# Connecting the Smithsonian American Art Museum to the Linked Data Cloud

Pedro Szekely[1], Craig A. Knoblock[1], Fengyu Yang[2], Xuming Zhu[1], Eleanor E. Fink[1], Rachel Allen[3], and Georgina Goodlander[3]

[1] University of Southern California, Los Angeles, California, USA
{pszekely,knoblock}@isi.edu, xumingzh@usc.edu, efink@ifc.org
[2] Nanchang Hangkong University, Nanchang, China
frueyang@gmail.com
[3] Smithsonian American Art Museum, Washington, DC, USA
{rallen,goodlanderg}@si.edu

**Abstract.** Museums around the world have built databases with metadata about millions of objects, their history, the people who created them, and the entities they represent. This data is stored in proprietary databases and is not readily available for use. Recently, museums embraced the Semantic Web as a means to make this data available to the world, but the experience so far shows that publishing museum data to the linked data cloud is difficult: the databases are large and complex, the information is richly structured and varies from museum to museum, and it is difficult to link the data to other datasets. This paper describes the process and lessons learned in publishing the data from the Smithsonian American Art Museum (SAAM). We highlight complexities of the database-to-RDF mapping process, discuss our experience linking the SAAM dataset to hub datasets such as DBpedia and the Getty Vocabularies, and present our experience in allowing SAAM personnel to review the information to verify that it meets the high standards of the Smithsonian. Using our tools, we helped SAAM publish high-quality linked data of their complete holdings (41,000 objects and 8,000 artists).

## 1 Introduction

Recently, there have been a number of efforts to publish metadata about the objects in museums as Linked Open Data (LOD). Some notable efforts include the Euopeana project [7], which published data on 1,500 of Europe's museums, libraries, and archives, the Amsterdam Museum[3], which published data on 73,000 objects, and the LODAC Museum [11], which published data from 114 museums in Japan. Despite the many recent efforts, there are still significant challenges in publishing data about artwork to the linked data cloud. Mapping the data of a museum to linked data involves three steps:

1. **Map the Data to RDF.** The first step is to map the metadata about works of art into RDF. This involves selecting or writing a domain ontology with standard terminology for works of art and converting the data to RDF

P. Cimiano et al. (Eds.): ESWC 2013, LNCS 7882, pp. 593–607, 2013.
© Springer-Verlag Berlin Heidelberg 2013

according to this ontology. De Boer et al. [3] note that the process is complicated because many museums have rich, hierarchical or graph-structured data. The data often includes attributes that are unique to a particular museum, and the data is often inconsistent and noisy because it has been maintained over a long period of time by many individuals. In past work, the mapping is typically defined using manually written rules or programs.

2. **Link to External Sources.** Once the data is in RDF, the next step is to find the links from the metadata to other repositories, such as DBpedia or GeoNames. In previous work, this has been done by defining a set of rules for performing the mapping. Because the problem is difficult, the number of links in past work is actually quite small as a percentage of the total set of objects that have been published.
3. **Curate the Linked Data.** The third step is to curate the data to ensure that both the published information and its links to other sources within the LOD are accurate. Because curation is so labor intensive, this step has been largely ignored in previous work and as a result links are often inaccurate.

Our goal is to develop technology to allow museums to map their own data to LOD. The contributions of this paper are an end-to-end approach that maps museum source data into high quality linked data and the corresponding lessons learned in performing this mapping. In particular, we describe the process and the lessons learned in mapping the metadata that describes the 41,000 objects of the Smithsonian American Art Museum (SAAM). This work builds on our previous work on a system called Karma for mapping structured sources to RDF. However, in the real-world data provided by the Smithsonian, we discovered that there were complex structures that required new capabilities in Karma. In terms of linking, we found that mapping the entities, such as artist names, to DBpedia could not be easily or accurately performed using existing tools, so we developed a specialized mapping approach to achieve high accuracy matches. Finally, to ensure that the Smithsonian publishes high quality linked data, we developed a curation tool that allows the museum staff to easily review and correct any errors in the automatically generated links to other sources.

In the remainder of this paper, we describe our approach and present the lessons learned in mapping (Section 2), linking (Section 3), and curating (Section 4) the SAAM data. For each of these topics, we describe our approach, present lessons learned, and evaluate the effectiveness of our approach. We then compare our work to previous work (Section 5) and conclude with a discussion of the contributions and future work (Section 6).

## 2 Mapping the Data to RDF

### 2.1 The SAAM Database

SAAM stores collection metadata in a relational database managed by TMS[1], a comprehensive data management system for museums. The SAAM deployment of TMS consists of over 100 tables, containing significant amounts of data

[1] `http://gallerysystems.com/tms`

that needs to remain private (e.g., financial information). In order to avoid issues about private data, we only use the tables that the museum uses to populate their Web site. All the information in these eight tables already appears on the museum Web site, so the museum is amenable to publishing it as linked data. The structure and format of these data are tailored to the needs of the Web site and some fields need to be decoded to produce appropriate RDF. For example, descriptive terms are encoded in text such as "Authorities\Attributes\Objects\Folk Art". The database includes data about 41,267 objects and the 8,261 artists who created them.

For objects, the database contains the regular tombstone information including classification (e.g., sculpture, miniature), their role (e.g., artist, printer), available images, terms (e.g., Portrait Female – Turner, Tina). For artists, the database contains names, including multiple variants (e.g., married name, birth or maiden name), title and suffixes, biographical information and geographical information including city, county, state and country of relevant places (e.g., birth and death place, last known residence) and citation information.

