4 Q6, Q7, Q10, Q14, Q19, Q21 Q22, Q25, Q30, Q34, Q40, and Q44.
- Documentation and reproducibility can be achieved if all the necessary details about the actual dataset, metadata, or tools are present, such as in Table 4 Q9, Q16, Q17, Q23, Q24 Q29, and Q41.
- License and accessibility: information about the needed resources that were accessed, and licenses needed are present in Table 4 Q8, Q26, Q35, and Q36.

7. Future Work

The levels of abstraction that were introduced for processes could also be combined with the abstraction of the sources into generic ones by indicating the schema of the product or, in the extreme case, by only providing the topic category that they belong to. This information can be extracted from the metadata of the sources by looking at ISO 19115 metadata fields not directly related to lineage.

This paper presents a sketch of a provenance window that has been co-designed in collaboration with the MiraMon Map Browser implementers. The development has only completed the first loops of an agile methodology that has been prototyped and tested with the data presented in this paper. Further co-design sessions with more users may reveal the need for extra functionalities or the need to change some aspects of the user interface of the provenance query and filter panels.

Extending the queries presented in this paper to larger collections is a challenge for the visualization; however, when combined with the right queries, it could be applied to an organization to determine the most useful datasets and tools. This will help organizations that are facing preservation challenges: for the first time, they have to decide which datasets should be preserved from the organizational digital legacy that is too big, and what information can be forgotten and erased from the archives. A comprehensive provenance study can help to determine this.

8. Conclusions

The geospatial lineage is a necessary component in the metadata of spatial information distributed over the web. However, it is recognized that these benefits cannot be materialized if there are no proper tools to help users visualize and interpret the lineage. This paper makes two main contributions to overcome this situation.

On one hand, the introduction of four levels of abstraction of the process step description (process run, processing tool, algorithm, and functionality) has proven to be a valuable way to better describe lineage. The inclusion of functionalities and algorithm descriptions as a part of lineage provides high-level information and representation independent from the software used and the moment in time the step was executed. This solution has provided certain benefits: it allows for datasets originated with different workflows to be interconnected and makes it possible to compare processing chains at the methodology level. Therefore, it was demonstrated that the lineage model of the ISO19115 family (to express the object types of the geospatial objects that are involved in the production processes) can be combined with W3C PROV (to convey the relationship naming conventions). In this paper, a symbolization as a provenance graph instead of hierarchical tree was explored as a more flexible alternative.

The web tool presented in this paper helps users to interpret lineage by making connections among processes and making sources more visible, as well as making it possible to filter and query lineage elements. The tool facilitates the formulation of queries to interrogate the origins of geospatial data of a collection of datasets. The tool generates on-demand visualizations that provide answers to queries that emphasize the benefits of lineage data: information and transparency; trust and authority; data quality; documentation and reproducibility; and, license and accessibility.

The possibility to formulate queries regarding a collection of datasets gives added value to provenance and provides scientists and technicians with the opportunity to inspect dataset interrelations and processing chain performance. Provenance graphs could help in the difficult task of determining the most useful datasets and eventually deciding what information should be preserved for future generations.

Author Contributions: Conceptualization, Guillem Closa and Joan Masó; Research, Guillem Closa; Methodology, Guillem Closa and Joan Masó; Software, Joan Masó and Núria Julià; Writing—original draft, Guillem Closa; Writing—review & editing, Joan Masó, Núria Julià and Xavier Pons. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Catalan Government [SGR2017 1690]. This work was carried out under the ECOPOTENTIAL, e-shape, and ERA-PLANET projects. These projects have received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No. 641762-2, 776740 and 689443 respectively. This work has also been supported by the Spanish MCIU Ministry through the NEWFORLAND project (RTI2018-099397-B-C21/22 (MCIU/AEI/ERDF, EU)). Xavier Pons is the recipient of an ICREA Academia Excellence in Research Grant (2016–2020).

Acknowledgments: We would like to thank the land cover time series team of CREAF and UAB for their valuable help.

Conflicts of Interest: There are no conflict of interest to declare.

