Extracting and Semantically Integrating Implicit Schemas from Multiple Spreadsheets of Biology based on the Recognition of their Nature

Ivelize R. Bernardo, Matheus S. Mota e André Santanchè

University of Campinas, UNICAMP, Brazil
{ivelize, mota, santanche}@ic.unicamp.br

Abstract. Spreadsheets are popular among users and organizations, becoming an essential data management tool. The easiness to handle spreadsheets associated with the creative freedom resulted in an increase in the volume of data available in this format. However, spreadsheets are not conceived to integrate data from distinct sources and challenges arise involving systematization of processes to reuse and combine their data. Many related initiatives address the problem of integrating data inside spreadsheets, focusing on lexical and syntactical aspects. However, the proper exploitation of the semantics related to this data is still an opportunity. In this sense, some related work propose mapping spreadsheets contents to open interoperability standards, mainly Semantic Web standards. The main limitation of such proposals is the assumption that it is possible to recognize and make explicit the schema and the semantics of spreadsheets automatically, regardless of their domain. This work differs from related work by assuming the essential role of the context – mainly the domain in which the spreadsheet was conceived – to delineate shared practices of the biology community, which establishes building patterns to be automatically recognized by our system, in a data extraction process and schema recognition. In this article, we present the result of a practical experiment involving such a system, in which we integrate hundreds of spreadsheets belonging to the biology domain and available on the Web. This integration was possible due to observation that the recognition of a spreadsheet nature can be achieved from its tabular organization.

Categories and Subject Descriptors: H.2 [**Database Management**]: Miscellaneous; H.3 [**Information Storage and Retrieval**]: Miscellaneous; I.7 [**Document and Text Processing**]: Miscellaneous

Keywords: biology, data integration, information extraction, semantic web, spreadsheets

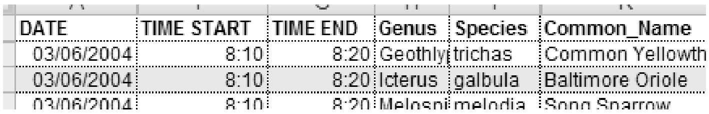
1. INTRODUCTION

Spreadsheets provide autonomy to end users to design their own tables, used to register and manage data [Chambers and Scaffidi 2010; Scaffidi et al. 2005]. Among the roles played by spreadsheets, this work focuses in a relevant subset in which they are applied as "popular databases". Chambers and Scaffidi [2010] noted that, among spreadsheets produced by end users, 25% are used as databases.

The growth of computing power associated with the advance of systems – which are able to handle increasingly larger spreadsheets – fostered a proliferation of these "popular databases" in different contexts. This phenomenon has as side effect the fragmentation of data, scattered in various files, containing informal and implicit schemas, which are designed to operate as isolated entities. These factors hamper data integration and the combination of data from distinct files.

There is a growing concern in transforming tabular data to open standards suitable for reuse and integration [Han et al. 2008; Langegger and Wöß 2009; O'Connor et al. 2010; Syed et al. 2010; Venetis

This work was partially supported by the Microsoft Research FAPESP, Virtual Institute (NavScales project), the Brazilian Institute for Web Sciences Research, CNPq (MuZOO project), PRONEX-FAPESP (Model and Methods in e Science for the Life and Agricultural Sciences), CAPES (AMIB project), as well as individual grants from CNPq.



DATE	TIME START	TIME END	Genus	Species	Common_Name
03/06/2004	8:10	8:20	Geothly	trichas	Common Yellowth
03/06/2004	8:10	8:20	Icterus	galbula	Baltimore Oriole
03/06/2004	8:10	8:20	Melospi	melodia	Song Sparrow

Fig. 1. Example of spreadsheet recording a collection [ecosystems.mbl.edu]

et al. 2011]. This process can be enhanced by associating elements of spreadsheets to concepts in knowledge bases available on the Web. One key issue is how to recognize implicit spreadsheet schemas to make them explicit. A schema goes beyond a set of fields. It is the visible materialization of an underlying conceptual model, which gives semantics to a set of fields, i.e. taking an ER perspective, there are entities and relationships hidden behind a pattern of fields and the way they are organized. An implicit schema and the respective hidden conceptual model – recognized through the pattern of the fields – inserted in a given domain are the essence of a *spreadsheet nature*.

Several initiatives try to recognize schemas in any context, resulting in an unbounded spectrum of possibilities. Therefore, they cannot exploit context specificities to drive their recognition process. Moreover, instead of identifying a construction pattern, usually characterized by the spreadsheet nature, these initiatives focus on the recognition of individual labels. For example, spreadsheets to catalog specimens in a museum (nature of the spreadsheet) usually share a construction pattern – not analyzed by related work – which can guide their recognition.