*Lesson 1: Satisfy the Legal Department First.* Much of the data in museums is proprietary and getting approval from the legal department can be challenging. We use the data that drives the Web site; it is not the raw data, but adequate and circumvents issues that could have stopped or delayed the project.

### 2.2 Europeana Data Model (EDM)

The Europeana Data Model (EDM[2]) is the metamodel used in the Europeana project[3] to represent data from Europe's cultural heritage institutions. EDM is a comprehensive OWL ontology that reuses terminology from several widely-used ontologies: *SKOS* [4] for the classification of artworks, artist and place names; *Dublin Core* [5] for the tombstone data; *FOAF* [6] and *RDA Group 2 Elements* [7] to represent biographical information; *ORE* [8] from the Open Archives Initiative, used by EDM to aggregate data about objects.

The SAAM ontology[9] (Figure 1) extends EDM with subclasses and subproperties to represent attributes unique to SAAM (e.g., identifiers of objects) and incorporates classes and properties from *schema.org* [10] to represent geographical data (city, state, country). We chose to extend EDM because this maximizes compatibility with a large number of existing museum LOD datasets.

[2] `http://www.europeana.eu/schemas/edm/`
[3] `http://europeana.eu`
[4] `http://www.w3.org/2004/02/skos/`
[5] `http://purl.org/dc/elements/1.1/` and `http://purl.org/dc/terms/`
[6] `http://xmlns.com/foaf/0.1/`
[7] `http://rdvocab.info/ElementsGr2`
[8] `http://www.openarchives.org/ore/terms/`
[9] `http://americanart.si/linkeddata/schema/`
[10] `http://schema.org/`

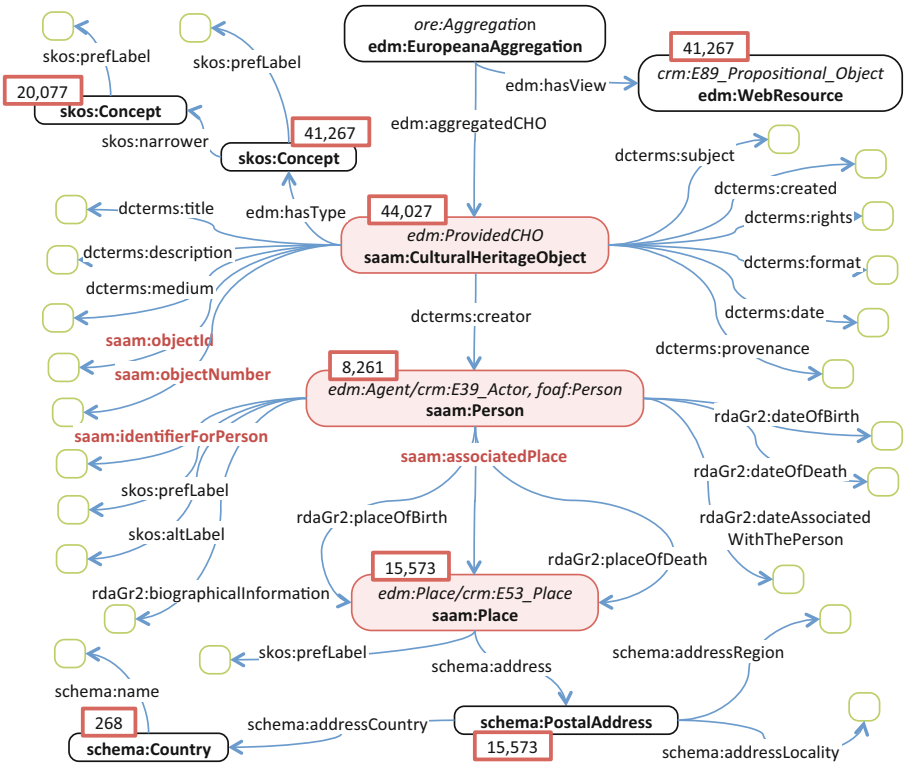


**Fig. 1.** The SAAM ontology. Named ovals represent classes, un-named green ovals represent literals, arcs represent properties, boxes contain the number of instances generated in the SAAM dataset, italicized text shows superclasses, all properties in the saam namespace are subproperties of properties in standard vocabularies.

One of the most challenging tasks in the project was selecting and extending the ontologies. We considered EDM and CIDOC CRM[11]; both are large and complex ontologies, but neither fully covers the data that we need to publish. We needed vocabularies to represent biographical and geographical information, and there are many to choose from. Following the lead of the Amsterdam Museum [3], we used RDA Group 2 Elements for the biographical information. We didn't find guidance for representing the geographical information in the cultural heritage community so we selected schema.org as it is a widely used vocabulary. Our extensions (shown in boldface/red in Figure 1) are subclasses or subproperties of entities in the ontologies we reuse.

*Lesson 2: A Library of Ontologies for Cultural Heritage Is Desperately Needed.* While EDM represents an excellent starting point for modeling cultural heritage data, the community can benefit from guidance on vocabularies to represent data not covered by EDM and an integrated library with the recommended ontologies.

