A Datalog$^+$ RuleML 1.01 Architecture for Rule-Based Data Access in Ecosystem Research

Harold Boley[1] Rolf Grütter[2] Gen Zou[1] Tara Athan[3] Sophia Etzold[2]

[1] Faculty of Computer Science, University of New Brunswick, Fredericton, Canada
{harold.boley, gen.zou}[AT]unb.ca
[2] Swiss Federal Research Institute WSL, Birmensdorf, Switzerland
{rolf.gruetter, sophia.etzold}[AT]wsl.ch
[3] Athan Services (athant.com), West Lafayette, Indiana, USA
taraathan[AT]gmail.com

Abstract. Rule-Based Data Access (RBDA) enables automated reasoning over a knowledge base (KB) as a generalized global schema for the data in local (e.g., relational or graph) databases reachable through mappings. RBDA can semantically validate, enrich, and integrate heterogeneous data sources. This paper proposes an RBDA architecture layered on Datalog$^+$ RuleML, and uses it for the ΔForest case study on the susceptibility of forests to climate change. Deliberation RuleML 1.01 was mostly motivated by Datalog customization requirements for RBDA. It includes Datalog$^+$ RuleML 1.01 as a standard XML serialization of Datalog$^+$, a superlanguage of the decidable Datalog$^\pm$. Datalog$^+$ RuleML is customized into the three Datalog extensions Datalog[$\exists$], Datalog[=], and Datalog[$\bot$] through MYNG, the RuleML Modular sYNtax confiGurator generating (Relax NG and XSD) schemas from language-feature selections. The ΔForest case study on climate change employs data derived from three main forest monitoring networks in Switzerland. The KB includes background knowledge about the study sites and design, e.g., abundant tree species groups, pure tree stands, and statistical independence among forest plots. The KB is used to rewrite queries about, e.g., the eligible plots for studying a particular species group. The mapping rules unfold our newly designed global schema to the three given local schemas, e.g. for the grade of forest management. The RBDA/ΔForest case study has shown the usefulness of our approach to Ecosystem Research for global schema design and demonstrated how automated reasoning can become key to knowledge modeling and consolidation for complex statistical data analysis.

1 Introduction

Ontology-Based Data Access (OBDA) has emerged as a major application area of Semantic Technologies for validating, enriching, and integrating heterogeneous databases (e.g., [1]). Complementary systems for Rule-Based Data Access

(RBDA) have been developed as well (e.g., [2]). For ontology-rule synergy, OBDA and RBDA have been generalized to Knowledge-Based Data Access (KBDA).[4]

While the earlier logic-database combinations, e.g. procedural Prolog-SQL interfaces, interleaved knowledge-based reasoning with data access, KBDA keeps these layers separate, using declarative mappings to bridge between the two. This way, the (higher-level) ontology and rule technologies can be advanced independently from, yet be combined with, the (lower-level) optimizations progressing for DB engines. KBDA can thus provide the urgently needed knowledge level for the growing number of data sources (e.g., about climate change) of big volume, variety, and velocity in a cost-effective manner.

KBDA builds on earlier work in knowledge-based information/data/schema integration (e.g., [3–5]). It *integrates* data complying to local (heterogeneous) schemas into data complying to a global (homogeneous) schema, usually employing Global-As-View (GAV) mappings. It also *validates* and *enriches* local-schema data with global-schema knowledge represented as ontologies or rulebases.

Some KBDA approaches use a *mediator* architecture for **query rewriting** [2, 6, 7] – corresponding to *top-down* processing and *backward* reasoning – while others use a *warehouse* architecture for **database materialization** [8] – corresponding to *bottom-up* processing and *forward* reasoning. Given that both have their advantages, we will propose a unified mediator/warehouse architecture.

KBDA KBs usually encompass rule knowledge to enrich the factual data mapped – again via rules – from the local (heterogeneous) schemas of one or more databases to a global (homogeneous) schema. Given these and other roles of rules, we will focus on RBDA in the following.

RuleML provides a family of rule (including fact) languages of customizable expressivity, a family-uniform XML format, and a suite of tools for rule processing, including the MYNG tool for generating serialization schemas in RNC and XSD. Deliberation RuleML 1.01 introduces a standard XML serialization of Datalog$^+$, a superlanguage of the decidable Datalog$^\pm$, which is being increasingly used for RBDA. Section 2 will present a unified architecture for KBDA, examine KBs and Mappings in Datalog$^+$ RuleML, and discuss relational-graph transformations for the global schema.

WSL creates knowledge and publishes data about Swiss forests, giving an integrated federal perspective on heterogeneous databases of various (e.g., geographically and thematically) specialized sources. In particular, the WSL project addressed in this work is about the susceptibility of forests to climate change [9]. Section 3 will show how this RuleML-WSL collaboration, termed ΔForest, is bringing the RBDA technologies of Section 2 to bear on WSL knowledge and databases.

[4] An overview is at http://www.cs.unb.ca/~boley/talks/RulesOBDA.pdf.

2 RBDA Technology

We will now examine RBDA technology, starting with 'the rules of OBDA' from a mediator perspective, continuing with a unification of mediator and warehouse architectures for KBDA, and then expanding on Datalog$^+$ RuleML and PSOA RuleML for our focus area of RBDA.

