Use of Semantic Technologies to Inform Progress Toward Zero-Carbon Economy

Stefano Germano[1(✉)], Carla Saunders[2], Ian Horrocks[1], and Rick Lupton[2]

[1] Department of Computer Science, University of Oxford, Oxford, UK
{stefano.germano,ian.horrocks}@cs.ox.ac.uk
[2] Department of Mechanical Engineering, University of Bath, Bath, UK
{cs2537,R.C.Lupton}@bath.ac.uk

Abstract. To investigate the effect of possible changes to decarbonise the economy, a detailed picture of the current production system is needed. Material/energy flow analysis (MEFA) allows for building such a model. There are, however, prohibitive barriers to the integration and use of the diverse datasets necessary for a system-wide yet technically-detailed MEFA study. Herein we describe a methodology exploiting Semantic Web technologies to integrate and reason on top of this diverse production system data. We designed an ontology to model the structure of our data, and developed a declarative logic-based approach to address the many challenges arising from data integration and usage in this context. Further, this system is designed for easy access to the needed data in terms relevant for additional modelling and to be applied by non-experts, allowing for a wide use of our methodology. Our experiments with UK production data confirm the usefulness of this methodology through a case study based on the UK production system.

Keywords: Semantic technology · Resource efficiency · Rule-based approach · Data integration · Material Flow Analysis · Ontology · Decision Support System

1 Introduction

A whole-systems understanding of production systems is essential to navigating the necessary rapid transition to a zero-carbon economy. Identifying opportunities and monitoring progress relies on having access to data about the production and consumption of physical resources (materials, products, energy, etc.) and their associated environmental impacts. However, due to the economy-wide yet detailed nature of these questions, they cannot be answered from single datasets collected by one entity, but must instead be based on many pieces of

This work was supported by the EPSRC project UK FIRES (EP/S019111/1), the SIRIUS Centre for Scalable Data Access (Research Council of Norway, project no.: 237889), and Samsung Research UK.

A. Hotho et al. (Eds.): ISWC 2021, LNCS 12922, pp. 665–681, 2021.
https://doi.org/10.1007/978-3-030-88361-4_39

data from different international and national organisations, individual companies, and academic research.

This data is incomplete, and defined using inconsistent categorisations of the types of resource and activities. It is thus challenging to obtain the clear, complete and robust picture that is needed of how our economies are functioning and could change [20]. In addition, the lack of well-defined data models for this type of data is limiting to data reuse and holding back academic research [9,19]. While progress has been made in developing shared data models [8,11,17,18] and data catalogues [14,16], which improve access to and reuse of relevant datasets, they do not yet confront the fundamental challenge of resolving conflicts where individual datasets are defined in inconsistent ways.

Semantic Web technologies are well placed to help with these types of problem, but there are some key challenges to their application. The knowledge representation and reasoning side requires complex modelling and expressive logic-based languages, due to the heterogeneity of the data. Furthermore, any solution must be accessible by people without specialised knowledge of Semantic Web technologies, requiring care in selecting an appropriate model and designing and implementing suitable technical solutions.

In this paper, we propose and develop a solution using a domain ontology and the RDFox triple store to efficiently implement Datalog rules integrating diverse data points into a consistent structure. This forms part of the "Physical Resources Observatory" (PRObs) system, being developed within the *UK FIRES* research programme[1], where it supports a wider research agenda on resource efficiency and decarbonisation in UK industrial strategy.

2 The Need for Monitoring the Physical Economy

Understanding how we produce and consume physical resources is fundamental to understanding the impacts human activity has, and how we can operate more efficiently. The following examples illustrate a range of uses for this knowledge.

Example 1 (Innovation in material efficiency). About half of industrial CO_2 emissions are due to production of just five major bulk materials [1]. Reducing scrap created during manufacturing processes would reduce overall demand for materials and hence emissions. But identifying the potential savings and opportunities for new manufacturing processes requires an understanding of how and where scrap is currently produced in the supply chain.

Example 2 (Reuse of building components). Components of buildings could be reused when the building is no longer needed [2], which would reduce emissions from recycling and production. But doing this requires knowledge of what components are available in existing buildings, which is generally not known directly. By monitoring materials going into construction and from demolition, the current composition of the building stock can be estimated.

[1] https://ukfires.org.

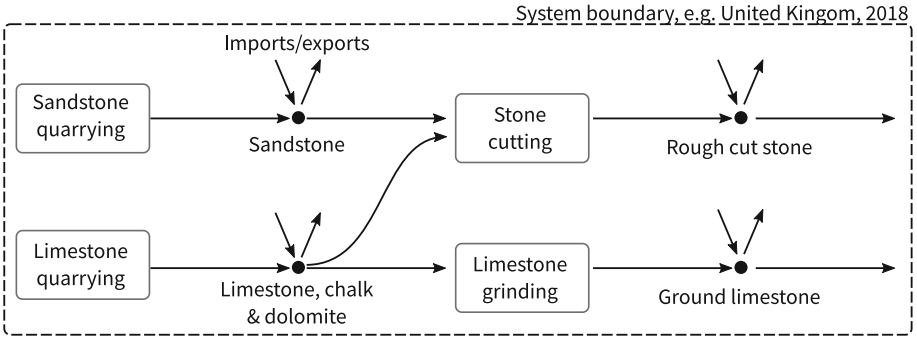

Fig. 1. MEFA system showing rock processing stages in the UK. Processes are shown by boxes. The arrows represent flows. The types of materials are shown by dots, with the vertical flows representing trade flows across the system boundary.