[11] `http://www.cidoc-crm.org`

### 2.3 Using KARMA to Map the SAAM Data to RDF

**Prior Work.** In previous work [9], we developed KARMA, a tool to map structured data to RDF according to an ontology of the user's choice. The goal is to enable data-savvy users (e.g., spreadsheet users) to do the mapping, shielding them from the complexities of the underlying technologies (SQL, SPARQL, graph patterns, XSLT, XPath, etc). KARMA addresses this goal by automating significant parts of the process, by providing a visual interface (Figures 2 to 4) where users see the KARMA-proposed mappings and can adjust them if necessary, and by enabling users to work with example data rather than just schemas and ontologies. The KARMA approach to map data to ontologies involves two interleaved steps: one, assignment of *semantic types* to data columns and two, specification of the relationships between the semantic types.

A semantic type can be either an OWL class or the range of a data property (which we represent by the pair consisting of a data property and its domain). KARMA uses a conditional random field (CRF) [10] model to learn the assignment of semantic types to columns of data from user-provided assignments [5]. KARMA uses the CRF model to automatically suggest semantic types for unassigned data columns (Figure 2). When the desired semantic type is not among the suggested types, users can browse the ontology to find the appropriate type. KARMA automatically re-trains the CRF model after these manual assignments.

The relationships between semantic types are specified using paths of object properties. Given the ontologies and the assigned semantic types, KARMA creates a graph that defines the space of all possible mappings between the data source and the ontologies [9]. The nodes in this graph represent classes in the ontology, and the edges represent properties. KARMA then computes the minimal tree that connects all the semantic types, as this tree corresponds to the most concise model that relates all the columns in a data source, and it is a good starting point for refining the model (Figure 3). Sometimes, multiple minimal trees exist, or the correct interpretation of the data is defined by a non-minimal tree. For these cases, KARMA provides an easy-to-use GUI to let users select a desired relationship (an edge in the graph). KARMA then computes a new minimal tree that incorporates the user-specified relationships.

**Challenge 1: Data Preparation.** We encountered multiple situations where we had to filter and transform data prior to modeling it and converting it to RDF. The following are the the types of data preparation tasks we encountered: *Filtering tables:* for example, the SAAM tables represent constituents, which includes both people and organizations. The ontologies for people and organizations are different so we defined database views to filter the tables accordingly. *Data extraction:* for example, the keywords associated with the art objects need to be extracted from text such as "Authorities\Attributes\Objects\Subject Specific\Animal\bird\owl". *Concatenating and formatting columns:* the SAAM tables represent people names, dates and places in a structured form (e.g., year, month and date in separate columns). We needed to concatenate these fields to construct values for single properties (e.g., dateOfBirth), taking care to insert separators and leading zeroes to format them appropriately.

We addressed these data preparation tasks before modeling the data in KARMA by defining views and stored procedures in the database. We then loaded the new tables and views in KARMA to model them. While data preparation is routine in database applications and powerful tools are available to support them, RDF mapping tools (including KARMA) lack the needed expressivity. Tools like ClioPatria [3] allow users to define expressions in a full programming language (Prolog in the case of ClioPatria) and invoking them within their mapping rules. Our approach is to enable users to use whatever tools they are familiar with in a prior data preparation step.

*Lesson 3: The Data Preparation/Data Mapping Split Is Effective.* The range of data preparation tasks is open-ended and ad hoc. It is wise to acknowledge this and to design a data mapping architecture that is compatible with traditional data preparation tools. This allows the data mapping language to remain relatively simple. KARMA integrates with a data preparation step by providing the ability to specify many aspects of the mapping in the data tables themselves (discussed below). We did the data preparation primarily using SQL views, other users of KARMA have used Google Refine[12].

**Challenge 2: Mapping Columns to Classes.** Mapping columns to the ontology is challenging because in the complete SAAM ontology there are 407 classes and 105 data properties to choose from. KARMA addresses this problem by learning the assignment of semantic types to columns. Figure 2 shows how users define the semantic types for the constituentid (people or organizations) and place columns in one of the SAAM tables. The figure shows a situation where KARMA had learned many semantic types. The left part shows the suggestions for constituentid. The SAAM database uses sequential numbers to identify both constituents and objects. This makes them indistinguishable, so KARMA offers both as suggestions, and does not offer other irrelevant and incorrect suggestions. The second example illustrates the suggestions for the place column and shows how users can edit the suggestions when they are incorrect.

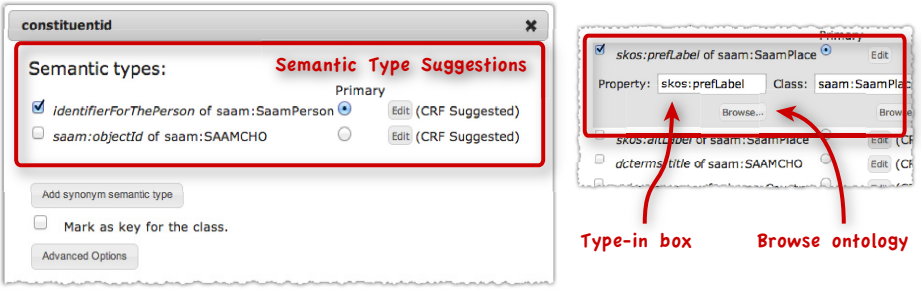


**Fig. 2.** Semantic types map data columns to classes and properties in an ontology. Left: KARMA suggestions to model the constituentid column in a SAAM table (the first choice is correct). Right: user interface for editing incorrect suggestions.