2.1 Kinds of Rules in KBDA

Motivated by rule-ontology synergies, we will discuss key mediator concepts of KBDA and their foundation in three kinds of (Datalog$^+$) rules, to be exemplified through the ΔForest case study in Section 3.
(1) A **conjunctive query** is a special Datalog rule whose conjunctive body can be rewritten as in (2) and unfolded as in (3), and whose n-ary head predicate instantiates the distinguished answer variables of the body predicates. OBDA ontologies beyond RDF Schema (RDFS) expressivity usually permit **negative constraints** for data validation, which are represented as Boolean conjunctive queries corresponding to RBDA integrity rules, e.g. in the extension Datalog[$\perp$] of Datalog$^+$ [10].
(2) OBDA ontologies support **query rewriting** through global-schema-level reasoning. They usually include the expressivity of RDFS, whose class and property subsumptions can be seen as single-premise Datalog rules with, respectively, unary and binary predicates, and whose remaining axioms are also definable by rules. Such ontologies often extend RDFS to the description logic DL-Lite [11] (as in OWL 2 QL [12]), including subsumption axioms that correspond to (head-)existential rules. RBDA rulebases are also being used for rewriting, e.g. via Description Logic Programs [13] (as in OWL 2 RL [12], definable in RIF-Core [14]), Datalog$^\pm$ [10], and Disjunctive Datalog [15]. We will refer to the store containing ontologies or rulebases for rewriting as the KB.
(3) KBDA data integration is centered on GAV mappings, which are safe Datalog rules for **query unfolding** of each global head predicate into a conjunction of local body predicates. These (heterogeneous) conjunctive queries can be further mapped to the database languages of the sources (e.g., to SQL or SPARQL). The store containing mappings for unfolding always is a rulebase.

2.2 A Unified Architecture for KBDA

Mediator and warehouse architectures for KBDA have often been considered in isolation. An architectural unification is achievable by using parts of the KB disjointly for mediator-style Query Rewriting [16–18] and warehouse-style DB Materialization [8], and using the Mappings reversely for mediator-style Query Unfolding and warehouse-style DB Folding. The unified architecture can thus be employed for a mediator, warehouse, and bidirectional strategy of KBDA (cf. Fig. 1), allowing for 'pluggable' domain refinements (cf. Fig. 2).

The architecture shows queries (as decorated Qs) and databases (as decorated DBs) explicitly while indicating answers (via solid triangular or diamond-shaped

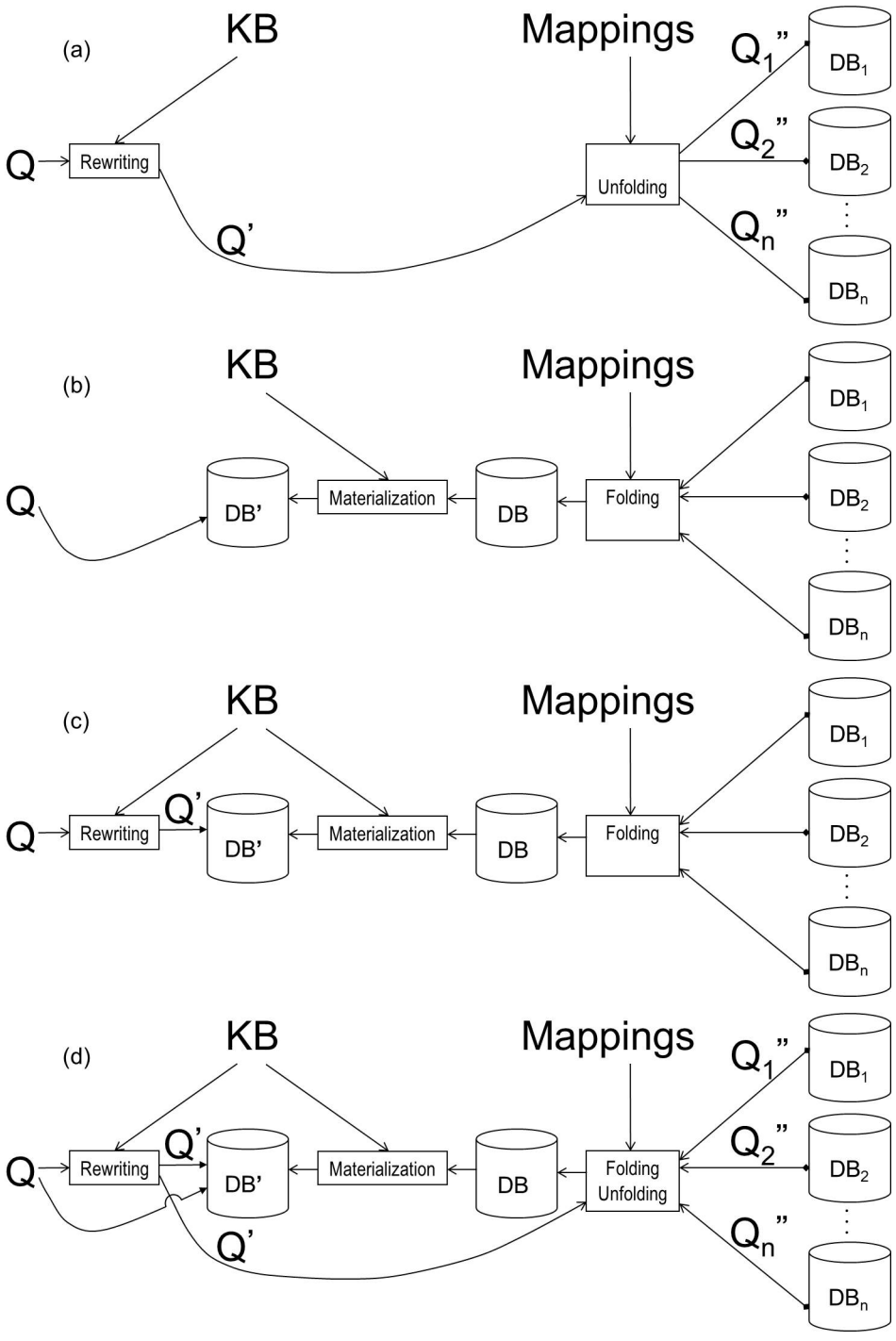

Fig. 1. From (a) mediator, (b) warehouse, and (c) bidirectional to (d) unified architecture.

arrow heads) implicitly. Each query Q_i'' targeting the local source DB_i abstracts from the relational/graph/... database level, but becomes grounded to, e.g., SQL/SPARQL[5]/... at the DB_i interface (indicated by a diamond head).

