Undefined 1 (2015) 1–5
IOS Press

Towards high quality data catalogues: addressing exploitability.

Enrico Daga, Alessandro Adamou, Mathieu d'Aquin, Enrico Motta
Knowledge Media Institute, The Open University
Walton Hall, Milton Keynes, United Kingdom
E-mail: {enrico.daga,alessandro.adamou,mathieu.daquin,enrico.motta}@open.ac.uk

Abstract.

Data quality is broadly defined as "fitness for use", and the increasing number of systems generally categorised as "Data Catalogues" are expected to support such a notion through enabling data discoverability. As such, one aspect that strongly relates to the quality and completeness of a Data Catalogue is the one we refer to as exploitability: The compatibility between the policies of the provided datasets and the task at hand. However, the current practice in Data Hubs and Data Stores is for their Data Catalogues to simply provide a link to the text of the licence associated to the original data, in its original data source. This is insufficient to effectively support exploitability, since it requires the data consumer to trace back the processing that might have been applied to the data, to manually assess how much it might have affected the policies described in the licences, and to finally check that these policies match the intended use. In this article we argue that a high quality data catalogue can better address exploitability by also considering the way policies propagate across the data flows applied in the system. We propose a methodology to deploy an end-to-end solution centred on a Data Catalogue to support the machine-processable representation of data policies and of the data flows in the system, to enable the propagation and validation of these data policies so to deliver them as exploitability information alongside the data itself.

Keywords: Data Catalogue, Provenance, Policies, Smart Cities, Semantic Web

1. Introduction

The amount of data available online is rapidly increasing, together with the scenarios, use cases and applications that rely on such shared data. Consequently, we also see an increase in the number of online platforms dedicated to registering, cataloguing and delivering large numbers of datasets, in specific domains or generally. These include Data Catalogues such as the ones from governmental organisations (e.g. `data.gov.uk`), general data repositories such as `datahub.io` and `OpenDataCommunities.org`, domain-/area-specific 'data stores' such as the ones dedicated to cities[1] or dataset catalogues dedicated to specific domains[2].

Many of these systems and of the associated processes are simple: They provide a way to register and describe existing datasets. It is clear however that, as the number and diversity of the datasets they need to handle is growing, there is a need for these systems to play a further role in fully supporting the delivery and reuse of datasets. In other words, the role of these *Data Catalogues* should no longer be restricted to provid-

[1] see for example `http://data.london.gov.uk/` for the London Data Store or `http://mksmart.org/data/` for the Milton Keynes Data Hub

[2] see for example `http://data.linkededucation.org/linkedup/catalog/` for the LinkedUp catalogue of datasets for education

ing a list of existing datasets, but should also include supporting the consumer of these datasets in exploiting them by appropriately curating this list to provide complementary information especially related to usage restrictions on the data. Therefore, maximising the *exploitability* of data is an issue of quality of the catalogue itself, since, as already described in [11], it implies dealing with datasets which are not only heterogeneous in content and coming from different sources, but also handling the large diversity of the datasets with respect to the rights and policies to which they relate.

For example, as motivation, we place our work within the context of the MK Data Hub (see [12]). The MK Data Hub is the data sharing platform of the MK:Smart project[3] that explores the use of data analytics to support Smarter Cities, taking the city of Milton Keynes (England) as a testbed. The data catalogue of the MK Data Hub contains information about a large number of datasets from many different sources, including open data from the local council and the UK government, as well as data from private sector organisations (e.g. utility companies) and individuals.

The main purpose of the MK Data Hub is to support applications that combine these different datasets in innovative scenarios. It therefore includes data access mechanisms (APIs) that provide an integrated view over the data. However, in order to enable the reuse of such data, not only technical integration mechanisms are required. Indeed, since the data as a result of the MK Data Hub APIs might be combined from diverse datasets, different parts of the data might have different exploitability conditions or requirements, propagated from the licences and policies associated with the original datasets. A data consumer (and application developer) might for example need to filter data for use in a commercial application, discarding any data from sources that explicitly, in the original data licence, specified that such use of the data was prohibited. Similarly, data consumers might need to check which original sources of the data need to be acknowledged because of an attribution requirement, and even whether the form of exposure or re-distribution they employ is allowed according to the policies attached to each individual piece of data they might obtain from the Data Hub.

These are of course only the most common examples of the kind of policies a data consumer might have to check when using data from multiple sources, through a Data Catalogue such as the one of the MK Data Hub. The issue of exploitability is therefore one that directly relates to providing the right level of information regarding the rights and policies that apply to the data being delivered by the Data Hub [12], while these data are integrated, after some processing, from a number of diverse and uncoordinated sources.

In this article, we propose a methodology through which the administrators of Data Catalogues can better support exploitability through tracing and propagating machine processable information about the rights and policies that apply to the included datasets alongside the traces of the applyed operations that might affect them, through the data on-boarding, acquisition, processing and delivery phases. The goal is the enable the propagation of such exploitability-supporting information up to the point where data can be delivered including precise information regarding the policies on requirements, prohibition and permissions that apply to the various parts of the delivered data. Using the MK Data Hub as a case study, we show how this methodology can be implemented in the design of a Data Catalogue through existing semantic technologies, and how it can be shown to address the kind of use cases described above. We also illustrate the applicability of this methodology in the context of the MK Data Hub, through discussing the assumptions on which the methodology relies and how they are validated in this context.

2. Related work

Though it may appear to be a restrictively defined data quality measure, there is no agreed-upon definition of *data exploitability*, mainly because of the different research problems that the qualitative assessments of leveraging data deal with. Most research concentrates on interoperability issues that are directly bound to either the shape of the data themselves, or the operational characteristics of the mechanisms to expose or exchange them. In such cases, exploitability is conceived as a notion akin to discoverability, security or semantic alignment [2,7,17,19,26]. The "data exploitability" wording itself is almost exclusively found in cyber-security parlance.

Given the fuzziness of this notion, we shall here concentrate on data exploitability as *the ability to determine which policies a unit of data is subject to, and their compatibility with the intended usage by a con-*

[3] see `http://mksmart.org`

sumer. This formulation will be reiterated later in the paper.

Data cataloguing platforms. We record very limited support for policy assessment in existing data cataloguing approaches. *CKAN*, one of the best-known data cataloguing platforms, adopts a package manager paradigm to implement dataset management[4]. A *package*, i.e. the basic unit whereupon policies are set in CKAN, is the dataset itself, on whose granularity (e.g. whether it is a single RDF graph in a multi-graph collection) the platform remains agnostic. CKAN also does not control the life-cycle of dataset contents. Policies are merely license attributes attached to datasets, and no propagation of them towards enclosing entities, such as *organisations*, is implied. A similar argument can be made for *Dataverse*[5], which adopts a granularity level and policy management system similar to those of CKAN. We were unable to directly assess the policy propagation and enforcement features, assuming any, of the proprietary and commercial *Socrata* data platform. [6] However, a survey of existing Socrata-based open data catalogues[7] accessed through the Socrata API has brought to the surface metadata about owner descriptions and roles, permissions - mostly related to the management platform - and grant inheritance policies, all using an in-house (presumably controlled) vocabulary. Custom metadata are also used for the specification of licenses, though their instances are for the most part in human-readable form.[8] The Socrata data API encompasses the consumption of dataset contents as well, effectively making Socrata an enabling platform for data hubs.

License expressions in data catalogues. Cataloguing platforms, as does CKAN since it introduced Linked Data support, rely upon legacy mechanisms for representing dataset policies and publishing them through their APIs. Fundamental to each of them are standard Dublin Core properties such as `dc:license` and `dc:rights`, which in general make no assumption as to whether their values should be machine-readable. The DC subschema for rights and licenses is incorporated in the *DCAT* standard of the W3C for the representation of the catalogue meta-level [14]. DCAT introduces a further level of concretisation via the `dcat:Distribution` class, which accommodates bespoke rights statements. CKAN combines DCAT by typically using the URIs of license descriptions (whose content is, more often that not, human-readable) as values for the DC properties of datasets and distributions. The optional `License` relation of the *HyperCat* specification follows a similar notion, however it enforces the use of URIs for values and contemplates machine-readable content as a possible form to which they dereference[9]. Even in eGovernment, where policy transparency is of the utmost importance, a fairly recent study assessed a degree of heterogeneity when it comes to expressing licenses in government data catalogues [22], though such a survey could be expected to deliver slightly more encouraging results if carried out today, if anything because of the standardisation efforts that have since been promoted.