[12] http://code.google.com/p/google-refine/

**Challenge 3: Connecting the Classes.** This is also challenging because there are 229 object properties in the ontology to choose from. Figure 3 illustrates how KARMA automatically connects the semantic types for columns as users define them. In the first screen the user assigns a semantic type for consitutentid. In the second screen, the user assigns a semantic type for place, and KARMA automatically adds to the model the associatedPlace object property to connect the newly added SaamPlace to the pre-existing SaamPerson. Similarly, when the user specifies the semantic type for column city, KARMA automatically adds the address object property.

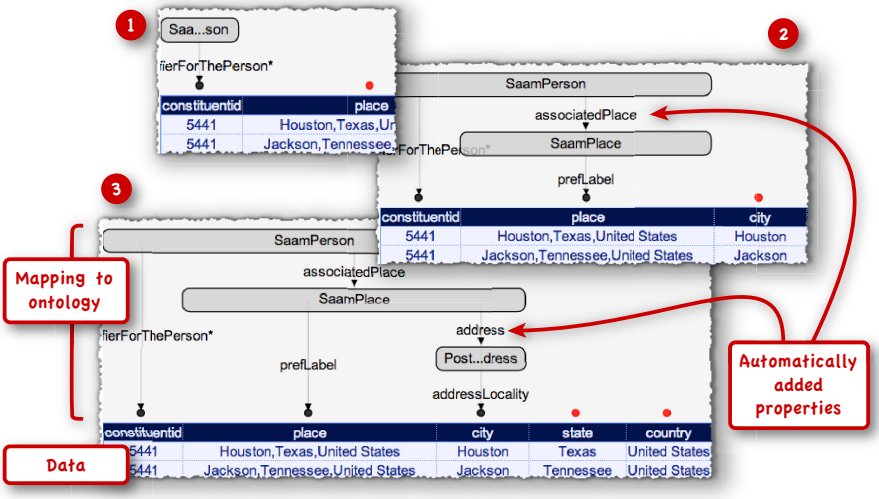


**Fig. 3.** Each time the user adds new semantic types to the model, KARMA connects them to the classes already in the model

Each time users model the semantic type of a column, KARMA connects it to the rest of the model. In the examples, the connections use a single property, but KARMA searches the whole graph and finds longer paths when appropriate. In addition, weights in the graph [9] bias the algorithm to prefer specific properties rather than general properties inherited from superclasses. Sometimes, multiple lowest-cost models exist, or the appropriate model is not the lowest-cost model. Users can easily adjust the proposed model by clicking on an incorrect property and selecting the appropriate one from a menu of all compatible properties.

*Lesson 4: Property Domain and Range Definitions Are Important.* KARMA leverages domains and ranges to automate modeling the relationships between the columns in the tables, often selecting the correct property. When KARMA proposed non-sensical, complicated paths to connect classes (e.g., subclass path via Thing), it was often because properties lacked domain or range information or because the classes we defined had not been defined as subclasses of the appropriate classes. This feedback helped us to integrate the SAAM-specific ontology with the large complex ontologies we are reusing.

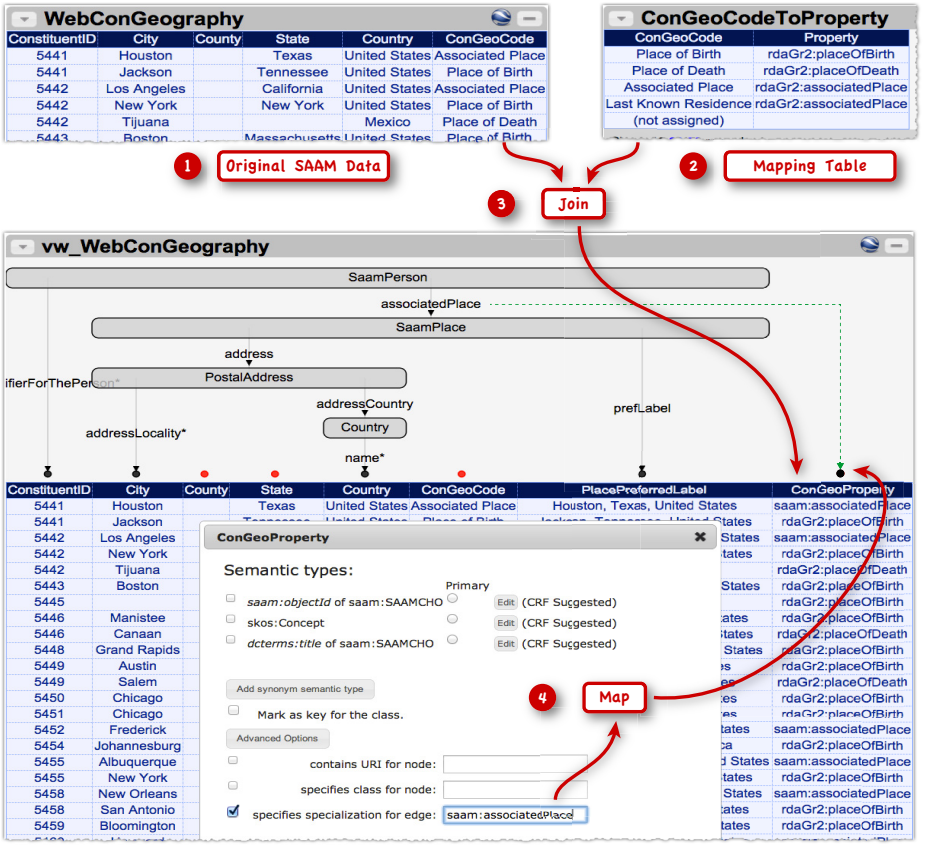


**Fig. 4.** Mapping columns where different rows must be mapped using different properties: 1) the data table; 2) a table to translate ConGeoCode to ontology properties; 3) join to add a column with the property URIs; 4) map values in the ConGeoProperty column to the associatedPlace property.