In (a), the **mediator strategy**, an incoming query Q undergoes Query Rewriting to Q' using (part or all of) the KB store. This Q' then undergoes Query Unfolding through the Folding/Unfolding transformation using the Mappings store, with results Q_1'', Q_2'', ..., Q_n''. The Q_i'' are finally grounded to (SQL/SPARQL/...) queries for the original databases DB_i, whose answers – ultimately for Q – are returned.

In (b), the **warehouse strategy**, databases DB_1, DB_2, ..., DB_n undergo database Folding through the Folding/Unfolding transformation, resulting in an integrated database DB. This DB then undergoes Database Materialization using (part or all of) the KB store, with result DB'. The original query Q is then sent to this DB', whose answers are returned.

In (c), the **bidirectional strategy**, databases DB_1, DB_2, ..., DB_n are transformed (in two steps) to a database DB' as in the warehouse strategy except that only part of the KB store is used. Independently, an incoming query Q undergoes Query Rewriting to Q' using a disjoint part of the KB store. This Q' is then sent to that DB', whose answers – ultimately for Q – are returned.[6]

The **unified strategy** (d) encompasses (a)-(c). This meets the needs of our ΔForest case study, where, e.g., R scripts materializing parts of the source data correspond to the warehouse strategy while the continuing extensions to the sources and the possible addition of new sources call for the mediator strategy, as focused in Section 3.

All strategies use the KB and the mapping store to perform (compositions of) transformations. The boundary between these stores, hence their transformations, is adjustable, both between the mediator-style transformations of Query Rewriting followed by Query Unfolding and between the warehouse-style transformations of DB Folding followed by DB Materialization. Intermediate forms can range between two normal forms. In the *KB-directed normal form* the KB store performs all deductions except atom-level local/global renamings, reserved to the mapping store. In the *mapping-directed normal form* the mapping store performs all deductions having local premises, leaving only purely global deductions to the KB store.

2.3 KB and Mappings in Datalog$^+$ RuleML

The RuleML language is based on a set of monotonic schema modules, each module providing the grammar of a syntactic feature that can be mixed-in to the language [19]. A language defined by a set of modules is always a superlanguage of a language defined by a subset of those modules, and the resulting structure

[5] Unlike the relational SQL for local data, the graph-oriented SPARQL plays an ambiguous role as a query language for local-schema data and global-schema knowledge.

[6] The two directions of the bidirectional strategy thus enable parallel processing with DB' acting as the synchronization point.

is called the RuleML language lattice. Over fifty schema modules are available, allowing for hundreds of thousands of highly customized languages tailored to specific applications, including Datalog customizations for RBDA. RuleML provides the MYNG GUI[7] as a tool for assembling an RNC schema by selecting syntactic features, as well as determining the closest lenient XSD schema for the desired sublanguage.

XML processing instructions of type "xml-model" refer to a schema that the document should validate against. This processing instruction can be used to provide an indication of the smallest RuleML sublanguage containing a RuleML document. Engines may take advantage of this information to optimize algorithms such as for rulebase transformation, query answering, and query rewriting.

Deliberation RuleML 1.01[8] introduces several new options for obtaining a more fine-grained customization of sublanguages. A small set of extensions of Datalog yields a major payoff: a standard XML serialization of Datalog$^+$, a superlanguage of the decidable Datalog$^\pm$ [10]. The highlight of Deliberation RuleML 1.01 is the ability to combine one or more of the following Datalog extensions which together define Datalog$^+$:

- Existential Rules, where the "then" part of a rule has existentially quantified variables,
- Equality Rules, where the "then" part of a rule is the "Equal" predicate, (this was already allowed in RuleML 1.0)
- Integrity Rules, where the "then" part of a rule is falsity, as a convenient way to express negative integrity constraints.

2.4 Relations and Graphs in PSOA RuleML

The two modeling paradigms of relational and graph languages can be used simultaneously in the global and local schemas of KBDA architectures.

Relational languages are used, e.g., for modeling knowledge in classical logic and data from relational databases. In these languages, a relationship among n entities becomes an n-ary predicate applied to n positional arguments. Some KBDA engines, e.g. Nyaya [7], use Datalog$^\pm$ for global relational querying. Graph languages are used, e.g., in frame logic and Semantic Web applications. In these languages, an object consists of a globally unique Object IDentifier (OID) typed by a class and described by an unordered collection of n attribute-value slots, where the value can identify an object. Other KBDA engines, e.g. Ontop [20], use SPARQL for global graph querying.

Mapping rules between the global and local schemas of the form $paradigm_1$:- $paradigm_2$ in KBDA can be within the same modeling paradigm or across the two paradigms, yielding four combinations of transformations: relational :- relational, relational :- graph, graph :- relational, and graph :- graph.