References

1. Baker, M. 1,500 scientists lift the lid on reproducibility. *Nat. Cell Biol.* **2016**, *533*, 452–454. [CrossRef]
2. Lemos, M.C.; Kirchhoff, C.J.; Ramprasad, V. Narrowing the climate information usability gap. *Nat. Clim. Chang.* **2012**, *2*, 789–794. [CrossRef]
3. Spiekermann, R.; Jolly, B.; Herzig, A.; Burleigh, T.; Medyckyj-Scott, D. Implementations of fine-grained automated data provenance to support transparent environmental modelling. *Environ. Model. Softw.* **2019**, *118*, 134–145. [CrossRef]
4. Brinckman, A.; Chard, K.; Gaffney, N.; Hategan, M.; Jones, M.B.; Kowalik, K.; Stodden, V. Computing environments for reproducibility: Capturing the "Whole Tale". *Future Gener. Comput. Syst.* **2019**, *94*, 854–867. [CrossRef]
5. Lewis, A.; Lacey, J.; Mecklenburg, S.; Ross, J.; Siqueira, A.; Killough, B.; Szantoi, Z.; Tadono, T.; Rosenavist, A.; Goryl, P.; et al. CEOS Analysis Ready Data for Land (CARD4L) Overview. In Proceedings of the IGARSS 2018—2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 7407–7410.
6. Giuliani, G.; Chatenoux, B.; Bono, A.D.; Rodila, D.; Richard, J.P.; Allenbach, K.; Peduzzi, P. Building an Earth Observations Data Cube: Lessons learned from the Swiss Data Cube (SDC) on generating Analysis Ready Data (ARD). *Big Earth Data* **2017**, *1*, 100–117. [CrossRef]
7. Fisher, P.F. Algorithm and Implementation Uncertainty: Any Advances? *Int. J. Geogr. Inf. Sci. Syst.* **2006**, 225–228.
8. Lutz, M.; Riedemann, C.; Probst, F. A Classification Framework for Approaches to Achieving Semantic Interoperability between GI Web Services. In Proceedings of the International Conference on Spatial Information Theory, Kartause Ittingen, Switzerland, 24–28 September 2008; Springer: Berlin/Heidelberg, Germany, 2003; pp. 186–203.
9. CEOS Interoperability Terminology, Version 1.0. CEOS—WGISS Interoperability and Use Interest Group. 2020. Available online: https://ceos.org/document_management/Meetings/Plenary/34/Documents/CEOS_Interoperability_Terminology_Report.pdf (accessed on 20 October 2020).
10. Jiang, L.; Yue, P.; Kuhn, W.; Zhang, C.; Yu, C.; Guo, X. Advancing interoperability of geospatial data provenance on the web: Gap analysis and strategies. *Comput. Geosci.* **2018**, *117*, 21–31. [CrossRef]
11. Yue, P.; Wei, Y.; Di, L.; He, L.; Gong, J.; Zhang, L. Sharing geospatial provenance in a service-oriented environment. *Comput. Environ. Urban Syst.* **2011**, *35*, 333–343. [CrossRef]
12. Zhang, M.; Yue, P.; Wu, Z.; Ziebelin, D.; Wu, H.; Zhang, C. Model provenance tracking and inference for integrated environmental modelling. *Environ. Model. Softw.* **2017**, *96*, 95–105. [CrossRef]
13. He, L.; Yue, P.; Di, L.; Zhang, M.; Hu, L. Adding Geospatial Data Provenance into SDI—A Service-Oriented Approach. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2015**, *8*, 926–936. [CrossRef]
14. Growth, P.; Moreau, L. PROV-Overview: An Overview of the PROV Family of Documents. *W3C*. 2013. Available online: https://eprints.soton.ac.uk/356854/ (accessed on 20 October 2020).
15. Lanter, D.P. Design of a Lineage-Based Meta-Data Base for GIS. *Cartogr. Geogr. Inf. Syst.* **1991**, *18*, 255–261. [CrossRef]
16. Spatial Data Transfer Standard (SDTS); American National Standards Institute's (ANSI). ANSI/NCITS320.1998. 1998. Available online: https://www.fgdc.gov/standards/projects/SDTS/sdts_cadd/finalcadd.pdf (accessed on 15 October 2020).
17. ISO. *Geographic Information—Metadata*; ISO 19115:2003; ISO: Geneva, Switzerland, May 2003; 140p.
18. ISO. *Geographic Information—Metadata—Part 1: Fundamentals*; ISO 19115-1:2014; ISO: Geneva, Switzerland, April 2014; 167p.
19. ISO. *Geographic Information—Metadata—Part 2: Extensions for Acquisition and Processing*; ISO 19115-2: 2019; ISO: Geneva, Switzerland, January 2019; 57p.
20. Di, L.; Shao, Y.; Kang, L. Implementation of Geospatial Data Provenance in a Web Service Workflow Environment with ISO 19115 and ISO 19115-2 Lineage Model. *IEEE Trans. Geosci. Remote Sens.* **2013**, *51*, 5082–5089. [CrossRef]
21. Di, L.; Yue, P.; Ramapriyan, H.K.; King, R.L. Geoscience Data Provenance: An Overview. *IEEE Trans. Geosci. Remote Sens.* **2013**, *51*, 5065–5072. [CrossRef]
22. Ivánová, I.; Armstrong, K.; McMeekin, D. Provenance in the next-generation spatial knowledge infrastructure. In Proceedings of the 22nd International Congress on Modelling and simulation (MODSIM 2017), Hobart, Tasmania, Australia, 3–8 December 2017; pp. 410–416.
23. Lopez-Pellicer, F.J.; Barrera, J. D16. 1 Call 2: Linked map VGI provenance schema. In *Linked Map Subproject of Planet Data. Seventh Framework Programe*; European Commission: Brussels, Belgium, 2014.