Example 3 (Supply constraints). Biomass is in demand for low-carbon energy supply and as a low-carbon building material, but supply is limited [6]. Reconciling this requires a whole-system view of total quantities of materials produced, together with all uses.

Since all the important characteristics of these systems cannot generally be measured directly, models are used to fill gaps and reconcile conflicts in data. Information is sparse, meaning that every piece of relevant data is valuable to confirm or improve our understanding of the system.

2.1 Material/Energy Flow Analysis

Although the general challenges of data access apply to a broader range of sustainability assessment methods, our focus is on system-level issues studied through Material/Energy Flow Analysis (MEFA). This is a systematic approach to understanding the flows (movements) and stocks (accumulations) of material within a system, typically defined by a spatial area (such as a country) and a time period (such as a year). It gives a clearer technological understanding of the system than economic models of the economy, and the principles of conservation of mass and energy allow for checking and reconciliation of the model [3]. Essentially an MEFA is an abstract representation of a system in terms of *processes*, *stocks*, and *flows*. A process is a part of the system where material/energy is transformed, transported or stored. A stock is the accumulation of material within a process. A flow represents the transfer of material/energy between processes, or between a process within the system and the surrounding environment. The system of processes and flows can be seen as a bipartite directed graph [17], as in Fig. 1.

Once the system is defined in this way, the available data can be mapped onto the relevant parts of the system. The MEFA approach is then essentially a constrained optimisation problem to find the size of the flows, subject to the

constraints set by conservation of mass/energy and the known technical characteristics of the processes, while matching as closely as possible the known data [4]. This paper focuses on the first step: finding and querying the available data in a form that can act as an input to the subsequent model solving stage.

2.2 Use Cases and Research Problems

To guide the development of the PRObs system, we identified use cases from the literature and from needs of researchers within the *UK FIRES* project.

Use Case 1 (Data Integration Including "System Context"). It is important that resource data can be associated with its "system context" [20], so it can be linked into a MEFA system and integrated with other data. For example, government statistics on material production should not be viewed simply as a table of numbers, but each value should be associated with the region and time period for which it was measured, and explicitly linked to the edge(s) in a system diagram like Fig. 1 to which it relates. However, datasets vary in the completeness and format of this metadata. A general data model for resource data has been proposed [16] which is largely sufficient to meet these requirements. The main barrier to allow its use with Semantic Technologies is the formalisation into a proper ontology which exploits the characteristics of this data model.

Use Case 2 (Access Diverse Data in a Consistent and Flexible Structure). Different data sources classify their information in different ways, and these classifications may evolve over time. Long-term time-series data are critical to understand the dynamics of past and future resource use, so it is important to be able to convert data published in different classification systems into one consistent set of categories. Differences in the measurement units also need to be harmonised.

Even if data were already reported in fully-consistent classification systems, there is still a need to alter the structure, since some data is more detailed than is needed for modelling the system. For example, production statistics provide information on pharmaceuticals at a high level of detail which is unnecessary and should be aggregated for a model focused on high-mass materials.

To enable flexible queries at the desired level of detail to be answered, a system is needed which can take account of the hierarchical structure of processes and materials/goods classification to aggregate data as needed. Aggregation must avoid double-counting values where data already exists at different levels of the hierarchy, and deal with missing values, which occur frequently in statistics due to confidentiality concerns or other lack of coverage.

Use Case 3 (Tracking the Provenance and Uncertainty of Data). Confidence in modelling results increases when data can be validated against independent sources. Different datasets are more or less credible, depending on their source and measurement methodology. During aggregation, uncertainty may increase due to missing data or dependence on lower-quality datasets. It is important to track the providence of values returned by queries, so they can be given a suitable measure of uncertainty, and independent data for validation can be identified.

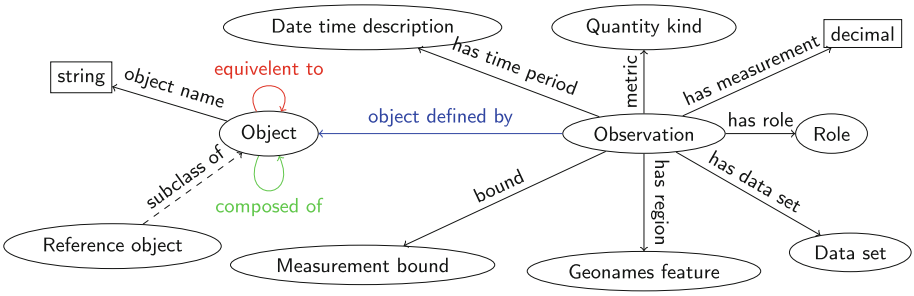

Fig. 2. The core concepts and relations in the PRObs ontology.