[7] http://deliberation.ruleml.org/1.01/myng
[8] http://deliberation.ruleml.org/1.01

Similarly, KB rules, which describe transformations within the global schema, can also be of the four forms. Hence, a language like PSOA RuleML [21], capable of knowledge and data modeling in both paradigms, can support the specification of these transformations. PSOA RuleML introduces positional-slotted, object-applicative (psoa) terms, which permit a relation application to have an OID – typed by the relation – and, orthogonally, its arguments to be positional or slotted. Psoa terms can be used as classical atoms without OIDs for relational modeling, and as frame atoms for graph modeling. Thus, all four kinds of transformations can be described in PSOA RuleML. In particular, graph :- relational transformations, which permit graph querying over relational databases, can be described by rules with frames in the conclusion and relations in the premise. Here, the positional argument that acts as the simple key in the relation becomes the OID of a frame, and the other positional arguments become slot values whose slot names correspond to relational column headings.

3 ΔForest Case Study

The WSL project [9] aims for an assessment of the susceptibility of forest ecosystems to the expected changing environmental conditions going along with climate change, such as temperature or precipitation. The susceptibility of a forest stand to climate change depends particularly on the change of the mortality rate. The death of single trees without a distinguishable reason and mortality of suppressed trees due to competition for nutrients or water are natural processes within the forest stand development, since only a limited number of trees can survive at one location depending on site properties, climate conditions, and tree species.

The higher the growth rate of a forest the higher is also the mortality. Accordingly, the absolute mortality is not a useful indicator to express the stand vitality. For dense forests a log-log linear relationship, called *self-thinning line*, exists for the density as number of trees per ha and the quadratic-mean tree diameter with slope and intercept (corresponding to maximum stand density) depending on tree species [22–24]. The relative mortality in a given period is defined as a shift in the self-thinning line. A change in relative mortality can then be attributed to changing environmental conditions.

The following working hypotheses are tested: (i) The relative mortality is a useful indicator for the susceptibility of forest stands to changing climatic conditions. (ii) At temperature-limited sites, increasing temperatures will increase the maximum stand density and relative mortality will decrease. (iii) At moisture-limited sites, increasing temperatures and frequency of drought events will reduce maximum stand density and relative mortality will increase.

Analysis is conducted for 285 pure and mixed forest stands in Switzerland, covering the five tree species groups of interest: beech (*Fagus sylvatica*), oak (*Quercus petraea* and *Quercus robur*), spruce (*Picea abies*), pine (*Pinus sylvestris*), fir (*Abies alba*), and several climatic regions. Data are derived from three main monitoring networks in Switzerland: yield plots (EKF) [25], monitoring of nature reserves (NWR) [26], and the Swiss Long-Term Forest Ecosystem

Research (LWF) network [27]. Data cover a time period from 1933 to 2010.

During the WSL project [9], a number of conditions have to be controlled which otherwise might impair the validity of the results. To achieve this, the following questions need to be answered, formalizations of which will be developed as queries in our ΔForest case study:

1. Are there sufficiently many eligible plots in order to perform an analysis per tree species group of interest?
2. Which eligible plots represent pure tree stands and which eligible plots represent mixed tree stands?

Re 1. To make a significant statement about how two or more variables are related, the sample size (i.e., the number of plots) must exceed a certain lower bound.

Re 2. The calculation of the self-thinning line assumes pure tree stands. Plots that represent mixed tree stands require a more complex analysis than those representing pure tree stands.

In what follows, the schema and rules for answering Questions 1 and 2 will be formalized, by ultimately mapping them to the forestry data sources.

3.1 Global Schema and KB Rules

Based on *local schemas* for the three data sources, the ΔForest *global schema* describes two kinds of predicates (cf. Fig. 2):

- External predicates of high arity (for knowledge consolidation): no dot prefix
- Internal predicates of low arity (for knowledge modeling): dot prefix

In order to construct relational global queries asking for eligible plots that represent tree stands, where a given tree species group is abundant (Question 1) or dominant (Question 2), we require the global schema to include the following external predicates (two tables of DB' in Fig. 2):

```
PlotsStatic(plot source x y altitude class)
SGAbundance(plot species-group percentage)
```

The external predicate `PlotsStatic`, which has a simple key, `plot`, can be directly transformed to frames using graph `:-` relation rules discussed in Section 2.4, hence allowing also graph querying over the global schema.

In order to model the knowledge domain of the study, we require the global schema to also contain the following internal predicates:

```
.EligiblePlot(plot)            .TreeStandAbundance(component percentage)
.IndependentPlot(plot)         .TreeStandKey(component plot species-group)
.PreEligiblePlot(plot)         .TreeStandMerged(plot species-group percentage)
.PossiblyDependentPlot(plot)   .TreeStandClass(stand class)
.LightlyManagedPlot(plot)      .PlotDistance(plot1 plot2 distance)
.ForestManagement(plot grade)  .Location(plot x y)
.PureTreeStand(component)      .Source(plot source)
.MixedTreeStand(component)     .Altitude(plot altitude)
.SpeciesGroupOfInterest(species-group)
```

