Semantic to Intelligent Web Era
Building Blocks, Applications, and Current Trends

Joe Tekli
School of Engineering,
Dept. of Elec. and Compt. Eng.
Lebanese American University
36 Byblos, Lebanon
joe.tekli@lau.edu.lb

Antoine Abou Rjeily [a-b]
[a] TICKET Laboratory
Faculty of Engineering
Antonine University
40016, Baabda, Lebanon
antoineabourjeily@gmail.com

Richard Chbeir
LIUPPA Lab., Univ. of Pau and Adour Countries
IUT de Bayonne, CS Dept.
64000 Anglet, France
rchbeir@acm.org

Gilbert Tekli
Research & Development Department
Pearl Advisers Offshore
Houston Office, 77251 Houston USA
gtekli@pearladvisers.com

Pélagie Houngue, Kokou Yetongnon
[b] LE2I CNRS, Univ. of Bourgogne
21000 Dijon, France
{pelagie.houngue, kokou}@u-bourgogne.fr

Minale Ashagrie Abebe
IT Doctoral Program
Addis Ababa University
3131 Addis Ababa, Ethiopia
minale2as@gmail.com

ABSTRACT
The Web has known a very fast evolution: going from the Web 1.0, known as Web of Documents where users are merely consumers of static information, to the more dynamic Web 2.0, known as social or collaborative Web where users produce and consume information simultaneously, and entering the more sophisticated Web 3.0, known as the Semantic Web by giving information a well-defined meaning so that it becomes more easily accessible by human users and automated processes. Fostering service intelligence and atomicity (the ability of autonomous services to interact automatically), remains one of the most upcoming challenges of the Semantic Web. This promotes the dawn of a new era: the Intelligent Web (Web 4.0), known as the Internet of Things (IoT), an extension of the Semantic Web where (physical/software) objects and services autonomously interact in a multimedia virtual environment, provided with embedded communication capabilities, common semantics and addressing schemes, promoting the concept of Digital Web Ecosystems where every where (human and software) agents collaborate, interact, compete, and evolve autonomously in order to automatically solve complex and dynamic problems. This paper briefly describes the recent evolution of the Web providing an overview of the technological breakthroughs contributing to this evolution, covering: knowledge bases and semantic data description, XML-based data representation and manipulation technologies (i.e., RDF, RDFS, OWL, and SPARQL) as well as the main challenges toward achieving the Intelligent Web: connectivity, semantic heterogeneity, collective knowledge management, collective intelligence, as well as data sustainability and evolution. We also present some of the main application domains characterizing the Intelligent (Semantic) Web, from information retrieval and content analysis, to systems status monitoring and improving business life-cycle through ubiquitous computing.

Categories and Subject Descriptors
A.1 [**General**] Introductory and Survey; C.0 [**Computer System Organization**] System Architectures; H.3.3 [**Information Storage & Retrieval**]: Information Search & Retrieval; I.7.1 [**Document & Text Processing**]: Document & Text Editing - *Document management*; I.7.2 [**Document Preparation**]: Document Preparation – *Markup languages*.

General Terms
Documentation, Design, Human Factors, Standardization, Languages.

Keywords
Web; Semantic Web; Internet of Things, Knowledge base; Data Semantics; XML; RDF; OWL; SPARQL; Intelligent Services.

1. INTRODUCTION
The *Semantic Web* (SW) is a collaborative movement guided by the World Wide Web Consortium (W3C), aiming to extend the Web (as we know it) by *giving information a well-defined meaning*, in order to improve data accessibility for humans and machines [17, 19]. Also known as *Web 3.0*, the SW is a vision of the Web where machines are able to automatically exploit the semantic meaning of information, available at different locations in a distributed environment, so as to allow more effective and intelligent Web data access, search and retrieval. Nonetheless, with the development of the mobile Internet (smart phone-based) and communication technology, the SW vision is being extended toward a new revolutionary stage: the *Intelligent Web* (IW), also known as *Internet of Things* (IoT), where semantically rich objects: i) either physical objects of the real world, with added digital components (e.g., smart phones, smart cars, robotic systems, etc.) which we identify as intelligent terminals; and/or ii) software agents (e.g., scripts, applications, APIs, etc.) autonomously interact, sustain themselves and evolve in a multimedia virtual environment, provided with embedded communication capabilities, common (collective) semantics and addressing schemes. However, the automatic capture and processing of semantic information remains a difficult task because of the well-known problems that machines have with processing semantics. For instance, a machine traditionally processes the expression "*university*" as a word consisting of 10 characters, rather than capturing the meaning of the word "*an academic institution of higher education*", unless some sort of semantic data processing is involved.

In this context, the SW vision aims to associate machine-readable semantic descriptions to Web data, using two major technological breakthroughs: i) knowledge bases (such as taxonomies and/or ontologies [14, 95]), which provide predefined semantic information references (similarly to dictionaries for human users) to allow the identification and extraction of semantic meaning from raw data, and ii) XML-based data representation technologies (namely RDF [38, 51] and RDFS [25, 64] for resource description, OWL [6, 37] for ontology definition, and SPARQL [50, 93] for semantic data manipulation and querying). These technologies are extensible, interoperable, and platform-independent [38], aiming to improve data modeling, annotation, manipulation, search and integration, and thus allowing intelligent information retrieval on the Web [14], which is at the core of the SW [38]. Yet, developing service intelligence and atomicity, i.e., the ability of software agents and services to interact and sustain themselves automatically, without human interaction, remains one of the most upcoming challenges of the SW. In addition, SW technologies and social networking services are promoting a new form of collaboration: nowadays, it is common for Web users to contribute their multimedia data and knowledge to the community, allowing the editing and manipulation of such public knowledge in a collaborative environment (e.g., Wikis, blogs, Foursquare[1], Google Latitude[2], etc.). As a result, the Web is becoming more than a distributed container of (raw and/or semantic) multimedia data, but is increasingly harnessing *Collective Knowledge* (CK), viewed as the combination of all known data, information, and meta-data concerning a given (set of) concept(s),

MEDES'13, October 29-31, 2013, Neumünster Abbey, Luxembourg.

[1] Location-based social website for mobile devices (http://Foursquare.com).
[2] Location-aware mobile application allowing users to view their contacts geographic locations (www.google.com/latitude). Note that Google Latitude is being recently retired, transforming most of its services to Google+.

fact(s), user(s), or processes (s), as well as the semantic links between them [9]. Hence, software agents (and/or intelligent terminals) are expected to automatically analyze and handle large collections of multimedia data with their contents, links and transactions, using the sum of their respective intelligence and knowledge, in order to improve data accessibility, management, and exchange between people and computers. Also, agents (terminals) in the IW vision are expected to guaranty autonomous data/services sustainability and evolution, e.g., predicting future events that might affect the data/services, and acting therefore in order to preserve or update them accordingly.

The goal of this paper is to provide a concise and comprehensive review on the technologies and tools contributing to the development and evolution of the SW, namely: knowledge bases and semantic data description, and XML-based data representation technologies (i.e., RDF, RDFS, OWL, and SPARQL). We then discuss some of the main challenges toward achieving the IW (IoT)[3] vision, namely: connectivity, semantic heterogeneity, collective knowledge management, as well as data sustainability and evolution. We also discuss the main application domains characterizing the IW. The remainder of this paper is organized as follows. Section 2 presents the SW vision, underlying technologies, and ongoing trends. Section 3 is devoted to the IW (IoT) vision and its main challenges with respect to the SW. Section 4 briefly emphasizes some application domains, and finally Section 5 concludes the paper.

2. SEMANTIC WEB VISION

Imagine having your favorite artist's new song downloaded automatically and added to your playlist, or imagine your phone automatically turning down the sound of all other local devices (e.g., television set, radio, laptop, etc. with a wireless volume control system) when you answer a phone call. Is that possible today, having some kind of software agent capable of understanding your needs and acting intelligently in order to fulfill them? And if there is such an agent, how could it gather all the data needed for these kinds of tasks? And most importantly, how would it understand the gathered data, analyze it and extract the bits and pieces needed for the task ahead? In the remainder of this section, we try to answer some of these questions by presenting and discussing the vision of the SW [17] which promises to resolve such scenarios, and more.

2.1 Overview: What is the Semantic Web?

To answer the question above, we first need to distinguish the concepts of: *data*, *information*, *knowledge* and *metadata*. The main difference lies in the level of abstraction of each concept. *Data* is viewed as the lowest abstraction and contains no meaning whatsoever (e.g., *"2001"* is considered as a number consisting of 4 digits, and highlights no information at all). For the data to be *informative*, it must be interpreted and given a well-defined meaning (such as *"the year of announcement of the Semantic Web"*) and can be therefore qualified as *information* [35]. In this context, *metadata* is viewed as a description about the data and information (such as who gave the data/information – e.g., *Wikipedia*, when was the data/information given – e.g., *published in 2002*, etc.) [35]. At a higher level of abstraction, *knowledge* is viewed as the combination of all known data, information, and meta-data concerning a given concept or fact, as well as the semantic links between them [103, 130] (like knowing that *"the year of announcement of the Semantic Web"* is *"2001"*, following *Wikipedia* in an article *published in 2002*).

In this context, we can generally distinguish between two kinds of files available on the Web: i) data documents (e.g., text files, media files, maps, graphs, etc.) designed to be accessed and understood by human users, and ii) information documents (e.g., calendars, contacts, registration info, traveling info, etc.) which can be stored and manipulated automatically by machines. This is why the original version of the Web (or the *Web 1.0*) was known as the *"Web of Documents"*, where documents are written in HTML (Hyper-Text Markup Language), uniquely identified by a URI (Uniform Resource Identifier) and linked together through hyperlinks [19], such as the documents are destined to be accessed by users. The traditional Web then gradually grew to meet the requirements and the needs of its users allowing them to better interact with the data and information published online. Websites became increasingly interactive, allowing users to easily exchange ideas, discuss topics, and publish information, which soon drove the Web to another level: the *Social Web* (Web 2.0) where people are involved in publishing and interacting with other users' published materials [5]. The Web 2.0 introduced a new era of distributed data/information management with double interaction: i) horizontal: *user-user*, and ii) vertical: *user-machine*.

