An ontology-based data integration approach for web analytics in e-commerce

María del Mar Roldán García, José García-Nieto [*], José F. Aldana-Montes [1]

Dept. de Lenguajes y Ciencias de la Computación, University of Málaga, ETSI Informática, Campus de Teatinos, Málaga - 29071, Spain

Keywords:
Semantic model
Ontology
E-commerce
Web analytics

ABSTRACT

Web analytics has emerged as one of the most important activities in e-commerce, since it allows companies and e-merchants to track the behavior of customers when visiting their web sites. There exist a series of tools for web analytics that are used not only for tracking and measuring web traffic, but also for analyzing the commercial activity. However, most of these tools focus on low level web attributes and metrics, making other sophisticated functionalities and analyses only available for commercial (non-free) versions.

In this context, the SME-Ecompass European initiative aims at providing e-commerce SMEs with accessible tools for high level web analytics. These software facilities should use different sources of data coming from digital footprints allocated in e-shops, to fuse them together in a coherent way, and to make them available for advanced data mining procedures. This motivated us to propose in this work an ontology-based approach to collect, integrate and store web analytics data, from many sources of popular and commercial digital footprints. As article's main impact, we obtain enriched and semantically annotated data that is used to properly train an intelligent system, involving data mining procedures, for the analysis of customer behavior in real e-commerce sites. In concrete, for the validation of our semantic approach, we have captured and integrated data from Google Analytics and Piwik digital footprints allocated in 15 e-shops of different commercial sectors and countries (UK, Spain, Greece and Germany), throughout several months of activity. The obtained results show different perspectives in customer's behavior analysis that go one step beyond the most popular web analytics tools in the current market.

1. Introduction

In the last few years, web analytics has emerged as one of the most important activities in e-commerce, since it allows companies and e-merchants to track the behavior of customers when visiting their e-shop sites. Web analytic applications can also help companies to measure the results of traditional print or broadcast advertising campaigns. Web analytics procedure is based on measuring a visitor's behavior once on a given e-shop site, which includes its drivers and conversions (to actual customer). These data are typically compared against key performance indicators and used to improve a website or marketing campaign's audience response.

In the current market, there exist a series of tools for web analytics, such as: Google Analytics, Piwik, Clicky, and StatCounter; that are widely used not only for tracking and measuring web traffic, but also for analyzing the commercial activity, hence to improve the effectiveness of a website. However, these tools often focus on low level and limited sets of web metrics and attributes, without the possibility of providing specialized analyses. In most of cases, high level web metrics and sophisticated functionalities are available only for commercial (non-free) versions, which are rarely accessible by SMEs or individual e-merchants.

In this context, the SME-Ecompass European initiative[2] aims at providing e-commerce SMEs with accessible tools for high level web analytics. These software facilities use the different sources of data coming from different digital footprints allocated in e-shops. However, integrating data from multiple heterogeneous sources entails dealing with different data models, schema and query languages. Therefore, there is a clear demand of integrative proce-

[*] Corresponding author.
E-mail addresses: mmar@lcc.uma.es (M.d.M.R. García), jnieto@lcc.uma.es, jmgarcianieto@gmail.com (J. García-Nieto), jfam@lcc.uma.es (J.F. Aldana-Montes).

[1] This work is partially funded by FP7 EU project SME E-COMPASS under Grant No: 315637. It is also partially funded by Grants TIN2014-58304 (Spanish Ministry of Sciences and Innovation) and Regional projects P11-TIC-7529/P12-TIC-1519. Authors thanks to involved e-shops to kindly offer web tracking data for testing and validation.

[2] SME-Ecompass FP7 European initiative http://www.sme-ecompass.eu/

dures for providing the advanced data mining algorithms with a uniform access to multiple heterogeneous web data sources.

The main hypothesis in this work is: **(H1) an ontology-based integration approach will help us to collect, fuse the data together in a coherent way, and store web analytics data, from many sources of popular and commercial digital footprints**. As a result, **(H2) we will obtain enriched and semantically annotated data that will be able to train data mining procedures for advanced analysis of customer behavior in real e-commerce sites**.

This motivated us to propose a semantic approach that uses an ontology as a mediated schema for the representation and consolidation in a knowledge base of the tracking data from web source's semantics. Semantic web ontologies become a key technology for intelligent knowledge processing, providing a framework for sharing conceptual models about a domain. Semantic mappings between the source schema and the ontology are then defined and used to transform the original data to RDF (Resource Description Framework) [3]. This way, data from heterogeneous sources are stored and integrated inside a single RDF repository, which can be now easily queried by high level algorithms. The goal is to properly feed artificial intelligence procedures capable of deciding how to perform marketing activities, such as: displaying a given advertisement targeted to certain category of clients, or decreasing the price of a product in a given region; then giving rise to sophisticated expert systems for e-commerce applications.

The main contributions of this study are summarized as follows:

- We have developed a semantic approach for the data integration and consolidation of multiple web analytics data sources. These data are daily accumulated from many heterogeneous digital footprints allocated on actual e-shops.
- We have designed and implemented for the first time an OWL (Web Ontology Language) ontology (Dean & Schreiber, 2004) for web analytics. This ontology considers a large and complemented set of attributes and metrics, which have been token from several representative web analytics tools in the market.
- To test hypothesis H1, we have captured and integrated data from Google Analytics and Piwik digital footprints allocated in 15 e-shops of different commercial sectors (retail, tourism, electronics, pharmacy, etc.) and countries (UK, Spain, Greece and Germany), throughout several months of activity. The data are integrated following the same (standard) format and stored in a common RDF repository.
- To test hypothesis H2, obtained "semantized" data are used to train advanced data mining algorithms to perform customer's profile analyses. In particular, these algorithms are tested with success in two cases of study to classify the visitor's behavior and product preference.

The remaining of this article is organized as follows. In Section 2, background and literature overview are presented. Section 3 presents the current state and practices in web analytics for e-commerce. In Section 4, the semantic approach is described, giving details of the service architecture and the OWL ontology. After this, the validation procedure is reported in Section 5. Finally, main conclusions and future work are given in Section 6.

2. Background and related work

This section describes the main background concepts. A review of current related works in the specialized literature is carried out to point out their main differences with regards to our approach.

2.1. Background concepts

- *Ontology*. Ontologies provide a formal representation of the real world, shared by a sufficient amount of users, by defining concepts and relationships between them (Gruber, 1993). In computer and information sciences, an ontology defines a set of representational primitives with which to model a domain of knowledge or discourse. These primitives are typically concepts (classes), attributes (properties), class members (class instances) and relationships (property instances). The definitions of the representational primitives include information about their meaning and constraints on their logically consistent application.

Ontologies are part of the W3C standards stack for the semantic web, in which they are used to specify standard conceptual vocabularies in which to exchange data between systems, provide services for answering queries, publish reusable knowledge bases, and offer services to facilitate interoperability across multiple, heterogeneous systems and databases.

- *RDF*. Resource Description Framework is a basic ontology language used for representing information about resources on the web (Staab & Studer, 2009). Resources are described in terms of properties and property values using RDF statements. Statements are represented as triples, consisting of a subject, predicate and object. RDF Schema (Staab & Studer, 2009) (RDFS) "semantically extends" RDF to enable us to talk about classes of resources, and the properties that will be used with them. It does this by giving particular meanings to certain RDF properties and resources. RDFS provides the means to describe application specific RDF vocabularies. RDF and RDFS provide basic capabilities for describing vocabularies that describe resources, metadata and ontologies.

- *SPARQL*. It is an RDF query language for ontology models and databases, capable of extracting and manipulating information stored in RDF format. Essentially, SPARQL is a graph-matching query language that can be used to extract knowledge from the model such as the one proposed in this article. Given a data source D, a query consists of a pattern, which is matched against D. The combinations of values resulting from this matching constitute the result of the query (Pérez, Arenas, & Gutierrez, 2009). SPARQL has strong support for querying semi-structured and tagged data, e.g. data with an unpredictable and unreliable structure. SPARQL supports queries to networked, web data sources identified by URIs. In fact, it is a W3C recommendation for RDF data.

- *OWL*. In 2004, the W3C ontology working group (Dean & Schreiber, 2004) proposed OWL as a semantic markup language for publishing and sharing ontologies on the World Wide Web. From a formal point of view, OWL is equivalent to a very expressive description logic where an ontology corresponds to a Tbox (Gruber, 1993). This equivalence allows the language to exploit description logic researcher results. OWL extends RDF and RDFS. When compared to RDF models, OWL adds more vocabulary for describing properties and classes: relations between classes (e.g. disjointedness), cardinality (e.g. "exactly one"), equality, richer typing of properties, characteristics of properties (e.g. symmetry), and enumerated classes (McGuinness & Harmelen, 2004).

- *OWL-DL*. Syntactic variant of the SHOIN (D) description logic (Haase & Stojanovic, 2005) with a different terminology to OWL, which is based on RDFS, hence the support for data values, data types and data type properties. OWL-DL restricts OWL into two distinct ways (Horrocks & Patel-Schneider, 2003): first, some syntactic constructs like recursive descriptions in them are not allowed; second, classes, individuals and properties (respectively concepts, individuals and roles in description logics) must all be disjoint. In this work, we use OWL-DL syntax to formalize the proposed ontology here for our semantic model. A summarized description of basic OWL-DL semantics syntax is shown in Table 1, where an informal logic syntax is represented (left

[3] RDF in W3C https://www.w3.org/RDF/

Table 1
Basic OWL-DL semantic syntax used to formally define the proposed ontology.

Descriptions	Abstract syntax	DL syntax
Operators	$intersection(C_1, C_2, \cdots, C_n)$ $union(C_1, C_2, \cdots, C_n)$	$C_1 \sqcap C_2 \sqcap \cdots C_n$ $C_1 \sqcup C_2 \sqcup \cdots C_n$

Restrictions	for at least 1 value V from C
	for all values V from C
	R is symmetric
		$\exists V.C$
		$\forall V.C$
		$R \equiv R^-$

Class Axioms	$A\ partial(C_1, C_2, \cdots, C_n)$ $A\ complete(C_1, C_2, \cdots, C_n)$	$A \sqsubseteq C_1 \sqcap C_2 \sqcap \cdots C_n$ $A \equiv C_1 \sqcap C_2 \sqcap \cdots C_n$

column) with regards to the corresponding OWL-DL equivalent (right).