Policy models. The heterogeneity in license descriptions raises the issue of modelling the license descriptions themselves in a machine-readable way. Since the early investigation carried out on using RDF to police resource access [6], the landscape of license models has witnessed the contributions of several actors in digital rights. The Creative Commons consortium itself publishes guidelines for describing permissions, jurisdictions and requirements on works in general.[10] Specifically for data, the Open Data Institute has proposed the *ODRS* vocabulary,[11] which addresses license compatibility and introduced the separation between data and content in the application of licenses. The *ODRL Policy Language* core model effectively made the leap from licenses to policies, by introducing the concepts of policy inheritance and profile, which instantiates common rights descriptions.[12] Coupled with these efforts are online repositories of licenses expressed in RDF. Among these we cite *LicenseDB*,[13] which mostly uses an in-house vocabulary in combination with DC and Creative Commons, and the Linked

[4]Comprehensive Kerbal Archive Network, `http://ckan.org`
[5]Dataverse, `http://dataverse.org`
[6]Socrata, `http://www.socrata.com`
[7]Open Data Monitor, `http://www.opendatamonitor.eu`
[8]Example at the time of writing: `https://opendata.camden.gov.uk/api/views/6ikd-ep2e.json`
[9]HyperCat specification, `http://www.hypercat.io/standard.html`
[10]Creative Commons Rights Expression Language, `https://creativecommons.org/ns`
[11]Open Data Rights Statement Vocabulary, `http://schema.theodi.org/odrs`
[12]W3C ODRL community, `https://www.w3.org/community/odrl/`
[13]LicenseDB, `http://licensedb.org`

Data license repository of the Universidad Politécnica de Madrid,[14] which uses ODRL.

Policy reasoning. In the remainder of this paper, we shall assume that the policies used to assess exploitability are formulated with the expressivity of ODRL, in that they describe permissions, prohibitions or duties to perform a given set of actions. Under this assumption, such an assessment is reduced to a problem of policy compatibility. This problem has been extensively studied in the literature [15,16,25] and tools that can perform such assessment do exist [21]. Specific forms of policy compatibility assessment are also found in fields whose primary focus is tasks rather than data, as in workflow modelling for task delegation [8]. Our previous work also addressed a form of policy reasoning, namely *policy propagation*. Policy Propagation Rules (PPR) are defined as Horn clauses on top of ODRL. We refer to this study as the reference method to manage a database of PPRs, in which the evolution of the requirements is tackled with an iterative process targeted to compress the rule base and refine the ontological description of the actions involved [10].

Provenance The exploitability of catalogued data is, to an extent, bound to the way their scheme for consumption is designed, and by extension, what conventions and APIs should be adopted and exposed. In an ecosystem like the one presented here, where datasets that are heterogeneous but share a semantic dimension such as the domain of interest are pooled together, data integration is a natural way of making a catalogue readily available for consumption. However, when data integration comes into play, the line that logically divides one dataset from another is blurred by the mappings, dependencies and integration rules that define and justify the introduction of a data integration system. So too does the traceability of integrated data become harder to determine, with consequent issues with their trust, usage rights, and more generally exploitability. As a determining element of these metrics, provenance has attracted attention from the data integration research community.

Provenance itself is an overloaded term, whose many aspects Ram and Liu tried to summarise in their W7 ontological model, where they also referred to it as *lineage* [27]. Even in this light, we note that the majority of work on data provenance addresses its conception as a description of their origin. However, efforts on managing provenance orthogonally have been recorded on the following fronts: (1) storage and querying, where approaches such as tSPARQL [18] aimed and embedding provenance data alongside actual data; and (2) representation, where vocabularies such as the VoIDp extension of VoID [24] or the W3C Recommendation *Prov-O* [28] of the Open Provenance Model [23] addressed the way traces should be modelled. Even those that made it to a standard are yet to see universal adoption, however, they helped define the scope of the problems at hand and reiterated the importance of the schools of thought behind traceability in data integration [29,20].

As dataset profiling began to gain attention in the context of the Semantic Web [13], the role of policy representations in the provenance element came under discussion, though the work carried out is still in preliminary stages on the modelling side [4]. Data integration, as one of the fields majorly concerned with provenance due to the setbacks of transformation processess, mostly appears to deal with provenance as the informative content required by users to design the integration rules and delegates to them the effort of assessing exploitability [31]. The introduction of *dataspaces* as a key concept in data integration, which denotes the units that are referenced by traceability information, has tremendously aided the integration processes themselves. Yet again, dataspaces do not take responsibility on managing the life-cycle of policies as part of their conceptual model [3].

Finally, we make use of the notion of Supply Chain Management, intended as the activity targeted to optimize networks of suppliers and customers in order to deliver superior value at less cost to the supply chain as a whole. In that setting, a network of interdependent actors mutually and co-operatively work together to improve the flow of materials and information from suppliers to end users. While we use this notion as a methaphor, where the materials are the data and the information the metadata, this comes useful to abstract from the complexity of grounded subproblems, like *data integration*, *metadata storage* or *policy management*, to mention a few. Altogether, they have led us to conclude that, to the best of our knowledge, there is no end-to-end solution for exploitablity assessment today.

3. Data cataloguing as a Metadata Supply Chain

A **data hub** is an infrastructure that manages a wide range of data sources and methods of delivering them,

[14] Licenses for Linked Data in RDF, `http://oeg-dev.dia.fi.upm.es/licensius/rdflicense/`

with the aim of providing users with services that rely upon data taken from the sources it manages. We also define (data) **exploitability** as the compatibility of the policies – inclusive of obligations, permissions and prohibitions – attached to the delivered data, with the requirements of the user's task.

In this article we argue that a prerogative of a high-quality data catalogue is that it must support users in assessing the exploitability of the data delivered by the various services of a data hub. Although exploitability assessment is ultimately for the end-user to perform, the objective is to support them by means of an end-to-end solution. In our proposal, such a solution is implemented within a *data cataloguing system* as an essential element of the Data Hub. We propose here a methodology to develop such an end-to-end solution, which role is to clarify: a) what is the general life-cycle of the data within a data hub; b) what are the actors involved in such a process, and what are their goals and tasks; c) what resources are needed, when and how they can be acquired and managed; and d) what operations have to be supported, in order for the exploitability assessment to be performed.

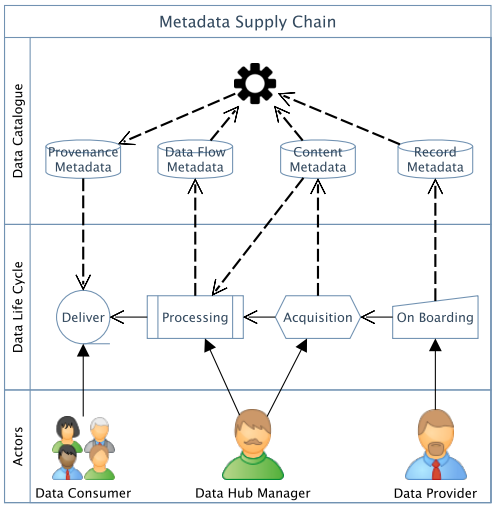

Fig. 1. Metadata Supply Chain Management (MSCM), overview.

The methodology that is introduced in this Section supports what we call "Metadata Supply Chain Management" (MSCM). It is based on a *data catalogue*, i.e. a running instance of a data cataloguing system, as a means to support MSCM.

