D2RCrime: A Tool for Helping to Publish Crime Reports on the Web from Relational Data

Júlio Tavares
University of Fortaleza - UNIFOR
Fortaleza/CE, Brazil
julio.at@gmail.com

Vasco Furtado
University of Fortaleza - UNIFOR
Fortaleza/CE, Brazil
furtado.vasco@gmail.com

Henrique Santos
University of Fortaleza - UNIFOR
Fortaleza/CE, Brazil
hensantos@gmail.com

Eurico Vasconcelos
University of Fortaleza - UNIFOR
Fortaleza/CE, Brazil
euricovasconcelos@gmail.com

***Abstract*—In the Law Enforcement context, more and more data about crime occurrences are becoming available to the general public. For an effective use of open data, it is desirable that the different sources of information follow a pattern, which allows reliable comparisons. In addition, it is expected that the task of creating a correspondence between the pattern and the internal representations of each source of information is not a steep learning curve. These two conditions are hardly found in the actual stage, where open data about crime occurrences refer to the data disclosed by each police department in its own way. This paper proposes an interactive tool, called D2RCrime, that assists the designer/DBA of relational crime databases to make the correspondence between the relational data and the classes and properties of a crime ontology. The ontology plays the role of a pattern to represent the concepts of crime and report of crime, and is also the interface to publish on-the-fly relational crime data. This correspondence allows the automatic generation of mapping rules between the two representations, what allows for access to relational data from SPARQL. An evaluation of D2RCrime is done with DBA/system analysts who used the tool for establishing correspondences between relational data and the ontology.**

***Index Terms*—Internet, Semantic Web, Knowledge Engineering, Law Enforcement, Open Government.**

I. Introduction

The culture of participation and collaboration on the Web could not leave out the public sector. New forms of relationships between citizens and governments are also emerging from a new attitude on the tract of government information and public service on the Internet. This new approach, understood here as Government 2.0 (while complying with the Web 2.0), relies on governments as open platforms to provide information [1].

In the Law Enforcement context, more and more data about crime occurrences are becoming available to the general public. In the U.S. and Britain in particular, police departments quickly realized that they should open data to encourage participation by the population. For an effective use of open information, it is desirable that the different sources of information follow a pattern, which allows, for instance, making reliable comparisons. Here, when we mention a pattern, we refer to a language with the power to represent information about both the provenance and the meaning of the concepts that should be available. Moreover, it is expected that the task of creating a correspondence between the pattern and the internal representations of each source of information is not a steep learning curve. These two conditions are hardly found in the actual stage in the context of opening data about crime occurrences. The usual process is each police department to define its own way to disclose its data by creating intermediary representations (typically spreadsheets[1]) that must constantly be updated. Alternatively, the police departments develop their own APIs[2] that are characterized by their specificity. In brief, each department spends time and resources to define its own way to disclose its data.

This paper proposes a method to guide the process of opening crime data that aims to mitigate the aforementioned problems. This method relies on ontologies for representing the concepts of crime and crime report. The *crime* ontology defines the basic concepts and properties used in the context of Law Enforcement to define a crime occurrence. The *crime report* ontology defines the basic information necessary to characterize the report of a crime occurrence such as the source of the report, the date and time of the report, its description, and so on.

We have designed an interactive tool that assists the designer/DBA to make the correspondence between the relational data and the classes and properties of the crime ontology. This correspondence allows us to automatically generate the mapping rules between the two representations, which conducts the process of accessing relational data from SPARQL. Unlike the majority of approaches that replicate the relational data into another repository, we based our proposal

[1] See http://www.atlantapd.org/crimedatadownloads.aspx in Atlanta
[2] See http://sanfrancisco.crimespotting.org/api for San Francisco

on the D2R Server [2]. D2R is a system for publishing relational data on the Web. The D2R Server enables Resource Description Framework (RDF) and HTML browsers to navigate the content of non-RDF databases, and allows applications to query a database using the SPARQL query language over the SPARQL protocol. This approach relieves the data owner of concerns about the integrity and consistency of the replicated data. Finally, an evaluation of D2RCrime is done with DBA/system analysts who used the tool for establishing correspondences between relational data and the ontology.