24. Closa, G.; Masó, J.; Proß, B.; Pons, X. W3C PROV to describe provenance at the dataset, feature and attribute levels in a distributed environment. *Comput. Environ. Urban Syst.* **2017**, *64*, 103–117. [CrossRef]
25. Closa, G.; Masó, J.; Zabala, A.; Pesquer, L.; Pons, X. A provenance metadata model integrating ISO geospatial lineage and the OGC WPS: Conceptual model and implementation. *Trans. GIS* **2019**, *23*, 1102–1124. [CrossRef]
26. Salton, G.; Allan, J.; Buckley, C.; Singhal, A. Automatic analysis, theme generation, and summarization of machine-readable texts. *Science* **1994**, *264*, 1421–1426. [CrossRef]
27. Konkol, M.; Kray, C. In-depth examination of spatiotemporal figures in open reproducible research. *Cartogr. Geogr. Inf. Sci.* **2019**, *46*, 412–427. [CrossRef]
28. Yazici, I.M.; Karabulut, E.; Aktas, M.S. A Data Provenance Visualization Approach. In Proceedings of the 2018 14th International Conference on Semantics, Knowledge and Grids (SKG), Guangzhou, China, 12–14 September 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 84–91.
29. Cohen-Boulakia, S.; Belhajjame, K.; Collin, O.; Chopard, J.; Froidevaux, C.; Gaignard, A.; Hinsen, K.; Larmande, P.; Le Bras, Y.; Lemoine, F.; et al. Scientific workflows for computational reproducibility in the life sciences: Status, challenges and opportunities. *Future Gener. Comput. Syst.* **2017**, *75*, 284–298. [CrossRef]
30. Yue, P.; Zhang, M.; Guo, X.; Tan, Z. Granularity of geospatial data provenance. In Proceedings of the 2014 IEEE Geoscience and Remote Sensing Symposium, Quebec, QC, Canada, 13–18 July 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 4492–4495.
31. Albrecht, J. Universal Analytical GIS Operations: A Task-Oriented Systematization of Data Structure-Independent GIS Functionality. *Geogr. Inf. Res. Transatl. Perspect.* **1998**, 577–591. Available online: https://www.researchgate.net/publication/228530780_Universal_analytical_GIS_operations_a_task-oriented_systematization_of_data_structure-independent_GIS_functionality (accessed on 20 October 2020).
32. Sun, Z.; Yue, P.; Di, L. GeoPWTManager: A task-oriented web geoprocessing system. *Comput. Geosci.* **2012**, *47*, 34–45. [CrossRef]
33. Goodchild, M.F. Geographic information systems. *Prog. Hum. Geogr.* **1991**, *15*, 194–200. [CrossRef]
34. Kuhn, W.; Ballatore, A. Designing a Language for Spatial Computing. In Proceedings of the Agile 2015, Washington, DC, USA, 3–7 August 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 309–326.
35. Yue, C.; Baumann, P.; Bugbee, P.; Jiang, L. Towards intelligent giservices. *Earth Sci. Inf.* **2015**, *8*, 463–481. [CrossRef]
36. ESRI. *ArcGIS Desktop: Release 10*; Environmental Systems Research Institute: Redlands, CA, USA, 2020.
37. Pons, X. *MiraMon: Geographical Information System and Remote Sensing Software*; Centre de Recerca Ecològica i Aplicacions Forestals: Barcelona, Spain, 2020.
38. GRASS Development Team. Geographic Resources Analysis Support System (GRASS) Software, Version 7.2. Open Source Ge-ospatial Foundation. 2017. Available online: http://grass.osgeo.org (accessed on 20 October 2020).
39. SNAP—ESA. Sentinel Application Platform v8.0.0. 2020. Available online: http://step.esa.int (accessed on 20 October 2020).