2.2. Related works

In the last decade, a series of studies have been appearing that semantically model certain domains or sub-domains of knowledge in the context of e-commerce.

A first attempt was proposed in Trastour, Bartolini, and Preist (2003), where a service description language was defined to be used throughout the life-cycle of a business-to-business (B2B) e-commerce interaction. In particular, they focused on DAML+OIL, as it is a sufficiently expressive and flexible service description language to be used not only in advertisements, but also in matchmaking queries, negotiation proposals and agreements.

After this, Tamma et al. (Tamma, Phelps, Dickinson, & Wooldridge, 2005) designed an approach to negotiation activities in e-commerce sites. In this work, the negotiation protocol does not need to be hard-coded in agents, but it is represented by an ontology, in terms of an explicit and declarative representation of the negotiation protocol. The ontology is also used to tune agents strategies to the specific protocol used.

A special case of e-commerce sites is e-tourism, for which Waralak (Waralakv, 2008) discussed some ontological trends that support the growing domain of online tourism. This study also gave some example concepts of existing e-tourism using ontologies display in graphical model and showed their descriptions in OWL and RDFS syntax.

Hepp defined two related ontologies (Hepp, 2008): GoodRelations and Product Ontology. GoodRelations is a standardized vocabulary for e-commerce (product, price, store, and company data) and the Product Ontology is an ontology for describing product types based on Wikipedia.

More recently, Gatchalee et al. (Gatchalee, Li, & Supnithi, 2013) proposed an ontology approach to cover the knowledge about the content and architecture of SMEs e-commerce websites. This knowledge is then used as input to a recommendation system for web design, that centered on the structure of e-shops in Thailand as sample group.

Finally, Akanbi (Akanbi, 2014) proposed LB2CO, an ontology which combines the framework of IDEF5 & SNAP as an analysis tool, for automated recommendation of product and services. This ontology is used to model a semantic framework for B2C transactions across different business domains that facilitates the interoperability and integration of transactions over the web.

As summarized in Table 2, all these works proposed semantic models focusing on different aspects in the domain of e-commerce, such as: web contents, structure, and life-cycle activities. However, to the best of our knowledge, there is still a lack of works where a semantic model is used to consider web analytic attributes and metrics from multiple and heterogeneous sources of data. This is a critical issue for current e-merchants that we try to cope, for the first time, with our approach.

In contrast to other past proposals, we validate our semantic model with real data coming from digital footprints and web scraping methods in a number of actual e-shops. As a result, some of these e-commerce SMEs have put in practice the generated data and analyses, leading them to change and improve their commercial strategies.

3. E-commerce web analytics: current practices

Previously to describe our semantic approach, we summarize in this section a series of activities we carried out in the context of SME-Ecompass project, with the aim of shedding light on the actual state of e-commerce companies.

In a first phase, we delivered an online survey to e-shop's owners (of associated chambers to SME-Ecompass project) with different questions with regards to: their company's profile, commerce activity, current practices when analyzing customer's and competitor's behaviors, etc. After that, face-to-face interviews were also conducted with a selection of e-merchants in order to obtain detailed information of their professional experience in e-commerce. A complete report of these questionnaires with the analysis of responses, statistics and conclusions can be found in Garía-Nieto and Roldán (2014). In concrete, more than 150 e-commerce SMEs completed the online surveys and 20 out of them were interviewed in private sessions. The following conclusions were extracted:

- Most of studied companies are micro enterprises with 1 to 5 employees and work in Business to Consumer (B2C) retailing sector. Most of them have a maximum number of 5000 orders per year (2013/2014) and a maximum annual revenue of 10,000 euros from online sales. Therefore, they are target candidates to be beneficiary of automatic (free or non-expensive) tools to obtain advanced e-commerce analysis.
- Concerning visitor's behavior analyses (see Fig. 1 left), 47% of companies do not use any tool for these kind of tasks. On the contrary, a percentage of 29% declared that they use automatic online tools and 22% make these tasks manually. Of course, most of them (> 80%) declared to be quite interested on using a service to discover tendencies and common habits in clients.
- Interestingly, as shown in Fig. 1 (right), it can be stood out that Google Analytics is the most used tool of interviewed companies (68%), although they also use additional tools like Piwik (16%) and other (16%). Therefore, the set of common metrics e-merchants usually analyze are those computed by Google Analytics, e. g., number of visits, average visit time, geo-localization, country, client device, etc. These metrics are usually checked weekly or monthly.

In the light of these results, we decided to focus our semantic model on attributes and metrics provided by Google Analytics and Piwik. We selected the former for being the most used analytic tool in the market. However, Google Analytics e-commerce advanced functionality is (to the date) not available for free users. This issue led us to complement our set of attributes with those of Piwik, since this last tool offers free access to advanced e-commerce attributes. Additional metrics for competitor and price monitor are also considered in this work, which are gathered from specific web scrapping processes of SME E-Compass tool. It is worth noting that the proposed ontology is aimed at covering as general as possible web analytic attributes, hence enabling the incorporation of new analytics tools in our semantic approach.

4. Semantic approach

One of the main aims of this work is to capture, clean, consolidate and integrate data from different sources of web tracking in

Table 2
Related approaches in the state of the art. The target area of application, the used ontology/vocabulary and the post-processing analysis, and the validation procedure are reported for each work.

Approach	Target area	Ontology/vocabulary	Analysis	Validation
Trastour et al. (2003)	Life-cycle in B2B	DAML+OIL	-	No
Tamma et al. (2005)	Negotiation in e-commerce	Negotiation protocol	Agents system	No
Waralakv (2008)	E-tourism	E-tourism concepts	Recommender	No
Hepp (2008)	Products in e-commerce	GoodRelations	-	Academic
Gatchalee et al. (2013)	E-commerce sites design	Website design structure	Recommender	Academic
Akanbi (2014)	B2C transactions	LB2CO	Recommender	Academic
Proposal	Web analytics digital footprints (Google/Piwik)	Web analytics Ontology (WAO)	Data mining Visitor/product	Real world

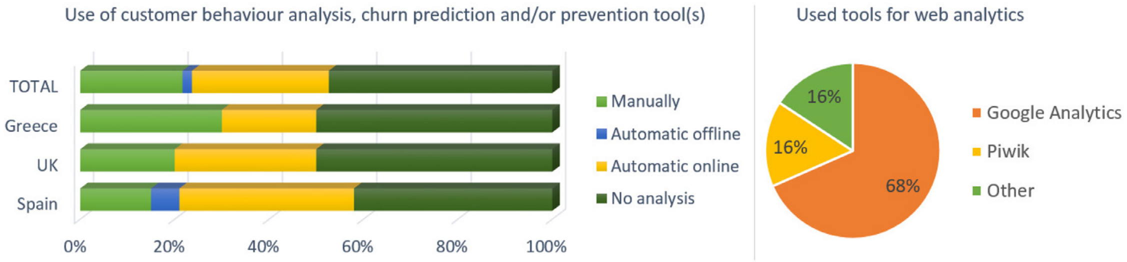


Fig. 1. Current practices in surveyed e-commerce SMEs with regards to the use of automatic tools for web analytics.

e-commerce sites. For this reason, we opted to design a semantic approach for sharing and reconciliation, whereby an agreed ontology model is used to archive a common understanding of the domain in which the system operates. In concrete, we have developed an OWL ontology to describe the e-shops main features by following the standard ontology 101 development process (Natalya, McGuinness, & Deborah, 2001) of seven steps:

(i) *Determine the domain and scope of the ontology*. As the starting point, to limit the scope of the ontology, we selected the kind of variables that the data mining algorithms need from Google Analytics and Piwik and also from competitors e-shops, for instance: visitors origin, visitors attributes, purchasing behavior, product and customer details, etc.

(ii) *Consider reusing existing ontologies*. As we examined in Section 2.2, there are no similar ontologies that have been previously proposed for modeling web tracking data in e-commerce. However, we partially considered two related ontologies: GoodRelations (Hepp, 2008), which is a standardized vocabulary for e-commerce and the Product Ontology (Hepp, 2008), which contextualizes product types based on Wikipedia.

(iii) *Enumerate important terms in the ontology*. Important terms in the ontology were extracted in a previous phase of requirements specification (Garía-Nieto & Roldán, 2014) from the minimum set of variables that are needed. Examples of such terms are: address, visitor, customer, device, browser, Geographical_origin, Number_of_visitors, Conversion_rate, etc.

(iv) *Define classes and the class hierarchy*. From the list of terms, we obtained the ontology classes. Fig. 2 shows the main set of classes in the hierarchy starting from the top class *Thing* (⊤). These main classes are related to other classes and some of them have subclasses. For instance, *Analytic_parameters* has a series of subclasses, such as: *Bounce_rate, Total_revenue, Number_of_returning_visitors*, and *Number_of_transactions*.

(v) *Define the properties of classes and slots*. In order to relate classes and to define attributes, we identified objects and data type properties based on the minimum set of variables previously defined. Examples of object properties are: an e-shop owner is owner of an e-shop, a visitor makes visits, a device has a browser, an IP address belongs to an organization, etc. Examples of data type properties are the title and URL of a page, the first and last name of an e-shop owner, the version of the operating system, the duration of a visit, etc. An object property is defined for each subclass to establish the correct relationship. For example, *Page* is related to *Bounce_rate* and *Date_of_last_visit; E-shop* is related to *Number_of_customers*. Tables 3–8, describe in OWL-DL representative subsets of object and data properties of a selection of the main classes.

(vi) *Define the facets of the slots*. This step includes the definition of cardinality constraints and value restrictions. Value restrictions are used in our ontology to specify the data type for the value in each subclass of the *Analytic_parameters* class. For example, the range of the property *hasValue* is restricted to *float*, when the class *Bounce_rate* is its domain; the range of the property *hasValue* is restricted to date, when the class *Date_of_last_visit* is its domain.