II. REPRESENTING CRIME REPORTS

Two ontologies are at the core of our proposal. They intend to represent the concepts of crime and report of crime. Our representation of crime is not restricted to the information that nowadays has been disclosed by police departments worldwide. However some information is mandatory to define a unique instance. A crime has at least a type, a date and time (imported from the time ontology [3], a precise address (geographical coordinates), and a description. Information about the people involved such as the perpetrator(s), the witnesses and the victim(s) may also be inserted, but it is not mandatory.

The crime ontology is basically a hierarchy for inferential purposes. It was modeled so that it is possible to map the various classifications of crime type. We define the crime events as specializations of the Event class, from the Event Ontology [4]. According to the Event Ontology, "an event is an arbitrary classification of a space/time region, by a cognitive agent. An event may have a location, a time, active agents, factors and products." To describe where a crime occurred geographically, we use the ontology wgs84[3] to express location in terms of latitude and longitude.

Typically, a detailed identification of the people involved is not open information due to privacy concerns. However, this varies according to different countries, sources and cultures. In Brazil, for instance, the media naturally discloses homicide victims. In the US, raw crime data does not include the victim's name.

We defined a crime ontology inspired by the Criminal Act Ontology in the context of the OpenCyC Project, and also took into consideration the FBI Uniform Crime Report[4] standard. The report of crime refers to a particular crime and has information about the reporting itself. The identification of the reporter, the time and date of the report, and links to external sources are examples of this kind of information. As a report of crime contains basic provenance information, in order to represent these latter features, we imported the Provenance Model Language 2 (PML2) ontology [5]. Even though the Open Provenance Model (OPM) [6] and its Open Provenance Model Ontology (OPMO) are becoming widely used for provenance exchange, we have chosen to use PML2 because it includes classes and properties to represent the trustworthiness of the sources and credibility of the information. These properties are important because our ultimate goal is to combine crime open data from a large variety of sources that sometimes can even be anonymous. The *CrimeReport* class is a subclass of *pmlp:Information*. We have also used some specific properties to describe a report, such as *pmlp:hasCreationDateTime* (hour of the report), *pmlp:hasDescription* (text of the report), and pmlp:hasSource (entity that published the report).

The complete ontology is described in [15]. Figure 1 shows a piece of this ontology describing a particular crime (homicide). This is the most refined level of detail that we have proposed. Doing so, we aim to keep the tradeoff between simplicity and generality while providing good coverage.

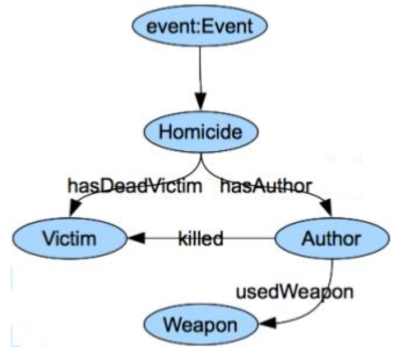

Fig. 1. Piece of the crime ontology for the description of homicide

III. ASSISTING THE MAP BETWEEN RELATIONAL DATA AND THE CRIME ONTOLOGY

The definition of a language to be used as a pattern for opening data on criminal incidents is only the first step of the proposed method. Patterns require community acceptance, therefore a key aspect is how friendly the use of the pattern is. Thus it is essential that the correspondence between information represented in the pattern and information represented in the databases of the police departments be easily established. In this section we describe how the proposed method seeks to accomplish this. It relies on two assumptions i) as crime data are originally stored in relational databases, the Web publication thereof should not require data replication, and ii) the task of associating the original data with the ontology should not require learning another programming language.

A. Publishing Relational Data on the Web

To achieve the first requirement, we have chosen to base our method on systems that map relational data to RDF on-demand such as Asio Semantic Bridge for Relational Databases[5], D2R[6] [2], SquirrelRDF[7], and UltraWrap[8] [7]. In these methods, an application (typically a Web server) takes requests from the Web and rewrites them to SQL queries. This on-the-fly translation allows the content of large

[3] http://www.w3.org/2003/01/geo/
[4] http://www.fbi.gov/about-us/cjis/ucr/ucr

[5] http://www.bbn.com/technology/knowledge/asio_sbrd
[6] http://www4.wiwiss.fu-berlin.de/bizer/d2r-server/
[7] http://jena.sf.net/SquirrelRDF
[8] http://www.cs.utexas.edu/~miranker/studentWeb/UltrawrapHomePage.html