Data can for example be imported by scheduled processes, or injested by external agents, reporting the activity of sensing devices, just to mention two intuitive examples. However, we can abstract the life cycle of the data in this environment in order to clarify the role that a data catalogue should have in the context of our work. Figure 1 gives an illustration of the elements of the methodology and their interaction. The primary assumption of our methodology is that a *Data Catalogue* exists, and is a shared resource on which all the different actors and phases rely on. Three are the actors involved in the methodology. A *Data Provider* aims to publish a new data source in order to provide value to the task of a given *Data Consumer*. A *Data Hub Manager* has the role to supervise the infrastructure in terms of configuration, maintenance and monitoring. Our methodology follows the *Data life-cycle*, which comprises four phases:

- **Onboarding**: data sources are registered with the Data Hub;
- **Acquisition**: data are imported in the Data Hub;
- **Processing**: data are processed, manipulated and analysed in order to generate a new dataset, targeted to support some data-relying task;
- **Delivery**: resulting data are delivered to an external system/end-user.

The Metadata Supply Chain Management (MSCM) activity follows the Data Life Cycle in parallel. In the following paragraphs we provide the details of each phase, focusing on:

- the *objectives* that need to be reached;
- the *roles* of the actors in this phase;
- the required *resources* to be managed;
- *operations* that need to be performed at the different stages;
- what are the *output* resources of each phase; and
- under what *assumptions* the above can be implemented by this methodology.

Tables 1-4 list in details each the components of each phase in the methodology, and serve as a guide to its implementation in concrete use cases.

3.1. Onboarding

The Onboarding phase is dedicated to acquiring and managing information about data sources. When a *Data Provider* wishes to publish a new dataset, the Data Hub has to provide the required facility to do that. From the point of view of the *Data life-cycle*, in this phase the provider registers a new data source (or modifies an existing one) in the *Data Catalogue*,

Table 1

Phase 1. — Onboarding

Objectives	Obtain information about a data source
Roles	A Data Provider and a Data Hub Manager
Resources	A Data Catalogue, including a Licenses Database, and a data source
Operations	Registration of the data source in the Data Catalogue.
Output	Structured information about the data source in the form of a Catalogue Record.
Assumptions	
1.1: The Data Provider associates a single License to the data source. *1.2*: The License is granted to whoever exploits the given data source. *1.3*: The License is described in the Licenses Database. *1.4*: Policies are set of binary relations between a deontic component (permission, prohibition, requirement) and an action. *1.5*: Policies are referenced by Policy Propagation Rules (PPRs), part of the Licenses Database.	

that is the space where dataset descriptions are managed. The Data Catalogue manages metadata about the data source as a *Catalogue Record*. These metadata must include information about the exploitability of the dataset in form of a data *License*, and details about the ownership of the data and a potential attribution statement.

The output of this component is metadata about the data source represented as a catalogue entry following the W3C DCAT specification [14]. This description includes details about how the dataset will be populated, and more importantly includes information about ownership (`dc:creator`) and licensing (`dc:license`), as well as attribution statement. The range of `dc:license` is meant to be a structured description of a license according to the ODRL Ontology [30], included in a *Licenses Database*. Licenses are described as set of *policies*, each being a binary association between a deontic component and an action (eg: requirement+attribution, prohibition+commercial_use), according to the definition in [10].

The onboarding process relies on the assumption that the licensor states a single license (Assumption 1.1), applicable to whoever exploits the given data source (1.2). The terms and conditions of the data sources are in the set of the available licenses in the data catalogue. This includes the other assumption that licenses can be described as set of ODRL policies, each one of them as a binary association betwen a deontic component and an action (Assumption 1.4). Existing policies are included in the set of Policies Propagation Rules (PPR) [10], also part of the Licenses Database.

Table 2

Phase 2. — Acquisition

Objectives	Access the data source and collection of the related data.
Roles	The Data Hub Manager supervises and monitors the relevant procedures.
Resources	A Catalogue Record, containing information about how to access the data.
Operations	Collection of the data, inspection and eventually storage in a staging environment.
Output	Content Metadata, ready to be exploted by the required processes.
Assumptions	
2.1: The data source is accessible. *2.2*: Acquisition is performed by respecting the data source License.	

3.2. Acquisition

After onboarding a new data source, the data need to be acquired by the data hub. "Acquiring" means that the data hub is given a means to control the delivery cycle of the data whose awareness was granted through the onboarding phase. Access methods can have various form, according to the different kind of data sources. For example, data sources could be registered as web accessible resources (via HTTP or FTP), Web APIs, or uploaded files. Methods for acquisition can include collecting resources from external systems or requiring an ingestion API to be exposed. The configuration of these processes can be fully automated, or dedicated procedures could be developed in the Data Hub so that specific data sources could be acquired. It is the role of a *Data Hub Manager* to supervise this process and monitor the acquisition, including implementing the needed strategies for data update and quality control. This activity can be rather complex, including automatic and supervised methods, and going into the details of it is out of scope for this article. What is important for us is that this phase should provide a sufficient amount of *metadata* in order to support data processing. It is the integration strategy itself that provides the requirements for metadata acquisition. However, any data integration task directly depends on a *data discovery* task. *Content Metadata* (see Figure 1) refers to topical and structural information that might be established by accessing the actual data, for example the types of the entities included in the content, the set of attributes, local and global identifiers (and their structure or format), relations and references to external datasets, as well as statistics about them. These will serve the purpose of discovering data sources and support the configuration of integration strategies by the Data Hub Manager. This phase is based on the assump-

tions that the data source is actually accessible by the Data Hub (Assumption 2.1) and that acquisition is possible according to the data source license (Assumption 2.2).

3.3. Processing

In this phase the data are manipulated in order to fullfil a given task that relies upon them. This activity can be seen as supporting a traditional ETL [32] task. As already mentioned in the previous Section 3.2, there is a strong dependency between the actual data processing strategy with the *Content Metadata* collected, as features like the schema, format and size of the data has a clear impact on the implementation of ETL. So, a *Data Catalogue* should provide information about the data sources in order to support the configuration of these processes, whether it is an automatic method or a process supervised by the *Data Hub Manager*. However, here we focus on the metadata that the data processing phase must produce in order for the Data Hub to support the exploitability assessment by the end user. From that point of view, a *Data Catalogue* should be capable of collecting plans about the integration processes in order to answer the following question: *when this process is executed, what will the policies attached to its output be?* Metadata about possible *processes* should be collected and stored in the catalogue, in order to allow reasoning on policy propagation, and to attach the required policies to the resulting dataset. Processes can be described as relations between data objects (Assumption 3.1). This is the approach followed by Datanode [9]. Datanode is an ontology that allows to represent data flows in a way that makes it possible to reason on Policy Propagation Rules (PPRs) [10]. Processing pipelines can be annotated with *data flow descriptions* as RDF representation of the processes using Datanode, allowing to execute Policy Propagation Rules (PPRs) and determine what policies can be attached to the output of each process. In a general case, the *Data Hub Manager* is responsible of providing such information, as well as assessing that the processing itself is made respecting the policies of the data sources (Assumption 3.2). Data flow descriptions should not involve runtime information. These metadata should provide an abstract representation of the process so that, once combined with the actual input (a given data catalogue record and content metadata), it would be possible to generate the relevant policies. In other words, a given data flow description should be valid for all possible executions of a process (Assumption 3.3).

Table 3

Phase 3. — Processing

Objectives	Obtain a new dataset to support a specific data-relying task.
Roles	The Data Hub Manager to configure the processes and produce descriptions of the data flows.
Resources	A Catalogue Record linked to Content Metadata. Processing will need to exploit the former or the latter, on a case by case basis.
Operations	Processes must be described as networks of data objects relying on the Datanode ontology.
Output	Data flow descriptions to be registered in the Data Catalogue.
Assumptions	
3.1: Processes can be described as data flows with Datanode. *3.2*: ETL processes do not violate the License of the source. *3.3*: Process executions do not influence policies propagation.	

Table 4

Phase 4. — Delivery

Objectives	Deliver the set of policies associated with the data as part of the provenance information.
Roles	The Data Consumer.
Resources	Catalogue Record, Data flow metadata, Policy Propagation Rules base
Operations	Reason on PPRs given the data flow description and the rule base.
Output	Set of policies attached as part of the provenance information of the returned data.
Assumptions	
4.1: Data flow descriptions and License policies enable reasoning on Policy Propagation Rules. *4.2*: End-user access method includes provenance information. *4.3*: Returned policies allow the end user to perform the assessment on data exploitability.	