(vii) *Create instances*. Instances (individuals in OWL) correspond to the specific data obtained from a specific e-shop. Individuals will be obtained by mapping the data from Google Analytics, Piwik or competitors e-shops to RDF according to the ontology. Individuals can be also used in the ontology to define the exact members of a class. The range of the property *hasType* is restricted to values: ”ASIN, ”EAN or ”ISBN, when its domain is Article_number. ASIN, EAN and ISBN are then ontology individuals (see Table A.5 for further explanations).

4.1. Ontology model

The proposed ontology, called “wao.owl” (Web Analytics Ontology), resulting from the development process described above has a total of 62 classes (groups of individuals sharing the same attributes), 61 object properties (binary relationships between individuals), and 67 data properties (individual attributes), 33 restriction axioms and 3 individuals. The complete ontology is available in WebProtégé repository.[4]

[4] URL link http://stanford.io/1XHhHzr

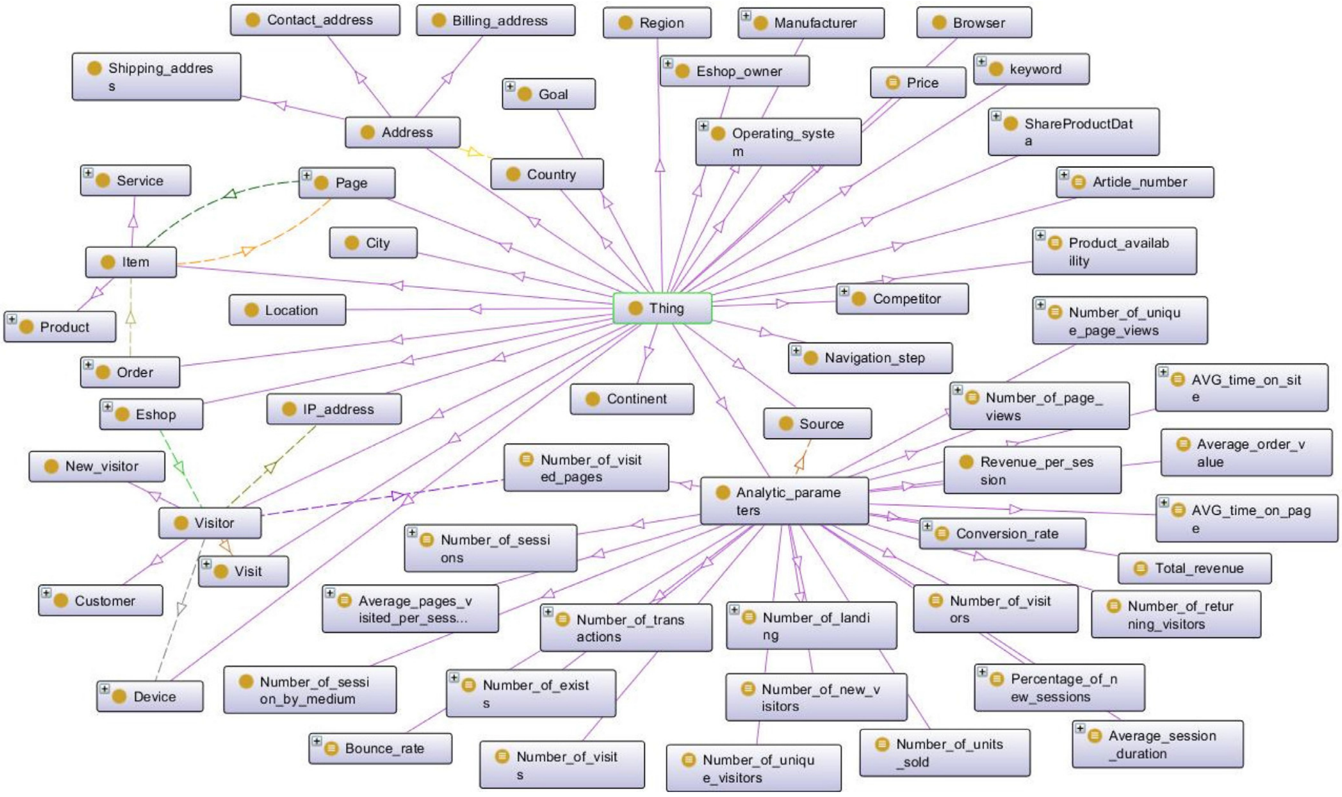


Fig. 2. General overview of the WAO ontology. Continuous arrows refer to sub class of. Dotted arrows refer to specific properties.

Table 3
Analytics_parameters group: object and data properties.

Object properties	Description logic
hasBrowser	∃ hasBrowser.Thing ⊑ Analytic_parameters ⊔ Device ⊤ ⊑ ∀ hasBrowser.Browser
hasCity	∃ hasCity.Thing⊑ Analytic_parameters ⊔ Location ⊔ Visitor ⊤ ⊑ ∀ hasCity.City
hasRegion	∃ hasRegion.Thing ⊑ Analytic_parameters ⊔ Location ⊔ Visitor ⊤ ⊑ ∀ hasRegion.Region
hasCountry	∃ hasCountry.Thing ⊑ Analytic_parameters ⊔ Location ⊔ Visitor ⊤ ⊑ ∀ hasCountry.Country
hasContinent	∃ hasContinent.Thing ⊑ Analytic_parameters ⊔ Location ⊤ ⊑ ∀ hasContinent.Continent
hasSource	∃ hasSource.Thing ⊑ Analytic_parameters ⊤ ⊑ ∀ hasSource.Source
Data Properties	Description Logic
hasDate	∃ hasDate.DatatypeLiteral ⊑ Analytic_parameters ⊔ Price ⊔ Product_availability ⊤ ⊑ ∀ hasDate.DatatypedateTimeStamp
hasHour	∃ hasHour.DatatypeLiteral ⊑ Analytic_parameters ⊤ ⊑ ∀ hasHour.Datatypetime
hasNetworkDomain	∃ hasNetworkDomain.DatatypeLiteral ⊑ Analytic_parameters ⊤ ⊑ ∀ hasNetworkDomain.Datatypestring
hasValue	∃ hasValue.DatatypeLiteral ⊑ Analytic_parameters ⊔ Article_number ⊔ Price ⊔ Product_availability

For simplicity, we describe here a representative subset of main classes including some of their most interesting object and data properties. These classes are: *Analytics_parameters, E-shop, Visitor, Page*, and *Item*. Each class requires a set of properties or conditions in order to be conceptualized. That is, an individual that satisfies those properties is considered to be a member of that class.

- *Analytics_parameters*. Those attributes provided by Google Analytics and Piwik that depend on time. Each analytic parameter has a value (*hasValue* in Table 3), which corresponds to the data provided by the analytic tool, and a date (*hasDate*), which corresponds to the date when the data was obtained. Subclasses in the ontology (⊑ Analytic_parameters) are, among others: *Average_order_value, Average_pages_visited_per_session, Average_session_duration, Average_time_on_site, Bounce_ rate, Conversion_rate, Number_of_transactions, Number_of_landings, Number_of_new_visitors, Number_of_page_views, Revenue_per_ session* and *Total_revenue*. Table 3 shows some representative object and data properties of *Analytics_parameters*. Each analytic parameter belongs to a data type. For instance, the value of *Number_of_transactions* is an non-negative integer and the value of *Conversion_rate* is a float. Data type restrictions are included in the ontology by means of data properties.

- *E-shop.* An *e-shop* has one or several pages and also an e-shop's owner. Each e-shop's owner has an address.

Table 4
E-shop group: object and data properties.

Object properties	Description logic
hasVisitor	> ≡ makesVisit > ⁻ ∃ hasVisitor.Thing ⊑ E-shop ⊤ ⊑ ∀ hasVisitor.Visitor
hasNumberOfVisitors	∃ hasNumberOfVisitors.Thing ⊑ E-shop ⊔ Page ⊤ ⊑ ∀ hasNumberOfVisitors.Number_of_visitors
hasNumberOfVisits	∃ hasNumberOfVisits.Thing ⊑ E-shop ⊔ Page ⊔ Visitor ⊤ ⊑ ∀ hasNumberOfVisits.Number_of_visits
isOwnerOf	∃ isOwnerOf.Thing ⊑ E-shop_owner ⊤ ⊑ ∀ isOwnerOf.E-shop
Data properties	Description logic
hasName	∃ hasName.DatatypeLiteral ⊑ Browser ⊔ Competitor ⊔ E-shop ⊔ Goal ⊔ Item ⊔ Operating_system ⊔ Page ⊔ Product ⊤ ⊑ ∀ hasName.Datatypestring
hasURL	∃ hasURL.DatatypeLiteral ⊑ Competitor ⊔ E-shop ⊔ Page ⊔ Price ⊤ ⊑ ∀ hasURL.Datatypestring

Table 5
Visitor group: object and data properties.

Object properties	Description logic
hasDevice	∃ hasDevice.Thing ⊑ Visitor ⊤ ⊑ ∀ hasDevice.Device
hasNumberOfVisits	∃ hasNumberOfVisits.Thing ⊑ E-shop ⊔ Page ⊔ Visitor ⊤ ⊑ ∀ hasNumberOfVisits.Number_of_visits
hasCity	∃ hasCity.Thing⊑ Analytic_parameters ⊔ Location ⊔ Visitor ⊤ ⊑ ∀ hasCity.City
makesVisit	hasVisitor> ≡ makesVisit>⁻ ∃ makesVisit.Thing ⊑ Visitor ⊤ ⊑ ∀ makesVisit.Visit
Data properties	Description logic
hasDaysSinceFirstVisit	∃ hasDaysSinceFirstVisit.DatatypeLiteral ⊑ Visitor ⊤ ⊑ ∀ hasDaysSinceFirstVisit.DatatypenegativeInteger
hasDaysSinceLastOrder	∃ hasDaysSinceLastOrder.DatatypeLiteral ⊑ Visitor ⊤ ⊑ ∀ hasDaysSinceLastOrder.DatatypenegativeInteger
hasDaysSinceLastVisit	∃ hasDaysSinceLastVisit.DatatypeLiteral ⊑ Visitor ⊤ ⊑ ∀ hasDaysSinceLastVisit.DatatypenegativeInteger ∃ IsReturningVisitor.DatatypeLiteral ⊑ Visitor ⊤ ⊑ ∀ IsReturningVisitor.Datatypeboolean

Table 6
Visit group: object and data properties.