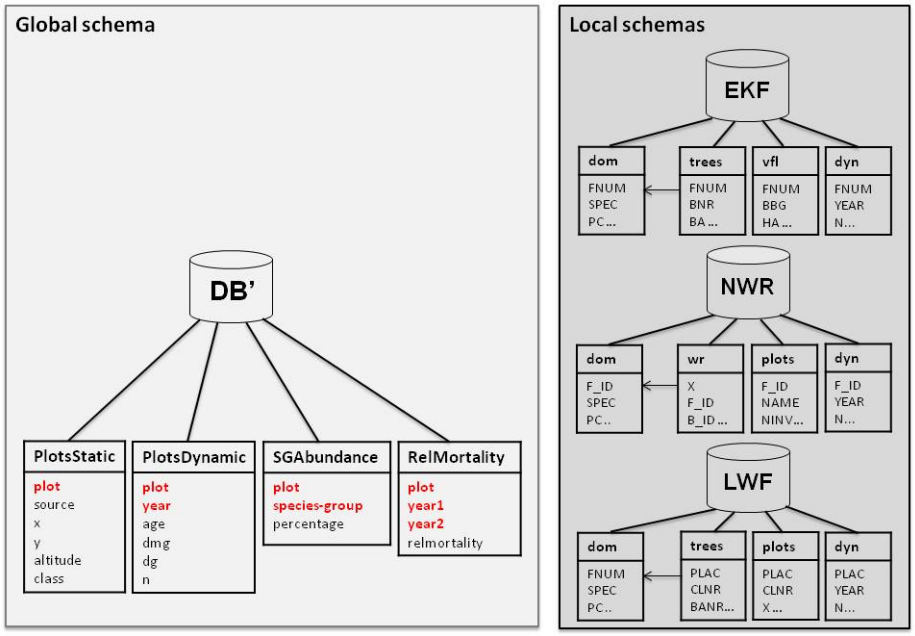

Fig. 2. Schemas of ΔForest, 'plugging' into DB', DB_1=EKF, DB_2=NWR, and DB_3=LWF of Fig. 1 (with n=3), where the KB partitioning in (c) and (d) becomes a split, e.g., between external :- internal vs. internal :- internal rules (the keys of the global schema – three being composite – are shown in bold red).

The internal predicates are transformed to the external predicates with the following consolidation (external :- internal) rules:[9]

```
PlotsStatic(?plot ?src ?x ?y ?alt ?class) :- .EligiblePlot(?plot)
                                             .Source(?plot ?src)
                                             .Location(?plot ?x ?y)
                                             .Altitude(?plot ?alt)
                                             .TreeStandClass(?plot ?class).
SGAbundance(?plot ?sg ?pct) :- .EligiblePlot(?plot)
                               .TreeStandKey(?id ?plot ?sg)
                               .TreeStandAbundance(?id ?pct)
                               ?pct>=lower.

lower = 15.
```

The external predicates have the following equality-rule key constraints.

```
?src1=?src2 ?x1=?x2 ?y1=?y2 ?alt1=?alt2 ?class1=?class2 :-
                  PlotsStatic(?plot ?src1 ?x1 ?y1 ?alt1 ?class1)
                  PlotsStatic(?plot ?src2 ?x2 ?y2 ?alt2 ?class2).
?pct1=?pct2 :-    SGAbundance(?plot ?sg ?pct1)
                  SGAbundance(?plot ?sg ?pct2).
```

Any violation of these key constraints indicates a key constraint violation in the source data.

The eligibility criteria take into account the following factors:

- The study assumes that the impact of forest management on tree mortality is negligible at the investigated sites. Nature reserves by definition prohibit all grades of forest management. Accordingly, none of the NWR plots need to be excluded from the study because of forest management. In the EKF data, forest management is graded as A, B, C, D, H, and P, where A has the lowest impact, D, H, and P the highest. In order for the study assumption to hold, forest management must not be of grade C, D, H, or P. Forest management is not recorded for LWF plots. This information must be obtained interactively by asking the respective forestry experts.
- Plots in the study must be statistically independent of each other. Plots that are located within a distance of 500 meters from each other are possibly dependent, because there is a high probability that stand characteristics are the same.
- Plots are ineligible for the study if they do not contain a time-averaged abundance greater than a threshold value of at least one of the following species groups of interest: oak, beech, spruce, pine, or fir.

The eligibility criteria are captured in the following concept-inclusion (internal :- internal) rules:

[9] In the study a number of lower bounds ranging around 15 percent are explored. Thus the global view may be considered to be parameterized by this quantity.

```
.EligiblePlot(?plot) :- .IndependentPlot(?plot)
                        .PreEligiblePlot(?plot).
.PreEligiblePlot(?plot) :- .LightlyManagedPlot(?plot)
                           .TreeStandKey(?id ?plot ?sg)
                           .PureTreeStand(?id).
.PreEligiblePlot(?plot) :- .LightlyManagedPlot(?plot)
                           .TreeStandKey(?id ?plot ?sg)
                           .MixedTreeStand(?id).
```

Additional rules among internal predicates assist in determining if the eligibility criteria are satisfied[10], and are related to each other in the following way (negative-constraint rules employ `Or()` conclusions to represent falsity – "$\bot$" of Datalog[$\bot$] – in a queryable manner):

```
.IndependentPlot(?plot) :- .Source(?plot "nwr").
.IndependentPlot(?plot) :- .Source(?plot "lwf").
.PossiblyDependentPlot(?plot1) :- .PlotDistance(?plot1 ?plot2 ?d)
                                  .PreEligiblePlot(?plot2)
                                  ?d < 500.
.IndependentPlot(?plot) :- .Source(?plot "ekf")
                           Naf(.PossiblyDependentPlot(?plot)).
.PlotDistance(?plot1 ?plot2 ?d) :- .Source(?plot1 "ekf")
                                   .Source(?plot2 "ekf")
                                   .Location(?plot1 ?x1 ?y1)
                                   .Location(?plot2 ?x2 ?y2)
                                   ?d = func:sqrt(func:pow(?x1-?x2,2) +
                                             func:pow(?y1-?y2,2))
                                   ?d > 0.
Or() :- .PlotDistance(?plot ?plot ?distance).
.LightlyManagedPlot(?plot) :- .ForestManagement(?plot "B").
.LightlyManagedPlot(?plot) :- .ForestManagement(?plot "A").
.LightlyManagedPlot(?plot) :- .ForestManagement(?plot "0").
.ForestManagement(?plot "0") :- .Source(?plot "nwr").
.PureTreeStand(?id) :- .TreeStandAbundance(?id ?pct)
                       .TreeStandKey(?id ?plot ?sg)
                       .SpeciesGroupOfInterest(?sg)
                       ?pct >= pure.
.pure = 70.
.MixedTreeStand(?id) :- .TreeStandAbundance(?id ?pct)
                        .TreeStandKey(?id ?plot ?sg)
                        .SpeciesGroupOfInterest(?sg)
                        Naf(.PureTreeStand(?id))
                        ?pct >= lower.
.TreeStandClass(?plot "pure") :- .PureTreeStand(?id)
                                 .TreeStandKey(?id ?plot ?sg).
```