Based on the need for more effective user-machine interactions, Tim Berners-Lee later introduced the vision of the SW, as an extension of the current Web, in which information is associated and augmented with well-defined meaning (i.e., knowledge), better enabling computers and people to interact and work in cooperation [17]. In addition, endowing machines with the ability to process knowledge has extended the concept of cooperation (introduced with the Web 2.0) into a new level: *machine-machine* interaction. This kind of interaction is not based on a simple exchange of *data*, but rather on an exchange of semantically meaningful *information* well understood by machines.

2.2 Motivations: Why the Semantic Web?

The vision of the SW spurs from a set of basic needs:

1) Improving Search Engines: Search engines have been developed and used for many years now on the Web. They have been improved throughout the years using many techniques, such as interactive querying (through user feedback) [83], exploratory search [23] also known as browsing, approximate querying [111], query expansion [30], and search result organization [124] (rearranging query results to facilitate the user's task in selecting relevant answers). Yet, search engines on the Web generally still suffer from a lack of accuracy in retrieving desired search results. That is mainly because queries are processed as text-based phrases in which keywords are found and matched to the results, which sometimes generates results far from the users' intentions expressed in the queries [14, 105]. In other words, given the concepts of data and information (introduced in Section 2.1), most existing systems process queries as data requests, neglecting most of the meaning (semantics) behind the information (e.g., what does the query mean? what does the user want from the query? etc.). Hence there is a central need to augment existing Web search engines with semantic-based processing functionalities in order to produce more accurate results [75].

2) Providing New Appropriate Services: In addition to improving search engines, new kinds of services are desperately needed, namely: intelligent services, personalized services, and domain-based services.

Intelligent services: As the Web revolves more around vertical interaction between men and machines (i.e., *user-machine*), the need for more intelligent services becomes evident, envisioning a scenario where the machine becomes more than a receiver (answering service requests), but rather an interlocutor, capable of initiating, negotiating, composing, and intelligently discovering new services [67, 115].

Personalized services: In addition, the Social Web experience has identified the need for a more personal engagement with the user, revolving around the user's needs, character, persona, beliefs, expectations, and ergonomics. For this to become real, the user needs to specify these characteristics for the machine so this latter can understand them and act accordingly [88, 128]. For instance, a personalized geo-service could detect the locations and identify the trajectories of users, and then mine the correlation between users and locations, allowing to i) connect users that share similar travel trajectories, ii) provide users with generic travel recommendations (e.g., most interesting locations), and iii) personalize friend and location recommendations [128].

Domain-based services: Geo-services [128] are also a good example of domain-based services, related to geographic information systems (*GIS*). In this context, knowledge specific to the domain at hand is usually shared between agents and services acting in the domain. For example, in the GIS domain, concepts such as "*GPS*" (Global Positioning System) or "*GML*" (Geographic Markup Language) are often used and designate (each) the same meanings. Hence, the need for domain-based services, built among domain-specific knowledge representations (e.g., domain-specific dictionaries or ontologies), in order to reduce the redundancy factor, improve service accuracy, and help speed up service processing [97, 125]. In other words, this requires

[3] We use both IW and IoT acronyms interchangeably in this paper.

a common framework for organizing knowledge in a specific domain, which is both accurate and complete in describing the semantics of the information [69].

3) Improving Data Accessibility: One of the main problems with data published on the Web at the moment is that it is not in a form that can be effectively and easily used [17, 18]. Data can be stored in different ways (spreadsheets, databases, etc.) and is not usually posted on the Web in its original form. Rather, certain bits of information deduced from the data itself are published online (following certain constraints, e.g., target audience, online storage space, etc.), and presented often in a more user-friendly format, such as plain text, graphs, charts, tables, etc. Hence, online information is often specific (to a certain audience) and might not be reusable by different users and/or applications [18, 39]. For example, when searching for a map on the Web, one user might be interested in street names, while another user searches for restaurants. Hence, the corresponding online geographic data has to be complete, endowed with flexible and efficient data access services providing each user with the information which best answers her needs [39].

4) Better Data Integration & Presentation: As mentioned earlier, the Web is a sum of contents linked together via hyperlinks. These hyperlinks reflect (in one way or another) certain semantic relationships between documents [63], which have been proven effective in answering queries and identifying relevant Web pages [26]. Sophisticated algorithms such as PageRank [26] and HITS [63] have been developed to analyze link structures in order to rank Web pages. Yet, most documents published on the Web remain flat, i.e., consisting of unstructured data, which limits the performance of existing (conventional) search engines. For example, a traveler seeking a train ticket to get to an airport would have to access two separate Web pages and cross-match the data to find the best tickets available. Yet, if the contents of Web pages were structured in a way to access specific information related to departure/arrival dates, then the data would be crisscrossed automatically, providing the user with the bundled data (e.g., a combination of a train and a plane ticket) in a single view. Hence, the need to structure data published on the Web in an appropriate form becomes critical, where not only whole documents but rather structured information within the documents are linked together. This is more recently known as the concept of: "Linked data" [20, 52]: interlinking structured data using Web technologies (such as HTTP, XML, RDF and URIs, cf. Section 2.3) in a way that can be processed and queried automatically by machines.

2.3. Technologies and Building Blocks

2.3.1. Overall Architecture

The SW vision is based on three main concepts: *objects*, *labels* and *links*. Objects designate any piece of data on the Web, e.g., Web pages, services, media files, etc. Each object is usually described with metadata, known as *data labeling* (augmenting data with descriptive labels), and is uniquely identified by a URI allowing the object to be unambiguously *linked* with other objects. Here, some golden rules need to be followed while realizing the SW architecture [19]: i) every object is labeled, ii) labels are readable by software agents and humans, iii) labels describe corresponding objects accurately, and iv) labels are located in a common environment for software agents and humans to explore, making objects accessible as resources.

In this context, a hierarchy of technologies, mainly: XML, RDF, RDFS, OWL, and SPARQL, were gradually normalized by the W3C in the last decade to fulfill the SW architecture (known as the "*The Semantic Web Stack*", as shown in Figure 1). Three layers can be distinguished in the stack: i) the naming and addressing layer, ii) the syntax layer, and iii) the semantic layers. The naming and addressing layer associates an object with a unique identifier, i.e., a URI (Universal Resource Identifier) or an IRI (Internationalized Resource Identifier) for multilingual Web addresses [58]. The syntax layer structures the data in a tree-like structure, using XML-based constructs and namespaces. Finally, to append semantic meaning to data, a semantic layer is added which associates labels to data objects, involving all other technologies, from ontology to query and rule-based languages [61]. In the following, we briefly present the main technologies making up the building blocks of the SW: knowledge bases, XML, RDF, RDFS, OWL, and SPAQRL.

2.3.2. Description Logics and Knowledge Bases

Description Logics (DLs) are a family of languages for Knowledge Representation (KR) and Knowledge Inference (KI) [33]. On one hand, KR in Artificial Intelligence (AI) provides a means to represent and describe *knowledge*, to be stored in Knowledge Bases (KBs), i.e., repositories of machine-readable *knowledge*, available for automated processes (software agents) to use and exploit, aiming to achieve intelligent processing capabilities. On the other hand, KI is the knowledge deduced by an inference engine, working within or alongside the automated process, based on a predefined KB. As for DLs, many languages have been proposed: Propositional Logic, First-Order Logic, Temporal Logic, Fuzzy Logic, etc. with specific properties and applications, mainly exploited in semantic data analysis [53, 110].

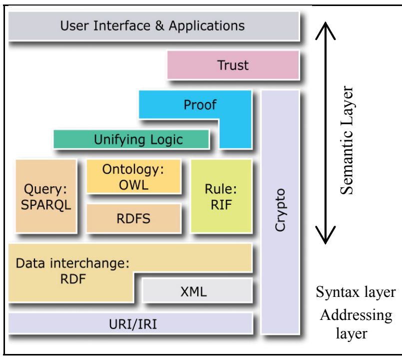

Figure 1. The Semantic Web Stack.

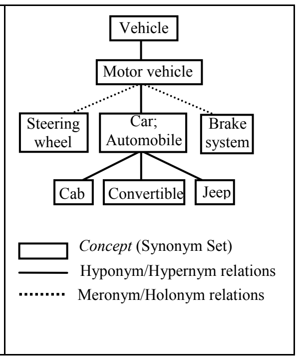

Figure 2. An extract from the WordNet semantic network [81].

In this context, every KB system based on DL is composed of a Terminology-Box (T-Box) and an Assertion-Box (A-Box). The *T-Box* underlines the set of concept definitions, while the *A-Box* consists of the collection of concept instances (also called individuals). In comparison with a relational database, the *T-Box* is similar to the structure of the tables (database schema) whereas the *A-Box* is more like the data rows (tuples) inserted into the tables [19, 33]. Here, KR structures such as taxonomies, thesauri, and ontologies, etc. have been investigated and developed (in the domains of natural language processing and information retrieval), in order to define, organize and link concepts in a KB [60]. A KB usually comes down to a *semantic network* which is basically a graph consisting of nodes and arcs, organizing words/expressions in a semantic space [96] (Figure 2). Each node represents a concept underlining a group of words/expressions (or URLs such as with ODP – Open Directory Project [75]). Arcs underline the semantic links connecting the concepts, representing semantic relations (synonymy, hyponymy (Is-A), Meronymy (Part-Of), etc. [81, 96]). Examples of lexical KBs are Roget's thesaurus [122] and WordNet [81] (Figure 2). In such structures, knowledge is usually processed as sets of triplets: *concept*$_1$*-relationship-concept*$_2$, or as more commonly known: *subject-predicate-object* triplets [33, 48]. This corresponds to the triplet representation: *objects*, *labels*, *links*, in the SW (Section 2.3.1).