**Challenge 4: Mapping Depends on Field Values.** Figure 4 illustrates a situation where the mapping from a table to the *desired* ontology cannot be specified at the schema level. The WebConGeography table contains information associated with people. Each row represents a place association: the first column (ConstituentID) represents the person identifier, the middle columns represent the place and the last column (ConGeoCode) represents the meaning of the place. The SAAM ontology defines a generic property associatedPlace to represent the relationship between a person and a place. This general property is appropriate for the first and third rows, but not for the others (e.g., the second row should use the more specific property rdaGr2:placeOfBirth).

To model these situations, users add a column that contains the required data. In the particular case illustrated in Figure 4, the user can define a table that maps the ConGeoCodes to the appropriate properties (step 2) and then do a join to add

the new column (step 3). Finally, when defining the semantic type for the new column (step 4), users can specify that the values in the column specialize the associatedPlace property. Analogous situations arise in the SAAM tables that represent data about dates associated with people and variant people names. This type of mapping can be defined in tools such as D2RQ[13], but requires an expert user to define multiple, complex conditional mapping rules. The ability to easily define these data-specific mappings is new since our prior work in KARMA.

*Lesson 5: Supporting Row-Level Metadata Solves Many Complex Mapping Problems.* The same mechanism we used to model row-specific properties can be used to model row-specific classes, URIs and language tags. It enables users to invoke arbitrary computation using their favorite tools to define data-dependent aspects of the mapping that cannot be cleanly represented in declarative representations. Other approaches such as D2RQ offer a limited set of built-in functions (e.g., concatenation, regular expression) that can be extended by writing Java classes. Our approach enables users to use whatever tools they are comfortable using.

**Evaluation.** We evaluated the effectiveness of KARMA by mapping 8 tables (29 columns) to the SAAM ontology (Table 1). We performed the mapping twice: in *Run 1*, we started with no learned semantic types, and in *Run 2* we ran KARMA using the semantic types learned in the first run. The author of the paper that designed the ontology performed the evaluation. Even though he knows which properties and classes to use, when KARMA didn't suggest them he used the browse capability to find them in the ontology instead of typing them in. It took him 18 minutes to map all the tables to RDF, even in the first run, when KARMA's semantic type suggestions contained the correct semantic type 24% of the time. The second run shows that the time goes down sharply when users don't need to browse the ontology to find the appropriate properties and classes. The evaluation also shows that KARMA's algorithm for assigning relationships among classes is very effective (85% and 91% correct in *Run 1* and *Run 2*).

*Lesson 6: Ontology Design Is the Hard Part.* Even though it takes about 8 to 18 minutes to map all the tables using KARMA, it took about 2 weeks after the initial design of the ontology to map all the tables. We spent the time designing and redesigning the ontology. During that period, we mapped the tables many times to slightly different ontologies. So, in Table 1 *Run 2* is typical as we spent significant type rerunning KARMA after slight changes to the ontology.

**Table 1.** Effectiveness of KARMA's automation capabilities

| | *# of times KARMA's top 4 suggestions contain the correct semantic type* | *# of times KARMA correctly assigns relationships among classes* | *Time (minutes)* |
|---|---|---|---|
| *Run 1* | 7 out of 29 (24%) | 30 out of 35 (85%) | 18 |
| *Run 2* | 27 out of 29 (93%) | 32 out of 35 (91%) | 8 |

[13] http://d2rq.org

## 3 Linking to External Resources

The RDF data will benefit the Smithsonian museum and the community if it is linked to useful datasets. We focused on linking SAAM artists to DBpedia[14] as it provides a gateway to other linked data resources and it is a focus for innovative applications. We also linked the SAAM artists to the Getty Union List of Artist Names (ULAN®) and to the artists in the Rijksmuseum dataset.

Museums pride themselves in publishing authoritative data, so SAAM personnel manually verified all proposed links before they became part of the dataset. To make the verification process manageable, we sought high-precision algorithms. We matched people using their names, including variants, and their birth dates and death dates. The task is challenging because people's names are recorded in many different ways, multiple people can have the same name, and birth dates and death dates are often missing or incorrect. For technical reasons we did not use other information, although we plan to do so in the future.

Our approach involves estimating the ratio of people in DBpedia having each possible value for the properties we use for matching (e.g., ratio of people born in 1879). For attributes compared using equality (birth/death years), we scan all people in DBpedia counting the number that have each specific value. For dependent attributes such as birth and death year, we also compute the ratios for pairs of values. We compare names using the Jaro-Winkler string metric [4], and for them compute the ratios as follows: we divide the interval $[0, 1]$ in bins of size $\epsilon$, and for each bin we estimate the number of pairs of people whose names differ by a Jaro-Winkler score less than $\epsilon$. Empirically, we determined that $\epsilon = 0.01$ and 10 million samples yield good results in our ground truth dataset.