3.4. Delivery

In this phase data are delivered to the end user or application. The *Data Catalogue* provides the required metadata to be distributed alongside the process output. Delivered data should include provenance information including: a) ownsership, b) attribution statement and c) policies (permissions, requirements, prohibitions). Delivered metadata should be included in the provenance information (Assumption 4.2), and support the user in assessing the data exploitability for the task at hand (Assumption 4.3). It is worth noting that the actual assessment of compatibility between the user's task and the policies of the output data is not part of this methodology, and is left to the end user. The exploitability task is indeed reduced to the assessment of the compatibility between the actions performed by the user's application and the policies attached to the datasets, with an approach similar to the one presented

in [16], for example using the SPIN-DLE reasoner[15], described in [21]. This methodology is targeted to offer the information required to perform such assessment, namely the usage policies attached to the offered data.

In the next Section we are going to validate our methodology by showing how it can achieve its objective with state of the art Semantic Web technologies, under the given set of assumptions. Secondly, we will perform a quantitative evaluation by inspecting the actual content of the Data Catalogue, to what extend the existing dataset descriptions meet the assumptions, and discuss existing limitations.

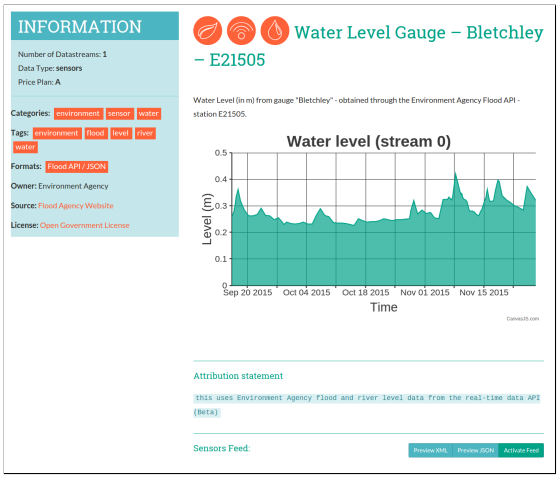

Fig. 2. Water Level Gauge - Bletchley - E21505.

4. System design: the MK:Smart Data Catalogue

Our hypothesis is that an *end-to-end* solution for exploitablity assessment can be developed by using state-of-the-art Semantic Web technologies. The MK Data Hub is the context in which we are going to validate the proposed methodology. In this Section we show how the phases of the methodology are supported by the MK Data Datalogue and how the implemented system verifies the assumptions so far introduced.

The MK:Smart project aims to provide citizens and companies with access to a wide range of data sources about the city of Milton Keynes (MK). These data sources include sensor data, public data extracted from the Web as well as data provided by public institutions and other organizations, for example Milton Keynes Council. These data sources, however, come with a set of policies regulating their usage. For example, the "Bletchley and Fenny Stratford" ward is a British electoral division that corresponds to an area in the South of the city. Located within this ward are a number of sensor devices that push data of varied nature to the Data Hub, including Air quality and Soil moisture (see an example in Figure 2). The National Museum of Computing is located in Bletchley park, and it is often a topic of interest in social platforms like Flickr. The Milton Keynes Council provides the MK Data Hub with statistics about population growth, crime, marital status, religion and employment, among others.

All these data sources are catalogued, consumed and stored as *datasets* by the Data Hub in order to provide the end-user with services that intensively rely upon these data. One of these services is the *Entity-Centric API* (ECAPI) of the Data Hub. The ECAPI offers an entity-based access point to the information offered by the Data Hub, on the rationale that the data themselves may, in their original form, be unable to convey their own relationships with real-world objects [1]. The aforementioned ward (see Figure 3 for some example data) and museum in Milton Keynes are examples of named entities the ECAPI may be queried for; but also, an arbitrary geographical area within a fixed radius of given geospatials coordinates (e.g. `51.998,-0.7436` in decimal degrees) could be an entity for an application to try to get information about (see Figure 4 for example data). The ECAPI will return a collection of items that are relevant for that location, selected from the appropriate datasets. In particular, the output will include information about the *provenance* of the data, detailing the policies attached to each unit thereof, in order to make the user aware of the possible permissions, requirements and prohibitions that restrict their usage.

In the rest of the section we will follow our methodology and provide insights as to how the MK:Smart Data Hub supports it in order to provide the users sufficient information to solve the data exploitability task. Figure 5 illustrates the components and their role in the data and metadata life-cycle of the MK Data Hub.

[15] `http://spin.nicta.org.au/spindle/index.html`

4.1. Onboarding

The Onboarding phase is the initial step of our methodology, and is supported by providing an *input interface* both for humans and external applications - implemented as a Data Hub Portal page and a Web API.

Following our guide use case, some data sources are Air Quality and Moisture Sensors in the Bletchley area, the Flickr API (including a number of images annotated with geocoordinates associated with the ward), the UK Food Estanblishments Info and Ratings API, as well as topographical information exposed by the Ordnance Survey and statistics from the Milton Keynes Council. Each one of these data sources have different licenses associated, kept from the collection of licenses described in RDF/ODRL in a *Licenses Database* (see Figure 5). For example, the metadata about the *Water Level Gauge - Bletchley - E21505* data source is one of the relevant data sources for the area. Figure 2 shows the Data Catalogue record as presented in the MK Data Hub web portal. As shown in Listing 1, the related description includes a reference to the *Open Government License*, described in the Licenses Database (Listing 2).

Listing 1: Dataset: Water Level Gauge - Bletchley - E21505: RDF description.

```
<http://datahub.mksmart.org/ns/dataset/water-level-gauge-
    bletchley-e21505>
      a       <http://www.w3.org/ns/dcat#Dataset> ;
      <http://datahub.mksmart.org/ns/schema/api>
              "https://datahub.beta.mksmart.org/data-
                  catalogue-api/?action=dataset&name=
                  water-level-gauge-bletchley-e21505"^^<
                  http://www.w3.org/2001/XMLSchema#
                  anyURI> ;
      <http://datahub.mksmart.org/ns/schema/attribution>
              "this uses Environment Agency flood and
                  river level data from the real-time
                  data API (Beta)" ;
      <http://datahub.mksmart.org/ns/schema/format>
              "Flood API / JSON" ;
      <http://datahub.mksmart.org/ns/schema/name>
              "water-level-gauge-bletchley-e21505" ;
      <http://datahub.mksmart.org/ns/schema/owner>
              "Environment Agency" ;
      <http://datahub.mksmart.org/ns/schema/policy>
              <http://datahub.mksmart.org/ns/policy/open-
                  government-license> ;
      <http://datahub.mksmart.org/ns/schema/theme>
              <http://datahub.mksmart.org/ns/category/s> ,
                   <http://datahub.mksmart.org/ns/
                  category/w> , <http://datahub.mksmart.
                  org/ns/category/l> ;
      <http://datahub.mksmart.org/ns/schema/uuid>
              "529bdc69-9874-4936-a94f-0036cb5a1e42" ;
      <http://purl.org/dc/terms/issued>
              "2015-04-02 09:50:46" ;
      <http://purl.org/dc/terms/modified>
              "2015-10-13 15:33:55" ;
      <http://purl.org/dc/terms/title>
              "Water Level Gauge - Bletchley - E21505" ;
      <http://www.w3.org/ns/dcat#distribution>
              <http://datahub.mksmart.org/ns/distribution
                  /3727593322> ;
      <http://www.w3.org/ns/dcat#landingPage>
              <https://datahub.beta.mksmart.org/dataset/
                  water-level-gauge-bletchley-e21505/> ;
      <http://xmlns.com/foaf/0.1/homepage>
              "https://datahub.beta.mksmart.org/dataset/
                  water-level-gauge-bletchley-e21505/" .
```

Listing 2: Open Government License: policy set

```
mks:ogl odrl:permission [ a odrl:Permission;
    odrl:action odrl:derive , odrl:distribute ,
        ldr:extraction , odrl:reproduce , odrl:read ,
        ldr:reutilization;
    odrl:duty [odrl:action odrl:attachPolicy , odrl:attribute]] .
```

Data sources like the *Flickr API* come with peculiar terms and conditions[16] (Listing 3). Some of them refer to the usage of the API, others to the assets the data are describing (like Flickr images). In these cases we limit the descriptions to the policies that are applicable to the accessed data, and describe them in the Licenses database. The description always include a reference to the document from which the policies have been extracted.

