BOOTOX: Bootstrapping OWL 2 ontologies and R2RML mappings from Relational Databases*

Ernesto Jiménez-Ruiz[1] Evgeny Kharlamov[1] Dmitriy Zheleznyakov[1] Ian Horrocks[1]
Christoph Pinkel[2] Martin G. Skjæveland[3] Evgenij Thorstensen[3] Jose Mora[4]

[1] University of Oxford; [2] fluid Operations; [3] University of Oslo; [4] Sapienza University of Rome

Abstract. In this demo paper we present BOOTOX, a system facilitating ontology and mapping development by their automatic extraction (i.e., bootstrapping) from relational databases. BOOTOX has a number of advantages: it allows to control the OWL 2 profile of the output ontologies, and to bootstrap complex and provenance mappings, which are beyond the W3C direct mapping specification. Moreover, BOOTOX allows to import pre-existing ontologies.

1 Motivation

The main idea behind exposing relational data via an ontology is to provide the user with access to the data store via a domain specific vocabulary of classes and properties that the user is familiar with. This vocabulary is related to the database schema via view definitions, called *mappings*; thus, technical details on how the data is actually stored in the database are hidden from end-users. Developing appropriate ontologies and mappings for a given relational database is a challenging and time consuming task. To aid this process, tools that can extract a preliminary ontology and mappings from database schemata play a critical role. In the literature one can find a broad range of approaches to bootstrap an ontology and mappings from a relational database. The interested reader may have a look, e.g., at the survey [7].

Our BOOTOX system [3] is (not) yet another ontology and mapping bootstrapper. BOOTOX, unlike existing approaches, allows to define different "profiles" depending on the application scenario and the required Semantic Web technologies. For example, if the bootstrapped ontology is to be used in a so-called Ontology Based Data Access (OBDA) scenario where the ontology provides a virtual access layer to the data, OWL 2 QL will be chosen as the ontology language as it is required by OBDA query processing techniques. If the data is materialised, one could opt for other OWL 2 profiles[1] depending on the used query processing engine. BOOTOX also allows to import domain ontologies, which will be integrated to the bootstrapped one via alignment [2] or directly mapped to the database [5]. BOOTOX bootstraps W3C R2RML[2] mappings following the W3C direct mapping directives[3] to connect the ontological vocabulary to the relational database; moreover, it offers a suit of advanced techniques for bootstrapping R2RML mappings that are beyond the direct ones. Furthermore, it extends the bootstrapped mappings with provenance information[4] that can be used to explain users where the database origin of answers computed by queries formulated over ontologies.

* Funded by the EU project Optique, and the EPSRC projects MaSI[3], Score! and DBOnto.

[1] OWL 2 profiles: `http://www.w3.org/TR/owl2-profiles/`

[2] R2RML language: `http://www.w3.org/TR/r2rml/`

[3] W3C direct mappings: `http://www.w3.org/TR/rdb-direct-mapping/`

[4] Based on the W3C recommendation PROV-O `http://www.w3.org/TR/prov-o/`.

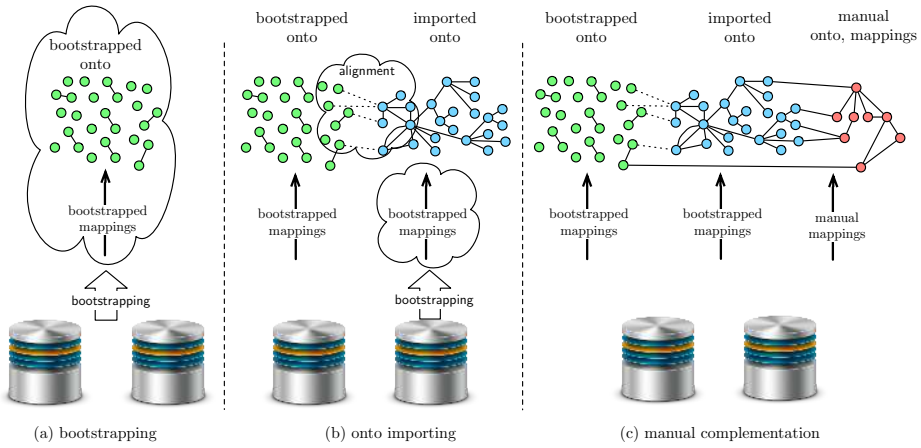

Fig. 1. General workflow of BOOTOX

In this demo paper we present BOOTOX and the attendees will be able to experience the system in several real world and synthetic preconfigured scenarios: MusicBrainz from the music domain, RODI benchmark databases that cover several domains, and our relational database from an enterprise domain that is specifically designed to show the benefits of BOOTOX. The attendees will be able to bootstrap ontologies and mappings, merge them with external ontologies, and verify quality of bootstrapped assets.