Object properties	Description logic
hasNavigationStep	∃ hasNavigationStep.Thing ⊑ Visit ⊤ ⊑ ∀ hasNavigationStep.Navigation_step
hasRefererKeyword	∃ hasRefererKeyword.Thing ⊑ Visit ⊤ ⊑ ∀ hasRefererKeyword.Referer_keyword
makesVisit	hasVisitor> ≡ makesVisit>⁻ ∃ makesVisit.Thing ⊑ Visitor ⊤ ⊑ ∀ makesVisit.Visit
Data properties	Description logic
hasDuration	∃ hasDuration.DatatypeLiteral ⊑ Visit ⊤ ⊑ ∀ hasDuration.Datatypetime
hasReturningVisitor	∃ hasReturningVisitor.DatatypeLiteral ⊑ Visit ⊤ ⊑ ∀ hasReturningVisitor.Datatypeboolean

Attributes for the e-shop are latitude, longitude and time zone. The e-shop's owner can have competitors, who are e-shop's owners of other e-shops. The analytic parameters of an e-shop are: *average_order_value, average_pages_visited_per_session, average_ session_duration, average_time_on_site, conversion_rate, date_of_last_transaction, number_of_customers, number_of_failed_transactions, number_of_successful_transactions, number_of_new_customers, number_of_new_visitors, number_of_sessions_by_medium, number_of_transactions, number_of_unique_visitors, number_of_units_sold, number_of_visitors, number_of_visits, percentage_of_new_sessions, revenue_per_session, total_revenue*, and *number_of_returning_visitors*. All the analytic parameters related to an e-shop are time dependent. Therefore, they are modeled as classes and related to the e-shop by the corresponding object property. Table 4 shows a subset of properties with classes in the e-shop group as domain.

- *Visitor and visit*. Class *visitor* has two subclases: *customer* and *New_visitor*. A customer is a visitor who makes a purchase. If it is the first purchase of this customer, he/she is a new customer. Customers have an address and name, whereas visitor do not. A visitor visits the e-shop by using a device. The analytic parameters for visitors are: *bounced_rate, number_of_visits*, and *number_of_visited_pages*. The analytic parameters for customers are *number_of_transactions*. Visitors visit pages.

Visits are essential to capture the behavior of a visitor when visiting the e-shop. A visit has an entry page and an exit page. It also has a referrer page, which is the way the visitor has accessed the site, i.e. search engine, social network, web-advertisement etc. If the referrer page is a search engine, the keywords used to find the site are also associated with the visit. A visit has a given duration. The attributes of a visit are the times when the entry page and the end page were accessed, duration, back link, whether or not an order was placed and the total goals converted during the visit, number of actions, number of events and number of searches. During a visit, transactions are made. A visit follows a path which has a next page, a previous page and a number. The class *Navigation_step* is used to model the path that the user follows from the entry page to the exit page. Each navigation step has only one

Table 7
Page group: object and data properties.

Object properties	Description logic
hasNumberOfVisits	$\exists$ hasNumberOfVisits.Thing $\sqsubseteq$ E-shop $\sqcup$ Page $\sqcup$ Visitor $\top \sqsubseteq \forall$ hasNumberOfVisits.Number_of_visits
hasNumberOfVisitors	$\exists$ hasNumberOfVisitors.Thing $\sqsubseteq$ E-shop $\sqcup$ Page $\top \sqsubseteq \forall$ hasNumberOfVisitors.Number_of_visitors
hasTotalRevenue	$\exists$ hasTotalRevenue.Thing $\sqsubseteq$ E-shop $\sqcup$ Page $\top \sqsubseteq \forall$ hasTotalRevenue.Total_revenue
isOnPage	$\exists$ isOnPage.Thing $\sqsubseteq$ Item $\top \sqsubseteq \forall$ isOnPage.Page
Data properties	Description logic
hasName	$\exists$ hasName.DatatypeLiteral $\sqsubseteq$ Browser $\sqcup$ Competitor $\sqcup$ E-shop $\sqcup$ Goal $\sqcup$ Item $\sqcup$ Operating_system $\sqcup$ Page $\sqcup$ Product $\top \sqsubseteq \forall$ hasName.Datatypestring
hasURL	$\exists$ hasURL.DatatypeLiteral $\sqsubseteq$ Competitor $\sqcup$ E-shop $\sqcup$ Page $\sqcup$ Price $\top \sqsubseteq \forall$ hasURL.Datatypestring
hasTitle	$\exists$ hasTitle.DatatypeLiteral $\sqsubseteq$ Page $\top \sqsubseteq \forall$ hasTitle.Datatypestring

Table 8
Item group: object and data properties.

Object properties	Description logic
hasItem	$\exists$ hasItem.Thing $\sqsubseteq$ Page $\top \sqsubseteq \forall$ hasItem.Item
hasPrice	$\exists$ hasPrice.Thing $\sqsubseteq$ Item $\sqcup$ ShareProductData $\top \sqsubseteq \forall$ hasPrice.Price
includes	$\exists$ includes.Thing $\sqsubseteq$ Order $\top \sqsubseteq \forall$ includes.Item
isOnPage	$\exists$ isOnPage.Thing $\sqsubseteq$ Item $\top \sqsubseteq \forall$ isOnPage.Page
Data properties	Description logic
hasCategory	$\exists$ hasCategory.DatatypeLiteral $\sqsubseteq$ Item $\top \sqsubseteq \forall$ hasCategory.Datatypestring
hasName	$\exists$ hasName.DatatypeLiteral $\sqsubseteq$ Browser $\sqcup$ Competitor $\sqcup$ E-shop $\sqcup$ Goal $\sqcup$ Item $\sqcup$ Operating_system $\sqcup$ Page $\sqcup$ Product $\top \sqsubseteq \forall$ hasName.Datatypestring
hasItemID	$\exists$ hasItemID.DatatypeLiteral $\sqsubseteq$ Item $\top \sqsubseteq \forall$ hasItemID.DatatypenonNegativeInteger
hasQuantity	$\exists$ hasQuantity.DatatypeLiteral $\sqsubseteq$ Item $\top \sqsubseteq \forall$ hasQuantity.DatatypenonNegativeInteger

attribute number. Tables 5 and 6 show the properties with classes in the visitor and visit group as domain, respectively.

- Page. Pages contain items, i.e. product and/or services to be sold. The analytic parameters for Page are: *average_order_value, average_time_on_page, bounce_rate, date_ of_last_ visit, number_of_exits, number_of_landings, number_of_new_visitors, number_ of_page_views, number_of_returning_visitors, number_of_sessions_ by_medium* (mediums are direct link, social media and search engine), *number_of_sessions, number_of_unique_page_views, number_of_unique_visitors, number_of_units_sold, number_of_visitors, number_of_visits, revenue_ per_session_and_total_revenue*. Attributes of page are title and URL. A series of representative properties whose domain is page are shown in Table 7. Interestingly, we can observe in this table that the property *hasTotalRevenue* is related to the *Page*, as well as the to the whole e-shop, as this value can be calculated for both classes.

- Item. As commented before, an Item is a *product* or a *service* which is sold in an *e-shop*. Specific items of an *e-shop* are modeled by defining a domain ontology for a specific domain, i.e., travel, books, music, etc. Table 8 contains some representative object and data properties of class *item*. According to this, Items have a *price* (*hasPrice*). Prices are valid on a certain date. Therefore, attributes for prices are *value, currency* and the date for the price *validity*. The attributes of *Items* are *category* and whether or not it has been deleted. *Products* have a *manufacturer*. The attributes for products are: name, type, availability on a specific date and article number. The article number can be “ASIN”, “EAN” or “ISBN”.

4.2. Data sources: mapping and querying

As we explained in Section 3, we have focused on three main sources of data coming from different web tracking methods, namely: Google Analytics, Piwik, and specific web scrapping methods in the scope of SME E-Compass project.

The process of translating the collected data from different sources to RFD is carried out by means of mapping functions. Each data source has a different set of methods to gather, harmonize, store and provide access to the analytical data. Therefore, a different set of mapping functions is required to parse the information coming from each data source to RDF, according to the ontology. Fig. 3 illustrates an general overview of the mapping process to store data from different sources in a common RDF repository. Each set of mappings is then composed by functions to translate the attributes with their values into their corresponding triple form in RDF. In fact, for most of the attributes, a corresponding mapping function has been developed, involving its corresponding class in the ontology. Nevertheless, as a number of analytic attributes shares a common structure in the ontology, they have been mapped by using generic functions, hence taking advantage of the ontology’s design.