Listing 3: Flickr TOS

```
mks:flickrtos odrl:prohibition [ a odrl:Prohibition;
    odrl:action odrl:sell , odrl:sublicense , cc:CommercialUse
        ]
    odrl:duty [ odrl:action odrl:attribute ];
        mks:attributionStatement "This product uses the
            Flickr API but is not endorsed or certified by
            Flickr." ;
        dct:source <https://www.flickr.com/services/api/tos
            />
```

The *UK Food Estanblishments Info and Ratings* dataset includes a snapshot of the food hygiene rating data published at `http://www.food.gov.uk/ratings`. The data provide the food hygiene rating or inspection result given to a business and reflect the standards of food hygiene found on the date of inspection or visit by the local authority. The use of the API is open and the data can be displayed without restrictions, except modifications. Moreover, the policies include a peculiar attribution requirement (Listing 4). In this case, we simplify the ODRL description to make it compliant with a flat representation of policies, restricted to the policies about the *data* (Listing 5).

Listing 4: Terms and Conditions for the website and services at www.food.gov.uk/ratings

```
: a odrl:Agreement ;
```

[16] Flickr: `https://www.flickr.com/services/api/tos/`

```
  rdfs:label "Terms and conditions for information and
       services at food.gov.uk/ratings" ;
  tos:source <http://www.food.gov.uk/about-us/data-and-
       policies/aboutsite/termsandconditions/hygiene-rating
       -data> ;
  odrl:permission [
    a odrl:Permission;
       odrl:action odrl:display;
       odrl:target :data;
    odrl:duty [
      odrl:action odrl:display ;
      odrl:target :liveOrStaticDataStatement ;
      tos:excerpt "However, it is your responsibility to
           ensure that you make clear whether the data you
           are using is either from the live data or static
            data that is updated daily."@en ;
    ]
  ],[
    a odrl:Permission;
       odrl:target :api;
       odrl:action odrl:use
  ];
  odrl:prohibition [
    a odrl:Prohibition;
       odrl:action odrl:modify;
       odrl:target :data
  ];
  .

:liveOrStaticDataStatement
  a odrl:Asset, tos:Singleton ;
  rdfs:label "a statement displaying whether the data is
       taken from the live data or static data"
  .
```

Listing 5: Terms and Conditions for the website and services at www.food.gov.uk/ratings (simplified)

```
: a odrl:Agreement ;
  rdfs:label "Terms and conditions for information and
       services at food.gov.uk/ratings" ;
  tos:source <http://www.food.gov.uk/about-us/data-and-
       policies/aboutsite/termsandconditions/hygiene-rating
       -data> ;
  odrl:permission [
    a odrl:Permission;
       odrl:action odrl:display
  ];
  odrl:prohibition [
    a odrl:Prohibition;
       odrl:action odrl:modify
  ];
  :attributionStatement "Static data extracted from http://
       www.food.gov.uk/ratings"
  .
```

Statistics from the MK Council come with an Open Government License, thus usage of them should include an attribution statement.

4.2. Acquisition

The Acquisition phase is the stage of the methodology that covers the execution of the processes required to populate the dataset from the sources. This can be achieved in different ways in a Smart Cities Data Hub. For each type of source the data cataloguing system implements a dedicated *metadata extractor* with the objective to complement the *Dataset Record* with more metadata to supporting the data processing. This can include: data schemas and vocabularies, content partitions and their relationships, among others. For example, a dataset discovery tool might use information about the type of entities, properties and their statistics, or even content samples. The role of this component in the data cataloguing system is directly related to support the requirements of data processing. For example, air quality and soil moisture sensors push regular streams of data in the Data Hub. The Flickr API is invoked on demand and information stored at query time in temporary datasets. Twitter feeds are regularly collected and relevant tweets stored for a limited time. During these processes, metadata about the geolocalization of the related items are extracted and stored in the Data Catalogue. *Content Metadata* includes the location of the flickr images, while geocoordinates of the

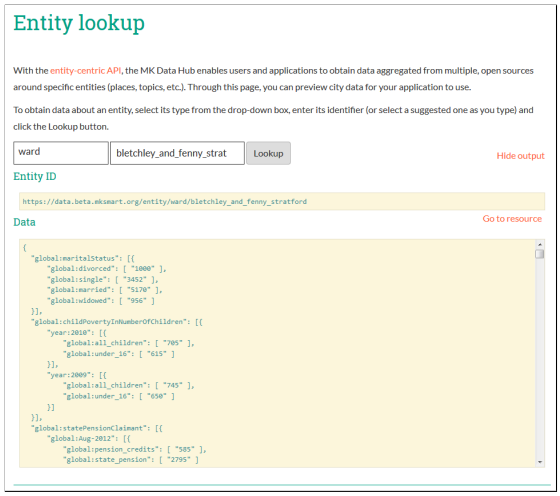

Fig. 3. MK:Smart Data Hub: example of a British ward.

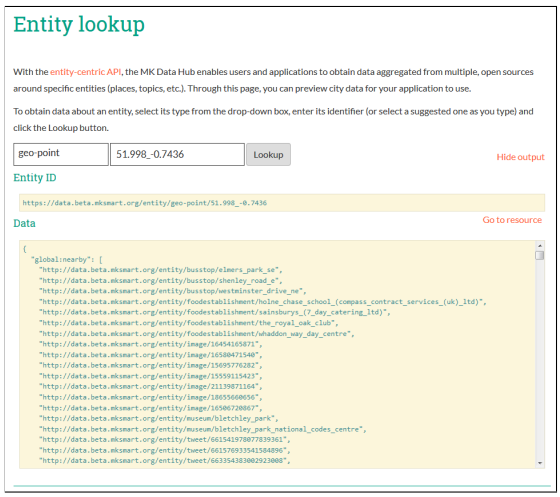

Fig. 4. MK:Smart Data Hub: example of a geographical point.

sensors are part of the *Dataset Record*. Statistical data from the MK Observatory has been imported in a single process, and the geocoordinates of the wards associated with each observation registered in the Content Metadata area of the Data Catalogue. This information is stored and used to configure the ECAPI with a manual process supervised by the *Data Hub Manager*.

4.3. Processing

In the Processing phase, data is extracted, transformed and loaded (ETL) in datasets using dedicated *pipelines*. Each pipeline performs a number of operations on the data sources in order to select the relevant information and transform it in a format suitable for the task at hand. A supervised process produces: a) a configuration for the processes to be executed and b) a description on the process capable of supporting the execution of Policy Propagation Rules using the Datanode ontology. Listing 6 shows the description of the processing pipeline of a file data source from Milton Keynes Council. The file is downloaded from the remote location and a copy is stored locally in a staging area (see also Figure 5). The content is then transformed into RDF using the CSV2RDF approach[17]. After that, as SPARQL query remodels the data applying the W3C Datacube Vocabulary[18] data model. This data is accessed by a SPARQL query, which selects a relevant portion of the data for the task at hand.

Listing 6: Processing pipeline for a CSV file.

```
:input a dn:Datanode;
   mks:format mks:csv;
   dn:hasCopy [
      dn:refactoredInto [
         mks:format mks:rdf;
         dn:usesSchema csvOntology: ;
         dn:remodelledInto [
            dn:usesSchema qb:
            dn:hasSelection :output .
         ] .
      ] .
    ] .
```

The descriptions of the data flows excuted in this phase allow the Data Catalogue to execute PPRs and associate to each dataset the right policies to be exposed to the users. These models represent in an abstract way the process, and they are agnostic with respect to the actual input.

[17] http://www.w3.org/TR/csv2rdf/
[18] http://www.w3.org/TR/vocab-data-cube/

4.4. Delivery

The Data Hub exposes a number of APIs to access the data in various forms. For example, sensor data can be extracted as streams by providing temporal constraints. The Entity Centric API is a specialized service for data discovery, that aggregates information summaries from several datasets about a given entity. In our guide examples, an application requests information about a location in Milton Keynes, in the form of geocoordinates: `51.998,-0.7436`. The output includes an aggregated view of items related to that geolocation as well as provenance information for each one of them, including the policies relevant to assess the exploitability of each item, thanks to a Data Cataloguing system that supports exploitability. The PPR Reasoner will be queried providing the actual input as a specific dataset in the catalogue, according to the user's query.