Use Case 4 (Streamline Usage of Semantic Web Tools for Domain Experts). The PRObs system is intended to be used to support MEFA modelling by domain experts unfamiliar with semantic web technologies. As such, they should be supported to enter information (e.g. about materials of interest and their hierarchical structure) and retrieve results without becoming experts in RDF and complex SPARQL queries. Because defining the system is subjective (a different definition could be chosen for different modelling goals), users should be supported in clearly documenting their choices. It should be possible to use the system as far as possible on typical researchers' computers without many cores and RAM, and integrate with typical modelling workflows involving e.g. Python notebooks.

The rest of this paper addresses these use cases as follows:

Use case 1**:** An ontology, building on an existing data model for the domain, for describing specific data points and their relationships (Sect. 3)
Use cases 2 & 3**:** Datalog rules/algorithms to infer new information and convert data between different classification systems (Sect. 4).
Use case 4**:** A system wrapping the RDFox implementation with Python packages to ease application by domain experts (Sect. 5).

3 The PRObs Ontology

To allow quantified data points on resource use to be expressed in RDF, we build on the data model proposed by Pauliuk et al. [16]. This describes three components of a data point: value, metadata, and "system location". The value can be a simple numerical value with associated physical units, or could account for uncertain values by defining probability distributions or bounds. The metadata includes provenance information. The system location is the component specific to MEFA: it associates the data point with its context, as in Use Case 1 above.

To represent this in RDF, we introduce the concept of an *Observation* to represent an individual data point and its value, linked to its system location (Fig. 2). We then introduce concepts describing types of materials/goods, and

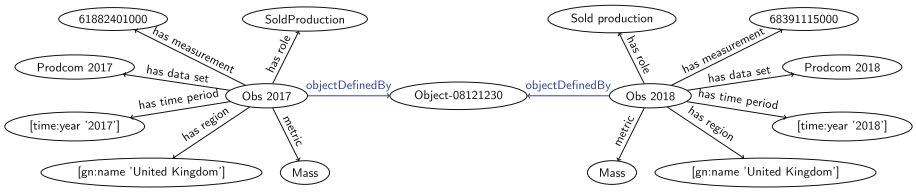

Fig. 3. Example observations representing data from the Prodcom database.

how they are related. Full details are available in Ref. [7] and the online documentation[2]. The ontology links to several external vocabularies: *PROV*[3] for data provenance, *QUDT*[4] for physical units, *Geonames*[5] for spatial regions, and *OWL-Time*[6] for time.

Example 4 (Stone, sand and gravel example). To explain the ontology, we use a small subset of a model of the UK production system as a running example. This example describes the production of "crushed stone" and "sand & gravel". To illustrate the way that data can be expressed at a coarser or finer level of detail, three sub-types of "crushed stone" are distinguished, and all these materials are collectively described as "aggregates". Two datasets are used in the example: "Prodcom" provides statistics on the production of manufactured goods, while "BGS" refers to the British Geological Survey Minerals Yearbook. Full details are available online[7]. This example features in Figs. 3, 4 and 5, described below.

3.1 Observations

An *Observation* represents a single data point. Every *Observation* is associated with the geospatial location and time period for which it was measured (defined using terms from the *Geonames* and *OWL-Time* vocabularies). Figure 3 shows how two example data points from the Prodcom database are represented, describing equivalent data for the United Kingdom recorded in different years.

The system context (i.e. the edge(s) in a system diagram like Fig. 1 to which the data relates) is defined by a *Role*, *Process* and/or *Object*. *Object* refers generically to any type of thing, including materials, goods and substances, but also non-material things that can flow through the system such as energy and services. *Process* refers to a type of activity. *Role* defines which element of the MEFA system is being measured. For simplicity all examples in this paper use the role "sold production", i.e. the total production of an *Object*.

[2] https://ukfires.github.io/probs-ontology.
[3] http://www.w3.org/TR/prov-o.
[4] http://qudt.org/2.1/vocab/citation.
[5] https://www.geonames.org/ontology.
[6] https://www.w3.org/TR/owl-time.
[7] Ref. [12], viewable at https://ukfires.github.io/probs-ISWC2021-example.

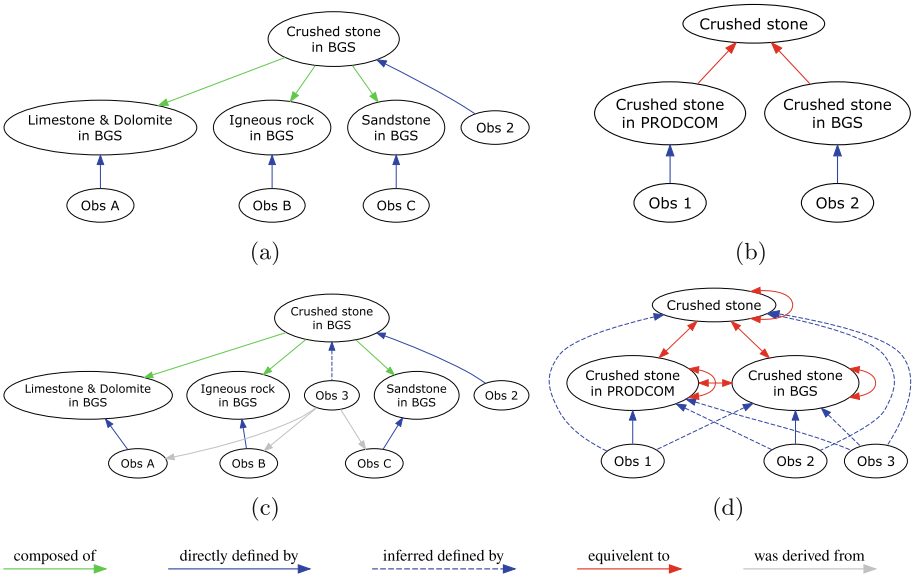

Fig. 4. Composition (a) and equivalence (b) of objects from Example 4, with only original observations shown. In (c–d) new inferred observations are included.