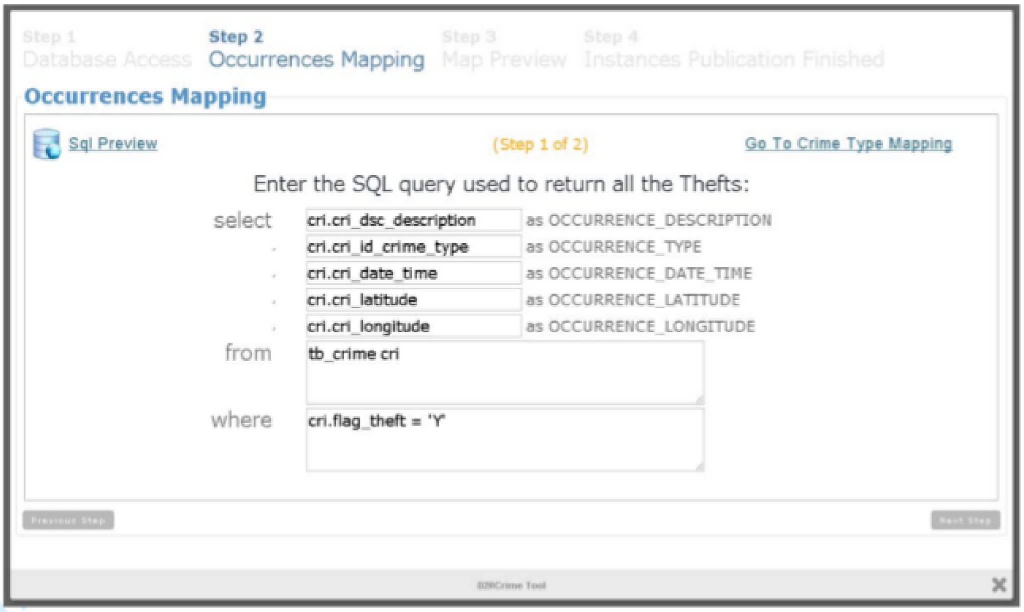

Fig. 2. Example of a SELECT clause to define the concept of THEFT

databases to be accessed with acceptable response times without requiring data replication.

The World Wide Web Consortium (W3C) has recognized the importance of mapping relational data to the Semantic Web by starting the RDB2RDF incubator group (XG) to investigate the need for standardization. In particular, we have chosen to use an approach based on the D2R server. D2R is an open and free system for publishing relational data on the Web. It enables RDF and HTML browsers to navigate the content of non-RDF databases, and allows applications to query a database using the SPARQL query language over the SPARQL protocol.

The operation of D2R is through the interpretation and execution of rules, described in the Data to Relational Query language (D2RQ [8]), for mapping the equivalence between an ontology and a relational database.

D2RQ consists of a mapping language between relational database schema and RDFS/OWL ontologies. The D2RQ platform creates an RDF view of the relational database, which can be accessed through Jena, Sesame, and the SPARQL query language. D2RQ's main elements are *ClassMap* and *PropertyBridge*. The *ClassMaps* represent the classes of an ontology and associates them with a table or a view of a database. The *PropertyBridges* are linked to one or more *ClassMaps* and are mainly used to connect the columns in a table with the properties (attributes) present in an ontology. Usually, they are filled with literal values, but can also make references to URIs that designate other resources.

With *PropertyBridges* it is possible to specify conditional restrictions that can be used to filter a specific domain or range of information. Using the Join structure, it is also possible to specify the mapping between multiple tables and a class or a property in the ontology. Another quite usual feature is the *TranslationTable* structure, which allows 1 to n mapping (table to classes).

The performance of more complex mappings, whereby it may be necessary to access a Web service or to use conditional structures and external sources of data, can be made through the *javaClass* structure, which allows the use of Java classes to perform the mapping.

In practice, it is very difficult to implement mapping just with simple correspondences like one-to-one table to classes. There is often the need to handle more complex structures, including the javaClass, which requires an effort that the designer is not always able to make. For instance, a tuple of a table that describes crime data must be mapped into instances of different classes such as robbery, theft, homicide, etc. Our idea then was to provide a tool that facilitates this process of mapping to the case of criminal data.