[10] In our study, a stand whose dominant species group is not a group of interest may be treated as a mixed stand, while in the general ecological domain it would be considered a pure stand. We distinguish between these general and study-specific concepts of the same name through an implicit namespace.

```
.TreeStandClass(?plot "mixed") :- .MixedTreeStand(?id)
                                  .TreeStandKey(?id ?plot ?sg).

.SpeciesGroupOfInterest("oak"). .SpeciesGroupOfInterest("beech").
.SpeciesGroupOfInterest("fir"). .SpeciesGroupOfInterest("spruce").
.SpeciesGroupOfInterest("pine").
```

The following existential rule is employed to introduce a simple key, `?id`, for the predicate `.TreeStandMerged`, which has a composite key, ⟨`?plot, ?sg`⟩. The existential variable `?id` is used in predicates `.PureTreeStand` and `.MixedTreeStand` to uniquely identify a single-species-group vegetative component on a plot. It can act as an object identifier in graph representations. In the existential rule conclusion, the predicate `.TreeStandKey` associates the original composite key ⟨`?plot, ?sg`⟩ with the introduced key `?id`, and the predicate `.TreeStandAbundance` replaces the composite key of `.TreeStandMerged` with the new key `?id`.

```
Exists ?id (.TreeStandKey(?id ?plot ?sg) .TreeStandAbundance(?id ?pct)) :-
  .TreeStandMerged(?plot ?sg ?pct).
```

3.2 Query Processing and Mappings

Question 1. In order to answer the first question, the **trees** tables of EKF, NWR, and LWF (not shown) are preprocessed using the statistical package R.[11] Preprocessing results in three instances of the table `dom(?plot ?sg ?pct)`, where `?pct` is the percentage, based on the basal area, of a tree species group (argument `?sg`) on a plot (argument `?plot`).

The following rules map the *local schemas* of EKF, NWR, and LWF to the *internal global predicates*, employing the KB-directed normal form except for merging oak species into a single species group and adding their percentages:

```
.TreeStandMerged(?plot "beech" ?pct) :- EKF.dom(?plot "Fagus sylvatica" ?pct).
.TreeStandMerged(?plot "beech" ?pct) :- NWR.dom(?plot "Fagus sylvatica" ?pct).
.TreeStandMerged(?plot "beech" ?pct) :- LWF.dom(?plot "Fagus sylvatica" ?pct).
.TreeStandMerged(?plot "oak" ?pct) :- EKF.dom(?plot "Quercus petraea" ?pct1)
                                      EKF.dom(?plot "Quercus robur" ?pct2)
                                      ?pct = ?pct1 + ?pct2.
.TreeStandMerged(?plot "oak" ?pct) :- NWR.dom(?plot "Quercus petraea" ?pct1)
                                      NWR.dom(?plot "Quercus robur" ?pct2)
                                      ?pct = ?pct1 + ?pct2.
.TreeStandMerged(?plot "oak" ?pct) :- LWF.dom(?plot "Quercus petraea" ?pct1)
                                      LWF.dom(?plot "Quercus robur" ?pct2)
                                      ?pct = ?pct1+?pct2.
```

The rules for spruce, pine and fir, not shown, are similar to those for beech. Additional plot characteristics are mapped as follows (`.Location :- NWR` mapping not shown):

[11] http://www.r-project.org/

```
.Location(?plot ?X ?Y) :- EKF.vfl(?plot ?BBG ?area ?Z ?X ?Y ?ORT ?GDE ?KT).
.Location(?plot ?X ?Y) :- LWF.Plots(?PLAC ?plot ?X ?Y ?Z ?LAT ?LON ?area).
.ForestManagement(?plot ?grade) :-
  EKF.trees(?plot ?BNR ?BA ?AJ ?grade ?AHC ?D1 ?D2 ?V7 ?SOZ ?HGEM ?HBER).
```

Question 1 for oaks is rephrased in terms of eligible plots representing tree stands where oaks are abundant, i.e., above the lower bound for the kinds of tree stand considered. This is formalized as a query using the external predicate `SGAbundance`:

```
q(?plot) :- SGAbundance(?plot "oak" ?pct).
```

In order to expand the query, the `SGAbundance`-headed KB rule and the fact regarding the value of `lower` are used to rewrite `q` as follows:

```
q(?plot) :- .EligiblePlot(?plot)
            .TreeStandKey(?id ?plot "oak")
            .TreeStandAbundance(?id ?pct)
            ?pct >= 15.
```

and then, using the existential rule, we obtain:

```
q(?plot) :- .EligiblePlot(?plot)
            .TreeStandMerged(?plot "oak" ?pct)
            ?pct >= 15.
```

This conjunctive query may be split as follows:

```
q(?plot)  :- q1(?plot) q2(?plot).
q1(?plot) :- .EligiblePlot(?plot).
q2(?plot) :- .TreeStandMerged(?plot "oak" ?pct)
             ?pct >= 15.
```