The way in which the data is measured is defined by the *Metric* (e.g. mass or volume), represented using the *QuantityKind* concepts from the *QUDT* vocabulary. Since conversions between alternative physical units for a given *Metric* are lossless and well-defined (e.g. to convert kilograms to tonnes), we normalise all values to a single reference unit for each metric type. The value is described by the *measurement* property. The presence of data whose value has been redacted (e.g. for confidentiality) is represented by an *Observation* with no *measurement*.

3.2 Composition and Equivalence of Objects

The next set of relations in Fig. 2 describes the relationships between *Objects*, allowing data from different sources at different levels of detail to linked.

Composition. When an *Object* can be broken down into several smaller categories, the *composite* object is linked to the *component* objects via the *objectComposedOf* object property. This relationship is stronger than simply a part-whole relationship. The *components* are implied to be *Mutually Exclusive, Collectively Exhaustive* (MECE) with respect to the *composite*; i.e. there are no other *components* of the *composite* parent which are not explicitly mentioned.

This allows *compatible* observations of the *components* to be aggregated to infer new observations for the *composite*. Observations are *compatible* if they share the same *Role*, *Region*, *Time Period*, and *Metric*. If any *components* are

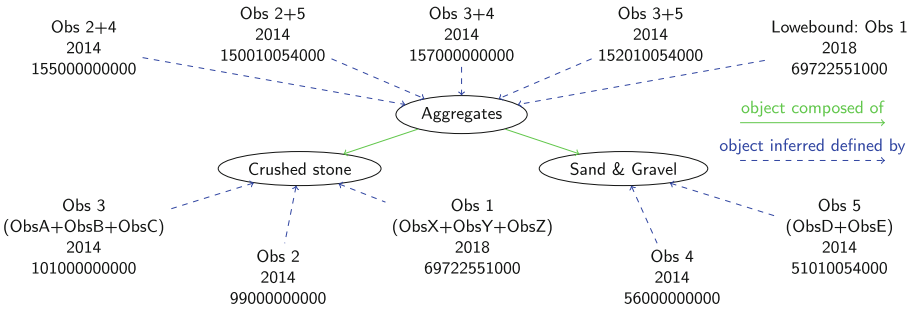

Fig. 5. Composition of *Aggregates*. The observations of *Crushed stone* are those in Fig. 4d. Those at the top arise from the different combinations of *components*. The *TimePeriod* is shown in square brackets, above the *measurement*.

missing measurement values, the result is only a lower bound, and if any *components* have multiple conflicting *compatible* observations (e.g. from independent data sources), there are multiple possible aggregated values that can be inferred.

In the running example, data on production of *Crushed stone* is reported in Prodcom as a single category, but the equivalent data in BGS is also split into three smaller categories. Figure 4a shows how the *component* and *composite Object*s are related. The three observations *Obs A*, *Obs B*, and *Obs C* are *compatible* and can be aggregated to infer a new observation (*Obs 3*) for the *composite* object *Crushed stone in BGS*, as shown in Fig. 4c. We use the relations *objectDirectlyDefinedBy* and *objectInferredDefinedBy* to denote, respectively, the observations we load directly from the datasets, and those we infer using equivalence or composition. They are subclasses of *objectDefinedBy*.

Equivalence. Different *Object* instances may be used in different datasets which in fact refer to the same type of thing. The matching instances are linked by the *objectEquivalentTo* relation, which is an equivalence relation (it is reflexive, symmetric, and transitive). When an *Observation* is linked to an *Object*, it should also be linked to any *Object* that is equivalent to the original (i.e. equivalent objects share the same observations).

In the running example, there are two dataset-specific *Object* instances for "crushed stone". To easily refer to these, a *ReferenceObject* is defined which gives the canonical representation of several equivalent individuals. In Fig. 4b, there is a *ReferenceObject* called simply *Crushed stone* which is equivalent to both the dataset specific instances. Figure 4d shows that both the original direct observations (*Obs 1* and *Obs 2*) and the inferred observation generated by composition (*Obs 3*) are propagated to the equivalent objects. In this way, the original data can be accessed via alternative terms.

Further Example of Composition and Equivalence. Figure 5 shows more complex cases of composition. The object *Crushed stone* and the object *Sand & Gravel* have 4 observations that are all *compatible* with each other (*Obs 2–5*). They are combined in all possible ways, generating 4 observations (shown

in the upper part of the figure). On the other hand, the observation *Obs 1* of the object *Crushed stone* is not *compatible* with any observation of the object *Sand & Gravel*, being defined for a different time period, so it generates a lower bound observation. If this lower bound observation is used to generate other observations, then they will also be lower bound observations.