B. The D2RCrime Tool

D2RCrime provides resources to support the publication of reports of crimes in RDF, from relational databases. In particular, the goal is to help designers and/or DBA who do not have extensive knowledge in semantic technologies. The ontology of crimes described above is used to guide an interactive process with a designer/DBA. The basic premise is that D2RCrime mapping between the ontology classes and the database tables can be obtained interactively by asking the designer to write SQL queries for retrieving tuples from the database that describe a particular class (or property) of the ontology. The aim is thus to use a language largely dominated by designers/DBA and allows them to easily describe the concepts represented in the ontology of crimes. Figure 2 shows an example of how this dialog occurs in D2RCrime.

It asks the designer to complete a SELECT clause to retrieve all the thefts from the database of crime occurrences (tb-crime in the Figure). The tool also asks that the response contain the date, time, location and description of each theft. For each SELECT clause made by a designer/DBA, D2RCrime transforms the query into an N3 rule. The process is iterative and new questions will be carried out until all the classes and properties of the ontology have been described in terms of SELECT clauses. At the end of the process, the entire mapping is performed using D2RQ and therefore can be executed on the D2R Server. Frame 1 illustrates the mapping between tables and classes. The crime report and theft classes are mapped there.

D2RCrime transforms the SQL into D2RQ elements. To do this, the following mapping is done: Aiming to accelerate the elicitation of the requirements for the mapping, D2RCrime identifies which database field is associated with the type of crime. It then proposes a customized interface in which it is possible to associate the values of crime type with the corresponding ontology classes.

```
// CrimeReport - In the ClassMap below
   it is defined that the instances are
   generated with the class
   "crime:CrimeReport"

map:CrimeReport a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:uriPattern "crimereport/
     @@tb_cri_crime.CRI_IDCRIME@@";
  d2rq:class crime:CrimeReport;
  d2rq:classDefinitionLabel "CrimeReport";
  map:CrimeReport__label a
  d2rq:PropertyBridge;

  d2rq:belongsToClassMap map:CrimeReport;
  d2rq:property rdfs:label;
  d2rq:pattern "CrimeReport
  #@@tb_cri_crime.CRI_IDCRIME@@";

// Theft [OCURRENCE_TYPE] -
   In the ClassMap below, it is defined
   that the instances are generated with
   the class "crime:Theft".
   Note the d2rq:condition for
   selecting the adequate type of crime

map:Theft a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:uriPattern "Theft/@@tb_cri_crime.
     CRI_IDCRIME@@";
  d2rq:class crime:Theft;
  d2rq:condition "tb_cri_crime.
    tcr_idtipo_crime=1 or
    tb_cri_crime.tcr_idtipo_crime=4";
  d2rq:classDefinitionLabel "Theft";

map:Theft__label a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:Theft;
  d2rq:property rdfs:label;
  d2rq:pattern "Theft #@@tb_cri_crime.
     CRI_IDCRIME@@";
```

Frame 1. Example of the code in D2RQ generated by D2RCrime

During the dialogue process, D2RCrime offers the possibility for the designer to see how the instances of the classes (crime reports) have been built. A widget to plot crimes on the spot where they occurred shows the values of each report. Figure 3 shows an example of this.

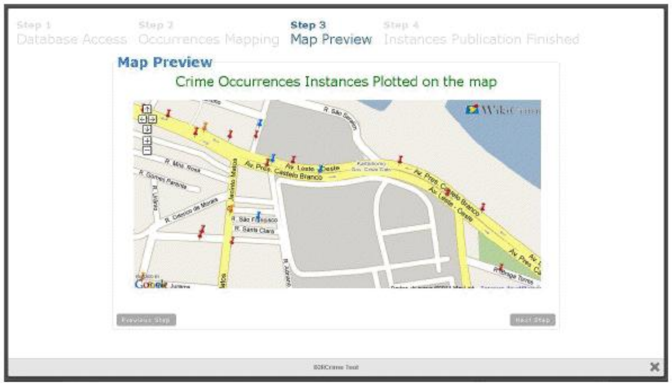

Fig 3 Preview of the instances of crime reports plotted in the map

IV. EVALUATION

Our approach proposes a new method of mapping between relational databases and structured data in RDF. We are not aware of similar tools or approaches that are able to perform the RDF2RDF mapping intuitively using SQL clauses. Because of this, we had difficulty choosing what would be the most appropriate way to validate our hypothesis for the comparison and experiments. To alleviate this issue, we decided to compare D2RCrime with the D2RServer tool itself, which automates the generation of D2RQ code for mapping the relational data into RDF.