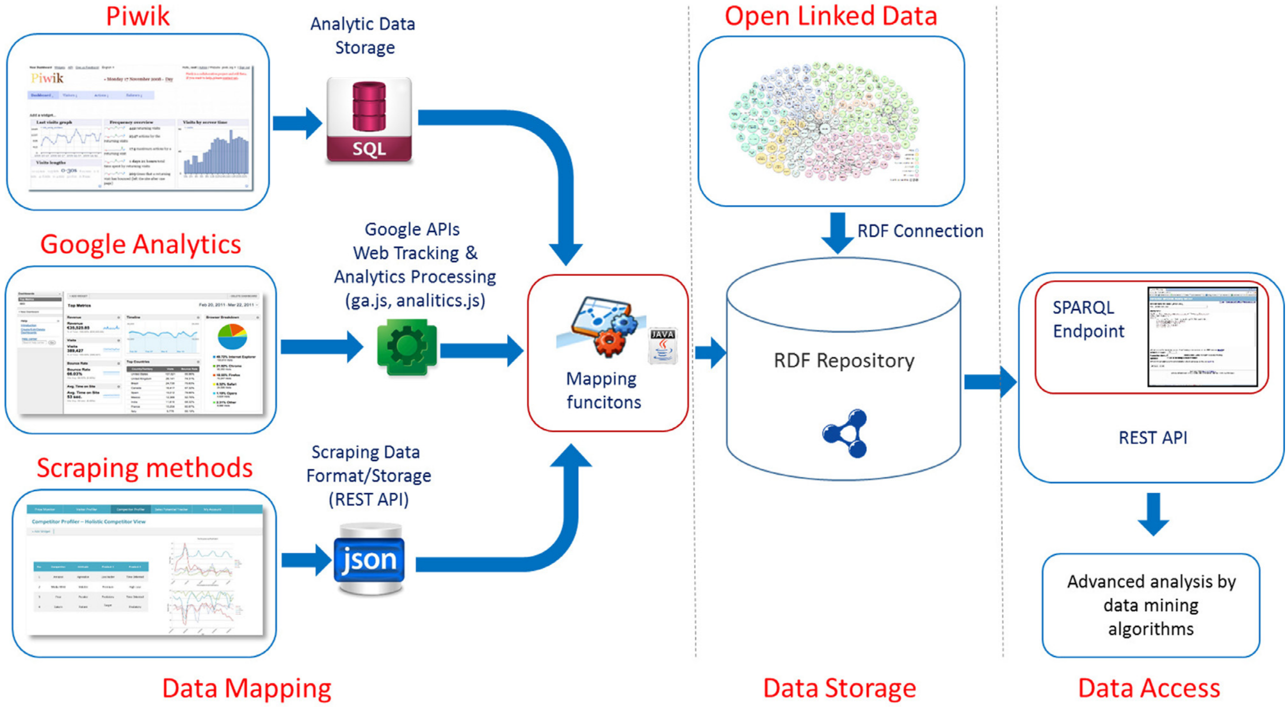


Fig. 3. General overview of the mapping process injecting data from different sources into the RDF repository.

4.2.1. Google Analytics

Google Analytics[5] is a partially free web analytics service that provides statistics and basic analytical tools for Search Engine Optimization (SEO) and marketing purposes. The service is available to anyone with a Google account, although advanced e-commerce functionalities are only available for restricted users. Google Analytics is geared toward small and medium-sized retail websites.

The web tracking procedure in Google Analytics is performed by a "snippet" or digital footprint component, that provides the developer with an API of functions for accessing to each attribute value. This digital footprint is a small piece of JavaScript code that is pasted into the e-shops HTML source code and deployed in the web server where the e-shop is hosted. It activates Google Analytics tracking by inserting the JavaScript `ga.js/analytics.js` into the page. As illustrated in Fig. 3, the JavaScript component is then instantiated by our mapping functions by means of a series of java classes to generate RDF triples.

Table A.1 in Appendix A contains the set of Google Analytics attributes that are currently tracked by our semantic approach. In this table, each attribute is listed with regards to its corresponding ontology class, data type, and description. This is a representative subset of the whole set of possible attributes (and its combinations) in the Google Analytics's API specification, that covers all our preliminary requirements for visitor's behavior and products' analysis. However, it is worth mentioning that the proposed ontology can be easily extended to consider any of the attributes worked with Google Analytics.

4.2.2. Piwik

Piwik [6] is a free and open source web analytics application running on a PHP/MySQL web server. Piwik tracks online visits to one or more websites and displays reports on these visits for analysis. Piwik analytic features include, among others: real-time data updates, free advanced e-commerce analytics features, goal conversion tracking, event tracking, geolocation, pages transitions views and page overlay.

Similarly to Google Analytics, the web tracking procedure in Piwik is also performed by a digital footprint script, that is allocated in the e-shops HTML source code. In the case of Piwik, the analytical data is automatically stored in a relational database (SQL). Therefore, as we have the possibility to access to this relational database, we have developed the mapping functions to directly query the analytic attributes. These attributes are described in Tables A.2–A.4 of Appendix A with regards to their corresponding ontology classes. The obtained data is then translated to RDF according to the ontology by means of specific mapping methods, as shown in Fig. 3.

4.2.3. Web scrapping methods

In the scope of the SME E-Compass project, there exist a series of methods for scraping product and price data from the competitors e-shop websites. This way, a given e-shop's owner is able to compare their products' prices with those ones of their direct competitors automatically.

This specific functionality provides a REST API service from with we can obtain attributes of competitor's profile in JSON[7] format, which is a compact and easily readable data format for the purpose of data exchange. Table A.5 contains the competitors attributes that are mapped to RDF in our semantic model (see Fig. 3), with regards to the corresponding ontology classes.

4.2.4. RDF repository

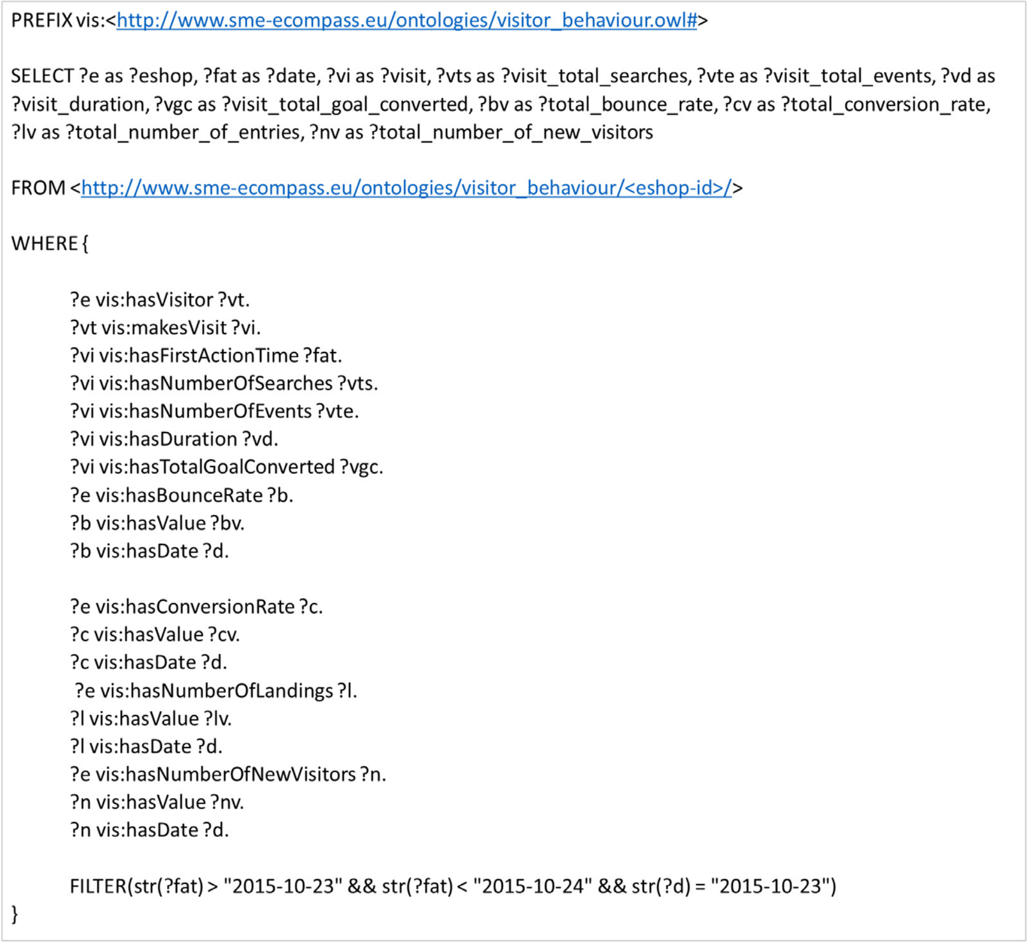
Finally, an RDF repository is used to integrate the analytic data collected and mapped from the different sources. Therefore, by means of an SPARQL endpoint, it is now possible to query the analytic data unambiguously and independently of the source.

As an instance of data access, let us consider an scenario in which the analytic module requires information concerning the

[5] http://www.google.com/analytics/

[6] http://piwik.org/

[7] http://json.org



```
PREFIX vis:<http://www.sme-ecompass.eu/ontologies/visitor_behaviour.owl#>

SELECT ?e as ?eshop, ?fat as ?date, ?vi as ?visit, ?vts as ?visit_total_searches, ?vte as ?visit_total_events, ?vd as
?visit_duration, ?vgc as ?visit_total_goal_converted, ?bv as ?total_bounce_rate, ?cv as ?total_conversion_rate,
?lv as ?total_number_of_entries, ?nv as ?total_number_of_new_visitors

FROM <http://www.sme-ecompass.eu/ontologies/visitor_behaviour/<eshop-id>/>

WHERE {

        ?e vis:hasVisitor ?vt.
        ?vt vis:makesVisit ?vi.
        ?vi vis:hasFirstActionTime ?fat.
        ?vi vis:hasNumberOfSearches ?vts.
        ?vi vis:hasNumberOfEvents ?vte.
        ?vi vis:hasDuration ?vd.
        ?vi vis:hasTotalGoalConverted ?vgc.
        ?e vis:hasBounceRate ?b.
        ?b vis:hasValue ?bv.
        ?b vis:hasDate ?d.

        ?e vis:hasConversionRate ?c.
        ?c vis:hasValue ?cv.
        ?c vis:hasDate ?d.
         ?e vis:hasNumberOfLandings ?l.
        ?l vis:hasValue ?lv.
        ?l vis:hasDate ?d.
        ?e vis:hasNumberOfNewVisitors ?n.
        ?n vis:hasValue ?nv.
        ?n vis:hasDate ?d.

        FILTER(str(?fat) > "2015-10-23" && str(?fat) < "2015-10-24" && str(?d) = "2015-10-23")
}
```

Fig. 4. Example of SPARQL query that returns disaggregated data attributes, as the ones provided by Piwik, as well as calculated metrics, as those obtained from Google Analytics.

Table 9
Two samples of the query result (Fig. 4) of a certain time slot (day 2015-10-23) of a real e-shop.