Classification Systems. While not every dataset is linked to well-defined classification systems for *Objects*, there are several important systems in use, for example for international trade data. In these cases the classification system has been used to create the composition and equivalence relations described above.

4 Reasoning with the PRObs Ontology

The ontology described in the previous section provides a data model for *Observations*, allowing data from sources in diverse formats to be integrated together with the necessary system context (Use case 1). However, if different data points have been defined using different classification systems, they cannot yet be easily and transparently retrieved for reuse in new analyses (Use case 2 & 3). New information needs to be inferred from the raw data using rules that implement the semantics of MEFA systems. Generally, this involves converting data between different definitions of time, location, activity, and object type. In this section, we describe our approach to this, focusing specifically on converting definitions of object types, since this is the most pressing issue in the use of the system so far.

We decided to use the Datalog language with stratified negation and aggregates to perform these computations. This allows the complex behaviours required to be expressed in simple rules, while benefiting from the efficient solvers available for evaluating Datalog programs. Although more expressive/complex logic-based language exist, they are not likely to work in our scenario due to the large amount of data and the huge number of combinations that arise from their evaluation.

4.1 Equivalence and Composition

Equivalence. As described in Sect. 3, equivalent objects are linked by the `:objectEquivalentTo` object property. These objects should share the same observations (Fig. 4d). Although this may seem trivial, it has several subtleties reflected in the Datalog rule we used:

```
[?Obs, :objectInferredDefinedBy, ?O1] :-
            [?O1, :objectEquivalentTo, ?O2] ,
            FILTER(?O1 != ?O2) ,
            [?Obs, :objectDefinedBy, ?O2] ,
            NOT [?Obs, :objectDirectlyDefinedBy, ?O1] .
```

Rule set 1.1. Equivalence propagation

Propagation of observations has several advantages over duplication; for instance, it allows saving memory and to have more consistent answers. In rule 1.1 we identify the equivalent objects `?O1` and `O2` (line 2), avoiding the reflexive links (line 3), and for each observation of `?O2` (line 4), that is not a direct observation of `?O1` (line 5), we add it as a new inferred observation of `?O1`.

This rule may seem overcomplicated for the simple task of sharing the observations among equivalent objects, but it is required to avoid unwanted behaviours. Given that `:objectEquivalentTo` is an equivalence relation, negation as failure is required to avoid deriving `:objectInferredDefinedBy` relations for objects that are already defined by direct observations. Figure 4 shows an example of the correct behaviour needed in this case; a naive definition of this rule would have derived that two additional `:objectInferredDefinedBy` relations from *Obs 1* and *Obs 2* to *Crushed stone in Prodcom* and *Crushed stone in BGS* respectively.

Composition. If an object is composed of multiple *component* objects, we want to create new inferred aggregated observations derived from all combinations of *compatible* observations of the *components*, as explained in Sect. 3.2. Although this type of computation is not possible in Datalog in general, we found the peculiar characteristics of our problem do allow a solution. To confirm this, we designed and implemented an "algorithm" called PCSC, discussed below.

If each object always had only one observation, then we could have used the aggregation feature of Datalog to infer the composed observations, but, as Fig. 5 shows, in general this is not the case. Finding all possible results requires aggregating values from the Cartesian product of an unbounded number of facts, but Datalog, as most logic-based languages, does not include an operator for this.

Moreover, since what we are computing is inherently recursive, we cannot achieve it using stratified rules. Aggregation and negation-as-failure are non-monotonic extensions of Datalog [5], but a simple stratification condition ensures a monotonic behaviour. Languages with non-monotonic operators are known to be much harder to evaluate, and thus not suitable for applications involving large amounts of data that may be involved in a combinatorial explosion.[8]

PCSC "Algorithm". The main idea behind PCSC is to avoid the unboundedness over the branches of the `:objectComposedOf` relation by building a tree (T) that transposes the breadth of the composition hierarchy into the depth of T. This solves the aggregation issue mentioned in Sect. 4.1. In particular, starting from a root node that represents the *composite*, after choosing an order among its *components*, we iteratively add as children the *Observation*s of each *component*. Figure 6a shows the T constructed from the example shown in Fig. 4c.

[8] A detailed explanation of the reasons to prefer monotonic reasoning over a non-monotonic one is beyond the scope of this paper, but we want to point out that in the context of this paper we are running specific calculations over our data while non-monotonic approaches are typically designed to solve combinatorial problems.