The query `q2` is *unfolded* using the mapping rules introduced above.

```
q2(?plot) :- EKF.dom(?plot "Quercus petraea" ?pct1)
             EKF.dom(?plot "Quercus robur" ?pct2)
             ?pct1+?pct2 >= 15.
q2(?plot) :- NWR.dom(?plot "Quercus petraea" ?pct1)
             NWR.dom(?plot "Quercus robur" ?pct2)
             ?pct1+?pct2 >= 15.
q2(?plot) :- LWF.dom(?plot "Quercus petraea" ?pct1)
             LWF.dom(?plot "Quercus robur" ?pct2)
             ?pct1+?pct2 >= 15.
```

The full rewriting of `q1` is not detailed here for space reasons. Partial database materialization, e.g. for `.PlotDistance`, would improve the efficiency of the query processing. On the other hand, full materialization of `.EligiblePlot` is not reasonable because the extension of this class is dependent on the value of the `lower` parameter, so a different materialization would be needed for each parameter value. Hence, the unified RBDA strategy explained in Section 2.2, which combines rewriting and materialization, fits the needs of the study.

Question 2. The second question is formalized with two queries using the external predicate `PlotsStatic`:

```
qPure(?plot) :- PlotsStatic(?plot ?src ?x1 ?y1 ?alt1 "pure" ).
qMixed(?plot) :- PlotsStatic(?plot ?src ?x2 ?y2 ?alt2 "mixed").
```

Query rewriting and unfolding work in a way similar to Question 1 except that abundance is compared to a bound of 70 (percent) using constant `.pure`. Eligible plots with abundance of a species group of interest above this value represent pure tree stands; the remaining eligible plots represent mixed tree stands.

4 Conclusions

In this paper, OBDA is complemented by Rule-Based Data Access (RBDA) and generalized to Knowledge-Based Data Access (KBDA). RBDA is founded on three kinds of rules: Query rules (including integrity rules), KB rules (for query rewriting and DB materialization), as well as mapping rules (for query unfolding and DB folding). A unified KBDA architecture is presented with mediator, warehouse, and bidirectional data-access strategies. Datalog$^+$ RuleML 1.01 is used for customizing rule expressivity, XML-based rule serialization, and platform-independent rule processing.

The ΔForest study applies RuleML techniques to real-world RBDA by formalizing two questions of a WSL project on ecosystems facing climate change. This case study has already shown the usefulness of our approach to Ecosystem Research, e.g. for the project's global schema design, and demonstrated how automated reasoning can become key to knowledge modeling and consolidation for complex statistical data analysis.

In the context of the open RBDA/ΔForest collaboration between RuleML and WSL, various avenues for future work are being explored, described as part of the RBDA wiki page.[12] Implementations of the specified architecture can reuse the (open source) KBDA technology referenced in this paper and the wiki page. In particular, relevant KBDA efficiency techniques [28] could be adapted to ΔForest. Moreover, our RBDA architecture could be applied to other areas of Ecosystem Research such as oceanography (ΔOcean). Finally, while our current RBDA focus is on Data Querying (RBDQ), Reaction RuleML 1.0[13] can also express updates as needed for Data Management (RBDM).

The RuleML blog[14] can contribute in bringing together the communities in Datalog$^\pm$, RuleML 1.x, RBDA, and Ecosystem Research.

[12] http://wiki.ruleml.org/index.php/Rule-Based_Data_Access
[13] http://wiki.ruleml.org/index.php/Specification_of_Reaction_RuleML_1.0
[14] http://blog.ruleml.org/

References

1. Calvanese, D., Giese, M., Haase, P., Horrocks, I., Hubauer, T., Ioannidis, Y.E., Jiménez-Ruiz, E., Kharlamov, E., Kllapi, H., Klüwer, J.W., Koubarakis, M., Lamparter, S., Möller, R., Neuenstadt, C., Nordtveit, T., Özçep, Ö.L., Rodriguez-Muro, M., Roshchin, M., Savo, D.F., Schmidt, M., Soylu, A., Waaler, A., Zheleznyakov, D.: Optique: OBDA solution for big data. In Cimiano, P., Fernández, M., Lopez, V., Schlobach, S., Völker, J., eds.: ESWC (Satellite Events). Volume 7955 of Lecture Notes in Computer Science., Springer (2013) 293–295
2. Baget, J.F., Croitoru, M., da Silva, B.P.L.: ALASKA for ontology based data access. In Cimiano, P., Fernández, M., Lopez, V., Schlobach, S., Völker, J., eds.: The Semantic Web: ESWC 2013 Satellite Events. Volume 7955 of Lecture Notes in Computer Science. (July 2013)
3. Kühn, E., Puntigam, F., Elmagarmid, A.K.: Multidatabase transaction and query processing in logic. In Elmagarmid, A.K., ed.: Database Transaction Models for Advanced Applications, Morgan Kaufmann Publishers (1991)
4. Lakshmanan, L.V.S., Sadri, F., Subramanian, I.N.: On the logical foundations of schema integration and evolution in heterogeneous database systems. In Ceri, S., Tanaka, K., Tsur, S., eds.: Deductive and Object-Oriented Databases. Volume 760 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (1993) 81–100
5. Bassiliades, N., Vlahavas, L., Elmagarmid, A.K., Houstis, E.N.: InterBase-KB: Integrating a knowledge base system with a multidatabase system for data warehousing. IEEE Transactions on Knowledge and Data Engineering **15**(5) (Sept 2003) 1188–1205
6. Calvanese, D., Giacomo, G.D., Lembo, D., Lenzerini, M., Poggi, A., Rodriguez-Muro, M., Rosati, R., Ruzzi, M., Savo, D.F.: The MASTRO system for ontology-based data access. Semantic Web Journal **2**(1) (2011) 43–53
7. De Virgilio, R., Orsi, G., Tanca, L., Torlone, R.: NYAYA: A system supporting the uniform management of large sets of semantic data. In: IEEE 28th International Conference on Data Engineering. (April 2012) 1309–1312
8. Motik, B., Nenov, Y., Piro, R., Horrocks, I.: Parallel materialisation of Datalog programs in centralised, main-memory RDF systems. To appear in AAAI 2014.
9. Rigling, A., Zingg, A.: Relative mortalität als indikator für die sensitivität von waldbeständen. WSL Projekt, Bew-Pin 201104N0134
10. Calì, A., Gottlob, G., Lukasiewicz, T.: A general Datalog-based framework for tractable query answering over ontologies. Journal of Web Semantics **14** (July 2012) 57–83
11. Calvanese, D., Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: Tractable reasoning and efficient query answering in description logics: The DL-Lite family. Journal of Automated Reasoning **39**(3) (2007) 385–429
12. Motik, B., Cuenca Grau, B., Horrocks, I., Wu, Z., Fokoue, A., Lutz, C.: OWL 2 web ontology language profiles (October 2009) W3C Recommendation. http://www.w3.org/TR/owl2-profiles/.
13. Grosof, B.N., Horrocks, I., Volz, R., Decker, S.: Description logic programs: Combining logic programs with description logic. In: Proceedings of the 12th International Conference on World Wide Web. WWW'03 (2003) 48–57
14. Boley, H., Hallmark, G., Kifer, M., Paschke, A., Polleres, A., Reynolds, D.: RIF core dialect (February 2013) W3C Recommendation. http://www.w3.org/TR/rif-core/.