Attribute/metric	Visit75688	Visit75692
timestamp	14:19:44	14:21:41
visit_total_searches	0	0
visit_total_events	0	0
visit_total_duration	2071	12
visit_total_goal_converted	1	0
total_bounce_rate	52.6066	
total_conversion_rate	34.1232	
total_number_of_entries	211	
total_number_of_new_visitors	145	

visits of a given e-shop, in a certain date or period of time. The required information of visits should consist of both: disaggregated data attributes, as the ones provided by Piwik, and calculated metrics, as those obtained from Google Analytics.

The SPARQL query represented in Fig. 4 unifies the encoding of such logic, for which a couple of result samples are displayed in Table 9. In concrete, these results correspond to two consecutive visits to the e-shop with ID `<eshop-id>`, that were performed at date 2015-10-23. The visit IDs are 75688 and 75692, and they were captured at timestamps 14:19:44 and 14:21:41, respectively. As shown in this table, the visit with a prolonged duration led to one goal conversion (usually a successful sale), whereas the visit with a short duration finished without any conversion, which represents a visitor that leaves the site prematurely.

In the case of aggregative attributes, they are calculated for all the visits in the time period of the SPARQL query. Therefore, as shown in the second half of Table 9, the e-shop registered a bounce rate close to 53% with conversion rate[8] of 34.12%, that corresponds to all visits, bounces and purchases of the queries time period.

Another important attribute is the number of new visitors, that for this e-shop and for this date is 145, e. g., 68.72% of total entries. This information could be now used to feed predictive algorithms that help the e-merchant to adopt a given marketing strategy to catch clients.

In order to automatize and simplify the accesses to the stored data, our semantic approach includes a specific REST API service with methods that implement predefined SPARQL queries. These methods are used as input of the data mining algorithms as described in the following step of validation.

As an additional advantage of this semantic approach, it is possible to connect our RDF repository with other/s external open linked data repository/ies. In this regard, a minimum adaptation has to be done in terms of deciding which class/classes are directly linked from the two repositories with similar semantic meaning. In fact, this is one of the most powerful features when using the semantic structure induced by the ontology.

[8] Conversion rate: proportion of visitors converted into paying customers.

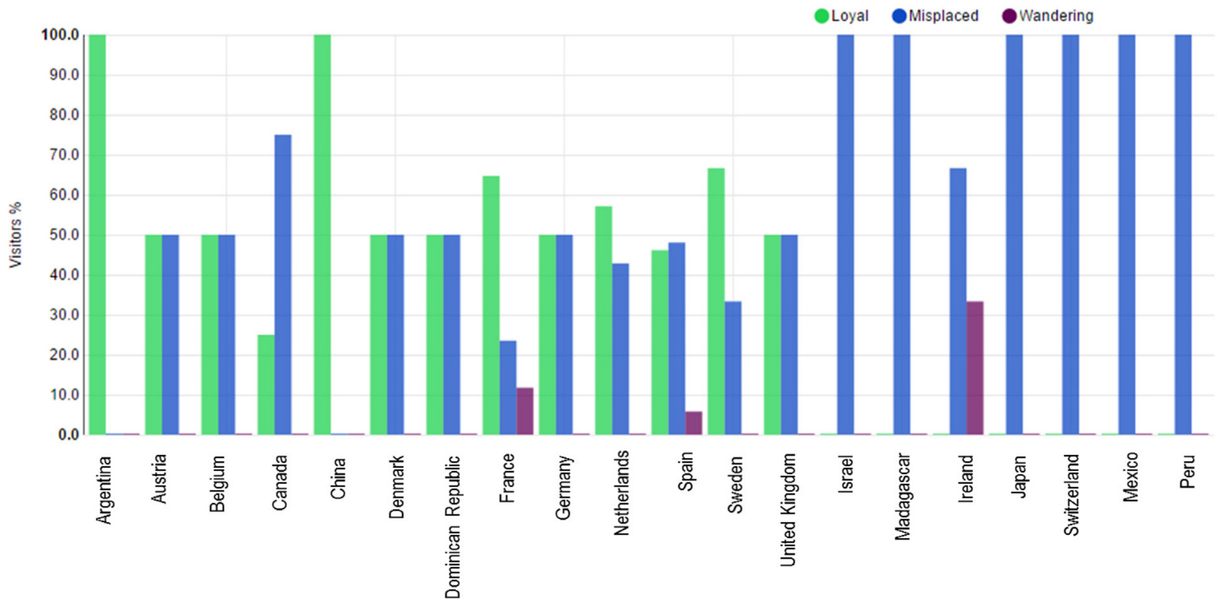


Fig. 5. Percentage of visitors classified by typologies (misplaced, loyal and wandering) for country of origin. The percentages are relative to each country.

5. Validation

For the validation of our semantic approach, we have captured and integrated data from digital footprints (Google Analytics and Piwik) allocated in more than 15 e-shops of different commercial sectors (retail, tourism, electronics, pharmacy, etc.) and countries (UK, Spain, Greece and Germany), throughout several months of activity. The resulting approach provides the data mining algorithms with a large and complemented set of web attributes and metrics, that enable them to perform advanced analyses.

In particular, we focus in this section on two different cases of study, for which we perform analyses regarding the visitor behavior and the product profile. Both kind of analyses entail a series of unsupervised learning procedures to classify visitors and products into different predefined types. First, a clustering algorithm is performed to determine groups of visitors/products. After this, a decision tree is generated with rules to assign samples in clusters to predefined types. Finally, a classification procedure is used to assign the new incoming data to each predefined type.

5.1. Case study I: visitor behavior

The behavioral patterns concerning to visitors have been classified into 3 types (misplaced, loyal, wandering) according to the features that they contain in common, such as length of visit, shopping number, whether they are recurrent regarding their visits/shopping on the web, or not, etc.

– Loyal: The loyal visitors visit more pages than the average visitor, navigate the site frequently and make purchases more often.
– Misplaced: This visitor stays in the site for a very short period of time and visits a small amount of pages. They rarely make transactions.
– Wandering: The wandering visitor stays in the site for quite a long period of time and visits an amount of pages close to the "loyal" visitor. However, (s)he is usually new and makes less purchases than the loyal visitor.

This kind of analysis is generated from a series of Google Analytics attributes, such as: users, entrances, exits, pageviews, uniquepageviews, sessionduration, newusers, sessions, and bounces; in combination with dimensions: region, source, networkdomain, browser, hour, date, and city. These attributes are modeled in our ontology model as specified in Section 4.1 and in Table A.1.

The following charts (Figs. 5–7) show analytical visualizations of visitor profiler provided by the SME E-Compass application for a given e-shop (Spanish). In these figures, the distribution of predefined types are plotted depending on variables such as the origin (continent, country, region, city, etc.) or time (in a range of dates). In concrete, Fig. 5 shows the percentage of loyal, misplaced and wandering visitors to the analyzed e-shop, in a time period (from 2015-08-15 to 2015-09-15), for each country of origin. We can observe in this figure that 100% of visitors from Israel, Madagascar, Japan, Switzerland, Mexico, and Peru are misplaced, whereas in the case of China and Argentina, all visitors are loyal. For other countries like Denmark, Dominican Republic, France, Germany, Netherlands, Sweden, United Kingdom, and Spain, the proportion of misplaced and loyal visitors are balanced. Wandering visitors are detected in the cases of France, Spain, and Ireland.

More in depth, if we focus on the global distribution of percentages per regions for a given typology of visitors, as shown in Fig. 6, it is clearly observable that the highest percentage of misplaced visitors are from the Spanish region of Valencia. In spite of existing regions of other countries (different to Spain) that registered low percentages of misplaced visitors, e. g., England (UK), State of Sao Paulo (Brazil), Ile-de-France (France), it is worth noting that the remaining regions of other countries registered global percentages lower than 1%, and therefore they are not registered in sector chart of Fig. 6. Therefore, focusing on this classification and origin study, a given e-merchant might be interested in looking at the generated impact by a marketing campaign on those regions with the aim of converting misplaced visitors to loyal ones.

In terms of time evolution, we can also focus on a specific range of chronology and check whether a customer loyalty strategy is able to obtain improvements in the overall sales or not, over the time. In this regards Fig. 7 plots evolution lines with respect to the three types of classified visitors. In this figure, it is easily observable that misplaced visitors evolve generally close to loyal ones, although we can inspect, by selecting a specific timeframe (e. g. from 2015-08-15 to 2015-09-15) which days the number of loyal is higher than the number of misplaced visitors, in order to find

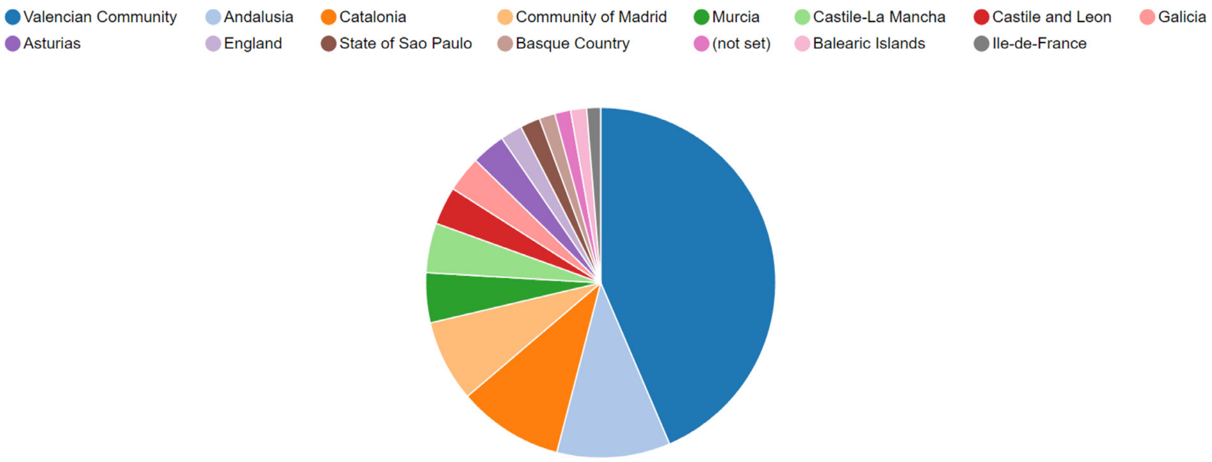


Fig. 6. Sectors chart focused by ”misplaced” visitor typology and region.