2.3.3. XML & Interoperability

The distributed nature of the Web, as a decentralized system running over multiple platforms and exchanging information between multiple heterogeneous sources, has underlined the need to manage *semantic interoperability*, i.e., the ability to automatically interpret *information* in Web documents exchanged between different sources, in a semantically meaningful way in order to produce useful results for efficient information management and search applications [48].

<person fname= "" lname= "" age= ""/>	<person> <name first ="" last= ""/> <age></age> </person>
a. Serialization with attributes only.	b. Serialization with elements and attributes.

Figure 3. Different XML serializations of an entity *person*.

In this context, XML was introduced as a data representation model that simplifies the tasks of interoperation and integration among heterogeneous data sources [24]. It allows to represent data in a (semi-) structured document, consisting of hierarchically nested information, made of a set of atomic and complex elements (i.e., containing sub-elements) as well as atomic attributes, incorporating structure and content in one entity (cf. Figure 3). In contrast with HTML initially designed for visual markup, XML tagging concerns data contents, and is not limited to a fixed vocabulary but rather allows flexible and extensible application-based vocabularies using dedicated grammar definitions (such as DTD – Document Type Definition [24] or XSD – XML Schema Definition [43]), specifying allowable combinations and nesting of tag names, attribute names, and the rules they adhere to in the documents. Making use of XML to index and retrieve complex objects has been proven successful, especially in multimedia applications (e.g., SVG, SMIL, X3D and MPEG-7), and has been adopted as a common data serialization and exchange format between programming languages (e.g., PHP, JSP, ASP, Java, C#, etc.).

However, while XML was shown most effective in exchanging data (*data interoperability*), it has been proven limited when it comes to handling semantics (*semantic interoperability*). For example, an object such as "*person*" in Figure 3, with properties: "*first name*", "*last name*", etc. can be serialized in different ways in XML. While semantically identical, these serializations are treated differently by different XML engines, since XML only specifies the syntactic and structural features of the data without any further semantic meaning.

2.3.4. RDF(S) & Semantics

While XML addresses the syntactic/structural properties of data, RDF (Resource Description Framework) [76] builds on XML to better manage *semantic interoperation*. RDF is a data model designed to standardize the definition and use of metadata, in order to better describe and handle data semantics. RDF was designed to meet the following goals: i) having a simple data model with formal semantics and provable inference, ii) using an extensible URI-based vocabulary, iii) supporting XSD data-types, and iv) allowing a transparent description of Web resources.

The RDF data representation model is based on triplets (*Object*, *Attribute*, *Value*), more commonly known as $A(O,V)$. A triplet binds an attribute value to an object, giving the relationship a semantic meaning. Objects, attributes and values underline any kind of Web resources, identified using URIs. Values can also contain literal (text) contents. For example, consider two resources: *person* and *name*. These can be instantiated as follows: *person*: p_1, p_2, and *name*: n_1, n_2. Basic triplets that can be modeled here are: $name(p_1, n_1)$, and $name(p_2, n_2)$. Yet to allow more semantic expressiveness, RDF allows the creation of so-called predicates or (semantic) properties, which underline (more specific) subsets of the resources. For instance, to link the concept *person* to the concept *name*, a property *hasName* can be used, such as $hasName(p_1, n_1)$ precisely indicates - without ambiguity - that p_1 is a *person* and n_1 is her/his *name*. Note that in the SW, the triplet is no longer called: *object-attribute-value*, but rather by *resource-property-value* [38].

In fact, RDF has a formal semantics [51], with a predefined namespace (i.e., *rdf*) and elements prefix tags (such as *rdf:type*, *rdf:Property*, *rdf:XMLLiteral*, etc.), along with rigorously defined notions of semantic relations and dependencies, which provide a solid basis for creating and reasoning about the meaning of an RDF expression [64]. Nonetheless, it is important to note that RDF by itself is just a data model, i.e., an *A-Box*. What really defines the intended semantics behind this data model is the use of a solid, rigorous and well-defined vocabulary, i.e., a *T-Box*. In this context, RDFS (RDF Schema) [25] was introduced as the *T-Box* for RDF (the schema of a KR, cf. Section 2.3.2) shaping the model in which the RDF data instances will be inserted. RDFS models and manipulates classes, similarly to an object-oriented programming language. The main difference lies in the definition of classes and properties: instead of defining a class in terms of the properties its instances may have, RDFS describes properties in terms of classes of resources to which they apply [25] (as briefly mentioned with the *HasName* example discussed above, where the property was defined in terms of its resource). In other words, RDFS is a semantic extension of RDF [76] (with a dedicated namespace: *rdfs*), providing mechanisms for describing groups of related resources and the semantic relationships between these resources [25], especially at the object-oriented level with hierarchy and heritage implementations (using constructs such as *rdfs:Class*, *rdfs:SubClassOf*, *srds:SubPropertyOf*), and at the predicate's level specifying the property's domain and range of application (using *rdfs:domain* and *rdfs:range*). In short, an effective semantic inference engine requires both the RDF data instances (the A-Box) and their RDF Schemas (the T-Box) to run properly [51, 64].

Yet, despite its expressiveness, RDFS carries some limitations [6]: i) local scope of properties: it does not allow restrictions or generalizations of properties, ii) disjointness of classes: two classes cannot be formally identified as disjoint, iii) Boolean combinations of classes: it does not allow Boolean set operators (e.g., union, intersection, complement, etc.) when creating classes, iv) cardinality restrictions: it is not possible to define a restriction on how many distinct values a property may or must take, and v) special characteristics of properties: it does not allow transitive, unique and/or inverse properties. While RDFS is semantically more expressive than RDF in describing basic Web resources (e.g., Web pages), it still lacks in expressiveness specially when describing complex resources such as ontologies [6, 51], which is where OWL comes to play.

2.3.5. Web Ontology Language (OWL)

While basic Web resources can be effectively described using binary ground predicates (using RDF) and/or subclass and property hierarchies (using RDFS), yet Web experts have identified the need for more semantic expressiveness: building common semantic information sources, or so-called Web Ontologies, serving as knowledge references for software agents when automatically processing Web resources (similarly to dictionaries and encyclopedias serving as knowledge references for human agents) [37, 44]. Hence, filling the gaps of RDF and RDFS, W3C has introduced OWL (Ontology Web Language) as a standard for describing ontologies on the Web. OWL is built upon RDF and RDFS, and inherits most basic RDFS elements (including constructs such as *owl:Class*, *owl:ObjectProperty*, *owl:DataTypeProperty* which extend the expressiveness of their *rdfs:Class* and *rdfs:Property* ancestors) allowing property specialization, identification of disjoint classes, specification of cardinality or data-type restrictions, etc. [6].

Due to the semantic richness of OWL constructs which might be complex to use in everyday scenarios, OWL has been presented in three W3C specifications [37, 44, 79], depending on the reasoning level: i) OWL Lite, ii) OWL-DL, and iii) OWL full. OWL Full is the basic version using all the OWL language primitives (as briefly describe above). It is fully compatible with RDF(S), both syntactically and semantically, but can be complex to handle given its powerful expressiveness. OWL-DL (OWL-Description Logic) is a sub-language of OWL Full which restricts the way the constructs from OWL and RDF(S) can be used [44, 54]. For instance, OWL-DL requires that every resource either be a class, object property, data-type property or instance, but cannot be treated as both a class and an instance at the same time. The category of each resource must be explicit in all ontologies (each resource must have an *rdf:type* statement). One cannot use a resource as a *class* without explicitly describing it as such. Note that such restrictions do not exist in OWL Full (where it is possible to treat a class as an instance, and there is no need to explicitly declare the type of each resource). OWL-DL is not compatible with RDF but is more efficient in reasoning as a Description Logic (DL) [33]. As for OWL Lite, it is a sublanguage of OWL-DL which excludes enumerated classes, disjointness statements and arbitrary cardinality (among others). It is intended for users with simple modeling needs and is user and implementation friendly [44].

In addition, in the most recent: OWL 2 specification, three new OWL sub-language specifications have been tailored: i) OWL 2 EL is dedicated to applications employing ontologies containing large classes and property descriptions, aiming to guaranty high scalability (typically maintaining polynomial time with respect to ontology size) in solving basic reasoning problems (EL stands for the use of the EL family of description logics [11], i.e., logics providing only Existential qualification), ii) OWL 2 QL is aimed at applications using very large volumes of instance data where query answering is critical (thus the acronym QL). It allows queries to be straightforwardly performed using conventional relational databases (reaching LOGSPACE querying with respect of the data size), and iii) OWL 2 RL is designed to allow scalable reasoning using rule-based engines (thus the acronym RL: Rule Language), trading (simplifying) the expressivity of the language in order to gain in efficiency, while maintaining the ontology's consistency (e.g., class satisfiability, class subsumption, instance checking, etc.).

Note that all OWL (Lite, DL, and full) ontologies come down to OWL 2 ontologies. Hence, they can be viewed as profiles of OWL 2.

Having defined powerful languages and constructs (RDF, RDFS, and OWL) to describe elaborate semantic information, the W3C has identified the need for finding a powerful means to accessing and querying them [44, 61], which is where SPARQL comes to play.