Listing 7: Policy Propagation Rules.

```
propagates(dn:remodelledTo,duty cc:ShareAlike)
propagates(dn:hasSelection,duty cc:ShareAlike)
propagates(dn:hasCopy,duty cc:ShareAlike)
```

The dataflow description will be complemented by the related dataset record metadata and associated policies from the licenses database. Listing 7 shows a subset of the rules that are activated in relation to the dataflow (Listing 6) and policies set (Listing 2). The propagated policies are displayed in Listing 8.

Listing 8: Policies associated with the returned data processed from the original Milton Keynes council CSV file.

```
[] a dn:Datanode ;
    odrl:duty [odrl:action odrl:attachPolicy, odrl:attribute
        ]
```

5. Discussion

We described how the MK:Smart Data Catalogue supports the methodology proposed in this article. Table 5 summarizes the assumptions upon which the methodology relies. In this Section we are going to discuss *to what extent* the assumptions are valid for the MK Data Hub.

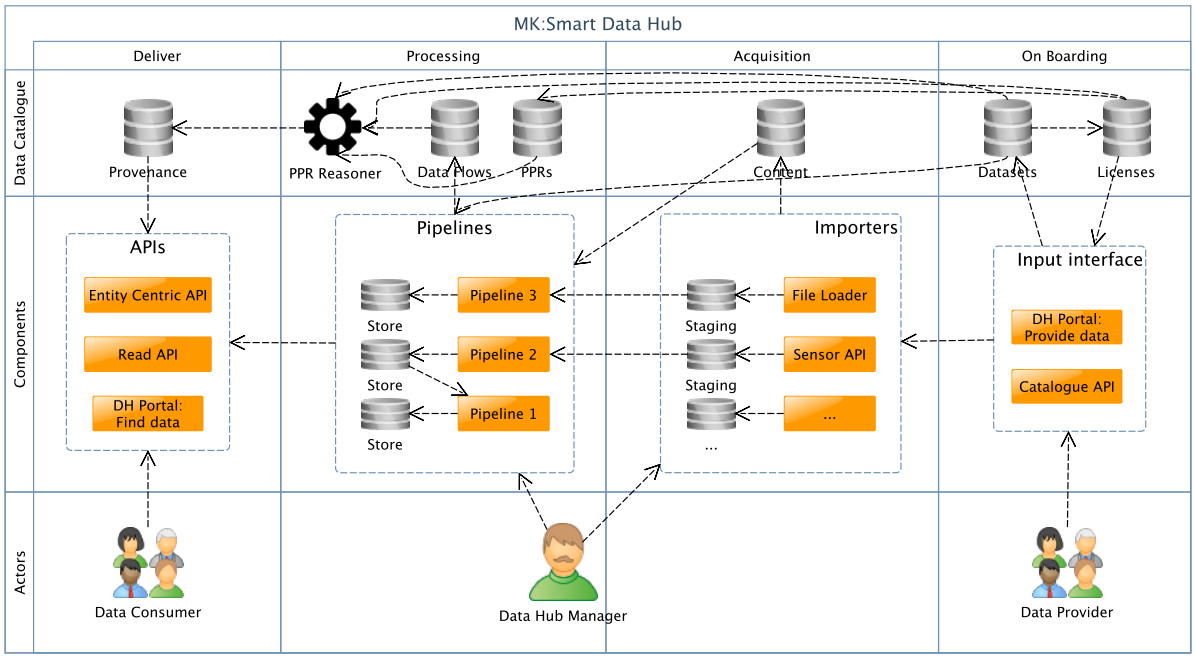

Fig. 5. MK Data Hub overview. The figure shows the phases of the methodologies and how they are supported by the MK Data Hub. The *Data Catalogue* is the component responsible for managing the *Metadata Supply Chain*, interacting with the other components of the system. On the right side of the image, an *Input interface* is exposed to allow *Data Providers* to register data sources and selecting the appropriate license. A *Data Hub Manager* is responsible for the description of licenses, and supervises the activity of *importers* and processing *pipelines*, including the curation of data flow descriptions (*Data Flows*) and policy propagation rules (*PPRs*). *Data Consumers* invoke APIs in the *Delivery* phase and associated *Provenance* information is provided from the *Data Catalogue*, exploiting a *PPR Reasoner* that relies on *Data Flows* descriptions and *PPRs*.

Table 5
Assumptions

Id	Assumption
1.1	The Data Provider associates a single License to the data source.
1.2	The License is granted to whoever exploits the given data source.
1.3	The License is described in the Licenses Database.
1.4	Policies are set of binary relations between a deontic component (permission, prohibition, requirement) and an action.
1.5	Policies are referenced by Policy Propagation Rules (PPRs), part of the Licenses Database.
2.1	The data source is accessible.
2.2	Acquisition is performed by respecting the data source License.
3.1	Processes can be described as data flows with Datanode.
3.2	ETL processes do not violate the License of the source.
3.3	Process executions do not influence policies propagation.
4.1	Data flow descriptions and License policies enable reasoning on Policy Propagation Rules.
4.2	End-user access method includes provenance information.
4.3	Returned policies allow the end user to perform the assessment on data exploitability.

Assumption 1.1 — The Data Provider associates a single License to the data source. Each Dataset is supposed to be annotated with a *single* license. The MK Data Hub contains today[19] 202 datasets. All of them specify a single license. This assumptions is fully valid for existing datasets. However we can expect cases where the license can change depending on the type of user or the context of applications. This is case of data from the BBC Weather Service, which terms and conditions[20] for commercial and non commercial use are different, and are also specified in different documents. While we do not support complex policies at the moment, we could deal with it by user profiling (with a commercial or non commercial account), or by including a taxonomy of usage contexts to consider separately, thus obtaining multiple policy sets depending on the usage context. Delivered metadata could include multiple policy sets associated with the related contextual information.

Assumption 1.2 — The License is granted to whoever exploits the given data source. The methodology as-

[19] November 2015.
[20] `http://www.bbc.co.uk/terms/`

Table 6
Licenses and their use.

N	License
71	"Open Government License"
31	"Other"
27	"Creative Commons Attribution License"
20	"Netatmo API Terms of use"
8	"Open Database License (ODbL) v1.0"
4	"OS Open Data License"
2	"Flickr APIs Terms of Use"
1	"Terms and conditions for information and services at food.gov.uk/ratings"

sumes the license of the data source to refer to any possible user having access to the data source. However, we can imagine situations in which the terms of use may vary depending on different kind of users, because of private agreements between the parties. While the MK Data Hub does not support this facility, it is possible to envisage an extension of the methodology in which the License Database contains associations between licenses and (classes of) users, thus enabling the configuration of the PPR reasoner in order to select the relevant License between the set of possible ones. This can be supported particularly because the License is part of the input of the PPR reasoner, together with Dataflow description.

Assumption 1.3 — The License is described in the Licenses Database. In the MK Data Catalogue, the number of datasets that do not specify a license is 33 ("Other" in Table 6). There can be many reasons for that. In some cases Data providers do not want (yet) to redistribute the content of the dataset, and rely on the MK Datahub solely for their own applications. Sometimes the intended license is not present in the current selection of licenses. When this happens, the user can contact the Data Hub Manager and discuss her specific requirement. In the future, we plan to allow the users to create entirely customized policies to be associated with their data, as supervised by the Data Hub Manager, and in cooperation with the legal team of the Data Hub. Table 6 summarizes the licenses currently used.