15. Eiter, T., Gottlob, G., Mannila, H.: Disjunctive Datalog. ACM Trans. Database Syst. **22**(3) (September 1997) 364–418
16. Gottlob, G., Orsi, G., Pieris, A.: Ontological queries: Rewriting and optimization. In Abiteboul, S., Böhm, K., Koch, C., Tan, K.L., eds.: Proceedings of the 27th International Conference on Data Engineering, ICDE 2011, April 11-16, 2011, Hannover, Germany, IEEE Computer Society (2011) 2–13
17. Calvanese, D., De Giacomo, G., Lenzerini, M., Vardi, M.Y.: Query processing under GLAV mappings for relational and graph databases. Proc. of the VLDB Endowment **6**(2) (2012) 61–72
18. Cuenca Grau, B., Motik, B., Stoilos, G., Horrocks, I.: Computing Datalog rewritings beyond Horn ontologies. In: Proc. of the 23rd Int. Joint Conf. on Artificial Intelligence (IJCAI 2013). (2013)
19. Athan, T., Boley, H.: Design and implementation of highly modular schemas for XML: Customization of RuleML in Relax NG. In Olken, F., Palmirani, M., Sottara, D., eds.: Rule - Based Modeling and Computing on the Semantic Web. Volume 7018 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2011) 17–32
20. Rodríguez-Muro, M., Kontchakov, R., Zakharyaschev, M.: Ontology-based data access: Ontop of databases. In Alani, H., Kagal, L., Fokoue, A., Groth, P., Biemann, C., Parreira, J., Aroyo, L., Noy, N., Welty, C., Janowicz, K., eds.: The 12th International Semantic Web Conference (ISWC 2013). Volume 8218 of Lecture Notes in Computer Science. (2013) 558–573
21. Boley, H.: A RIF-Style Semantics for RuleML-Integrated Positional-Slotted, Object-Applicative Rules. In: Proc. 5th International Symposium on Rules: Research Based and Industry Focused (RuleML-2011 Europe), Barcelona, Spain. Lecture Notes in Computer Science, Springer (July 2011) 194–211
22. Pretzsch, H., Biber, P.: A Re-evaluation of Reineke's rule and stand density index. For. Sci. **51** (2005) 304–320
23. Reineke, L.: Perfecting a stand density index for even-aged forests. J. Agric. Res. **46** (1933) 627–638
24. Schütz, J.P., Zingg, A.: Improving estimations of maximal stand density by combining Reineke's size-density rule and yield level, using the example of spruce (Picea abies (L.) Karst.) and European Beech (Fagus sylvatica L.). Ann. For. Sci. **67** (2010)
25. Zingg, A., Bachofen, H.: Wachstumsforschung an der WSL. Schweizer Wald **134**(9) (1998) 15–23
26. Brang, P., Commarmot, B., Rohrer, L., Bugmann, H.: Monitoringkonzept für Naturwaldreservate in der Schweiz. Eidg. Forschungsanstalt für Wald, Schnee und Landschaft WSL; ETH Zürich, Professur für Waldökologie, Birmensdorf, Zürich (February 2008) Available from: http://www.wsl.ch/publikationen/pdf/8555.pdf.
27. Dobbertin, M., Kindermann, G., Neumann, M.: Analysis of forest growth data on intensive monitoring plots. In Fischer, R., Lortenz, M., eds.: Forest Condition in Europe: Technical Report of ICP Forests and FutMon. Institute for World Forestry, Hamburg (2011) 115–127
28. Bak, J., Brzykcy, G., Jedrzejek, C.: Extended rules in knowledge-based data access. In Olken, F., Palmirani, M., Sottara, D., eds.: Rule-Based Modeling and Computing on the Semantic Web, 5th International Symposium, RuleML 2011- America, Ft. Lauderdale, FL, Florida, USA, November 3-5, 2011. Proceedings. Volume 7018 of Lecture Notes in Computer Science., Springer (2011) 112–127