2.3.6. Data Manipulation & Querying

For any database to be useful (other than for data storage), it needs to be queryable. In other words, the value of (semantic information) contents depends on how easy it is to search, access and manage [90]. Hence, several solutions have been proposed for querying XML, RDF and OWL instances, namely XQuery [32] for XML-based (and RDF/XML) documents, and SPARQL [93] for RDF-based (and OWL) documents. These languages are specially designed to resemble SQL in their grammars and constructs, to facilitate their usage by programmers. For instance, a simple query in SPARQL would have the following form: `SELECT * WHERE {S, P, O}`, where (*S*, *P*, *O*) are the RDF triplets (subject *S*, predicate *P*, object *O*). More complex queries can be created in SPARQL to search about anything in an RDF document [50]. By adding other namespaces such as RDFS or OWL, SPARQL can be extended to query more expressive ontology documents. Yet syntactically speaking, XQuery and SPARQL are not easy-to-use (straightforward) query languages, and generally require deep knowledge and special skills in XML and RDF-based languages to be manipulated efficiently. Recent research efforts have focused on developing simplified tools or alternatives, including visual interfaces [4], keyword-based querying [99], and eventually programming APIs (Jena [7], OWL API [92], etc.) to help create, manipulate and query structured, semantically rich and ontology-based documents. In addition, XQuery and SPARQL are based on exact matching and do not support ranked queries via textual/structural similarity. Hence, several attempts have been made to extend these query languages in order to support ranked results [77, 111].

2.4. Open Issues and Current Trends

Despite the technological achievements in semantic data representation and manipulation, realizing the SW vision still faces many challenges:

- *Creating comprehensive ontologies* containing all concepts required in a given domain remains a very difficult task, owing to the difficulty of managing a huge KB (e.g., computing the semantic similarity between two concepts using the WordNet taxonomy requires several hours [75, 109]). This problem could be solved by creating several domain-based ontologies, which is currently a hot topic [114, 123].
- *Compromising between expressiveness and reasoning* is a very delicate issue: the more the language is expressive, the harder is the reasoning to be achieved [53]. Hence, choosing the DL to be used, and the level of semantic details (in the KBs) remains a critical task.
- Since ontologies are created separately by different developers, some concepts are being redefined constantly. *Mapping structurally and semantically-rich documents* (XML/RDF-based) [8, 109] and ontologies [82, 100] is needed to lower redundancy and increase efficiency.
- *Simplifying the use and manipulation of KBs* and semantic references, including visual interfaces [4], approximate querying [111], keyword-based querying [99], and eventually programming APIs (Jena, OWL API, etc.) to help create, manipulate and query semantically rich documents, leading to the study of semantic mashups [45, 107].
- *Improving privacy protection* strategies to reduce the information disclosure caused by data sharing and linkage. It is in essence worthy to note that data sharing and linkage are not always beneficial and could be dangerous in several situations (social networks, health, etc.). New relevant privacy protection solutions must be provided to protect sensitive information (which might be multimedia-based) especially those that Web 3.0 users would like to keep private [1, 42].
- Promoting *service intelligence* and *atomicity* (the ability of services to interact automatically, central in *user-machine* and *machine-machine* interactions), remains one of the most upcoming challenges of the SW. Allowing software agents to perform intelligent tasks relies on the awareness (intelligence) of the software agents, and their ability to learn, act and evolve with time. This promotes the dawn of the *Intelligent Web* (*Web 4.0*) era [67, 115].

3. TOWARD DIGITAL WEB ECOSYSTEMS

With the dawn of the SW era on one hand, and the development of the mobile Internet (smart phone-based) technology on the other hand, we are heading toward a new Internet revolution: the *Intelligent Web* (IW), also known as the *Internet of Things* (IoT) vision, where objects of the real world (with added digital components) are linked with objects of the virtual world (software agents on the Web), enabling connectivity at anytime/anyplace, using collective knowledge and ubiquitous computing [98] (integrating computing capabilities of intelligent network devices).

In the following, we briefly present the IoT (IW) vision, and then discuss some of the main challenges toward achieving it.

3.1 Internet of Things Vision

The concept of *Internet of Things* (IoT) can be defined as a dynamic global network infrastructure; merging computer networks, communications, Web, and artificial intelligence (hence the label: *Intelligent Web*); to manipulate physical and/or virtual objects or "things" (such as computers, sensors, people, TVs, vehicles, smart phones, passports, luggage, etc.) having unique identities, physical attributes, virtual personalities, capable of intelligent processing, interaction, and self-sustainability [47].

In the vision of the IoT, it is foreseeable that any (physical or virtual) object will have at least one unique way of identification (directly by a "Unique Identifier" or indirectly by some "Virtual Identifier" techniques, e.g., URI), creating an addressable set of objects having the capabilities of addressing, communicating and exchanging information (knowledge) and processes (services) with each other. Also, the growth of storage capacity at lower costs, with enhanced processing capabilities in intelligent mobile terminals (PDAs, smart phones, tablets, etc.), and ubiquitous connectivity will allow terminals to handle larger amounts of information (knowledge) and perform collective knowledge processing, i.e., combining processing capabilities to perform common tasks [104]. This will promote a more advanced view of the Web: connecting physical and/or virtual (software-based) objects via network architectures exhibiting high levels of heterogeneity, with different models, addressing schemes, communication protocols, and processing capabilities, which are expected to interact and evolve in the same environment [104]. Hence, SW experts and pioneers are expected to deal with different challenges in order to promote the IoT (IW) vision.

3.2 Main Challenges with respect to the Semantic Web

3.2.1 Connectivity and Network Infrastructure

The IoT vision underlines the concept of a network of networks, linking public/private infrastructures, dynamically extended by connection points consisting of the "things" (terminals) connecting to one another. Enhanced processing capabilities and always-on connectivity, will make terminals gain a central role in communications: terminals (deemed henceforth intelligent) will be able to form bridges between existing network structures thus extending the overall infrastructure capacity.

In this context, developments in network technologies such as RFID (Radio Frequency Identification), short-range wireless technologies, and sensor networks, coupled with enhancements in network addressing techniques, such as IPv6 to expand address space, become critical to the IoT vision, allowing to reach more objects in the physical/virtual worlds and connect them together through the Internet. Yet, scalability and cross platform compatibility between diverse networked systems remain an open problem. Network technologies have to offer solutions that allow ubiquitous access, i.e., connecting any terminal to the network, which will require dedicated network protocol translation gateways (defining the correspondences between diverse network and communication protocols), in comparison with today's IP (Internet Protocol) which only allows end-to-end communication between devices without any intermediate protocol translation. Here, enhancements in wireless and sensor communication protocols (from direct transmission and minimum transmission energy [112], to multi-hop routing, multi-path routing, and cluster-based routing [121]) can be exploited to improve scalability and robustness for dynamic networks, reducing the amount of information that must be transmitted between terminals (e.g., integrating data fusion within the routing protocol [40]) aiming to enhance connectivity.

In addition, ensuring effective and ubiquitous connectivity for terminals requires extensive terminal design efforts covering: i) *mobility*, allowing occasional or continuous mobility of terminals in the selected environment, ii*) resources,* and *energy efficiency*, since the terminals' resources availability might vary from limited (e.g., with sensor terminals) to unlimited (e.g., with Cloud computing systems), iii) supporting different *communication modalities*, ranging over electromagnetic communication (radio frequency), to optical, acoustic, as well as inductive and capacitive coupled communications, and iv) supporting different *network topologies*, such as single hop, star, multi-hop, mesh and/or multitier. Yet, the sheer diversity of terminals which will be supported stipulates that no single hardware and software platform can hope to support the whole design space [47]. Hence, heterogeneous systems and interoperability will have to be addressed.

3.2.2 Semantic Interoperability and Interaction

As described in the previous section, the IoT (IW) vision will exhibit high levels of heterogeneity as different terminals (in terms of functionality and technology) are expected to connect and interact in the same communication environment. Hence, *semantic interoperability* becomes an essential requirement: allowing terminals to communicate "meaningfully" with each other, exchanging data (knowledge) and processes (services) despite the heterogeneous nature of the underlying information structures and communication protocols. Fortunately, the issue of semantic interoperability has been at the center stage of SW-related studies, introducing standard data representation and manipulation technologies (e.g., XML, RDF, OWL, and SPARQL) to facilitate information and knowledge interchange (cf. Section 2.3); and has also been investigated in related domains namely: Service Oriented Architectures (SOA), aiming to improve communication and information exchange between heterogeneous service providers and requestors [85].

Service requestors/providers in SOA are generally dynamic, operating on the "publish-find-bind" paradigm principle, where services are dynamically added and described (published) in a service registry. The service descriptions are then used to search (find) and associate (bind) the service to the service requestor. The problem of semantic interoperability is far more acute in such dynamic situations, due to the lack of predefined relationships between the requestors/providers [47]. Hence, experts have suggested leveraging work in the SW to address the issue of semantic interoperability for SOA environments, which could be henceforth extended to the IoT (IW) [47]. On one hand, the development of comprehensive shared information models using SW technologies (e.g., shared RDF or OWL reference ontologies defining common semantics following the SW vision) can enhance semantic interoperability among the participant terminals [65]. However, the problems with this approach remain: i) the complexity of producing a universal ontology (encompassing all semantic descriptions concerning all possible terminals and processes), and ii) semantic rigidity, underlining the difficulty in updating/extending the reference ontology once it is defined (in order to handle new terminals, new processes, and new information [44]). It is worthy to note that there is an interesting ongoing work currently conducted by W3C Semantic Sensor Network Incubator Group aiming to provide an ontological representation of a sensor network to order to solve these problems[4]. On the other hand, semantic interoperability can be achieved by providing appropriate semantic mediators (translators) at each terminal's end, to allow the conversion to the information (knowledge) format which the terminal understands. A combination of context independent shared information models can be utilized (using SW data representation technologies such as XML, RDF, and OWL), coupled with context specific information specialization approaches (using SW data manipulation technologies such as XQuery and SPARQL) to achieve semantic interoperability [129]. This semantic mediation approach avoids imposing a unique information model (e.g., a unique reference ontology) that has to be agreed and adopted by everybody, thus allowing intelligent Web terminals (agents) to select the formats best suited for their needs, while still being able to interact and use services offered by other terminals [129]. Yet, defining semantic mediators for each Web terminal (agent) does not seem feasible in practice and remains an open issue. Hence one can predict that a certain compromise between shared semantic references and semantic mediators can be made.