In this article we assume that such recognition and mapping process will be more effective if we consider the domain in which the spreadsheet was created – e.g., a usage domain of biology – share practices which result in construction patterns. In previous work [Bernardo et al. 2012] we demonstrated that many of these patterns are likely to be recognized by computer programs and we have introduced our strategy for automatic recognition of such patterns. This article presents how our process to recognize biology schemas making them explicit was applied in the construction of a system able to transform several spreadsheets into a unified and integrated data repository. Our process includes automatic schema recognition and association between spreadsheets fields with concepts available in ontologies like Geospecies. Unlike related work, it is able to recognize the nature of several spreadsheets, producing data with associated semantics, which can provide a guideline for choosing appropriate operations for their integration.

This research is part of a larger project involving cooperation with biologists to build databases that integrate biodiversity data. We observed that biologists maintain a significant portion of their data in spreadsheets. In parallel, there are initiatives aimed at making biological data more flexible [Yang et al. 2005; Ponder et al. 2001] and shareable. They point out that although information sources are rich in semantics, it is not properly exploited, as the heterogeneity of formats adopted by sources hampers their access and manipulation. For this reason, this research adopted the context of biology and spreadsheets oriented to data management as its specific focus. The article is organized as follows. Section 2 presents an overview of related work, also introducing the distinctive features of our approach. Section 3 introduces our process to make spreadsheet schemas explicit. Section 4 presents our system that integrates data from several spreadsheets based on the recognition of their nature. Section 5 presents the conclusion and future work.

2. RELATED WORK

There are several initiatives aimed to provide semantic interoperability for tabular data, in order to subsidize integration of data from different sources. Data management on spreadsheets can be treated as a specialized subset of this universe. In this section, we present some relevant works in this direction. Figure 1 presents a spreadsheet containing data about bird specimens collected in the field, which will be used as an example to illustrate the analysis of related work.

The main factor to transform data from a spreadsheet into an open standard representation is the recognition of its schema in order to make it explicit. A usual approach involves mapping spreadsheet fields to ontology concepts.

The mapping process can be automatic, manual or semiautomatic. In the manual process, the user must locate spreadsheet elements that represent specific fields and associate them to elements of an ontology. Han et al. [2008] applies the *entity-per-row* [O'Connor et al. 2010] manual mapping approach, suitable only for tables with simple schemas. The focus of this and all approaches, which we will present in this section, is the recognition and mapping of attributes individually, without considering how they are combined to form an entity. Our approach goes beyond by considering that in spreadsheets – as in other kinds of data management artifacts – attributes are combined to reflect a hidden conceptual model, which is consequence of the *spreadsheet nature*. We are able to recognize the nature of several spreadsheets belonging to the biology usage domain, producing a semantically richer characterization of the generated instances.

The technique proposed by Langegger and Wöß [2009] is not limited to the *entity-per-row* perspective and propose mapping implicit hierarchies found in spreadsheet schemas. Abraham and Erwig [2006] identified a specific subset of spreadsheets, which are adopted by users as templates to produce new ones, in a copy and adapt approach. As people outside the creation context can produce errors and inconsistencies in the reuse process, they propose a life cycle for spreadsheets in two stages: development and use. The stages clearly devise the schema creation (development stage) from the data entry process (use stage). A schema created in the first stage cannot be changed in the second stage, reducing errors and inconsistencies. While this proposal changes the way users produce spreadsheets, our approach differs by exploiting the latent semantics in spreadsheets in their "natural habitat". They address an important phenomenon of spreadsheet reuse as templates, which contributes to establish building patterns [Abraham and Erwig 2006]. In our work we also exploit such building patterns, but for an automatic recognition of the spreadsheet schema, since a systematized production process, as proposed by them, requires controlled environments.

In most cases, manual semantic mapping is not feasible [Syed et al. 2010]. For this reason, some related work propose an automatic semantic mapping supported by external knowledge bases, as those provided by the Semantic Web. Syed et al. propose a generic mapping approach, which can be applied to any context. In order to map the attributes and values, found in a spreadsheet, to RDF properties and values, they associate spreadsheet attributes to concepts available in knowledge bases, as DBpedia (http://dbpedia.org) and Yago (http://www.mpiinf.mpg.de/yago-naga/yago/) [Syed et al. 2010]. One of its advantages is the fact that these databases are maintained and updated by people from all parts of the world. On the other hand, it can generate ambiguous and inconsistent links. Applying this strategy to the case of Figure 1, an inconsistency could be generated by analyzing the `Genus` column, which has different interpretations in different contexts.

Venetis et al. [2011] address the ambiguity problem making a correlation of table cells like a correlation between text fragments. Therefore, they will address the ambiguity of `Genus` by relating it with `Species`. Jannach et al. [2009] also apply a semantic mapping of terms during the Web tables extraction process. It involves three types of ontology: 1 - core: works as a meta schema describing a generic structure to be recognized; 2 - domain: elements of a schema in a specific domain to be recognized; 3 - ontology instance: elements extracted from tables mapped to instances of the domain ontology. This process clusters related elements to put them in a context, improving the proper association with ontologies. Although the work of Venetis et al. [2011] and Jannach et al. [2009] find correlations among attributes and enhance their association with concepts in ontologies, their focus stay fragmented in isolated attributes interpretation. Since our approach focus on a specific domain and the spreadsheet natures in this domain, the recognized schema+fields will fit in a pattern, preassociated with an OWL description (class+properties). The OWL properties inside the context of an OWL class further guide the proper association of values in the instances.