2 Ontology and Mapping generation in BOOTOX

A general workflow of BOOTOX is in Figure 1: the user first extracts an initial ontology and mapping from the input databases, and then can import and ontology, and finally complement the resulting ontologies and mappings manually. We now give a very brief overview of these steps implemented in BOOTOX and refer to [3] for details.

Ontology bootstrapping. The basic algorithm of BOOTOX creates ontological vocabulary from a relational schema by translating *(i)* each (non-binary) table into an OWL class; *(ii)* each attribute not involved in a foreign key into an OWL datatype property; *(iii)* each foreign key into an OWL object property. Advanced algorithms discover complex queries over input databases and associate them to classes and properties. For bootstrapping of axioms BOOTOX relies on a series of patterns to transform database features into OWL 2 axioms and puts special attemption to the OWL 2 profile of the generated ontology.

Mapping bootstrapping. BOOTOX follows the W3C direct mapping guidelines mappings generation and relies on the R2RML language to encode direct mappings. Intuitively, each R2RML mapping relates a valid SQL query to either a class or a property. Besides direct mappings, that involve only projection operator in their SQL part, BOOTOX can help user to build *complex* mappings, that may also include selection and/or join operators. Furthermore, BOOTOX automatically extends direct mappings with metainformation about provenance.

Ontology importing. BOOTOX also allows to import domain ontologies, by integrating them with, e.g., bootstrapped ontologies either *(i)* via ontology alignment or *(ii)* by directly mapping them to the database. For the ontology alignment we extended the LogMap [2] system: BOOTOX gives special care to unwanted logical consequences from aligned ontologies that may lead to unexpected answers [9].

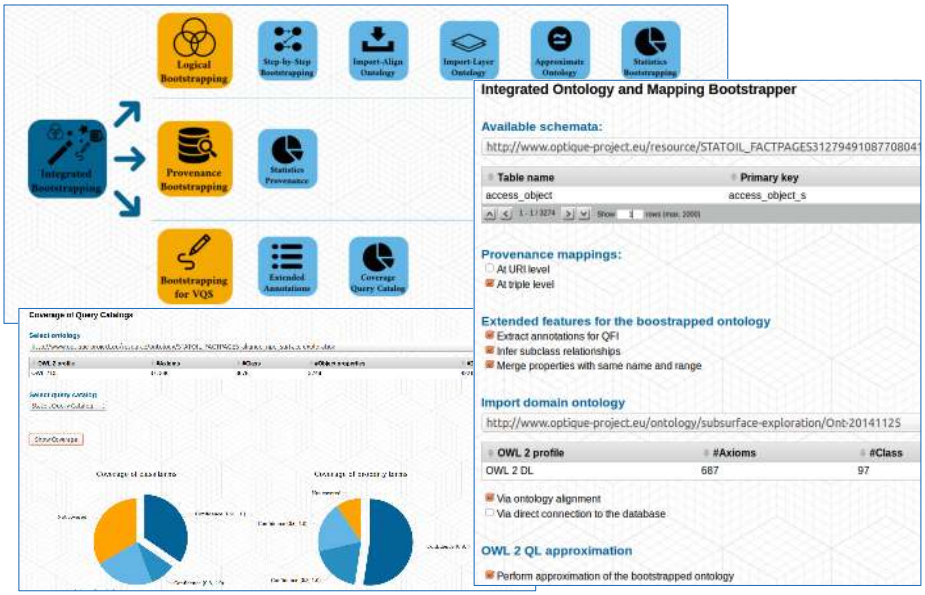

Fig. 2. Screenshots of BOOTOX

Provenance bootstrapping. BOOTOX extends direct mappings with metainformation about provenance.The provenance metainformation is modelled in the mapping assertion, adding, for instance, the source database from which the information is extracted, and more granular information, like table and column identifiers. Currently, provenance in BOOTOX comes into three different granularity levels: *URI*, *triple* and *graph*.

Bootstrapping for query formulation. A representative axiom for a database feature may not be always available in a required OWL 2 profile. In these cases, BOOTOX keeps this knowledge as OWL 2 annotation axioms. These axioms have no logical impact on the ontology, but they can potentially be used in user interfaces to guide the formulation of queries (e.g. in the OptiqueVQS [10, 11]).

Quality evaluation. BOOTOX evaluates the bootstrapped assets with respect to 4 metrics: *(i)* compliance with OWL 2 or one of its profiles, *(ii)* compliance with a standard mapping language like R2RML, *(iii)* suitability of the ontology vocabulary to formulate the queries that the user is interested in, and *(iv)* suitability of the ontology and mappings to enable the answering of queries.