In addition, to improve semantic interaction, intelligent Web terminals (software agents) are expected to communicate their state, location, and requirements to their counterparts, to be monitored or displayed in a common representation. This will allow remote authorized terminals (agents) to query or update the information or state of the terminal at hand [47]. Hence, there is a need in the IoT (IW) framework to handle and synchronize information concerning the collective of intelligent terminals (agents) interacting in a collaborative environment.

3.2.3 Collective Knowledge Management

Adding to the issues of connectivity and semantic interoperability, SW technologies and social networking services are promoting a new form of collaboration. Nowadays, it is common for Web users to contribute their multimedia data and knowledge to the community, allowing the editing and manipulation of such public knowledge in a collaborative environment (e.g., Wikis, blogs, Foursquare, Google+, etc.). As a result, the Web is becoming more than a distributed container of (raw and/or semantic) multimedia data, but is increasingly harnessing *Collective Knowledge* (CK). CK is the development of knowledge assets or (semantic) information resources from a distributed pool of contributions [9]. It is viewed as the combination of all known multimedia data, information, and meta-data concerning a given (set of) concept(s), fact(s), user(s), or processes (s), as well as the semantic links between them [9]. Hence, intelligent terminals (software agents) connected to the Web are expected to automatically analyze and handle large collections of multimedia data with their contents, links and transactions [17], using the sum of their respective intelligence and knowledge, producing new knowledge (new ontologies, new semantic annotations, new inference rules, etc.) in order to improve information management (indexing, storage, search, and retrieval) [71].

CK management has been addressed from different perspectives in different application scenarios mainly in Web-related research and applications [46] and has recently received strong attention. For example, collaborative social tagging of Web resources [46, 101] is viewed as an attempt of acquisition and sharing of so-called CK concerning a given community. Wikis have also become popular tools for collaboration on the Web among many vibrant online communities [36, 94] promoting CK extraction and representation [110].

Hence, extracting and handling CK requires intelligent Web terminals (agents) which are not only capable of understanding and meaningfully processing information (using SW technologies), but which are also capable of thoroughly collaborating and even "reasoning" together, as a collective, to produce and handle common CK, leading to more sophisticated (intelligent) services, as well as achieving the ultimate goal: *collective intelligence*, where Web agents are able to automatically sustain themselves and evolve without direct human intervention.

---

4 http://www.w3.org/2005/Incubator/ssn/XGR-ssn-20110628/

3.2.4 Collective Intelligence: Sustainability & Evolution

An intelligent Web terminal is viewed as a software agent capable of handling existing SW technologies (e.g., RDF, OWL, SPARQL) in order to enhance collaboration between humans and machines [41, 55, 57]. Yet, individual intelligence needs to be coordinated in order to enhance its own capabilities as well as the capabilities of its surrounding entities [68]. Here, a major problem facing autonomous as well as collaborative intelligent data processing on the Web is the nature of shared multimedia data sources, which often exist in loosely distributed environments, with unstructured and heterogeneous contents, created by different users (and/or terminals/agents) with different profiles, formatted following different standards, and using different kinds of techniques.

In this context, *Digital Ecosystems* are promoting a framework to effectively handle data and information in a distributed and heterogeneous environment [22]. A digital ecosystem can be viewed as a distributed, adaptive, and open system with properties of self-organization, scalability and sustainability inspired from biological ecosystems, where (human and software) agents collaborate, interact, compete, and evolve autonomously in order to automatically solve complex and dynamic problems [28, 29]. The concept and study of digital ecosystems promise to address various challenges, concerning knowledge management, related services, and knowledge manipulation.

Improving Knowledge Management

- Autonomous organization of CK (representation, indexing, filtering, and storage), generated based on data/metadata contents which are:
  - Heterogeneous (originating from different sources, with different schemas and properties),
  - Of dynamic nature (e.g., Web contents vary regularly, also Web services are regularly updated following service needs),
  - Distributed on different sites (such as the nature of the Web),
  - Unstructured or semi structured (data on the Web usually consists of free form text or text-rich XML which, which are usually more difficult to handle via automated processes than rigorously structured XML or relational tables),
  - Rich in multimedia contents (combining images, maps, sounds, animations, videos, etc.).
- Automatic knowledge evaluation and verification [117, 118] to:
  - Prevent erroneous knowledge manipulation and processing later on (e.g., recommending services which are not pertinent to the users, based on inaccurate services/users knowledge descriptions; or composing non-compatible services/mashups)
  - Compare the produced CK with real facts, in order to verify its accuracy in correctly describing the data/services (through analyzing user feedback, data mining, statistical analysis, etc.).
- Preservation of CK over time, through:
  - Versioning (maintaining different versions of the CK captured at different time stamps, hence allowing to maintain a history of CK variation and evolution over time) [113],
  - Identifying the minimum amount of CK necessary for preservation, in order to safeguard pertinent data - following the users' needs, or perform certain specific tasks – following system requirements (such as intelligent search, storage, indexing, etc.), instead of needlessly preserving the whole CK, hence drastically reducing recourse consumption.
- Extrapolation of CK in order to accommodate to the needs of users and the constraints of the system. Analysing CK history (CK versions) can prove to be crucial here, allowing to study the evolution and variation trends of CK, in order to consequently infer rules to automatically extrapolate new knowledge [10, 49].

Most of these challenges require combining efforts originating in different domains, namely: the Semantic Web, Service Oriented Architectures (SOA), Multi-Agent Systems (focusing the interaction among intelligent agents within an environment), and Evolutionary Systems (using evolutionary computing, e.g., evolutionary algorithms, to produce solutions [73]).

Promoting Sophisticated Services

In this perspective, *collective intelligence* will emerge, which consists in effectively mobilizing the skills of a group of agents in a digital ecosystem to emerge and handle the collective knowledge (CK) from all agents. The new form of CK will be automatically aggregated and recombined to create new knowledge, new rules, and/or new ways of learning what individual agents of the ecosystem cannot do individually [46]. For instance, knowledge recommendation methods [72] can be developed to identify form the large pool of CK maintained, the pieces of knowledge and data contents which are required by an agent (human user or automated process), based on explicit needs, past experiences, profile, and preferences.

In addition to CK exchange and manipulation, as intelligent Web terminals move and interact within their environment, events will be automatically generated (e.g., service requested from provider, action performed by client, at a certain location, etc.). These events need to be subsequently enhanced with relevant knowledge in order to describe the context in which each event happened [84] (such as why a thing was observed at a location, or how and why it interacted with another thing) and act accordingly. This highlights the need for innovation to automatically interpret events and processes related to Web terminals (agents) in given contexts, adding semantic annotations and predicting what will happen, and what precautionary measures could be taken to optimize data/services sustainability and evolution. Such issues can be handled through the use of evolutionary computing [73], as digital ecosystems will likely solve problems by evolving solutions, e.g., starting with a current set of semantically-rich events (i.e., a set of *solutions*), and then iteratively applying selection, replication, recombination, and mutation, to produce enhanced and more useful solutions [28]. Selection is based on answering to a fitness function which is determined by a specific problem, allowing to evolve better solutions over time [73].

Simplifying Knowledge Manipulation

As the Web requirements evolved - from simple data visualization (Web 1.0), through data insertion/removal (Web 2.0), to data/knowledge manipulation (Web 3.0), and then semi-automated CK manipulations (Web 4.0) - programming the Web has taken a turn from traditional experts in Web programming towards non-expert Web surfers behaving as providers and consumers simultaneously. Thus, digital ecosystems would simplify knowledge manipulations through the integration of "Web-based manipulation techniques by non-expert users" [108] in order to guaranty and preserve agents' collaborations, interaction and evolution (in particular human machine interactions). These techniques such as Mashups [74], Wrappers [12], Information Extraction [34], and Dataflow Visual Programming Languages [78] aim - either partially or fully - at providing non-expert programmers and users with means to develop and deploy their manipulation operations ranging from simple selections/visualizations to complex modifications/ alterations and semantic manipulations (i.e., semantic similarity comparisons and filtering). In other words, using these techniques with semantic capabilities, digital ecosystems promise to provide an application layer for non-experts to program the Web through the manipulation of CK.

In this context, we are still faced with majors difficulties in fostering knowledge sustainability and evolution following the IoT (IW) vision, namely: i) handling data, events, and intelligent terminals (agents) in a heterogeneous and distributed environment, ii) defining complex and dynamic fitness functions to sustain and evolve pertinent knowledge and processing solutions, and iii) self-organizing the obtained knowledge/processing arising from interactions among intelligent terminals (agents) and their environment (e.g., how to automatically produce new ontologies, new semantic mediators, or new events, etc.).

To sum up, the extension of the Web toward the IW (IoT) vision, where machines and software agents meaningfully and intelligently manipulate and exchange information and services without human interaction, remains at its early stages; as it faces many technological challenges, ranging over: always-on connectivity (requiring universal identification techniques as well as open network infrastructures), semantic interoperability (requiring comprehensive semantic references and mediation gateways between heterogeneous terminals/agents), CK management (in order to improve data indexing, storage and retrieval in a distributed and heterogeneous environment), leading to autonomous data/services sustainability and evolution (fostering the processing capabilities of terminals/agents on Web in order to produce and manipulate CK in solving problems).