Hermans et al. [2010] are able to automatically recognize the structure and content of a spreadsheet, transforming it in an UML representation. They adopt a three step approach: parse, prune and enrichment. In the first step, a parse tree is produced representing the internal structure of the spreadsheet, which is pruned in the second step, to maintain just the relevant elements. In the last step, the pruned parsing three is transformed in an UML class diagram through the recognition of patterns, represented as grammars. Even though this grammar-driven mechanism is a powerful approach to capture patterns, our approach goes beyond a grammar, by supporting weight-based approximate pattern association and other strategies to characterize patterns, e.g., according to the spatial position of the field in the spreadsheet. Moreover, our results are semantically richer, as they address OWL ontologies instead of UML. Besides the advantages provided by Semantic Web standards, RDF/OWL define properties as first class citizens. It was particularly relevant in our work, as the unified characterization of properties belonging to multiple classes enables us to recognize the same fields in distinct spreadsheets, which helps merging and articulation of data.

Limaye et al. [2010] adopt machine learning techniques to recognize the implicit schema. They start by associating a type to each attribute, followed by looking for binary relations between attributes. The recognized attributes are associated to concepts in the Yago knowledge base. Compared to our approach, Limaye et al. [2010] recognize only binary relations instead of the nature of the whole spreadsheet. However, their approach have relevant contributions, which can be complementary to our approach. Thus, we intend to explore them in a future work.

This research follows the same strategy of the In Loco Semantics [Santanchè and Silva 2010], it interprets organization patterns and the user behaviour in order to automate part of the process involving in the identification and semantic mapping. Previous works of the In Loco Semantics focused on the recognition and data extraction of textual documents.

3. NATURE-DRIVEN SCHEMA EXPLICITATION

In this article we present the implementation of a process to extract data from spreadsheets, making their schema explicit and transforming them into RDF/OWL. The result is stored in a repository and its semantics enables integrating schemas and combining data coming from several spreadsheets. This section describes the steps from the extraction until the production of OWL – the first three steps of Figure 2. The next section will focus on exploiting the schema integration and data combination possibilities of the resulting repository – last step of Figure 2 – and will also present the prototype and practical experiments comprising the whole process.

3.1 Schema Recognition

The schema recognition step involves analyzing the pattern users follow to organize data, which is strongly influenced by the spreadsheet nature in its domain. In a previous work [Bernardo et al. 2012] we have produced a systematic categorization of construction patterns observed in biology spreadsheets, which served as the basis to design a process to recognize these specific patterns. This work follows this design for developing a system for spreadsheets integration focused on their schemas with the steps illustrated in Figure 2. In our system, spreadsheet data is extracted by using a third-party software named *Document Data Extractor* (DDEx), developed in an associated work [Mota et al. 2009] and available at http://code.google.com/p/ddex/. It is able to read from several specialized formats – Excel, OpenOffice etc. – and convert them to an open representation. While our system receives the stream of data from DDEx, it tries to recognize the schema. By using a dictionary (English and Portuguese), the system maps input terms to lemmas related to ontology concepts. Lemmas are further categorized into six exploratory questions *(who, what, where, when, why, how)* [Jang et al. 2005], which will help to analyze patterns in a higher abstraction level.

Our process to recognize a building pattern and consequently the spreadsheet nature is focused on the schema. While a schema pattern is recognized, it is mapped to an OWL description, as illustrated

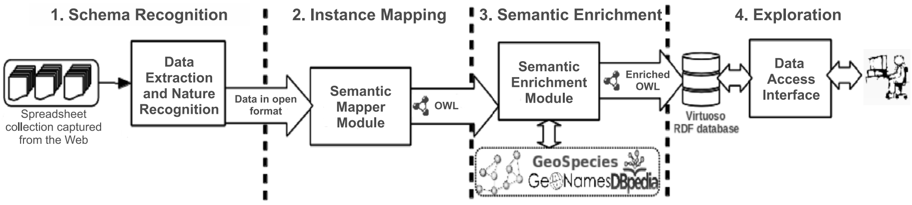


Fig. 2. Data integration cycle comprising data extraction and schema recognition/enrichment.