40. Lopez-Pellicer, F.J.; Lacasta, J.; Espejo, B.A.; Barrera, J.; Agudo, J.M. The standards bodies soup recipe: An experience of interoperability among ISO-OGC-W3C-IETF standards. In Proceedings of the Inspire-Geospatial World Forum, Lisbon, Portugal, 25–29 May 2015.
41. Masó, J.; Pons, X.; Zabala, A. Building the World Wide Hypermap (WWH) with a RESTful architecture. *Int. J. Digit. Earth* **2012**, *7*, 175–193. [CrossRef]
42. Erwig, M.; Schneider, M. Developments in spatio-temporal query languages. In Proceedings of the Tenth International Workshop on Database and Expert Systems Applications DEXA 99, Florence, Italy, 3 September 1999; IEEE: Piscataway, NJ, USA, 1999; pp. 441–449.
43. Koubarakis, M.; Karpathiotakis, M.; Kyzirakos, K.; Nikolaou, C.; Sioutis, M. Data Models and Query Languages for Linked Geospatial Data. In *Reasoning Web International Summer School*; Springer: Berlin/Heidelber, Germany, 2012; pp. 290–328.
44. Amann, B.; Scholl, M. Gram: A graph data model and query languages. In Proceedings of the ACM conference on Hypertext, Seattle, WA, USA, 14–18 November 1993; pp. 201–211.
45. Maso, J. OGC JSON Best Practice Draft. 2018. Available online: https://github.com/opengeospatial/architecture-dwg/tree/master/json-best-practice (accessed on 15 October 2020).
46. Masó, J.; Zabala, A.; Pons, X. Protected Areas from Space Map Browser with Fast Visualization and Analytical Operations on the Fly. Characterizing Statistical Uncertainties and Balancing Them with Visual Perception. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 300. [CrossRef]
47. Vis.js. 2020. Available online: https://visjs.org/ (accessed on 25 October 2020).
48. Generalitat de Catalunya; Departament de Territori i Sostenibilitat. *Land Use and Cover Open Data Page*. 2020. Available online: https://territori.gencat.cat/ca/01_departament/12_cartografia_i_toponimia/bases_cartografiques/medi_ambient_i_sostenibilitat/usos-del-sol/ (accessed on 20 October 2020).
49. Ibàñez, J.J.; Burriel, J.A. Mapa de cubiertas del suelo de Cataluña: Características de la tercera edición y relación con SIOSE. In *Tecnologías de la Información Geográfica: La Información Geográfica al Servicio de los Ciudadanos*; Ojeda, J., Pita, M.F., Vallejo, I., Eds.; Secretariado de Publicaciones de la Universidad de Sevilla: Sevilla, Spain, 2010; pp. 179–198, ISBN 978-84-472-1294-1.
50. González-Guerrero, Ò.; Pons, X.; Bassols-Morey, R.; Camps, F.X. Dinàmica de les Superfícies de Conreu a Catalunya Mitjançant Teledetecció en el període 1987–2012. *Quaderns Agraris* **2019**, 59–91.
51. González-Guerrero, Ò.; Pons, X. The 2017 Land Use/Land Cover Map of Catalonia based on Sentinel-2 images and auxiliary data. *Revista de Teledetección* **2020**, *55*, 81–92. [CrossRef]

52. Zhao, J.; Goble, C.; Stevens, R.; Turi, D. Mining Taverna's semantic web of provenance. *Concurr. Comput. Pract. Exp.* **2008**, *20*, 463–472. [CrossRef]
53. Theoharis, Y.; Fundulaki, I.; Karvounarakis, G.; Christophides, V. On Provenance of Queries on Semantic Web Data. *IEEE Internet Comput.* **2010**, *15*, 31–39. [CrossRef]
54. Viola, F.; Roffia, L.; Antoniazzi, F.; D'Elia, A.; Aguzzi, C.; Cinotti, T.S. Interactive 3D Exploration of RDF Graphs through Semantic Planes. *Future Internet* **2018**, *10*, 81. [CrossRef]