The matching algorithm is simple. Given a SAAM and a DBpedia person, their matching score is $s = 1 - d * n$ where $d$ is the *date score* and $n$ is the *name score*. If the dates match exactly, $d$ is the fraction of people in DBpedia with those dates. Otherwise, $d$ is the sum of the fractions for all the intervening years. $n$ is the fraction of people in DBpedia whose Jaro-Winkler score is within $\epsilon$ from the score between the given pair of people, computed using the estimates discussed above. We use a simple blocking scheme based on the first letter of the name. To increase efficiency, we discard pairs whose birth or death years differ by more than 10 and whose Jaro-Winkler score is less than 0.8.

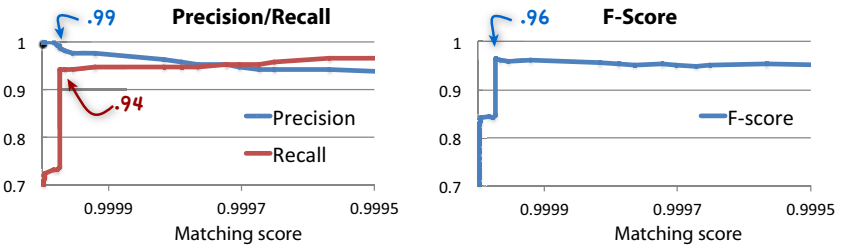


**Fig. 5.** Precision/Recall and F-score as a function of our algorithm's matching score

[14] `http://dbpedia.org`

**Evaluation.** To evaluate our algorithm we constructed ground truth for a dataset of 535 people in the SAAM database (those whose name starts with A). We manually searched in Wikipedia using all variant names and verified the matches using the text of the article and all fields in the SAAM record, including the biography. We found 176 matches in DBpedia.

Figure 5 shows the evaluation results on the ground truth (note that the matching score $s$ decreases from left to right). The highest F-score .96 achieves a precision of .99 and a recall of .94 (166 correct results, 1 incorrect result). At this threshold $s^*$ all names and the years for all but 5 people match exactly. The incorrect result is one where neither date matches, but interestingly, there are 4 results where the years are present but not equal. The sharp increase in recall comes at a score $s > s^*$ where suddenly 37 results for people missing a single date are above threshold (all these are correct results). The next interesting threshold is $\hat{s} = 0.9995$. Between $s^*$ and $\hat{s}$ are 13 results; of these, 4 are correct (2 with non matching names) and 9 incorrect, yielding .938 precision and .966 recall. For $s < \hat{s}$, the density of correct results decreases sharply, containing only 4 correct results in the next 286 candidates. Based on these results, we used $\hat{s}$ as the threshold to match the SAAM data against all DBpedia people (2,807 results), the Getty ULAN (1,759 results) and the Rijksmuseum (321 results).

## 4 Curating the Linked Data

Museums need the ability to ensure that the linked data they publish are of high quality. The first aspect of the curation process is to ensure that the RDF is correct. Museum personnel can easily browse individual RDF records on the Web, but without understanding the relationship between an RDF record and the underlying database records, it is hard to assess whether the RDF is correct. Karma helps museum personnel understand these relationships at the schema level by graphically showing how database columns map to classes and properties in the ontology (e.g., Figures 3 and 4). Karma also lets users click on individual worksheet cells to inspect the RDF generated for it, helping them understand the relationships at the data level. These graphical views also enabled SAAM personnel and the Semantic Web researchers to communicate effectively while refining the ontology and the mappings. Our goal by the end of the project is that SAAM personnel will use Karma to refine the mappings on their own.

The second aspect of the curation process is to ensure that links to external sources are correct. Our approach is to 1) record the full provenance of each link so that users (and machines) can record links and inspect them when the data sources or the algorithm change, and 2) make it easy for users to review the results of the linking algorithm. We use the PROV ontology[15] to represent provenance data for every link including revisions, matching scores, creation times, author (human or system/version) and data used to produce the link. Users review the links using the Web interface depicted in Figure 6. The interface is a visualization and editor of the underlying PROV RDF records. Each row