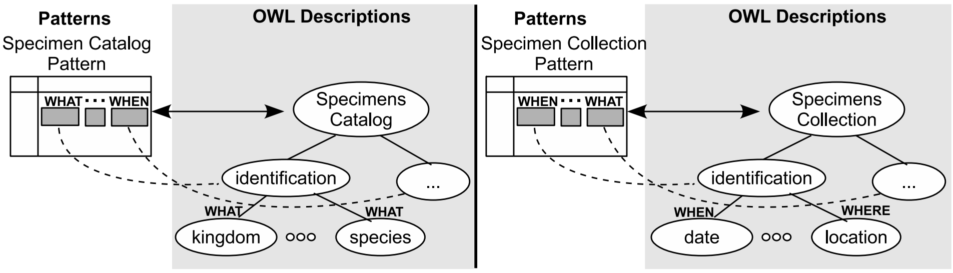


Fig. 3. Matching between patterns and OWL descriptions.

in Figure 3. Therefore, a challenge of our system is to find the proper schema pattern of a given spreadsheet. Each new recognized term and its position in the spreadsheet – properly mapped to a lemma and one of the six exploratory questions – is used to match a pattern among the candidates. As the position of the schema in a spreadsheet may vary, one parallel task is to devise it among the stream of terms. Our observations show that the schema usually appears in the first rows of the spreadsheet, before any instance. Therefore, the system tries to devise it in the beginning of the stream, by looking for a row with high adherence to a given pattern. This adherence is a threshold parameter derived from experimental trials. Only spreadsheets with recognized schema patterns will be selected to follow the remaining of the pipeline.

Each pattern has a profile represented in a file. It defines the lemmas which appear in the pattern, as well as their weights according to their: (a) absolute distinctive power; (b) relative distinctive power; (c) position. The absolute distinctive power expresses how singular the lemma is for a given template – e.g., a "species" lemma is much more rare and distinctive for a specimens catalog pattern than a "date" lemma, which is usual in many spreadsheets. The weight of the absolute distinctive power for each lemma was inferred based on observations presented in [Bernardo et al. 2012]. Since the distinctive power of a lemma is affected by other lemmas that have already been recognized, the relative distinctive power defines how much the distinctive power of a given lemma decreases compared to others already recognized. For example, a set of lemmas has the role of identifying a given species (the *what* question): kingdom, phylum, family etc. Therefore, the first recognized lemma answering *what* – e.g., kingdom – has a higher distinctive power, the second – e.g., phylum – just reinforces the first and has a lower distinctive power, and so on. Finally, we observed that the field position in the spreadsheet is important for determining its role in the pattern and consequently to recognize a given nature [Bernardo et al. 2012]. For example, the first fields in a spreadsheet are usually identification keys and they have high influence in the spreadsheet nature. For example, in Catalog patterns instances are usually identified by *what* fields – e.g., a species identifier in a species catalog. By attributing weights to the position of the lemmas – or their abstractions as six exploratory questions – it is possible to produce "signatures" that lead to the recognition of specific patterns. For example, *what* related lemmas receive the highest weights in the first positions of catalog templates and *when* related lemmas, on the other hand, receive highest weights in the first positions of event templates.

Once a pattern is recognized, the system starts mapping the schema to the respective OWL description, with a root class representing the spreadsheet nature and lemmas mapped to constituent elements. In the next section we describe a prototype – specialized in the biology domain – which aims to demonstrate how to exploit the integration and linkage of data extracted from spreadsheets, when its nature is recognized and made explicit.

3.2 Mapping Instances and Semantic Enrichment

While Step 1 of Figure 2 addresses the schema, Step 2 handle the instances. The *Semantic Mapper Module* (SM) maps spreadsheet instances to OWL supported by external knowledge bases. From the OWL schema description, produced in the previous step, the SM module can relate fields of each instance to a specific domain. The SM module has three main tasks: (i) filter consistent values and convert values in distinct formats; (ii) produce a URI based unique identification for each identifiable value; (iii) unify identifications referring to the same entity. The following example illustrate how the SM produces a corresponding OWL for the extracted data.

Consider that the SM module receives the recognized schema of the spreadsheet in Figure 4(b). The schema fields `Kingdom` and `Genus`, `Species` were related to OWL classes `gs:KingdomConcept` and `gs:SpeciesConcept` of the Geospecies ontology (http://bioportal.bioontology.org/ontologies/1247). Since `Animalia` and `Aix sponsa` are interpreted as instances of `Kingdom` and `Species` respectively, the SM will produce the URIs URI_{k1} and URI_{s1} for values `Animalia` and `Aix sponsa` respectively. The next instance values for the same fields are `Animalia` and `Icterus galbula`. The SM is able to verify that this second `Animalia` refers to the same kingdom of the first one, reusing the URI_{k1}.

While the previous step (SM) fuses data extracted in internal instances represented in RDF/OWL, this Semantic Enrichment (SE) step has the complementary role of linking them to representations in external knowledge bases. For example, the internal URI_{k1} representing animalia is linked to DBpedia or to Geospecies knowledge bases. In [Bernardo et al. 2012] we presented our first version of the semantic enrichment process and we adopted a simple string matching using labels of the instances to query and retrieve identifiers from open knowledge bases, in this work we exploit the relationship among properties related to the same class to enhance the matching. For example, since components of a taxon – i.e., kingdom, phylum, species etc. – are linked both in the original instances as well as in the knowledge base, the system will look for consistent match in which the links in the original instances match with equivalent elements linked in the knowledge base. Resulting data are stored in a Virtuoso RDF database[1], which allows access through a WebService[2].