3 BOOTOX demonstration

BootOX is currently integrated within the Optique's project platform [1, 4] and it can be tested via its public demonstrator.[5] However, in the close future, we plan to implement a standalone version of BootOX.[6]

Figure 2 shows an overview of the current BOOTOX related interfaces. The main menu presents to the user the available options currently supported in BOOTOX: automatic bootstrapper, guided bootstrapper, ontology alignment, provenance bootstrapping, bootstrapping related statistics, etc. The second screenshot shows the integrated bootstrapping form in BOOTOX. The bottom screenshot shows the coverage of a selected query catalog by the vocabulary of a bootstrapped ontology.

[5] Optique's public demonstrator: `http://optique-northwind.fluidops.net/`

[6] Check `http://www.cs.ox.ac.uk/isg/tools/BootOX/` for updates.

In the demonstration we will allow attendees to analyze the different techniques implemented in BOOTOX using several databases. The demonstration will put special attention to the following points:

(i) Bootstrapping of ontologies in different OWL 2 profiles.
(ii) Analysis of the impact of advanced bootstrapping features.
(iii) Importing of domain ontologies via ontology alignment.
(iv) Mapping domain ontologies directly to the relational database.
(v) Creation of provenance mappings.
(vi) Coverage of the vocabulary of a query catalog by the bootstrapped ontology.
(vii) Query formulation and answering using the bootstrapped ontology and mappings.

We will demonstrate BOOTOX over the scenarios listed below. Additionally, we will allow attendees to test BOOTOX with their own databases.

Northwind DB. is a demo database with easy-to-understand business data comprising customers, products, orders, employees, etc.[7]

NPD FactPages [8] is a database that is heavily used in the oil and gas industry. It consists of 70 tables, 276 different attributes, 96 foreign keys.

MusicBrainz. is an open music encyclopedia that contains information about 830,000 artists, 1.2 million releases, and 13.2 million recordings.[8]

RODI [6] is a benchmark designed to test relational-to-ontology mappings end-to-end. RODI currently provides several scenarios in the conference domain.

BOOTOX *database.* We prepared a database in order to show all benefits of BOOTOX.

4 References

[1] M. Giese, A. Soylu, G. Vega-Gorgojo, et al. Optique — Zooming In on Big Data Access. In: *Computer* 48.3 (2015).
[2] E. Jiménez-Ruiz, B. Cuenca Grau, Y. Zhou, et al. Large-Scale Interactive Ontology Matching: Algorithms and Implementation. In: *ECAI*. 2012.
[3] E. Jiménez-Ruiz, E. Kharlamov, D. Zheleznyakov, et al. BOOTOX: Practical Mapping of RDBs to OWL 2. In: *ISWC*. 2015.
[4] E. Kharlamov, E. Jiménez-Ruiz, D. Zheleznyakov, et al. Optique: Towards OBDA Systems for Industry. In: *ESWC (Selected papers)*. 2013.
[5] C. Pinkel, C. Binnig, E. Kharlamov, et al. IncMap: Pay as You Go Matching of Relational Schemata to OWL Ontologies. In: *OM*. 2013.
[6] C. Pinkel, C. Binnig, E. Jimenez-Ruiz, et al. RODI: A Benchmark for Automatic Mapping Generation in Relational-to-Ontology Data Integration. In: *ESWC*. 2015.
[7] J. Sequeda, S. H. Tirmizi, Ó. Corcho, et al. Survey of Directly Mapping SQL Databases to the Semantic Web. In: *Knowledge Eng. Review* 26.4 (2011).
[8] M. G. Skjæveland, E. Lian, and I. Horrocks. Publishing the NPD FactPages as Semantic Web Data. In: *ISWC*. 2013.
[9] A. Solimando, E. Jiménez-Ruiz, and G. Guerrini. Detecting and Correcting Conservativity Principle Violations in Ontology-to-Ontology Mappings. In: *ISWC*. 2014.
[10] A. Soylu, M. Giese, E. Jimenez-Ruiz, et al. Experiencing OptiqueVQS: A Multi-Paradigm and Ontology-Based Visual Query System for End Users. In: *UAIS Journal* (2015).
[11] A. Soylu, M. G. Skjaeveland, M. Giese, et al. A Preliminary Approach on Ontology-Based Visual Query Formulation for Big Data. In: *MTSR*. 2013.

[7] Northwind DB: `http://northwinddatabase.codeplex.com/`
[8] MusicBrainz:`http://musicbrainz.org/`