[15] `http://www.w3.org/TR/prov-o/`

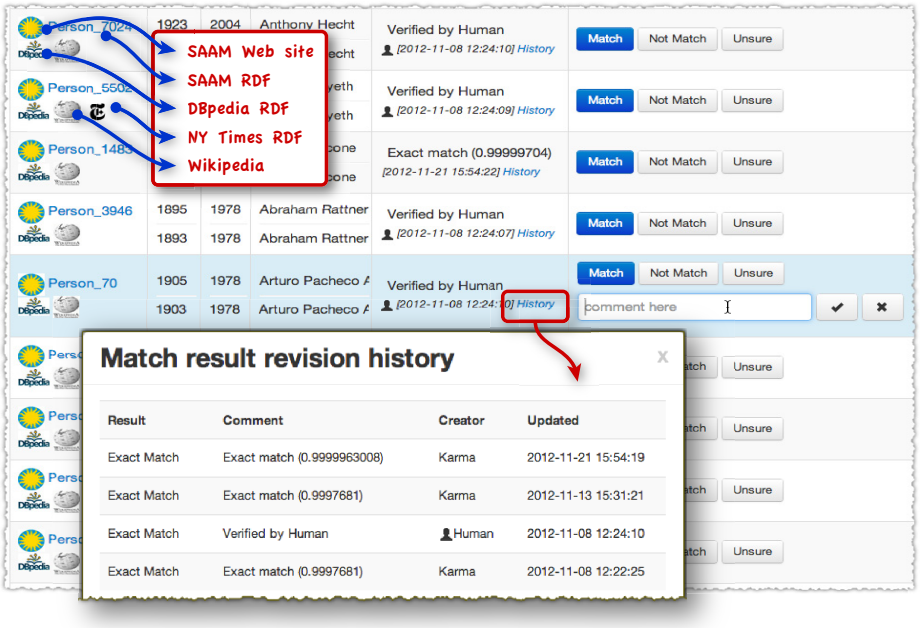


**Fig. 6.** The KARMA interface enables users to review the results of linking

represents a link. The first cell shows the records being linked: the top part shows links to information about the SAAM record and the bottom part shows links to information for a record in an external source. The next columns show the data values that were used to create the link and information about its revision history. The last column shows buttons to enable users to revise links and provide comments (recorded as PROV records). SAAM personnel used this interface to verify all 2,807 links to DBpedia.

*Lesson 7: PROV Is a Suitable Technology for Curating the Links.* In addition to supporting the user interface for human verification of links, the PROV representation affords other benefits. We can use SPARQL statements to construct a dataset of owl:same-as triples containing only those links that have been verified by a human (suitable for publication on the Smithsonian Web site) or a dataset containing containing all links with a matching score above a given threshold (suitable for other applications). Similarly, when the underlying data changes (e.g., there is a new version of DBpedia) or a new version of the matching software becomes available, it is possible to retrieve the affected links.

## 5 Related Work

There has been much recent interest in publishing museum data as Linked Open Data. Europeana[7], one of the most ambitious efforts, published the metadata on 17 million items from 1,500 cultural institutions. This project developed a

comprehensive ontology, called the Europeana Data Model (EDM) and used it to standardize the data that each organization contributes. This standard ontology enables Europeana to aggregate data from such a large number of cultural institutions. The focus of that effort was on developing a comprehensive data model and mapping all of the data to that model. Several smaller efforts focused on mapping rich metadata into RDF while preserving the full content of the original data. This includes the MuseumFinland, which published the metadata on 4,000 cultural artifacts[8] and the Amsterdam Museum [3], which published the metadata on 73,000 objects. In both of these efforts the data is first mapped directly from the raw source into RDF and then complex mapping rules transform the RDF into an RDF expressed in terms of their chosen ontology. The actual mapping process requires using Prolog rules for some of the more complicated cases. Finally, the LODAC Museum published metadata from 114 museums and research institutes in Japan. They defined a relatively simple ontology that consists of objects, artists, and institutions to simplify the mapping process.

In our work on mapping the 41,000 objects from SAAM, we went beyond the previous work in several important ways. First, we developed an approach that supports the mapping of complex sources (both relational and hierarchical) into a rich domain ontology [9]. This approach is in contrast to previous work, which first maps the data directly into RDF [1] and then aligns the RDF with the domain ontology [2]. As described earlier, we build on the EDM ontology, a rich and easily extensible domain ontology. Our approach makes it possible to preserve the richness of the original metadata sources, but unlike the MuseumFinland and the Amsterdam Museum projects, a user does not need to learn a complex rule language and only needs to do a data preparation step to define database views using SQL statements and simple stored procedures.

Second, we performed significantly more data linking than these previous efforts. There is significant prior work on linking data across sources and the most closely related is the work on Silk [14] and the work on entity coreference in RDF graphs [13]. Silk provides a nice framework that allows a user to define a set of matching rules and weights that determine whether two entities should be matched. We tried to use Silk on this project, but we found it extremely difficult to write a set of matching rules that produced high quality matches. The difficulty was due to a combination of missing data and the variation in the discriminability of different data values. The approach that we used in the end was inspired by the work on entity coreference by Song and Heflin [13], which deals well with missing values and takes into account the discriminability of the attribute values in making a determination of the likelihood of a match.

Third, because of the importance to the Smithsonian of producing a high-quality linked data, we developed a curation tool that allows an expert from the museum to review and approve or reject the links produced automatically by our system. Previous work has largely ignored the issue of link quality (Halpin et al. [6] reported that in one evaluation roughly 51% of the same-as links were found to be correct). The exception to this is the effort by the NY Times to map all of their metadata to linked data through a process of manual curation.

In order to support a careful evaluation of the links produced by our system, we developed the linking approach that allows a link reviewer to see the data that is the basis for the link and to be able to drill down into the individual sources to evaluate a link.

## 6 Conclusions and Future Work

In this paper we described our work on mapping the data of the Smithsonian American Art Museum to Linked Open Data. We presented the end-to-end process of mapping this data, which includes the selection of the domain ontologies, the mapping of the database tables into RDF, the linking of the data to other related sources, and the curation of the resulting data to ensure high-quality data. This initial work provided us with a much deeper understanding of the real-world challenges in creating high-quality link data.

For the Smithsonian, the linked data provides access to information that was not previously available. The Museum currently has 1,123 artist biographies that it makes available on its website; through the linked data, we identified 2,807 links to people records in DBpedia, which SAAM personnel verified. The Smithsonian can now link to the corresponding Wikipedia biographies, increasing the biographies they offer by 60%. Via the links to DBpedia, they now have links to the New York Times, which includes obituaries, exhibition and publication reviews, auction results, and more. They can embed this additional rich information into their records, including 1,759 Getty ULAN® identifiers, to benefit their scholarly and public constituents.