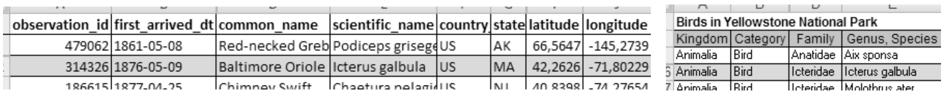
It is important to notice that the focus and contribution of this work address the schema recognition/mapping. Even though it reflects in a better mapping of instances, as they are recognized inside a schema, there are several additional challenges in instance integration beyond the scope of this work. We apply existing string matching algorithms to associate strings with URIs and with knowledge bases.

4. DATA EXPLOITATION AND PROTOTYPE

As we presented in the previous section, the recognition of the spreadsheet nature plays an important role in our semantic mapping process. We will show here that it can also be exploited to determine consistent operations over data. This section aims to show a practical prototype and experiments implementing the whole process described in this article, as well as illustrating distinctive features supported by our semantic recognition approach.

[1] `http://virtuoso.openlinksw.com`

[2] Available at `http://sparql.lis.ic.unicamp.br`



observation_id	first_arrived_dt	common_name	scientific_name	country	state	latitude	longitude
479062	1861-05-08	Red-necked Greb	Podiceps grisege	US	AK	66,5647	-145,2739
314326	1876-05-09	Baltimore Oriole	Icterus galbula	US	MA	42,2626	-71,80229
186615	1877-04-25	Chimney Swift	Chaetura pelagi	US	NJ	40,8398	-74,27654

(a) Example event record [https://www.pwrc.usgs.gov]

Birds in Yellowstone National Park

Kingdom	Category	Family	Genus, Species
Animalia	Bird	Anatidae	Aix sponsa
Animalia	Bird	Icteridae	Icterus galbula
Animalia	Bird	Icteridae	Molothrus ater

(b) Example catalog of specimens [www.greateryellowstonescience.org]

Fig. 4. Example spreadsheets to be merged/articulated

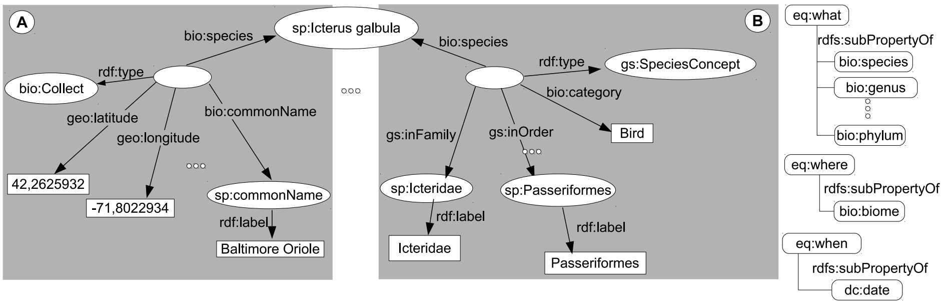


Fig. 5. Semantic mapping of spreadsheets in Fig.3.(a) and Fig.3.(b)

4.1 Integrating and Articulating Data by Nature

In order to illustrate the potential of the produced data, consider again the spreadsheet of Figure 1, recording events related to collections, created by biologists in the field. Most of related work are able to recognize individual attributes, but not the wider scenario, i.e. that each record refers to an event (collection). It has a direct impact on the possibilities of integration and articulation of the resulting set, e.g., if we wish to integrate an instance of this spreadsheet with spreadsheets illustrated in Figure 4. The spreadsheet of Figure 4(a) records collection events as the spreadsheet of Figure 1. An operation to combine both spreadsheets, compatible with their nature is a merge operation, in which data from one spreadsheet can complement the other.

The spreadsheet of Figure 4(b) has a different nature, as it contains a specimens catalog. Although it makes no sense merging this spreadsheet with data of Figures 1 and 4(a), their data can be articulated. For example, specific birds species indicated in a record collection can be linked to those of a catalog. Our proposal is able to recognize such nature of each spreadsheet, which works as a "glue", interrelating the semantics of each field with the semantics of the spreadsheet as a whole. The recognition of each nature will drive applications to apply consistent operations to data from spreadsheets.

The RDF graph of Figure 5 summarizes the result of our extraction process for both spreadsheets in Figure 4 according to their natures. The area highlighted in gray – identified as side (A) – represents the RDF mapping of the spreadsheet of Figure 4(a) (event) and side (B) represents the RDF mapping of the spreadsheet of Figure 4(b). Unlike related work, the instance was recognized as a collection record and materialized in the RDF graph as an instance of the class `bio:Collect` (see an edge representing the property `rdf:type`). Moreover, the instance in Figure 5(B) was recognized as a specimen in the museum and materialized as a RDF instance of the class `gs:SpeciesConcept`.