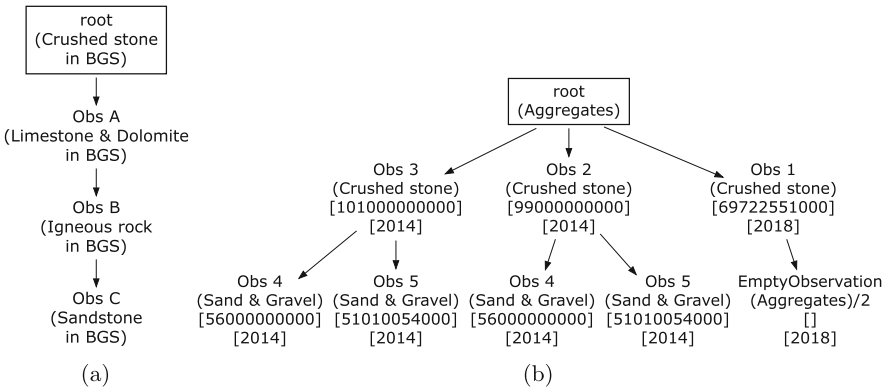

Fig. 6. Examples of trees build by the PCSC "algorithm". (a) and (b) show the trees corresponding to the examples in Fig. 4c and Fig. 5, respectively.

After building the tree T of the *composite* object O, aggregating the *measurement* in each path from the root to a leaf produces the new inferred observations of O.

To handle the case where some *components* have missing or 'not *compatible*' *Observation*s, we add an *EmptyObservation* for each 'missing node' in T. The *EmptyObservation*s do not affect the *measurement* value of the inferred observations, and are useful to identify the *lower bound* observations. The tree T constructed from the example shown in Fig. 5 is shown in Fig. 6b.

To compute the new aggregated inferred observations using stratified programs, we create a copy of all the classes and properties involved in the composition into a different named graph, and then use them to infer the information about the new observations. Because in our specific scenario we know in advance how many iterations are needed to derive all possible inferred observations, we can use a multi-step approach to derive all the inferred observations even for multi-level hierarchies of composed objects. This solves the monotonic behaviour issue mentioned in Sect. 4.1. To illustrate this, in the example we first derive *Obs3* for *Crushed stone in BGS* from its *components* (Fig. 4c), and which is then used in turn to derive the inferred observations of *Aggregates* (Fig. 5).

We designed and implemented several improvements, both from the conceptual and the technical sides, to make this "algorithm" work with a large amount of data. The complete version also derives additional relations capturing the provenance of the inferred observations. The full code can be found in the ontology repository.

5 System Implementation

Our system consists of a frontend interface for defining and documenting system definitions as input RDF data, and a backend implementation based on RDFox.

5.1 Defining and Documenting Input RDF Data

It should be possible to set up and use the PRObs system without a detailed knowledge of semantic web technologies (Use Case 4). To achieve this, we adopt a literate programming approach to produce code (RDF) and documentation (HTML) from a single source, by extending the Sphinx documentation system[9] with domain-specific extensions[10]. This allows for full documentation-writing features, including concept indices, cross-references, text formatting, and bibliographies, within Python executable notebooks.

5.2 Running RDFox to Answer Queries

The PRObs system runs RDFox scripts to load the input data and answer queries, supported by Python utilities to embed this within a testing or analysis workflow. The input data consists of system definitions in RDF as described above, with external datasets provided in the form of tabular data files and mapping scripts which are read during processing by RDFox.

RDFox was originally developed at the University of Oxford and is now being commercialised by a spin-out company, Oxford Semantic Technologies[11]. RDFox supports the RDF graph data model, the OWL 2 RL ontology language and the SPARQL query language. Rules in RDFox can be represented using a powerful extension to the Datalog language allowing, e.g. the use of much of SPARQL in rule bodies [15]. RDFox has a small memory footprint, is very efficient in its use of memory to store RDF triples, and exploits modern multi-core architectures for fast parallel reasoning. RDFox reasons by materialising all the triples implied by the data and rules, which allows for fast query answering [13]. RDFox has a scripting language which can sequentially run commands covering all features of the system[12], and exposes a REST API, which includes a SPARQL endpoint.

PRObs Ontology and RDFox Scripts. The ontology (Sect. 3) and the RDFox scripts implementing the rules and algorithm (Sect. 4) are published online (See footnote 4). To streamline use in a MEFA analysis, we have developed Python wrappers that assist with setting up the RDFox scripts to load the relevant datasets, and running RDFox as part of a wider workflow to answer queries retrieving relevant *Observations* for input to subsequent modelling and analysis steps. A utility called `rdfox_runner` provides generic support for interacting with RDFox processes[13].

Our current pipeline is shown in Fig. 7. We first run some preprocessing steps to transform the data and the ontology into a format that is more compatible with RDFox. Then we load the datasets and the ontologies, and we run the 'Conversion' phase to convert the data into RDF, enrich them with new information

[9] https://www.sphinx-doc.org.
[10] https://github.com/ricklupton/sphinx_probs_rdf.
[11] https://www.oxfordsemantic.tech.
[12] https://docs.oxfordsemantic.tech/command-line-reference.html.
[13] https://github.com/ricklupton/rdfox_runner.

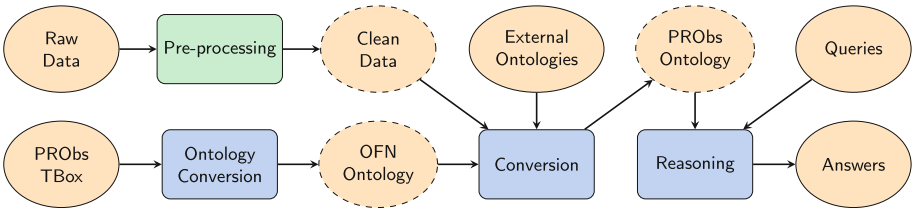

Fig. 7. Back-end pipeline. The rectangles represent the steps of our pipeline (green for Python scripts and blue for RDFox scripts). The ellipses represent inputs and outputs (dashed for internal results). (Color figure online)

(for instance, the new inferred *Observations* from equivalence and composition), and save them. Finally, in the 'Reasoning' phase the whole PRObs Ontology is loaded and a SPARQL endpoint is exposed to answer queries over it.