In order to analyze the hypotheses raised in this paper, an empirical study was conducted aimed at assessing: 1) the representational power of the proposed ontology to represent criminal events; 2) whether the task of creating correspondence by means of the proposed tool is not actually a “steep learning curve” and whether the tool is user friendly and intuitive, enabling and facilitating the proposed mapping process.

A. Methodology

The study was conducted in two stages. In the first stage, a battery of tests of “translation” of information on crimes was conducted in the laboratory, based on the proposed ontology. The battery was based on non-probabilistic and intentional samples (50 each) from police agencies. The choice of samples was based on two factors: the requirement that the police agencies have their information about crimes published, and the interest in evaluating the ontology in different countries (criminal law) and in different languages.

In the second stage, tests were conducted with users to analyze whether the D2RCrime tool softens the "steep learning curve" found in the data-opening process. For such, a sample of 10 users — 5 analysts and five DBAs, all with experience in DBMSs and SQL language — were invited to publish data on crimes in two sessions.

The first session used the D2RCrime tool in conjunction with the proposed ontology. The second session was conducted without introducing the tool, encouraging users to perform the publication without support of the tool. To do so, we used the automatic mapping generation resource (generate-mapping) available in the D2RServer software. This procedure automatically generates a mapping file expressed in D2RQ language, which reflects the structure of the relational database to be mapped.

All the users who took part in the tests had good knowledge on SQL language and little or no knowledge on semantic technologies, representing the scenario usually found in an IT staff. The proposed method takes this fact into account, utilizing the System Analysts' and DBAs' prior knowledge in SQL and not exposing them to the need to learn the set of tools required for publishing content on the Semantic Web.

As a methodology for performing the test, users were exposed to a document with different data models, which were aimed at representing the tables related to the storage of criminal occurrences. Thus, different data modeling was distributed among the user groups, so that there would be a significant representation of the main scenarios found in the databases of police departments. The use of different models was aimed at assessing the generality of this approach. The following performance factors were used for the tests conducted:

1) Success in the mapping activities, which indicates whether it was possible to complete the mapping test within the allotted time (30 minutes);

2) RDF Mapping, which reflects the quantity of concepts and properties of the ontology that were successfully mapped to RDF for those users who finished the tasks (item 1);

3) Correctness of the generated vocabulary, which reflects whether the published data met the main concepts described in the ontology;

4) Autonomy which is the number of users that have finished the activities without human guidance at the time (only with the specification of the activity).

B. Results: Ontology Coverage

As mentioned before, the proposed crime ontology was based on the current initiatives of open crime data. For the purpose of evaluating the completeness of the ontology coverage, we compared the concepts represented therein with four samples of crime datasets in different countries: Oakland, US; FBI, US; London, UK; and Fortaleza, BR. A table describing the main concepts used in this comparison is available at http://www.wikicrimes.org/ontology/table.htm. In general the main concepts were correctly mapped. Most of the types of reports open to the public refer to crimes against property (robbery, thefts, burglary, etc.) and crimes against life (murder, attempted murder, etc.). Problematic cases refer to types of crimes that are generic, such as "anti-social behavior" or "disturbing the peace." Typically this involves several types of crimes that differ from country to country. In US, for instance, prostitution is a crime that could be classified as anti-social behavior. In Brazil, prostitution is not crime. We decide not to drill down in each one of these cases; we created the generic classes to represent them.

C. Results: User Interaction

Figure 4a shows the results obtained from the tests, in which D2RCrime was used according to the indicators outlined in Section IV.A. Figure 4b shows the results for the case in which the D2R tool was used.