As illustrated in Figure 5, unlike related work, in our approach the value assigned to each property is not limited to labels. In the specimen instance, for example, Figure 5(B), it is possible to verify that the property value for `gs:inFamily` – which indicates the animal's family, represented using the GeoSpecies vocabulary – is an instance of a specimen that represents the family Icteridae (`biospread:Icteridae`). As detailed in the previous section, the system will link all specimens of the Icteridae family to the same (`biospread:Icteridae`) object. Thus, it is possible congregate all the data from the spreadsheets at any level of characterization of a living being. For example, it is possible to compile all the data from a particular species or from an entire family and so on. As illustrated in the right part of Figure 5, properties mapped into RDF are categorized as sub-properties

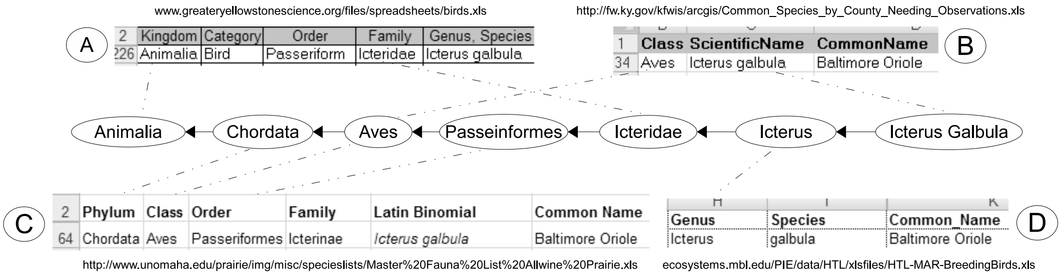

www.greateryellowstonescience.org/files/spreadsheets/birds.xls
http://fw.ky.gov/kfwis/arcgis/Common_Species_by_County_Needing_Observations.xls
A
2 Kingdom Category Order Family Genus, Species
226 Animalia Bird Passeriform Icteridae Icterus galbula
1 Class ScientificName CommonName
34 Aves Icterus galbula Baltimore Oriole
B
Animalia
Chordata
Aves
Passeinformes
Icteridae
Icterus
Icterus Galbula
C
2 Phylum Class Order Family Latin Binomial Common Name
64 Chordata Aves Passeriformes Icterinae Icterus galbula Baltimore Oriole
Genus Species Common_Name
Icterus galbula Baltimore Oriole
D
http://www.unomaha.edu/prairie/img/misc/specieslists/Master%20Fauna%20List%20Allwine%20Prairie.xls
ecosystems.mbl.edu/PIE/data/HTL/xlsfiles/HTL-MAR-BreedingBirds.xls

Fig. 6. Taxonomy chain created from spreadsheets

.

of properties representing the six exploratory questions. For example, the properties to characterize a specimen (`bio:species`, `gs:inFamily`, `gs:inOrder` etc.) are sub-properties of the `eq:what` property and so on. This property classification allows using the questions as a key for articulation. Collection instances can be articulated with specimen characterization instances around *what* properties, because their occurrence in both sides indicates common information – a specimen collected in one side is the specimen characterized in the other side.

4.2 Practical Experiments and Prototype

The practical experiment presented in this section aims to demonstrate the potential for integration and linkage of data extracted from spreadsheets when its nature is recognized and made explicit. They involved gathering approximately 11,000 spreadsheets from the Web. Spreadsheets were located through the Google search engine, using keywords in biology domain. The system automatically recognized and mapped 1,151 spreadsheets from which 806 were classified as Object spreadsheets and 345 were classified as Event spreadsheets. The process recognized 3 different kingdoms, 51 phyla and 33,808 species. Also, 55,248 different collection items were recognized, with 48,034 georeferences.

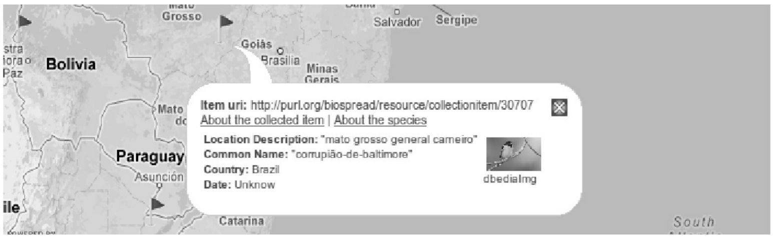
In order to evaluate the precision/recall of schemas recognition, we selected a random subset sample of 1,203 spreadsheets, which we annotated. The recognition percentages were approximately the same as those of the larger group. Our algorithm achieved a precision of 0.84 – i.e., 84% of the retrieved spreadsheets were relevant – and recall of 0.76 – the system recognized 76% of all relevant spreadsheets – having a F-measure of 0.8. It achieved the accuracy of 93% and the specificity of 95%.