Assumption 1.4 — Policies are set of binary relations between a deontic component (permission, prohibition, requirement) and an action. Policies can have a very diverse structures, including composite constraints involving actions, classes of users, conjunctions, disjunctions, etc. While these can be represented in ODRL, in this work we only focused on policies having a flat representation, i.e. a binary association between a deontic component and an action. However, Policy Propagation Rules treat the policy as an atom that can or cannot propagate through relations between datanodes. For this reason, the actual structure of the policy does not affect the behavior of the rule, and we can extend our framework to also work on more complex ones. This would have an impact on the life cycle of policies and licenses definition, which should be extended to also manage these kind of policies, when necessary. At the moment these policies are not represented in the Licenses Database. However, we performed an informal evaluation of this aspect using the RDF License Database, that contains a number of licenses expressed as RDF/ODRL. We observed that all the RDF/ODRL policies expressed in the database can be reduced to sets of binary associations between a deontic component and an action, thus supporting this assumption[21].

Assumption 1.5 — Policies are referenced by Policy Propagation Rules (PPRs), part of the Licenses Database. In order for the process to be succesful, all policies used to describe licenses in the License Database need to be referenced appropriately by Policy Propagation Rules. Policies introduced by new licenses should be also included in the set of rules. In [10], a methodology to manage a knowledge base of policies propagation rules is presented, and we rely on that approach to manage the evolution of the rule base in order to guarantee that any policy in the licenses database is properly represented by PPRs.

Assumption 2.1 — The data source is accessible. Data Catalogues are conceived as metadata repositories, that act as registries of existing datasets. In our methodology, ETL processes rely on Content Metadata that is generated by inspecting the data source, thus establishing a dependency between the Dataflow description and the access of the actual data. While it is obvious that derived datasets cannot be generated without accessing the input data source, we can envisage situations in which data flow descriptions can be generated with no need to access the data source. One example is when the structure of the data conforms an existing standard and the process itself is agnostic with respect to the population of the dataset. In these cases, process executions can be simulated by running the PPR Reasoner with the related data source (metadata) as input.

[21] However, we did not performed a validation of the accuracy of the RDF Licenses Database

Assumptions 2.2 — Acquisition is performed by respecting the data source License. and 3.2 — ETL processes do not violate the License of the source. The Data Hub Manager has the responsability to respect the terms and conditions on the data access method as well as the ETL procedures involved. This assessment can be performed by inspecting the data source licenses. While currently the MK Data Hub does not support ETL processes involving multiple datasets, these cases can be also supported by relying on the licenses compatibility approach. The need of setting up Dataflow descriptions guarantees that there exists one operation/phase under which this assessment will be performed.

Assumption 3.1 — Processes can be described as data flows with Datanode. The primary implication is that Datanode is capable of describing the data flow. Datanode is an evolving component and it can be extended by adding new relations in the ontology. This can also evolve the Policy Propagation Rules database, following the method described in [10]. For this reason, we can assume that Datanode will have enough expressivity to cover existing dataflows. The generation of the policy set to attach to the output is performed at runtime. This method allows for process descriptions to be reusable between different executions. However this implies that processes need to be careful not to change the implications of the policies at runtime. For example, if some policy applies to a specific section of the data, different runtime executions might have different policies depending on the selected data. This aspect is not currently supported and processes are designed in order to be agnostic with respect to runtime information (user's input). Without this assumption, process executions should be able to provide fine grained traces (eg: logs) that could be then transformed in Datanode graphs. This could be an interesting future work to experiment with.

Assumption 3.3 — Process executions do not influence policies propagation. This assumption is an implication of Assumption 1.1. If policies are attached to the whole dataset, different executions of the same process will always refer to the same set of policies. Dataflow descriptions are based on the operations performed by the ETL process on hipothetical inputs. At runtime, the concrete data source is selected, thus the set of policies of the related license. This is not necessarely always true. As a negative example, we can imagine a dataset including policies attached at instance level. The records referring the current year cannot be used for commercial purposes, while data about the past years are of public domain. Depending on the input of the query, a process might or might not select restricted data, thus changing at runtime the information required to assess policies propagation. We solve this problem by slicing the data source in different *Catalogue Record*s, with different licenses.

Assumption 4.1 — Data flow descriptions and License policies enable reasoning on Policy Propagation Rules. Following the approach in [10], and given the *Assumptions 1.5* and *3.1*, the PP Reasoner will have sufficient information to reason on policies propagation.

Assumption 4.2 — End-user access method includes provenance information. and 4.3 — Returned policies allow the end user to perform the assessment on data exploitability. Finally, the methodology assume the user has access to some metadata (Provenance information). The user's task need to be expressible in terms of ODRL policies, thus enabling reasoning on policies compatibility. However, while this assessment is part of an early analysis, when the user wants to assess whether a given dataset is eligible to be adopted, we expect this assessment to be performed manually, on a case by case basis. We plan to extend the MK Data Hub Portal to also support a friendly user interface that users can exploit to validate the policies with respect to her requirements.

6. Conclusions

The MK Data Hub indeed supports the methodology proposed in this article. A *Data Provider* registers a dataset in the Data Hub, and can indicate a license from the ones available in the Licenses Database, containing a set of licenses described in RDF/ODRL. These policies are mapped to Policy Propagation Rules following the approach described in [10]. During the import phase, *Content Metadata* is extracted, assuming that all the relevant information to setup data integration startegies is available. A supervised process produces: a) a configuration for the processes to be executed and b) a description on the process capable of supporting the execution of Policy Propagation Rules. Since policies and data flows are described according to the process in [10], they enable a *PPR Reasoner* to execute Policies Propagation Rules in relation to the process dataflow description and to generate the part of *Provenance Metadata* to be attached to the result

of the call to the ECAPI. According to this process, the end user will have enough information to select the appropriate dataset to fullfill her task, according to whether her requirements match the policies associated with the dataset descriptions, which are therefore supporting exploitability assessment.

While our work focuses on the metadata required to assess exploitability, a similar methodology can in principle be applied to other metadata-relying tasks.

Future work includes the support of multiple licenses by enabling "scopes" of use as additional metadata, user profiling in order to add more contextual information to the reasoning process, and expanding the data flow descriptions phase to also support articulate processes by adding process execution traces as part of the description.

In a complex environment like the one of MK:Smart, there might be other research questions related to policies and constraints with respect to the data sources, data flow and output, respectively. For example:

- How to automate the assessment of the compatibility of the data flow with the policies attached to the input of the process?
- How we can diagnose inconsistencies between a data flow and the related data policies?
- How to support the user to assess the consistency between the policies of multiple data objects the user wants to exploit in a single process (integration)?
- How to support the user on describing her task in a way to recommend relevant datasets or processing methods?

We plan to explore these questions further in an expanding framework for computationally handling data usage policies, of which the presented methodology is the foundation.

References

[1] A. Adamou and M. d'Aquin. On requirements for federated data integration as a compilation process. In Berendt et al. [5], pages 75–80.

[2] S. Aissi, M. S. Gouider, T. Sboui, and L. B. Said. Enhancing spatial datacube exploitation: A spatio-semantic similarity perspective. In G. Dregvaite and R. Damasevicius, editors, *Information and Software Technologies - 20th International Conference, ICIST 2014, Druskininkai, Lithuania, October 9-10, 2014. Proceedings*, volume 465 of *Communications in Computer and Information Science*, pages 121–133. Springer, 2014.

[3] D. W. Archer, L. M. L. Delcambre, and D. Maier. A framework for fine-grained data integration and curation, with provenance, in a dataspace. In J. Cheney, editor, *First Workshop on the Theory and Practice of Provenance, TaPP'09, San Francisco, CA, USA, February 23, 2009, Proceedings*. USENIX, 2009.

[4] A. Assaf, R. Troncy, and A. Senart. HDL - towards a harmonized dataset model for open data portals. In Berendt et al. [5], pages 62–74.

[5] B. Berendt, L. Dragan, L. Hollink, M. Luczak-Rösch, E. Demidova, S. Dietze, J. Szymanski, and J. G. Breslin, editors. *Joint Proceedings of the 5th International Workshop on Using the Web in the Age of Data (USEWOD '15) and the 2nd International Workshop on Dataset PROFIling and fEderated Search for Linked Data (PROFILES '15) co-located with the 12th European Semantic Web Conference (ESWC 2015), Portorož, Slovenia, May 31 - June 1, 2015*, volume 1362 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2015.

[6] B. Carminati, E. Ferrari, and B. M. Thuraisingham. Using RDF for policy specification and enforcement. In *15th International Workshop on Database and Expert Systems Applications (DEXA 2004), with CD-ROM, 30 August - 3 September 2004, Zaragoza, Spain*, pages 163–167. IEEE Computer Society, 2004.