Taking into account that the users had no prior knowledge in the use of the tool or semantic technologies, the tests showed that the tool is a viable alternative to easily provide for the opening of data. This strengthened our hypothesis that the use of the SQL metaphor is a good heuristic for the success of the method. The high percentage obtained in the "RDF mapping" and "Correctness of vocabulary" indicators can be used to demonstrate the effectiveness of the method. During the experiments, it was also proven that this approach obtained good acceptance due to the fact that it is not necessary to invest time in semantic technologies/tools that are often not of direct interest to such users.

Regarding the "the number of activities done in the time constraint" indicator, we found that each concept of the ontology was mapped, with the aid of the tool, taking one minute on average. It was also perceived that the process of mapping the last concepts was always performed faster than mapping the initial concepts: after mapping the first concepts, the users acquire the minimum experience in the tool, enough to perform the subsequent tasks even more quickly.

Regarding the "RDF mapping" indicator, there were slight indications of mapping and usability failures. In one of the tests, the tool did not properly format a string informed by the user for the "date" field, causing the respective property of the ontology not to be mapped successfully. The "date" field is more prone to situations such as this, because several SQL functions are applied thereto (e.g.: substring) to format the data.

In order to make a comparative analysis, we conducted the same test with other users, but this time using a different methodology. We chose to use the tool provided by the D2R itself, where — given a relational database — the automated mapping functionality (generate mapping) is responsible for generating the mapping file starting from the structure of a relational database. In order to do so, the tool generates an RDF vocabulary according to the database, taking into account the table names as the ontology class names and the table columns as the ontology properties. The following aspects drove the choice of the D2R tool:

1) Independence of paid license;

2) Ease of use;

3) Availability on the market;

4) Ability to be used in a 30-minute test without the need for special infrastructure.

Approaches such as the Asio Semantic Bridge for Relational Databases — ASBRD[9], SquirrelRDF[10], and RDBToOnto [9] are methods that are close to our approach, but require a considerable learning curve, due largely to the need for specific configurations and the need to manipulate the mapping file manually. Tools such as Oracle Semantic Technologies and the ASIO SBRD itself require paid software licenses.

As the methodology for conducting this second phase of testing, a document containing the information needed to perform the installation of D2R Server software was made available to the users, as well as the procedures to generate the automatic mapping of the relational database and test whether the publication of the data was successful. Before beginning the tests, the basic operation of the D2RQ mapping file was explained to the users, detailing its main structures and compulsory components (*ClassMaps* and *PropertyBridges*). After these procedures, the users then began the tasks related to publication of the data.

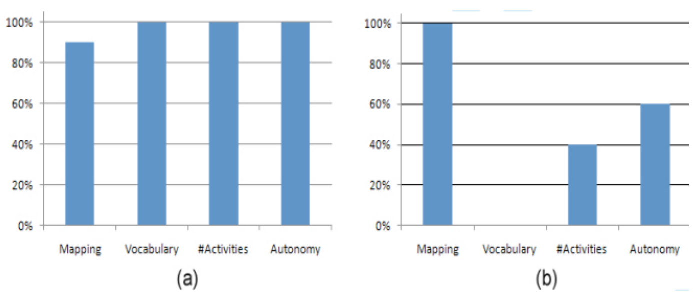

Fig. 4. Results of the evaluation (a) with the use of D2RCrime and (b) with the D2R standard tool

Figure 4b reflects the results of the testing, according to the same aforementioned indicators. The “RDF mapping” (100%) demonstrates that the approach is stable and is able to perform the mapping of the various types of data among the tables and columns involved. The “Correctness of vocabulary” indicator, however, got a very low percentage (0%). This is obviously due to the fact that using only the D2R, the classes and fields of the ontology cannot be generated. The D2R tool generates its own vocabulary created in an ad hoc way. This reflects a common fragility found in automated mapping approaches: although the data are mapped to RDF, in order for them to be able to actually represent the local domain and its respective relationships to be mapped, the mapping device must undergo a series of customizations to relate the generated instances efficiently.

The “the number of activities done in the time constraint” indicator (40%) shows that not all tests could be completed in the stipulated time. This is due to the fact that users had to learn how to configure the D2RServer software in order for the automatic mapping to be generated, confirming the fact that — even for a task that is simple to perform — a higher learning/difficulty curve is already shown to be present for the completion of the mapping tasks due to the need to learn about semantic tools.