As mentioned in subsection 3.2, the instance mapping and enrichment adopt existing matching algorithms. However, the recognized schema can support better instance mapping, as the system identifies instances inside a class context. The taxonomic tree built by the system with data from spreadsheets illustrates how the schema enhances the instance integration[3].

Figure 6 shows a chain of the taxonomic tree, populated combining partial fragments of spreadsheets "A", "B", "C" and "D". In spreadsheet "A", the system links the `Animalia` Kingdom and the `Icteridae` Family. The `Category` label is not recognized, because it does not belong to the set of terms recognized by the system. The instance in spreadsheet "B" links the `Aves` Class to the `Icterus galbula` Species – whose `ScientificName` term is related to the `Species` lemma – and the `Baltimore Oriole` Common Name. In spreadsheet "C", there are instances linking the `Chordata` Phylum, the `Aves` Class, the `Passeriformes` Order, the `Icteridae` Family and the `Baltimore Oriole` Common Name. Finally, in spreadsheet "D", the `Icterus` Gender is linked to `Baltimore Oriole` Common Name.

Even though each spreadsheet has a partial fragment of the chain, their recognized schemas support instances integration and linking. Since spreadsheets were captured from several repositories on the

[3] See `http://purl.org/biospread/?task=pages/txnavigator`



Item uri: http://purl.org/biospread/resource/collectionitem/30707
About the collected item | About the species
Location Description: "mato grosso general carneiro"
Common Name: "corrupião-de-baltimore"
Country: Brazil
Date: Unknow
dbediaImg

Fig. 7. Screenshot of the query interface of the prototype

.

Web, we observed that the instances showed greater diversity of format and data quality problems, whose proper integration is beyond the scope of this work and may be addressed in future work.

Many spreadsheets were not recognized due to the strategy adopted to locate them through a search engine, which returns many spreadsheets out of the context. The recognized nature of each record guided the application of consistent operations over it. In particular, it was possible to merge all catalog-typed records, extracted from the spreadsheets. Figure 7 shows a screenshot of our query and visualization prototype for data extracted from spreadsheets[4]. This interface represents the last step of Figure 2.

By recognizing the spreadsheet nature, it was possible to articulate data collected in the field with data describing species. Moreover, data of the same species were merged and aggregated in different levels taxonomic classification: kingdom, phylum, class, order, family, genus and species. Each aggregation level is filtered by a respective drop down box of the system's interface, as illustrated in Figure 7. Every time the user characterizes a taxonomic level – e.g., by selecting a specific kingdom – the system will filter the records of the respective level. The georeferenced records are plotted over an interactive map – see bottom of Figure 7 – and might be automatically related to the Geospecies database. In this prototype version the data will be plotted over the map only when the user defines all taxonomic levels in the interface. It has an interactive exploitation interface in JavaScript, using the OpenLayers framework for maps (http://openlayers.org).This prototype is available in http://purl.org/biospread/ and both ontology and resources (instances) resulted from the practical experiment can be accessed through http://purl.org/biospread/resource/ and http://purl.org/biospread/ontology/ respectively.

Each flag in the map of Figure 7 represents a data record collected in the field about a specimen. When the user clicks on the flag, it shows the respective record and enables linking this data with a summary of all data available concerning the informed species, by articulating data of this specific record with data coming from the same and other spreadsheets regarding the same species.

5. CONCLUSION AND FUTURE WORK

Spreadsheets have had great acceptance among users of various segments, becoming "popular databases" arranged in files, which are difficult to integrate. To tackle this problem, many authors proposed solutions that recognize implicit schemas and map them to Semantic Web representations. Our work differs from previous works by considering the context in which the spreadsheet was conceived essential to delineate the set of practices shared by the respective community, establishing building patterns to be automatically recognized by our system, to make schemas explicit.

We have implemented a prototype system, presented in this article, which can recognize and integrate schemas from hundreds of spreadsheets belonging to biology domain and obtained from the Web. By recognizing the spreadsheet nature, reflecting their semantics in the produced data, the system is able to perform consistent combinations among them. This is a preliminary experiment of

[4] Available at `http://purl.org/biospread/?task=pages/txnavigator`

data integration. We are aware of its limitations, especially regarding the quality of data coming from various sources. However, it demonstrates its potential for data integration.

This research has raised new challenges to be investigated, such as automatically discovering of articulation possibilities for data coming from different spreadsheets – even for data of different natures – and their respective integration. Such integration will enable inferences that emerge from the combination of these data and which could not be obtained from an analysis of documents individually.

In this article, we showed that in the biology domain, when the system identifies a construction pattern of spreadsheets we can infer their respective nature. We intend to generalize this association and to demonstrate that a construction pattern inside a domain may imply in the identification of the spreadsheet nature.