The larger goal of this project is not just to map the SAAM data to Linked Open Data, but rather to develop the tools that will enable any museum or other organization to map their data to linked data themselves. We have already developed the Karma integration tool, which greatly simplifies the problem of mapping structured data into RDF, a high-accuracy approach to linking datasets, and a new curation tool that allows an expert to review the links across data sources. Beyond these techniques and tools, there is much more work to be done. First, we plan to continue to refine and extend the ontologies to support a wide range of museum-related data. Second, we plan to continue to develop and refine the capabilities for data preparation and source modeling in Karma to support the rapid conversion of raw source data into RDF. Third, we plan to generalize our initial work on linking data and integrate a general linking capability into Karma that allows a user to create high-accuracy linking rules and to do so by example rather than having to write the rules by hand.

We also plan to explore new ways to use the linked data to create compelling applications for museums. A tool for finding relationships, like EverythingIsConnected.be [12], has great potential. We can imagine a relationship finder application that allows a museum to develop curated experiences, linking artworks and other concepts to present a guided story. The Museum could offer pre-built curated experiences or the application could be used by students, teachers, and others to create their own self-curated experiences.

**Acknowledgements.** This research was funded by the Smithsonian American Art Museum. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Smithsonian Institution.

## References

1. Bizer, C., Cyganiak, R.: D2R Server–publishing relational databases on the semantic web. Poster at the 5th International Semantic Web Conference (2006)
2. Bizer, C., Schultz, A.: The R2R Framework: Publishing and Discovering Mappings on the Web. In: 1st International Workshop on Consuming Linked Data, Shanghai (2010)
3. de Boer, V., Wielemaker, J., van Gent, J., Hildebrand, M., Isaac, A., van Ossenbruggen, J., Schreiber, G.: Supporting Linked Data Production for Cultural Heritage Institutes: The Amsterdam Museum Case Study. In: Simperl, E., Cimiano, P., Polleres, A., Corcho, O., Presutti, V. (eds.) ESWC 2012. LNCS, vol. 7295, pp. 733–747. Springer, Heidelberg (2012)
4. Cohen, W.W., Ravikumar, P., Fienberg, S.E., et al.: A comparison of string distance metrics for name-matching tasks. In: Proceedings of the IJCAI 2003 Workshop on Information Integration on the Web (IIWeb 2003), pp. 73–78 (2003)
5. Goel, A., Knoblock, C.A., Lerman, K.: Exploiting Structure within Data for Accurate Labeling Using Conditional Random Fields. In: Proceedings of the 14th International Conference on Artificial Intelligence, ICAI (2012)
6. Halpin, H., Hayes, P.J., McCusker, J.P., McGuinness, D.L., Thompson, H.S.: When owl:sameAs isn't the same: An analysis of identity in linked data. In: Patel-Schneider, P.F., Pan, Y., Hitzler, P., Mika, P., Zhang, L., Pan, J.Z., Horrocks, I., Glimm, B. (eds.) ISWC 2010, Part I. LNCS, vol. 6496, pp. 305–320. Springer, Heidelberg (2010)
7. Haslhofer, B., Isaac, A.: data.europeana.eu - The Europeana Linked Open Data Pilot. In: Multiple Values Selected, The Hague, The Netherlands (July 2011)
8. Hyvonen, E., Makela, E., Salminen, M., Valo, A., Viljanen, K., Saarela, S., Junnila, M., Kettula, S.: MuseumFinland - Finnish museums on the semantic web. Web Semantics: Science, Services and Agents on the World Wide Web 3(2-3) (2005)
9. Knoblock, C.A., Szekely, P., Ambite, J.L., Goel, A., Gupta, S., Lerman, K., Muslea, M., Taheriyan, M., Mallick, P.: Semi-automatically mapping structured sources into the semantic web. In: Simperl, E., Cimiano, P., Polleres, A., Corcho, O., Presutti, V. (eds.) ESWC 2012. LNCS, vol. 7295, pp. 375–390. Springer, Heidelberg (2012)
10. Lafferty, J., McCallum, A., Pereira, F.: Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In: Proceedings of the International Conference on Machine Learning (2001)
11. Matsumura, F., Kobayashi, I., Kato, F., Kamura, T., Ohmukai, I., Takeda, H.: Producing and Consuming Linked Open Data on Art with a Local Community. In: Proceedings of the Third International Workshop on Consuming Linked Data (COLD 2012). CEUR Workshop Proceedings (2012)
12. Sande, M.V., Verborgh, R., Coppens, S., Nies, T.D., Debevere, P., Vocht, L.D., Potter, P.D., Deursen, D.V., Mannens, E., Walle, R.: Everything is Connected. In: Proceedings of the 11th International Semantic Web Conference, ISWC (2012)
13. Song, D., Heflin, J.: Domain-independent entity coreference for linking ontology instances. ACM Journal of Data and Information Quality, ACM JDIQ (2012)
14. Volz, J., Bizer, C., Gaedke, M., Kobilarov, G.: Silk–a link discovery framework for the web of data. In: Proceedings of the 2nd Linked Data on the Web Workshop, pp. 559–572 (2009)