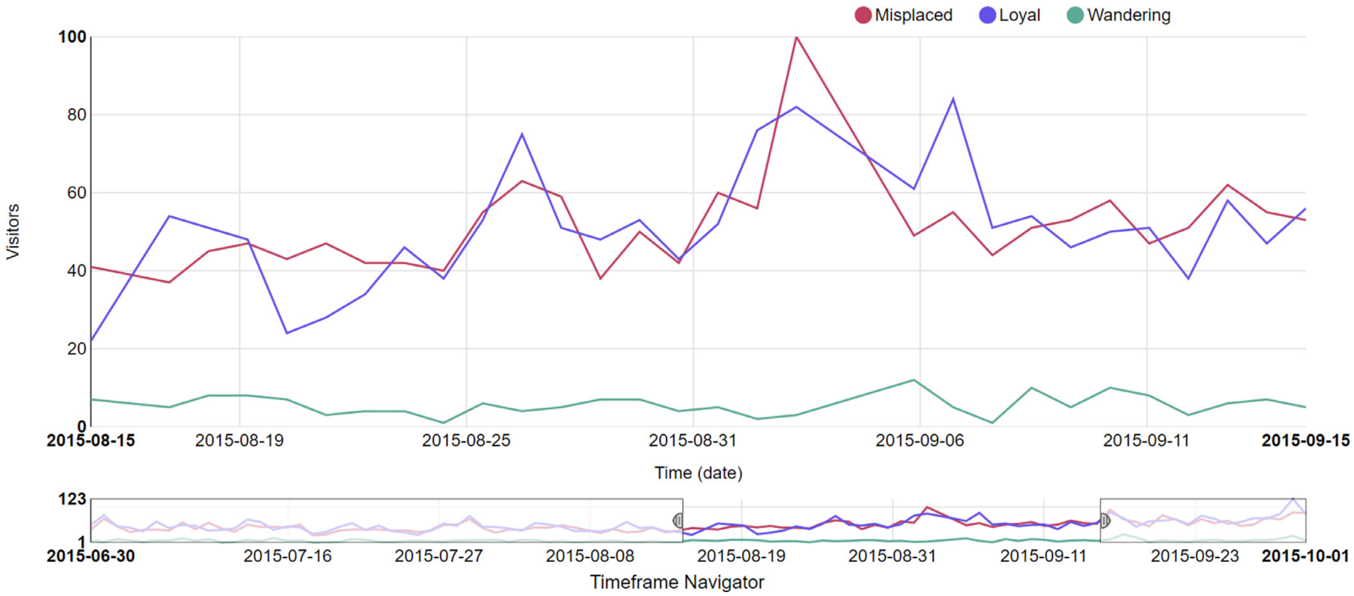


Fig. 7. Visitor typologies over time. The plot below shows the activity in a range of three months from 2015-06-30 to 2015-10-01, whereas the plot above is a specific timeframe selected for one month from 2015-08-15 to 2015-09-15.

any insight of why this happened, e. g., whether it is the effect of a marketing campaign launched during this time period, or not.

5.2. Case study II: product profiler

The intrinsic idea in product analysis is to show evidences about the relationships between visitors actions and purchased products, in order to guide the e-merchant to improve its benefits by updating the products visibility and their prices.

This kind of analysis requires information concerning, not only the navigation activity of visitors, but also the e-commerce or purchasing habits. Therefore, we now focus on captured data from Piwik attributes, such as: *idaction_sku* (Stock-keeping unit), *idaction_name, idaction_category, location_geoip, visit_first_action_time, visitor_days_since_order, visit_goal_buyer*, and *visit_goal_converted*. These attributes are modeled in our ontology model as specified in Section 4.1 and in Tables A.2–A.4.

In this regard, Fig. 8 shows the relationships between product conversions and visitors. It consists of a SWOT (strengths, weaknesses, opportunities and threats) diagram in which, horizontal axis of provides an indication of the amount of visitors for a product within the e-shop, whereas vertical axis shows the conversion

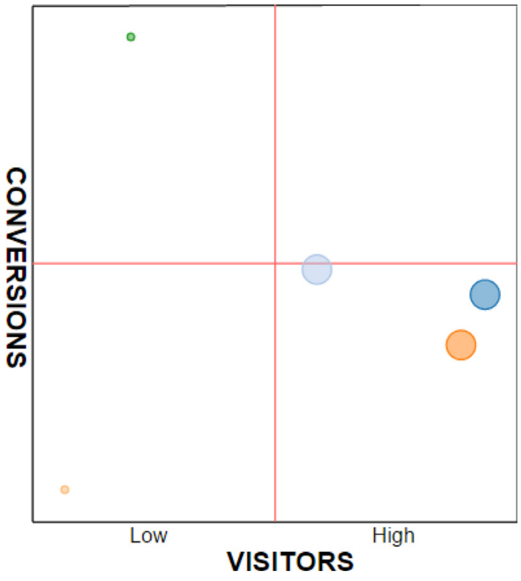


Fig. 8. Product SWOT analysis.

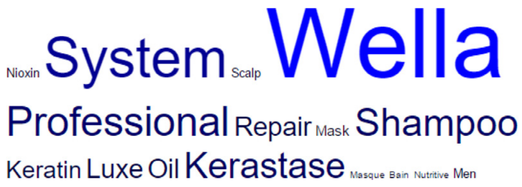


Fig. 9. Trending products.

rate of a product. The size of points represents the total revenue generated by a particular product. Therefore, for each product in a given e-shop, a different strategy can be assessed according to the 4 quadrants in this plot where it is located. This way, products with low visitors and low conversions show little interest from visitors, as well as few sales. Profitable products are located in Low Visitors-High Conversions quadrant, having little web traffic oriented to it. With a proper promotion, this product could be more visited and create even more sales. High Visitors-Low Conversions products might be interesting but somehow visitors are not totally convinced to buy it. These products are candidate to revise web positioning, features, price, etc. Finally, High Visitors-High Conversions and attractive and profitable products, that could be successful by themselves.

In general, it might be desirable to check if products are behaving as expected, regarding their position in the chart. For this specific case of study, we are using data from footprints allocated in an e-shop of cosmetic and hairdresser products, which products in High Visitors-High Conversions quadrant are: *Wella SP Luxe Shampoo 1000 ml* and *Wella Professional Repair 400 ml*. These preferences of customers correspond to the text cloud tool as shown in Fig. 9, where the e-shop's owner can rapidly view the trending product characteristics.

Finally, it is worth saying that without integration data from many sources, such results could not be obtained, since they are computed by using data queried for specific visitor's features with complementary attributes, as shown in Section 4.2.4. In particular, for the cases of study worked here, they use aggregated and disaggregated attributes that are jointly provided by Google Analytics and Piwik, respectively.

6. Conclusions

In this work, we propose a semantic approach that uses an ontology as a mediated schema for the representation and consolidation of the tracking data from web source's semantics. Semantic mappings between the source schema and the ontology are then defined and used to transform the original data to RDF. In this way, data from heterogeneous sources are stored and integrated inside a single RDF repository, which can be now easily queried by high level algorithms.

The main hypothesis guiding us is that: **(H1) an ontology-based integration approach will help us to collect, fuse the data together in a coherent way, and store web analytics data, from many sources of popular and commercial digital footprints**. In order to test it, we have designed and implemented for the first time an OWL (Web Ontology Language) Ontology for web analytics. This ontology considers a large and complemented set of attributes and metrics, which are given from several representative web analytics tools in the market, as well as other specific attributes of e-commerce analysis, such as the competitor's behavior and prices monitors.

In this regard, we have conducted a series of analyses to validate our semantic approach. We have captured and integrated data from Google Analytics and Piwik digital footprints allocated in 15 e-shops of different commercial sectors (retail, tourism, electronics, pharmacy, etc.) and countries (UK, Spain, Greece and Germany), throughout several months of activity. The data are integrated following the same (standard) format, stored in a common RDF repository.

As secondary hypothesis, we expect **(H2) to obtain enriched and semantically annotated data able to successfully train data mining procedures for advanced analysis of customer behavior in real e-commerce sites**. The obtained "semantized" data are used to train advanced data mining algorithms to perform customer's profile analyses. In particular, these algorithms are tested with success in two cases of study to classify the visitor's behavior and product preference. The resulting approach provides data mining algorithms with a large, complemented, and well-grounded set of web attributes and metrics that enable them to perform advanced analyses.

As future work, we plan to include new kinds of analytic footprints to our semantic approach. This probably entails new updates of our ontology model to consider further attributes for advanced web analytics. We are also interested in incorporating open linked data to enrich our semantic model with new perspectives of information, such as: meteorological information, product descriptions and/or social sector's affinities.

Appendix A. Analytic metrics and attributes

The complete set of used attributes and metrics from Google Analytics, Piwik and Scrapping methods are described in Tables A.1–A.5. The corresponding ontology class of each attribute are located in the first column of these tables.

Table A.1
Google Analytics used metrics and attributes in the ontology model.

Ontology class	Attribute	Type	Description
E-shop	transactions/sessions	float∗	Number of transactions divided by number of visits
Page	sessions	int*	Number of sessions
Page	entrances	int*	Number of sessions starting at this page
Page	exits	int*	Number of sessions ending at this page
Page	pageviews	int*	Number of page views
Page	uniquepageviews	int*	Number of unique page views
Page	bouncerate	float∗	Number of sessions with just one page view
Visit	sessionduration	int*	Session duration in seconds
E-shop	avgsessionduration	float∗	Average session duration in seconds
E-shop	percentNewSessions	float∗	Percentage of new sessions
E-shop	pageviewspersession	float∗	Average number of pages visited per session
E-shop, Page	users	int*	Number of unique users/visitors
E-shop, Page	newusers	int*	Number of new visitors
E-shop, Page	users newusers	int*	Number of returning visitors

(*continued on next page*)

Table A.1 (*continued*)

Ontology class	Attribute	Type	Description
E-shop	transactions	int*	Number of e-commerce transactions
E-shop, Page	itemRevenue	float*	Total e-commerce revenue
E-shop, Page	itemQuantity	int*	Total number of units sold
E-shop, Page	transactionRevenuePerSession	float*	E-commerce revenue per Session
E-shop, Page	revenuePerTransaction	float*	Average order value
E-shop, Page	bounces	int*	Total number of single page (or single engagement hit) sessions
Visit	uniquePurchases	int*	Number of product sets purchased

All these metrics are combined with dimensions: date, hour, city, region, browser, networkDomain, and source.
Besides, sessions are combined with dimensions: city, region, country and continent.