4. APPLICATION DOMAINS

Despite the challenges to realizing the Semantic (and Intelligent) Web vision(s), experts highlight many potential applications, which can be grouped as: i) SW (software-based), and ii) IoT (real world based).

The main SW application domains range over: information retrieval and extraction, machine translation, content analysis, and lexicography.

Information Retrieval: As mentioned earlier, state-of-the-art search engines do not use explicit semantics to prune out documents which are not relevant to a user query [14, 87]. Hence, semantic document and query indexing (i.e., associating accurate semantic labels to document/ query concepts, with respect to a reference KB) would allow it to eliminate documents containing the same words used with different meanings (thus increasing precision) and to retrieve documents expressing the same meaning with different wordings (thus increasing recall), e.g. [14, 70, 111].

Information Extraction: Extracting semantically related concepts form a corpus (also called semantic categories or labels), which is particularly useful for part-of-speech tagging (i.e., the assignment of parts of speech to target words, i.e., concepts, in context) [3], named entity resolution (i.e., the classification of target textual items into predefined categories, i.e., semantic concepts) [91], and text categorization (i.e., the assignment of predefined labels, i.e., concepts, to target texts) [13, 21].

Machine Translation: The automatic identification of the correct translation of a word in context (called *machine translation*) is a critical task in the SW vision, as it requires word sense disambiguation (i.e., associating right sense for the right word, among a set of possible senses given a reference KB) [56, 80], given that the disambiguation of texts should help translation systems choose better candidates. It also becomes central for live speech translation techniques [31, 116].

Content analysis: It underlines the analysis of the general content of a text in terms of its ideas, themes, etc. which is gaining importance in various applications such as i) blog classification (e.g., introducing simple and effective methods to semantically classify blogs, determining their main topics, and identifying their semantic connections [15, 16], and ii) semantic social network analysis (e.g., disambiguation of entities in social networks, and identifying semantic relations between users based on their published materials [2, 89]).

Lexicography: It underlines the creation of dictionaries or ontologies (i.e., semantic references). While lexicography was restricted to human experts, with the advent of the SW, there has been a growing interest in the field of automatic ontology generation, using empirical sense groupings and data analysis (statistically significant indicators of context for new or existing senses [62, 127]), and the integration/combination of existing semantic references and structured documents to produce a new one (creating domain or application-specific ontologies, etc. [86, 114]).

The main IoT application domains range over: systems status monitoring, intelligent transportation systems, intelligent buildings monitoring, improving medical technology and business environments.

Systems status monitoring: it leads to vehicle/aircraft monitoring, detecting and analyzing conditions such as pressure, vibrations, temperature, etc., using intelligent sensors available inside and/or outside the vehicle/aircraft, connected to the main monitoring system using sensor networking capabilities [27]. The data collected in such environments gives access to customized usage trends, facilitates maintenance planning, allows condition-based maintenance, reduces waste, and can be used as input for evaluating and reducing energy consumption during vehicle aircraft operations [102, 104].

Intelligent Transportation Systems: it involves public and private transportation interaction. Here, a vehicle is perceived as a “thing”, combining intelligent processing capabilities (vehicles’ computer system) with wireless communication technology and RFID, allowing it to automatically communicate and interact with other vehicles as well as with surrounding network systems [120], e.g., placing emergency calls or breakdown calls when necessary, collecting data from surrounding “things”: vehicles and buildings in the vicinity, transportation infrastructure (e.g., road, rail, etc.), sensors in the load it is carrying (humans, goods, etc). In this context, vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) connectivity can significantly improve intelligent transportation systems (ITS) applications such as vehicle safety services and traffic management [104].

Intelligent buildings monitoring and management: it allows automatic energy metering, measuring energy consumption and transmitting the information to the energy provider. Here, sensors for temperature, humidity, etc. can provide useful data to automatically adjust the comfort level within the building, as well as to optimize energy consumption (for heating or cooling for instance) [106]. Also, monitoring and reacting to human activity within a building, can allow the detection and handling of exceptional situations namely when people need help or assistance (which is extremely useful in supporting the elderly, especially in an aging society) [104, 106].

Improving medical technology and healthcare: Implantable wireless identifiable devices (combining sensor, RFID, Bluetooth, ZigBee, and WiFi technologies) could be used to store health records that could save a patient's life in emergency situations, especially for people with diabetes, cancer, coronary heart disease, etc., as well as people with complex medical device implants, such as pacemakers and organ transplants. As “things” become more integrated within the human body, so-called *body area networks* can be formed [66, 126], capable of sensing and processing sensor data, as well as communicating with treating physicians and emergency services. An example is the automated internal Cardioverter-Defibrillator, built into the human heart, allowing to autonomously decide on when to administer shocks to defibrillate (namely when heart activity is failing), such as the device is always connected and monitored by the medical officer involved [104, 119].

Improving business life-cycle: It increases mobile and smart device capabilities and connectivity by allowing business logic to be executed on the edges of a network, enabling certain business processes to be decentralized in order to enhance performance, scalability, and local decision-making [67, 104]. Hence, more productive business environments emerge, such as factories become smart and interactive, allowing production-related data relayed in real-time. Each object under production is expected to be attached to information processing component, from production to the end of the lifecycle, hence keeping track of the history of the item and its current status, stored on the object tag or in the information system, reflecting valuable information for product design, marketing and the design of related services [59, 104].

The reader can refer to [104] for a more detailed presentation concerning IW (IoT) application domains and scenarios.

5. CONCLUSIONS

In this paper, we gave a brief overview on the Semantic Web vision and its evolution towards the Intelligent Web (Internet of Things), covering the main technological breakthroughs ranging from: knowledge bases and semantic representation, to extensible and interoperable XML-based data representation technologies (namely RDF, RDFS, OWL and SPARQL), aiming to improve data modeling, annotation, manipulation, search and integration, on the Web. We discussed some of the main challenges toward achieving the Intelligent Web vision: connectivity, semantic heterogeneity, collective knowledge management, as well as data sustainability and evolution. We also briefly described some of the main application domains. To conclude, while the traditional view of Web was confined to a virtual (software-based) world handling software agents, the Intelligent Web is promoting a new era: that of ubiquitous computing and ambient intelligence, where people are surrounded with intelligent terminals and electronic environments which are sensitive and responsive to their demands and desires.

6. ACKNOWLEDGMENT

This study is partly funded by the CEDRE research collaboration program, project *AO 2011*, entitled: *Easy Search and Partitioning of Visual Multimedia Data Repositories*, funded by the French CNRS (National Center for Scientific Research) and the Lebanese CNRS.

7. REFERENCES

[1] Al Bouna B., Chbeir R., Gabillon A., and Capolsini P., *A Fine-Grained Image Access Control Model.* SITIS'12, 2012. pp. 603-612.

[2] Aleman-Meza B. *et al.*, *Scalable Semantic Analytics on Social Networks for Addressing the Problem of Conflict of Interest Detection.* ACM Transaction on the Web (TWeb), 2008. 2(1):7.

[3] Allan J. and H. Raghavan, *Using Part-of-Speech Patterns to Reduce Query Ambiguity.* Inter. ACM SIGIR Conference, 2002. pp. 307-314.

[4] Ambrus O. *et al.*, *Konduit VQB: a Visual Query Builder for SPARQL on the Social Semantic Desktop.* Proceedings VISSW'10 Workshop, 2010.

[5] Anderson P., *What is Web 2.0? Ideas, technologies and implications for education.* JISC Technology and Standrads Watch, 2007. pp. 64.

[6] Antoniou G. and Van Harmelen F., *Web Ontology Language: OWL.* Handbook on Ontologies, 2004. pp. 67-92.

[7] Apache Jena. *ARQ - A SPARQL Processor for Jena.* http://jena.apache.org/documentation/query/index.html [cited Jan. 2012].

[8] Araújo S. *et al.*, *SERIMI - Resource Description Similarity, RDF Instance Matching and Interlinking.* Inter. Semantic Web Conf.(ISWC'11), 2011.

[9] Armbrecht R. *et al.*, *Knowledge Management in Research And Development*, Research-Technology Management, 2001. 44(4):28-48.

[10] Auer S. and Herre H., *A Versioning and Evolution Framework for RDF Knowledge Bases.* Perspectives of Systems Informatics, 2007. 4378:55-69.

[11] Baader F. *et al.*, *Pushing the EL Envelope.* Inter. IJCAI'05 Conf., 2005.

[12] Barrero F. et al., *Adapting Searchy to extract data using evolved wrappers.* . Expert Syst. Appl. 3, 2012. 9(3): 3061-3070.

[13] Bawakid A. and Oussalah M., *A Semantic-based Text Classification System.* IEEE Inter. Conf. on Cybernetic Intelligent Systems (CIS'10), 2010, 1-6.

[14] Baziz M. *et al.*, *A concept-based approach for indexing documents in IR.* INFORSID 2005, 2005. pp. 89-504, Grenoble, France.

[15] Berendt B. & Trumper D., *Semantics-based Analysis & Navigation of Heterogeneous Text Corpora: The Porpoise News and Blogs Engine.* Web Mining Apps in E-comm. & E-services Studies in Compt. Intel., 2009. 172:45-64.

[16] Berendt B. and Navigli R., *Finding Your Way Through Blogspace: Using Semantics for Cross-Domain Blog Analysis.* AAAIS, 2006. pp. 1–8.

[17] Berners-Lee T. *et al.*, *The Semantic Web.* Scientific American, 2001. 284(5):1:19.

[18] Berners-Lee T. and Lohr S., *Tim Berners-Lee Looks Back: the "//" in Web Addresses Was Unnecessary.* http://tinyurl.com/bgse2cw., 2009.