6 Case Study and Evaluation

To illustrate the use of the system, we describe a case study of mapping flows through the UK production system.

6.1 Case Study: UK Production System

This case study forms part of the ongoing research within the *UK FIRES* programme, motivated by seeking opportunities for innovation in manufacturing processes. The goal is to obtain a detailed understanding of how supply chains are dependent on different manufacturing processes, and where scrap is currently arising within the system, in order to quantify the benefits of innovation in different areas. To this end, a MEFA model is used to define the structure of supply chains and estimate the pattern of flows through the system which best matches the available measurements. The role of the PRObs ontology described here is to provide access to data from a diverse set of external datasets in a coherent structure aligned to the required inputs of the optimisation model.

Since there is no standard system definition of UK manufacturing supply chains at the level of technical detail required for this analysis, a major element of the project is to describe a suitable set of *Processes* and *Objects* to which the available data can be mapped, and which describe entities of relevance to the study's research questions. These are defined and documented using the system described in Sect. 5.1. Figure 1 illustrates a very small extract of the MEFA system; the whole project includes 701 processes and 617 object types. The datasets used include Prodcom and Comtrade. As mentioned in Sect. 3, a working example of a small extract of the case study system is available online (See footnote 9).

6.2 Queries

The example repository includes a set of queries which demonstrate how each of the original use cases is satisfied. For example, all data about production of a particular object can be retrieved by a query such as the following:

```
SELECT ?Value
WHERE {
    ?Observation :objectDefinedBy
                [ a :ReferenceObject ;
                :objectName "Crushed stone" ] ;
          :hasRegion [gn:name "Great Britain"] ;
          :hasTimePeriod
                [ time:unitType time:unitYear ;
                time:year "2014"^^xsd:gYear ] ;
          :hasRole :SoldProduction ;
          :metric quantitykind:Mass ;
          :measurement ?Value .
}
```

In the full case study dataset and model, we use similar queries to access data linked to the processes and objects forming the MEFA model, enabling data from different sources to be transparently and easily linked into the modelling process.

Due to the use of concepts from the ontology and the Datalog rules, the queries are straightforward, easy to read and fast to evaluate.

7 Related Work

The data model proposed by Pauliuk et al. [16] provided the starting point for the ontology described here. A key difference is that the original data model focused on describing results already in the form of a modelled, consistent MEFA system, whereas we aim to represent raw data about the system. Because of this, the PRObs ontology includes additional concepts such as the "sold production" role which do not map one-to-one to the *flows* described by the original data model. On the other hand, the original data model include some other data types such as ratios and metrics which are out of scope of the PRObs ontology.

The existing implementation of the data model does not yet aim to deal with the issues discussed here about harmonising individual data points between datasets based on composition and equivalence of object types. It allows for keyword searching, but the specified materials within each dataset are not standardised or consistent. A search for "steel" returns 75 data sets, but, for example, two sources use "iron ore" and "iron ore, in ground" respectively to mean the same thing. Our system provides a way to formally link these datasets.

Within the broader field of sustainability assessment, several efforts have been made to apply semantic web technologies for Life Cycle Assessment (LCA) in particular. Kuczenski et al. [11] describe the history of ontology development

for LCA, and present an overall ontology for LCA based on previous "ontology design patterns" [10,21]. They demonstrate how multiple LCA datasets can be catalogued and analysed using this metadata. While the ontology design patterns have elements of overlap with the ontology presented here, especially with regard to "spatio-temporal scope" of processes, their concepts are tightly bound to the LCA modelling approach. More recently, the BONSAI project [8] has been developing a broader ontology which aims to catalogue a range of datasets relevant to sustainability assessment. They acknowledge the problems of working with actual data points defined with differing terminology, but also stop short of harmonising individual data points.

8 Conclusion

We presented a novel solution to integrate and reason on different production system data using Semantic Technologies. We introduced an ontology based on a general data model for resource data, and we presented an original technique to generate new information about related objects. Finally, we provided some details about the implementation of our method and its effectiveness.

The proposed solution is the basis of the "Physical Resources Observatory", which has been developed and applied initially to support analysis within the UK FIRES research programme. However, this approach applies generally to MEFA-type analysis, and the ontology and data integration approach are currently being applied in a further project to study worldwide petrochemicals emissions. The core ontology is a starting point for more specific additions, and further development of concepts needed for flexible data visualisation is underway.

Applying Semantic Web technologies in this context has raised interesting challenges. Through the integration of existing ontologies and the adoption of logic-based approaches, we have been able to tackle a rather demanding knowledge integration and completion task and automate many of the processes that have been performed manually so far. However, we have also seen that such technologies can pose a significant obstacle for those without a specialised background. To mitigate this we developed a customised set of solutions, which we found simplifies the information retrieval process even for non-experts.