Table A.2
Piwik used metrics and attributes in the ontology model.

Ontology class	Attribute	Type	Description
Address	location_geoip_latitude	decimal (7,4)	Latitude where the visitor lives
	location_geoip_longitude	decimal (7,4)	Longitude where the visitor lives
	location_geoip_city	varchar (100)	City where the visitor lives
	location_geoip_region	varchar (2)	Region where the visitor lives
Browser	config_browser_name	varchar(10)	Browsers name acronym
	config_browser_version	varchar (20)	Browsers version
City	location_geoip_city	varchar (100)	City where the visitor lives
Continent	location_geoip_continent	varchar (100)	Continent where the visitor lives
Country	location_geoip_country	varchar (100)	Country where the visitor lives
Device	config_resolution	varchar (9)	Devices resolution
	config_device_type	tinyint(100)	Kind of device
	config_device_brand	varchar(100)	The brand of the device
	config_device_model	varchar(100)	The model of the device
IP address	location_ip	varbinary (16)	Connection IP address
ISP provider	location_provider	varchar(100)	ISP provider
Keyword	referrer_keyword	varchar(255)	set of words that has hit the web via search engine
Navigation step	idaction_url_ref	int(10) unsigned	The previous action URL
	Idaction_name_ref	varchar (1000)	The previous action name
	time_spent_ref_action	int(10) unsigned	The duration of the action
Operating system	config_os	char (3)	Operating systems name acronym
	config_os_version	varchar(100)	Operating systems version
Page	idaction_name	varchar (100)	Pages URL name
	server_time	datetime	Time when the action has happened
Path	Idaction_url	varchar (1000)	Page's URL
Product	idaction_sku	int (10)	Product ID
	idaction_name	varchar (100)	Page's URL name
	idaction_category	varchar (50)	Product's category
	price	float	Product's price
	quantity	int(10)	Amount of products
	deleted	tinyint(1)	if deleted of the active part of the catalogue
Region	location_geoip_region	varchar (2)	Region where the visitor lives

Table A.3
Piwik used metrics and attributes in the ontology model.

Ontology class	Attribute	Type	Description
Visit	idVisit	int (10) unsigned	Visit ID
	visit_first_action_time	datetime	Time when the first action of the visit happens
	visit_last_action_time	datetime	Time when the last action of the visit happens
	visit_exit_idaction_url	varchar (1000)	The last action's URL of the visit
	visit_exit_idaction_name	varchar (255)	The last action's name of the visit
	visit_entry_idaction_url	varchar (1000)	The first action's URL of the visit
	visit_entry_idaction_name	varchar (255)	The first action's name of the visit
	referer_name	varchar (70)	Website's name referring to the landing page of the site
	referer_url	text	Website's URL referring to the landing page of the site
	referer_type	tinyint(1) unsigned	The kind of referrer link (search engine, social net., etc.)
	visit_total_actions	smallint(5) unsigned	Number of visit's actions
	visit_total_searches	smallint(5) unsigned	Number of visit's searches
	visit_total_events	smallint(5) unsigned	Number of visit's events
	visit_total_time	smallint(5) unsigned	Total time of the visit
	visit_goal_converted	tinyint(1)	Whether or not this visit converted a goal

(*continued on next page*)

Table A.3 (*continued*)

Ontology class	Attribute	Type	Description
	visit_goal_buyer	tinyint(1)	If the visitor ordered something during this visit
Visitor	idVisitor	binary(8)	Visitor ID
	visitor_localtime	time	The time of the machine that the visitor use
	visitor_returning	tinyint(1)	Whether or not the visitor is recurrent
	visitor_count_visits	smallint(5) unsigned	Number of visits carried out by the visitor
	visitor_days_since_last	smallint(5) unsigned	Number of days since the last visit of this visitor
	visitor_days_since_order	smallint(5) unsigned	Number of days since the order of this visitor
	visitor_days_since_first	smallint(5) unsigned	Number of days since the first visit of this visitor
Site	idSite	int(10) unsigned	Site ID
	name	varchar (90)	Site's name
	main_url	varchar (255)	The main URL of the site
	timezone	varchar(50)	The site's time zone (UTC)
	currency	char(3)	The currency of the site
Idiom	location_browser_lang	varchar(20)	Browser's language

Table A.4
Piwik used metrics and attributes in the ontology model.

Ontology class	Attribute	Type	Description
Goal	idGoal	int(10) unsigned	Goal ID
	name	varchar (50)	Goal's name
	match_attribute	varchar (20)	Related attribute with the goal
	pattern	varchar (255)	How the goal can be converted
	pattern_type	varchar (10)	The kind of the pattern
	revenue	Float	The revenue per visit for each goal
	deleted	tinyint(4)	Whether or not it has been deleted from the goals collection
	allow_multiple	tinyint(4)	Allow goal to be triggered more than once per visit
Order	idOrder	varchar (100)	*order's ID
	idGoal	int(10)	the ID of the goal this conversion is for
	revenue	float	The amount of revenue a conversion generates (if any)
	revenue_subtotal	float	*total cost of the items in the order/cart
	items	smallint unsigned	*number of items in the order/cart
	revenue tax	float	*total tax applied to the items in the order/cart
	revenue_shipping	float	*total cost of shipping
	revenue_discount	float	*total discount applied to the order

* If this conversion is for an e-commerce order or abandoned cart.

Table A.5
Attributes of the web scrapping methods in the ontology model.

Ontology class	Attribute	Type	Description
E-shop owner	E-shop ID	Integer	ID of E-shop owner given by E-COMPASS Cockpit (user management)
	Last name	String	Name of the person in charge (employee of the e-shop)
	First name	String	Name of a person (employee of the e-shop)
	E-Mail address	String	E-Mail address of the person in charge (employee of the e-shop)
E-shop	URL	String	Start page of the e-shop
	E-shop owner ID	Integer	ID of E-shop owner given by E-COMPASS Cockpit (user management)
	Competitor ID	E-shopID	E-shop ID of all competitors
Product	Product ID	Integer	Product ID of the E-COMPASS System
	Name	String	Product Name given by E-Shop owner (e.g. as a search query)
Article Number	Type	String	Type of article number (ASIN, EAN and/or ISBN)*
	Value	String	The value of product (ASIN, EAN and/or ISBN)*
Price	Value	Double	Price value on scraping date
	Currency	String	Currency of Price
	Date	Date	Scraping date of product price
Availability	Value	String	Availability of product available or "not available
	Date	Date	Scraping date of availability

* ASIN: Amazon Standard Identification Number, a ten-digit alpha-numerical product code;
EAN: European Article Number, 8-digit or 13-digit number for product identification;
ISBN: International Standard Book Number, 10-digit or 13-digit number for book identification.

References

Akanbi, A. K. (2014). Lb2co: a semantic ontology framework for b2c ecommerce transaction on the internet. *International Journal of Research in Computer Science, 4*(1), 1–9.

Dean, M., & Schreiber, G. (2004). OWL web ontology language reference. *Technical Report*. W3C Recommendation, 10 February 2004.

Garía-Nieto, J., & Roldán, M. (2014). D2.1 SME-E-COMPASS requirements analysis. *Technical Report*. Public Deliverable.

Gatchalee, P., Li, Z., & Supnithi, T. (2013). Ontology development for smes e-commerce website based on content analysis and its recommendation system. In *Computer science and engineering conference (icsec), 2013 international* (pp. 7–12).

Gruber, T. R. (1993). A translation approach to portable ontologies. *Knowledge Acquisition,, 5*(2), 199–220.

Haase, P., & Stojanovic, L. (2005). Consistent evolution of owl ontologies. In A. Gmez-Prez, & J. Euzenat (Eds.), *The semantic web: research and applications*. In *Lecture Notes in Computer Science: 3532* (pp. 182–197). Springer Berlin Heidelberg.

Hepp, M. (2008). Goodrelations: an ontology for describing products and services offers on the web. In *Proceedings of the 16th international conference on knowledge engineering and knowledge management (ekaw2008)* (pp. 332–347). Springer LNCS, Vol 5268.

Horrocks, I., & Patel-Schneider, P. (2003). Reducing owl entailment to description logic satisfiability. In *The semantic web - iswc 2003*. In *Lecture Notes in Computer Science: 2870* (pp. 17–29). Springer Berlin.

McGuinness, D., & Harmelen, F. (2004). OWL web ontology language overview. *Technical Report*. W3C Recommendation.

Natalya, N., McGuinness, F., & Deborah, L. (2001). DOntology Development 101: A Guide to Creating Your First Ontology. *Technical Report*. tanford University Knowledge Systems Laboratory Technical Report KSL-01-05.

Pérez, J., Arenas, M., & Gutierrez, C. (2009). Semantics and complexity of sparql. *ACM Transactions on Database Systems, 34*(3), 1–45.

Staab, S., & Studer, R. (2009). *Handbook on Ontologies*. International Handbooks on Information Systems. Springer.

Tamma, V., Phelps, S., Dickinson, I., & Wooldridge, M. (2005). Ontologies for supporting negotiation in e-commerce. *Engineering Applications of Artificial Intelligence, 18*(2), 223–236.

Trastour, D., Bartolini, C., & Preist, C. (2003). Semantic web support for the business–to-business e-commerce pre-contractual lifecycle. *Computer Networks, 42*(5), 661–673.

Waralakv, S. (2008). Learning semantic web from e-tourism. In N. Nguyen, G. Jo, R. Howlett, & L. Jain (Eds.), *Agent and multi-agent systems: Technologies and applications*. In *Lecture Notes in Computer Science: 4953* (pp. 516–525). Springer Berlin Heidelberg.