[19] Bertails A., Herman I., and Hawke S., *The Semantic Web: The Internet and Tomorow's Web.* Industrial Realities (in French), 2010. pp. 80-89.

[20] Bizer C., Heath T., Berners-Lee T., and Hausenblas M., *WWW'12 Workshop on Linked Data on the Web.* CEUR Workshop 937, CEUR-WS.org, 2012.

[21] Bloehdorn S. and Hotho A., *Text Classification by Boosting Weak Learners based on Terms and Concepts.* ICDM, 2004. pp. 331-334.

[22] Boley H. and Chang E., *Digital Ecosystems: Principles and Semantics.* Digital EcoSystems and Technologies Conf. (DEST'07), 2007, pp. 398-403.

[23] Bozzon A. *et al.*, *Liquid Query: Multi-Domain Exploratory Search on the Web.* Proc of Inter. WWW '10 Conf., ACM NY, USA, 2010. pp. 161-170.

[24] Bray T. *et al.*, *Extensible Markup Language (XML) 1.0 - 5th Edition.* W3C recommendation, 2008; Available from: http://www.w3.org/TR/REC-xml/.

[25] Brickley D. and Guha R. V., *RDF Vocabulary Description Language 1.0: RDF Schema.* W3C Recomm., 2004. http://www.w3.org/TR/rdf-schema/.

[26] Brin S. and Page L., *The Anatomy of a Large Scale Hypertextual Web Search Engine.* In Computer Networks and ISDN Systems, 1998. 30 (1-7):107-117.

[27] Brintrup A., R. D.C., Kwan S., Parlikad A., and Owens K., *Roadmap to Self-Serving Assets in Civil Aerospace.* CIRP IPS2 Conf., 2009, 323–331.

[28] Briscoe G. and De Wilde P., *Computing of Applied Digital Ecosystems* The International ACM MEDES'09 Conf., 2009. pp. 28-35.

[29] Briscoe G. *et al.*, *Digital Ecosystems: Ecosystem-Oriented Architectures.* Natural Computing, 2011. 10(3):1143-1194.

[30] Carpineto C. and Romano G., *A Survey of Automatic Query Expansion in Information Retrieval.* ACM Computing Survey, ACM NY, 2012. 44(1):1.

[31] Carpuat M. and Andwu D., *Improving Statistical Machine Translation using Word Sense Disambiguation.* EMNLP-CoNLL'07, 2007. pp. 61-72.

[32] Chamberlin D. *et al.*, *XQuery : A Query Language for XML, http://www.w3.org/TR/2001/WD-xquery-20010215.* 2001 [May 2011].

[33] Champavère J., *From Knowledge Representation to the Semantic Web: An Overview.* pp.14 (in French), 2010.

[34] Chang H. and Lui S.-C., *IEPAD: Information Extraction Based on Pattern Discovery.* In Proc. of Inter. WWW Conference, , 2001. pp.681--688.

[35] Chen M. *et al.*, *Data, Information and Knowledge in Visualization.* IEEE Computer Graphics and Applications, 2009. 29(1):12-19.

[36] Dahlander L. *et al.*, *Online Communities and Open Innovation : Governance and Symbolic Value Creation.* Industry and Innovation, 2008. 15(2):115-123.

[37] Dean M. and Schreiber G., *OWL Web Ontology Language Reference.* W3C Recommendation, http://www.w3.org/TR/owl-ref/. 2004.

[38] Decker S. *et al.*, *The Semantic Web: The Roles of XML and RDF.* IEEE Internet Computing, 2000. 4(5):63-74.

[39] Ding L. and Finin T., *Boosting Semantic Web Data Access using Swoogle.* Proceedings of the 20th Inter. AAAI'05 Conf., 2005. (4): pp. 1604-1605

[40] Feng C. H. *et al.*, *UPS: Unified Protocol Stack for Emerging Wireless Networks.* Elsevier Ad Hoc Networks Special Issue on Cross-layer Design in Ad Hoc and Sensor Networks, 2013. Vol. 11, pp. 687-700.

[41] Fong T., Thorpe C., and Baur C., *Collaboration, Dialogue, and Human-Robot Interaction.* 10th Inter. Symp. of Robotics Research, 2001.

[42] Gabillon A. and Letouzey L., *A View Based Access Control Model for SPARQL.* NSS'10, 2010. pp. 105-112.

[43] Gao S. *et al.*, *W3C XML Schema Definition Language (XSD) 1.1 Part 1: Structures*. W3C recomm., http://www.w3.org/TR/xmlschema11-1/ 2009.

[44] Garcia-Castro R. and Gomez-Perez A., *Interoperability Results for Semantic Web Technologies using OWL as the Interchange Language.* Journal of Web Semantics (JWS), 2010. 8(4):278-291.

[45] Grammel L. and Storey M.-A. D., *A Survey of Mashup Development Environments.* The Smart Internet, 2010. 137-151.

[46] Gruber. T., *Collective knowledge systems: Where the Social Web meets the Semantic Web.* Journal of Web Semantics (JWS), 2008. 6(1):4-13.

[47] Guillemin P. and Friess P., *The Internet of Things: Strategic Research Agenda.* CERP-IoT - Vision and Challenges for Realizing the Internet of Things, Ch. 3, pp. 41-42, 2010.

[48] Guo Y. *et al.*, *A Requirements Driven Framework for Benchmarking Semantic Web Knowledge Base Systems.* IEEE TKDE, 2007, 19(2): 297-309.

[49] Harrer A. *et al.*, *Visualizing wiki-supported knowledge building: co-evolution of individual and collective knowledge.* 4th Inter.Symp. on Wikis, 2008, p.19.

[50] Hartig O. *et al.*, *Executing SPARQL Queries over the Web of Linked Data.* Inter. Semantic Web Conference (ISWC'09), 2009. pp. 293-309.

[51] Hayes P., *RDF Semantics.* W3C Recommendation, http://www.w3.org/TR/rdf-mt/. 2004.

[52] Heath T. and Bizer C., *Linked Data: Evolving the Web into a Global Data Space.* Lectures on the Semantic Web: Theory and Technology, 2011.

[53] Heflin J., *Knowledge Representation on the Internet: Achieving Interoperability in a Dynamic, Distributed Environment.* PhD Thesis, University of Maryland, USA., 2000. AAAI/IAAI 2000:1074

[54] Heflin J., *An Introduction to the OWL Web Ontology Language.* http://www.cse.lehigh.edu/~heflin/IntroToOWL.pdf., 2007.

[55] Hoc J.M., *From Human–Machine Interaction to Human–Machine Cooperation.* Ergonomics, 2000. 43(7), pp.833-43.

[56] Ide N. and Veronis J., *Introduction to the Special Issue on Word Sense Disambiguation: The State of the Art.* Computational Linguistics, 1998. 24(1):1-40.

[57] Inagaki T., *Smart Collaboration between Humans and Machines based on Mutual Understanding.* Annual Reviews in Control, 2008. 32(2):253-61.

[58] Ishida R., *An Introduction to Multilingual Web Addresses.* International/articles/idn-and-iri/, http://www.w3.org/, 2008.

[59] James T., *Smart Factories.* Engineering & Technology, 2012. 7(6): 64 - 67

[60] Jiang J. and Conrath D., *Semantic Similarity based on Corpus Statistics and Lexical Taxonomy.* Inter. Conf. on Research in Compt. Linguistics, 1997.

[61] Joo J., *Adoption of Semantic Web from the perspective of technology innovation: A grounded theory approach.* International Journal of Human-Computer Studies, 2011. 69(3):139-154.

[62] Kilgarriff A., *Word Senses.* Word Sense Disambiguation: Algorithms and Applications, 2006. pp.29–46, Springer, New York, NY.

[63] Kleinberg J., *Authoritative Sources in a Hyperlinked Environment.* Journal of ACM, 1999. 46(5):604–632.

[64] Klyne G. and Carroll J., *Resource Description Framework (RDF): Concepts and Abstract Syntax.* . W3C Recommendation REC-rdf-concepts-20040210, 2004. http://www.w3.org/TR/rdf-concepts/.

[65] Krogstie J. *et al.*, *Integrating Semantic Web Technology, Web Services, and Workflow Modeling: Achieving System and Business Interoperability.* Intern. J. of Enterprise Information Systems (IJEIS), 2007. 3(1): 22-41.

[66] Latré B. *et al.*, *A survey on wireless body area networks.* JWS 2011, 17:1-18

[67] Lau R., *Towards a Web Services and Intelligent Agents-based Negotiation System for B2B ECommerce.* Electronic Commerce Research and Applications archive, 2007. 6(3):260-273

[68] Levy P., *Collective Intelligence: Mankind's Emerging World in Cyberspace.* Basic Books, 1999. pp. 312.