D. Discussion

As a general result, the data obtained showed the proposed method as a viable alternative to easily provide for the opening of data on the Semantic Web. The D2RCrime tool is shown to be an effective alternative to lessen the steep learning curve required in this process.

It is important to stress that the automatic mapping generated by the D2R Server software does not provide integration with standardized ontologies accepted by the community (e.g.: GeoNames, Time, PMLP, Sioc, etc.), which somewhat hinders the context of data integration and reuse of information. Using the D2RCrime tool, the data are published using a proposed ontology that foresees this entire scenario of integration/mash-up of information.

It is also important to highlight that in order for semantic applications to be integrated more deeply to the published data, it’s necessary to replace the vocabulary generated automatically with RDF vocabularies that are standardized, accepted by the community, widely known, and publicly accessible. The generated mapping can be freely edited. However, in order to do so, the user must have all of the knowledge about how the mapping method and syntax work.

V. RELATED WORK

Metatomix’s Semantic Platform[11] and RDBtoOnto[12] [9] are examples of automatic tools that generate a populated ontology in RDF. In the case of the first, the mapping is done through a graphical eclipse plugin. Other structured sources can map to the same ontology allowing data integration under the same ontology. DB2OWL [10] automatically generates ontologies from database schemas, but it does not populate the ontology with instances. The mapping process is performed from the detection of particular cases for conceptual elements in the database, then the conversion is realized through the mappings from these components present in the database to their counterparts in the ontology.

Triplify [11] is a lightweight plug-in that exposes relational database data as RDF and Linked Data on the Web. There is no SPARQL support. The desired data to be exposed is defined in a series of SQL queries. Triplify is written only in PHP but has been adapted to several popular web applications (WordPress, Joomla, osCommerce, etc.).

ODEMapster[13] is a plugin for the NeOn toolkit, which provides a GUI to manage mappings between the relational database and RDFS/OWL ontologies. The mappings are expressed in the R2O language.

---

[9] http://www.bbn.com/technology/knowledge/asio_sbrd
[10] http://jena.sourceforge.net/SquirrelRDF
[11] http://www.metatomix.com
[12] http://www.tao-project.eu/researchanddevelopment/demosanddownloads/RDBToOnto.html
[13] http://neon-toolkit.org/wiki/ODEMapster

Asios' SBRD (Semantic Bridge for Relational Databases) enables integration of relational databases to the Semantic Web by allowing SPARQL queries over the relational database. An initially OWL ontology is generated from the database schema, which can then be mapped to a defined domain OWL ontology. The refinement of the ontology is done by means of Snoogle [12]. Snoogle converts the initial mappings to SWRL/RDF or SWRL/XML. It also allows two ontologies to be viewed on screen and then the correspondence between their classes can be generated, as well as attributes thereof. This whole process of mapping is accomplished via a visual interface.

This two-step approach followed by Asio requires a significant effort by the user compared with the approach we have proposed. For non-experts, it requires learning of two sets of tools. SquirrelRDF8 is a tool that allows relational databases to be queried using SPARQL. This tool takes a simplistic approach by not performing any complex model mapping like D2RQ. One of the most significant limitations of this approach is that it is not possible to use SPARQL queries searching for properties.

VI. CONCLUSION

In this paper we have described a method that relies on the representation of ontologies as a pattern to represent the concepts of crime and report of crimes. Besides a pattern, the ontologies are the interface to publish relational crime data on-the-fly. We have also proposed an interactive tool, called D2RCrime, which assists the designer/DBA to make the correspondence between the relational data and the classes and properties of the crime ontology. This correspondence allows automatic generation of the mapping rules between the two representations that conduct the process of access of relational data from SPARQL.

Open issues persist and will drive our future research. Open data may come from different sources. It will be necessary to have mechanisms to compare and check whether the information refers to the same fact. Creating mechanisms to automatically identify these repetitions is a challenge to be pursued. Another challenge, also due to the fact that information comes from different sources, is the need to account for the credibility of information automatically. When sources are known, such as official sources, the attribution of credibility is natural. However, the credibility of non-official information sources is difficult to be assigned. Methods for computing reputation and trustworthiness of the sources as in [13] [14] are examples of how this can be addressed.