[7] A. Chappell, S. Choudhury, J. Feo, D. J. Haglin, A. Morari, S. Purohit, K. Schuchardt, A. Tumeo, J. Weaver, and O. Villa. Toward a data scalable solution for facilitating discovery of scientific data resources. In X. Sun, Y. Chen, and P. C. Roth, editors, *Proceedings of the 2013 International Workshop on Data-Intensive Scalable Computing Systems, DISCS 2013, Denver, Colorado, USA, November 18, 2013*, pages 55–60. ACM, 2013.

[8] J. Crampton and H. Khambhammettu. Delegation and satisfiability in workflow systems. In I. Ray and N. Li, editors, *SACMAT 2008, 13th ACM Symposium on Access Control Models and Technologies, Estes Park, CO, USA, June 11-13, 2008, Proceedings*, pages 31–40. ACM, 2008.

[9] E. Daga, M. d'Aquin, A. Gangemi, and E. Motta. Describing semantic web applications through relations between data nodes. Technical Report kmi-14-05, Knowledge Media Institute, The Open University, Walton Hall, Milton Keynes, 2014.

[10] E. Daga, M. d'Aquin, A. Gangemi, and E. Motta. Propagation of policies in rich data flows. In *Proceedings of the 8th International Conference on Knowledge Capture*, page 5. ACM, 2015.

[11] M. d'Aquin, A. Adamou, E. Daga, S. Liu, K. Thomas, and E. Motta. Dealing with diversity in a smart-city datahub. In T. Omitola, J. Breslin, and P. Barnaghi, editors, *Proceedings of the Fifth Workshop on Semantics for Smarter Cities, a Workshop at the 13th International Semantic Web Conference (ISWC 2014)*, Riva del Garda, Italy, 19 October 2014. CEUR-WS.org.

[12] M. d'Aquin, J. Davies, and E. Motta. Smart cities' data: Challenges and opportunities for semantic technologies. *Internet Computing, IEEE*, 19(6):66–70, 2015.

[13] E. Demidova, S. Dietze, J. Szymanski, and J. G. Breslin, editors. *Proceedings of the 1st International Workshop on Dataset PROFIling & fEderated Search for Linked Data co-located with the 11th Extended Semantic Web Conference, PROFILES@ESWC 2014, Anissaras, Crete, Greece, May 26, 2014*, volume 1151 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2014.

[14] J. Erickson and F. Maali. Data catalog vocabulary (DCAT). W3C recommendation, W3C, Jan. 2014. http://www.w3.org/TR/2014/REC-vocab-dcat-20140116/.

[15] G. Governatori, H.-P. Lam, A. Rotolo, S. Villata, G. Atemezing, and F. Gandon. Checking licenses compatibility between vocabularies and data. In *Proceedings of the Fifth International Workshop on Consuming Linked Data (COLD2014)*, 2014.

[16] G. Governatori, A. Rotolo, S. Villata, and F. Gandon. One license to compose them all. In *The Semantic Web–ISWC 2013*, pages 151–166. Springer, 2013.

[17] K. Gunaratna, S. Lalithsena, and A. P. Sheth. Alignment and dataset identification of linked data in semantic web. *Wiley Interdisc. Rew.: Data Mining and Knowledge Discovery*, 4(2):139–151, 2014.

[18] O. Hartig. Querying trust in RDF data with tSPARQL. In L. Aroyo, P. Traverso, F. Ciravegna, P. Cimiano, T. Heath, E. Hyvönen, R. Mizoguchi, E. Oren, M. Sabou, and E. P. B. Simperl, editors, *The Semantic Web: Research and Applications, 6th European Semantic Web Conference, ESWC 2009, Heraklion, Crete, Greece, May 31-June 4, 2009, Proceedings*, volume 5554 of *Lecture Notes in Computer Science*, pages 5–20. Springer, 2009.

[19] S. S. Kemperman, B. Brembeck, E. W. Brown, A. de Langevan Oosten, T. Fons, C. Giffi, N. Levin, A. Morrison, C. Ruschoff, G. A. Silvis, and J. White. Success strategies for electronic content discovery and access. White paper, OCLC, 2014. http://www.oclc.org/content/dam/oclc/reports/data-quality/215233-SuccessStrategies.pdf.

[20] H. Kondylakis, M. Doerr, and D. Plexousakis. Empowering provenance in data integration. In J. Grundspenkis, T. Morzy, and G. Vossen, editors, *Advances in Databases and Information Systems, 13th East European Conference, ADBIS 2009, Riga, Latvia, September 7-10, 2009. Proceedings*, volume 5739 of *Lecture Notes in Computer Science*, pages 270–285. Springer, 2009.

[21] H.-P. Lam and G. Governatori. The making of spindle. In *Rule Interchange and Applications*, pages 315–322. Springer, 2009.

[22] F. Maali, R. Cyganiak, and V. Peristeras. Enabling interoperability of government data catalogues. In M. Wimmer, J. Chappelet, M. Janssen, and H. J. Scholl, editors, *Electronic Government, 9th IFIP WG 8.5 International Conference, EGOV 2010, Lausanne, Switzerland, August 29 - September 2, 2010. Proceedings*, volume 6228 of *Lecture Notes in Computer Science*, pages 339–350. Springer, 2010.

[23] L. Moreau and P. Missier. PROV-dm: The PROV data model. W3C recommendation, W3C, Apr. 2013. http://www.w3.org/TR/2013/REC-prov-dm-20130430/.

[24] T. Omitola, N. Gibbins, and N. Shadbolt. Provenance in linked data integration. In *Future Internet Assembly*, December 2010. Event Dates: 16-17 December 2010.

[25] J. Padget and W. W. Vasconcelos. Policy-carrying data: A step towards transparent data sharing. In E. M. Shakshuki, editor, *Proceedings of the 6th International Conference on Ambient Systems, Networks and Technologies (ANT 2015), the 5th International Conference on Sustainable Energy Information Technology (SEIT-2015), London, UK, June 2-5, 2015*, volume 52 of *Procedia Computer Science*, pages 59–66. Elsevier, 2015.

[26] L. Pérez-Freire and F. Pérez-González. Exploiting security holes in lattice data hiding. In T. Furon, F. Cayre, G. J. Doërr, and P. Bas, editors, *Information Hiding, 9th International Workshop, IH 2007, Saint Malo, France, June 11-13, 2007, Revised Selected Papers*, volume 4567 of *Lecture Notes in Computer Science*, pages 159–173. Springer, 2007.

[27] S. Ram and J. Liu. A new perspective on semantics of data provenance. In J. Freire, P. Missier, and S. S. Sahoo, editors, *Proceedings of the First International Workshop on the role of Semantic Web in Provenance Management (SWPM 2009), collocated with the 8th International Semantic Web Conference (ISWC-2009), Washington DC, USA, October 25, 2009*, volume 526 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2009.

[28] S. Sahoo, T. Lebo, and D. McGuinness. PROV-O: The PROV ontology. W3C recommendation, W3C, Apr. 2013. http://www.w3.org/TR/2013/REC-prov-o-20130430/.

[29] Y. Simmhan, B. Plale, and D. Gannon. A survey of data provenance in e-science. *SIGMOD Record*, 34(3):31–36, 2005.

[30] S. Steyskal and A. Polleres. Defining expressive access policies for linked data using the ODRL ontology 2.0. In H. Sack, A. Filipowska, J. Lehmann, and S. Hellmann, editors, *Proceedings of the 10th International Conference on Semantic Systems (SEMANTiCS 2014)*, pages 20–23. ACM, 2014.

[31] B. Tomazela, C. S. Hara, R. R. Ciferri, and C. D. de Aguiar Ciferri. Empowering integration processes with data provenance. *Data Knowl. Eng.*, 86:102–123, 2013.

[32] P. Vassiliadis, A. Simitsis, and S. Skiadopoulos. Conceptual modeling for ETL processes. In *Proceedings of the 5th ACM international workshop on Data Warehousing and OLAP*, pages 14–21. ACM, 2002.