[69] Li H., Du X., and Tian X., *Towards Semantic Web Services Discovery with QoS Support using Specific Ontologies* 3rd SKG Conf. 2007. pp. 358 - 36.
[70] Li Y. *et al.*, *Term Disambiguation in Natural Language Query for XML.* In Proc. of Inter. FQAS Conf., 2006. LNAI 4027, pp. 133–146.
[71] Liang P. *et al.*, *From collective knowledge to intelligence: pre-requirements analysis of large and complex systems.* Proc. of Web2SE, 2010, 26-30, NY.
[72] Lianga T. *et al.*, *A semantic-expansion approach to personalized knowledge recommendation.* Decision Support Systems, 2008. 45(3):401–412.
[73] Liu J. *et al.*, *A Multiagent Evolutionary Algorithm for Combinatorial Optimization Problems.* IEEE Trans. on Systems, Man, & Cyber., 2010.
[74] Lorenzo G. D. et al., *Data integration in mashups* SIGMOD Record, 2009. 38:59-66.
[75] Maguitman A. *et al.*, *Algorithmic Detection of Semantic Similarity.* Proceedings of the Inter. WWW Conf., 2005. pp. 107-116.
[76] Manola F. and Miller E., *Resource Description Framework (RDF) Primer : Model and Syntax Specification.* W3C Recommendation, 2004. http://www.w3.org/TR/rdf-primer/.
[77] Marian A. *et al.*, *Adaptive Processing of Top-k Queries in XML.* ICDE, 2005.
[78] Marttila-Kontio M., *Visual Data Flow Programming Languages: Challenges and Opportunities.* Publications of the University of Eastern Finland, Dissertations in Forestry and Natural Sciences, 2011. N 30, pp. 101.
[79] McGuinness D.L. and Van Harmelen F., *OWL 2 Web - Ontology Language Document Overview.* W3C Proposed Edited Recommendation, 2012. http://www.w3.org/TR/owl2-overview/.
[80] Mihalcea R., *Knowledge-based Methods for WSD.* In Word Sense Disambiguation: Algorithms and Applications, 2006, 107–131, Springer.
[81] Miller G., *WordNet: An On-Line Lexical Database.* International Journal of Lexicography, 1990. 3(4).
[82] Ming M. *et al.*, *A Harmony Based Adaptive Ontology Mapping Approach.* In Proc. of International SWWS Conf., 2008. pp. 336-342.
[83] Mishra C. and Koudas N., *Interactive Query Refinement.* EDBT'09, 862-873.
[84] Mrissa M. *et al.*, *Context-based Semantic Mediation in Web Service Communities.* Weaving Services and People on the WWW, 2008. pp. 49-66.
[85] Nagarajan M. *et al.*, *Semantic Interoperability of Web Services - Challenges and Experiences.* IEEE ICWS'06 Conf., 2006. pp. 373-382.
[86] Navigli R. and Velardi P., *Learning Domain Ontologies from Document Warehouses and Dedicated Websites.* Compt. Linguistics, 2004. 30:151–179.
[87] Navigli R. and Velardi P., *An Analysis of Ontology-based Query Expansion Strategies.* Proc. of Inter. IJCAI'03 Conf., 2003.
[88] Pallis G. *et al.*, *Online Social Networks: Status and Trends.* New Directions in Web Data Management, 2011. (1): 213-234.
[89] Passant A. *et al.*, *Special issue on real-time and ubiquitous social semantics.* Semantic Web 2012. 3(2):113.
[90] Pereira F., *Technologies for Digital Multimedia Communications: An Evolution Analysis of MPEG Standards.* China Comm. Journal, 2006.
[91] Pilz A. and Paaß G., *From names to entities using thematic context distance.* CIKM 2011, 2011. pp. 857-866.
[92] Protege-OWL API, Stanford Center for Biomedical Informatics Research. http://protege.stanford.edu/plugins/owl/api/. Retrieved January 2012.
[93] Prudhommeaux E. and Seaborne A., *SPARQL Query Language for RDF.* W3C Recommendation, 2008. http://www.w3.org/TR/rdf-sparql-query/.
[94] Resnik M. *et al.*, *Scratch: Programming for All.* Communications of the ACM, 2009. 52(11):60-68, Doi:10.1145/1592761.1592779.
[95] Resnik P., *Using Information Content to Evaluate Semantic Similarity in a Taxonomy.* Proceedings of Inter. IJCAI Conf., 1995. Vol 1, pp. 448-453.
[96] Richardson R. and Smeaton A., *Using WordNet in a Knowledge-based approach to information retrieval.* BCS-IRSG Colloquium, 1995.
[97] Rocco D. *et al.*, *Domain-Specific Web Service Discovery with Service Class Descriptions.* IEEE Inter. Conf.on Web Services (ICWS), 2005, 481-488.
[98] Santucci G., *The Internet of Things: Between the Revolution of the Internet and the Metamorphosis of Objects.* CERP-IoT – Vision and Challenges for Realizing the Internet of Things, Ch. 1, pp. 11-24.
[99] Shekarpour S. *et al.*, *Keyword-Driven SPARQL Query Generation Leveraging Background Knowledge.* IEEE/ACM WI-IAT, 2011. (1)203-210.
[100] Shvaiko P. and Euzenat J., *Ten challenges for ontology matching.* Proc. of the OTM 2008 Confederated International Conferences, 2008, 1164–1182.
[101] Sigurbjörnsson B. and Van Zwol R., *Flickr tag recommendation based on collective knowledge.* Proc. of Inter. WWW'08 Conf., 2008. pp. 327-336.
[102] Smith J.R. *et al.*, *Revisiting Smart Dust with RFID Sensor Networks.* Seventh ACM HotNets-VII Workshop, 2008. Calgary, Alberta, Canada.
[103] Spiegler I., *Technology and knowledge: Bridging a "Generating" Gap.* Information & Management, 2003. 40(6), 533-539.
[104] Sundmaeker H. *et al.*, *Vision and Challenges for Realizing the Internet of Things.* CERP-IoT, European Commission - Info. Society and Media, 2010.
[105] Tagarelli A. and Greco S., *Semantic Clustering of XML Documents.* ACM Transactions on Information Systems, 2010. 28(1):3.
[106] Taylor A., Harper R., Swan L., Izadi S., Sellen A., and P. M., *Homes that Make Us Smart* Personal and Ubiquitous Computing, 2007. Vol. 11, Nb. 5.
[107] Tekli G. *et al.*, *XA2C: a framework for manipulating XML data* International IJWIS'11 Journal, 2011. 7(3):240-269.
[108] Tekli G. *et al.*, *A Visual Programming Language for XML manipulation* Journal of Visual Languages and Computations, 2013. 24(2): 110-135.
[109] Tekli J. *et al.*, *A Novel XML Document Structure Comparison Framework based-on Sub-tree Commonalities and Label Semantics.* Elsevier Journal of Web Semantics (JWS), 2012. 11: 14-40.
[110] Terzi E. *et al.*, *Knowledge Representation, Ontologies, and the Semantic Web.* Asia-Pacific Web Conference (APWeb 2003), 2003. pp. 382-387.
[111] Theobald M. *et al.*, *TopX: Efficient and Versatile Top-k Query Processing for Semistructured Data.* The VLDB Journal, 2008. 17:81–115.
[112] Tilak S. *et al.*, *A Taxonomy of Wireless Micro-Sensor Network Models.* ACM Mobile Computing and Communications Review (MC2R), 2002. 6(2).
[113] Tzitzikas Y. *et al.*, *On Storage Policies for Semantic Web Repositories That Support Versioning.* Semantic Web: Research and Apps, 2008. 5021:705-719
[114] Umer Q. and Mundy D., *Semantically Intelligent Semi-Automated Ontology Integration.* Proc. of the World Congress on Eng., 2012. London, U.K. .
[115] Van Den Heuvel W.J. & Maamar Z., *Intelligent Web Services Moving Toward a Framework to Compose.* Comm. of the ACM, 2003, 46(10):103-109.
[116] Vickrey D. *et al.*, *Word Sense Disambiguation for Machine Translation.* Proc. of the Inter. EMNLP Conf., 2005. pp. 771–778, Vancouver.
[117] Völker J. and Rudolph S., *Fostering Web Intelligence by Semi-automatic OWL Ontology Refinement.* Web Intelligence 2008. pp. 454-460.
[118] Völker J., Vrandecic D., and Sure Y., *Automatic Evaluation of Ontologies (AEON).* International Semantic Web Conference (ISWC'05), 2005. 716-731.
[119] Wamba S.F., *RFID-enabled Healthcare Applications, Issues, and Benefits: an Archival Analysis* Journal of Medical Systems, 2012. 36(6):3393-3398.
[120] Wang B. *et al.*, *An Exact Markov Process for Multihop Connectivity via Intervehicle Communication on Parallel Roads.* IEEE TWC, 2012.
[121] Wang T. *et al.*, *Link Energy Minimization for Wireless Sensor Networks.* Elsevier Ad Hoc Networks, 2012. 10(3):569-585.
[122] Yaworsky D., *Word-Sense Disambiguation Using Statistical Models of Roget's Categories Trained on Large Corpora.* Proceedings of the Intern. COLING Conf., 1992. Vol 2, pp. 454-460. Nantes.
[123] Ye S. and Chua T.S., *Automatically Integrating Heterogeneous Ontologies from Structured Web Pages.* Inter. IJSWIS Journal, 2007. 3(2):96-111.
[124] Zhang B. *et al.*, *Improving Web Search Results using Affinity Graph.* Proc. of Inter. ACM SIGIR Conf., 2005. pp. 504-511, NY.
[125] Zhang H. *et al.*, and Lakshminarayanan M., *Domain-Specific Web Services for Scientific Application Developpers.* GCE'10 Workshop, 2010. pp. 1-7.
[126] Zhang Y. *et al.*, *Energy Efficient Design for Body Sensor Nodes.* Journal of Low Power Electronics and Applications, 2011. 1(1):109-130, 2011.
[127] Zhao L. and Ichise R., *Mid-Ontology Learning from Linked Data.* Proc. of Inter. COLING'11 Conf., 2011. pp. 1098-1102, Montreal, Canada.
[128] Zheng Y. *et al.*, *GeoLife: A Collaborative Social Networking Service among User, Location and Trajectory.* IEEE Data Eng. Bull. , 2010. 33(2): 32-39.
[129] Zhu W. and Vij S., *Extending SOA Infrastucture for Semantic Interoperability.* DoDSOA & Semantic Technology Symp., 2011. Alion.
[130] Zins C., *Conceptual Approaches for Defining Data, Information, and Knowledge.* JASIST Journal, 2007. 58(4), 479-493.