Finally it is important to point out that the main advantage of having open crime data is the possibility that it will be used to provide services to citizens. Examples of this are alerts about how dangerous a certain place is and suggestions of safe routes. Such information can be enriched with data coming from popular participation, for example, via collaborative mapping. An example of collaborative mapping in Law Enforcement is WikiCrimes[14] [13]. WikiCrimes aims to offer a common interaction space among the public in general, so that people are able to report criminal facts as well as keep track of the locations where such crimes occur. We have integrated D2RCrime to WikiCrimes in which the instances retrieved by WikiCrimes from the Police Department's relational databases via D2RCrime are plotted directly on the digital map (for further details see [15]). Doing so, a set of services provided by WikiCrimes is available to the citizens. It is possible to receive alerts about dangerous places and to receive alerts by email as well. Apps for running on iPhones and Android smartphones also exist.

[14] http://www.wikicrimes.org

ACKNOWLEDGMENT

This work was supported in part by the CNPq under Grants 55977/2010-7 and 304347/2011-6 .

REFERENCES

[1] D. Lathrop, L. Ruma, "Open government: Collaboration, transparency, and participation in practice", in O'Reilly Media, 2010.

[2] C. Bizer, R. Cyganiak, "D2R Server - Publishing Relational Databases on the Semantic Web", in Poster at the 5th International Semantic Web Conference, 2006.

[3] J.R. Hobbs, F. Pan, "An ontology of time for the semantic web". In ACM Transactions on Asian Language Information Processing (TALIP), 66–85, ISSN 1530-0226.

[4] Y. Raimond, S. A. Abdallah, "The event ontology", 2006. Available: http://purl.org/NET/c4dm/event.owl.

[5] D. McGuinness, L. Ding, P. Pinheiro da Silva, C. Chang, "PML2: A Modular Explanation Interlingua", in Proceedings of the AAAI 2007 Workshop on Explanation-Aware Computing, Vancouver, British Columbia, Canada, July 22-23, 2007.

[6] L. Moreau, B. Clifford, J. Freire, J. Futrelle, Y. Gil, P. Groth, N. Kwas-nikowska, S. Miles, P. Missier, J. Myers, B. Plale, Y. Simmhan, E. Stephan, and J. Van Den Bussche, "The Open Provenance Model — Core Specification (v1.1)", in Future Generation Computer Systems, 2010.

[7] J. F. Sequeda, R. Depena, D. Miranker, "Ultrawrap: Using SQL Views for RDB2RDF", in Poster at the 8th International Semantic Web Conference (ISWC2009). Washington DC, US, 2009.

[8] C. Bizer, A. Seaborne, "The D2RQ Platform v0.7 - Treating Non-RDF Relational Databases as Virtual RDF Graphs". Available: http://www4.wiwiss.fu-berlin.de/bizer/d2rq/spec/20090810.

[9] F. Cerbah, "RDBToOnto: un logiciel dédié à l'apprentissage d'ontologies à partir de bases de données relationnelles", Strasbourg, 2009.

[10] N. Cullot, R. Ghawi, K. Yétongnon, "DB2OWL: A Tool for Automatic Database-to-Ontology", in Proceedings of the 15th Italian Symposium on Advanced Database Systems (SEBD), 2007.

[11] S. Auer, "Triplify – Light-Weight Linked Data Publication from Relational Databases", in Proceedings of the 18th World Wide Web Conference (WWW2009).

[12] H. Wang, C. Tan, Q. Li, "Snoogle: A search engine for the physical world", in IEEE Infocom, 2008.

[13] V. Furtado, L. Ayres, L. de Oliveira, M. Vasconcelos, C. Caminha, J. D'Orleans, J, "Collective Intelligence in the Law

Enforcement: The WikiCrimes System", in Information Science, 2010.

[14] I. Pinyol, J. Sabater-Mir, G. Cuni, "How to talk about reputation using a common ontology: From definition to implementation", in Proceedings of the Ninth Workshop on Trust in Agent Societies, Hawaii, USA. pp: 90-101. 2007.

[15] . J. Tavares, V. Furtado, H. Santos, "Open Government in Law Enforcement: Assisting the publication of Crime Occurrences in RDF from Relational Data", in AAAI Fall Symposium on Open Government Knowledge: AI Opportunities and Challenges, Arlington, VA, 2011.