REFERENCES

Abraham, R. and Erwig, M. Inferring Templates from Spreadsheets. In *Proceedings of the International Conference on Software Engineering*. Shanghai, China, pp. 182–191, 2006.

Bernardo, I. R., Mota, M. S., and Santanchè, A. Extraindo e Integrando Semanticamente Dados de Múltiplas Planilhas Eletrônicas a Partir do Reconhecimento de sua Natureza. In *Proceedings of the Brazilian Symposium on Databases*. São Paulo, Brazil, pp. 256–263, 2012.

Bernardo, I. R., Santanchè, A., and Baranauskas, M. C. C. Reconhecendo Padrões em Planilhas no Domínio de Uso da Biologia. In *Proceedings of the Brazilian Symposium on Information System*. São Paulo, Brazil, pp. 360–371, 2012.

Chambers, C. and Scaffidi, C. Struggling to Excel: a field study of challenges faced by spreadsheet users. In *IEEE Symposium on Visual Languages and Human-Centric Computing*. Madrid, Spain, pp. 187–194, 2010.

Han, L., Finin, T., Parr, C., Sachs, J., and Joshi, A. RDF123: from spreadsheets to RDF. In A. Sheth, S. Staab, M. Dean, M. Paolucci, D. Maynard, T. Finin, and K. Thirunarayan (Eds.), *The Semantic Web*. Lecture Notes in Computer Science, vol. 5318. Springer, pp. 451–466, 2008.

Hermans, F., Pinzger, M., and van Deursen, A. Automatically Extracting Class Diagrams from Spreadsheets. In T. DHondt (Ed.), *Object-Oriented Programming*. Lecture Notes in Computer Science, vol. 6183. Springer, pp. 52–75, 2010.

Jang, S., Ko, E.-J., and Woo, W. Unified User-Centric Context: who, where, when, what, how and why. In *Proceedings of the International Workshop on Personalized Context Modeling and Management for UbiComp Applications*. Tokyo, Japan, pp. 26–34, 2005.

Jannach, D., Shchekotykhin, K., and Friedrich, G. Automated Ontology Instantiation from Tabular Web Sources - the AllRight System. *Journal of Web Semantics* 7 (3): 136–153, 2009.

Langegger, A. and Wöss, W. XLWrap – Querying and Integrating Arbitrary Spreadsheets with SPARQL. In A. Bernstein, D. Karger, T. Heath, L. Feigenbaum, D. Maynard, E. Motta, and K. Thirunarayan (Eds.), *The Semantic Web*. Lecture Notes in Computer Science, vol. 5823. Springer, pp. 359–374, 2009.

Limaye, G., Sarawagi, S., and Chakrabarti, S. Annotating and Searching Web Tables using Entities, Types and Relationships. *Proceedings of the VLDB Endowment* 3 (1-2): 1338–1347, 2010.

Mota, M. S., Oliveira, N., Costa, D. P., Santanchè, A., and Dalforno, C. Geração Semanticamente Dirigida e Apresentação Dinâmica de Objetos digitais complexos na web. In *Anais do Workshop de Trabalhos de Iniciação Científica do Simpósio Brasileiro de Sistemas Multimídia*. Fortaleza, Ceará, Brazil, pp. 1–3, 2009.

O'Connor, M., Halaschek-Wiener, C., and Musen, M. Mapping Master: a flexible approach for mapping spreadsheets to OWL. In *Proceedings of the International Semantic Web Conference*. Shanghai, China, pp. 194–208, 2010.

Ponder, W. F., Carter, G. A., Flemons, P., and Chapman, R. R. Evaluation of Museum Collection Data for Use in Biodiversity Assessment. *Conservation Biology* 15 (3): 648–657, 2001.

Santanchè, A. and Silva, L. A. M. Document–Centered Learning Object Authoring. *IEEE Learning Technology Newsletter* vol. 12, pp. 58–61, 2010.

Scaffidi, C., Shaw, M., and Myers, B. Estimating the Numbers of End Users and End User Programmers. In *IEEE Symposium on Visual Languages and Human-Centric Computing*. Dallas, Texas, USA, pp. 207 – 214, 2005.

Syed, Z., Finin, T., Mulwad, V., and Joshi, A. Exploiting a Web of Semantic Data for Interpreting Tables. In *Proceedings of the Web Science Conference*. Raleigh, North Carolina, USA, 2010.

Venetis, P., Halevy, A., Madhavan, J., Paşca, M., Shen, W., Wu, F., Miao, G., and Wu, C. Recovering Semantics of Tables on the Web. *Proceedings of the VLDB Endowment* 4 (9): 528–538, 2011.

Yang, S., Bhowmick, S. S., and Madria, S. Bio2X: a rule-based approach for semi-automatic transformation of semi-structured biological data to XML. *Data & Knowledge Engineeering* 52 (2): 249–271, 2005.