Acknowledgements. We would like to acknowledge the support of José Azevedo and Christopher Cleaver, whose work on the UK production system case study has provided essential context for the development of this work, and the Oxford Semantic Technologies team for their support.

References

1. Allwood, J.M., Ashby, M.F., Gutowski, T.G., Worrell, E.: Material efficiency: a white paper. Resour. Conserv. Recycl. **55**(3), 362–381 (2011). https://doi.org/10.1016/j.resconrec.2010.11.002

2. Arora, M., Raspall, F., Cheah, L., Silva, A.: Buildings and the circular economy: estimating urban mining, recovery and reuse potential of building components. Resour. Conserv. Recycl. **154**, 104581 (2020). https://doi.org/10.1016/j.resconrec.2019.104581
3. Brunner, P.H., Rechberger, H.: Practical Handbook of Material Flow Analysis. CRC/Lewis, Boca Raton (2004)
4. Cencic, O.: Nonlinear data reconciliation in material flow analysis with software STAN. Sustain. Environ. Res. **26**(6), 291–298 (2016). https://doi.org/10.1016/j.serj.2016.06.002
5. Ceri, S., Gottlob, G., Tanca, L.: Logic Programming and Databases. Surveys in Computer Science. Springer, Heidelberg (1990). https://www.worldcat.org/oclc/20595273
6. Committee on Climate Change: Biomass in a low-carbon economy. Technical report, CCC (2018). https://www.theccc.org.uk/wp-content/uploads/2018/11/Biomass-in-a-low-carbon-economy-CCC-2018.pdf
7. Germano, S., Saunders, C., Lupton, R.: ukfires/probs-ontology: probs-ontology v1.5.2, July 2021. https://doi.org/10.5281/zenodo.5052739
8. Ghose, A., Hose, K., Lissandrini, M., Weidema, B.P.: An open source dataset and ontology for product footprinting. In: Hitzler, P., et al. (eds.) ESWC 2019. LNCS, vol. 11762, pp. 75–79. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-32327-1_15
9. Hertwich, E., et al.: Nullius in verba: advancing data transparency in industrial ecology. J. Ind. Ecol. (2018). https://doi.org/10.1111/jiec.12738
10. Janowicz, K., et al.: A minimal ontology pattern for life cycle assessment data. In: Proceedings of the Workshop on Ontology and Semantic Web Patterns (6th Edition), Wop 2015 (2015)
11. Kuczenski, B., Davis, C.B., Rivela, B., Janowicz, K.: Semantic catalogs for life cycle assessment data. J. Clean. Prod. **137**, 1109–1117 (2016). https://doi.org/10.1016/j.jclepro.2016.07.216
12. Lupton, R., Germano, S., Saunders, C.: ukfires/probs-ISWC2021-example: initial release for ISWC2021 paper, April 2021. https://doi.org/10.5281/zenodo.5052758
13. Motik, B., Nenov, Y., Piro, R.E.F., Horrocks, I.: Incremental update of datalog materialisation: the backward/forward algorithm. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, Texas, USA, 25–30 January 2015, pp. 1560–1568. AAAI Press (2015)
14. Myers, R.J., Fishman, T., Reck, B.K., Graedel, T.E.: Unified materials information system (UMIS): an integrated material stocks and flows data structure. J. Ind. Ecolo. (2018). https://doi.org/10.1111/jiec.12730
15. Nenov, Y., Piro, R., Motik, B., Horrocks, I., Wu, Z., Banerjee, J.: RDFox: a highly-scalable RDF store. In: Arenas, M., et al. (eds.) ISWC 2015. LNCS, vol. 9367, pp. 3–20. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-25010-6_1
16. Pauliuk, S., Heeren, N., Hasan, M.M., Müller, D.B.: A general data model for socioeconomic metabolism and its implementation in an industrial ecology data commons prototype. J. Ind. Ecol. (2019). https://doi.org/10.1111/jiec.12890
17. Pauliuk, S., Majeau-Bettez, G., Müller, D.B.: A general system structure and accounting framework for socioeconomic metabolism. J. Ind. Ecol. **19**(5), 728–741 (2015). https://doi.org/10.1111/jiec.12306
18. Pauliuk, S., Majeau-Bettez, G., Müller, D.B., Hertwich, E.G.: Toward a practical ontology for socioeconomic metabolism. J. Ind. Ecol. **20**(6), 1260–1272 (2016). https://doi.org/10.1111/jiec.12386

19. Pauliuk, S., Majeau-Bettez, G., Mutel, C.L., Steubing, B., Stadler, K.: Lifting industrial ecology modeling to a new level of quality and transparency: a call for more transparent publications and a collaborative open source software framework. J. Ind. Ecol. **19**(6), 937–949 (2015). https://doi.org/10.1111/jiec.12316
20. Petavratzi, E., et al.: A roadmap towards monitoring the physical economy. Technical report, MinFuture Team, November 2018. https://minfuture.eu/downloads/D5.3_Roadmap.pdf
21. Yan, B., et al.: An ontology for specifying spatiotemporal scopes in life cycle assessment. In: Diversity++@ ISWC, pp. 25–30 (2015)