Proceedings

IECON 2013 - 39th Annual Conference of the IEEE Industrial Electronics Society

Austria Center Vienna
Vienna, Austria
10 - 14 November, 2013

Sponsored by

The Institute of Electrical and Electronics Engineers (IEEE)
IEEE Industrial Electronics Society (IES)

Co-sponsored by

Austrian Institute of Technology (AIT), Austria
Vienna University of Technology (TU Vienna), Austria

IEEE Catalog Number: CFP13IEC-ART
ISBN: 978-1-4799-0224-8

Welcome Message of the IECON 2013 Chairs

The 39th Annual Conference of the IEEE Industrial Electronics Society (IECON2013), November 10-13, 2013 at Austria Centre Vienna, Austria, is focusing on industrial and manufacturing theory and applications of electronics, controls, communications, instrumentation and computational intelligence. The purpose of IECON 2013, following the footsteps of the previous editions, is to promote activities in various areas of industrial electronics by providing a forum for exchange of ideas, presentation of technical achievements and discussion of future directions. The IECON2013 brings together an international community of experts to discuss the state-of-the-art, new research results, perspectives of future developments, and innovative applications relevant to Power Electronics & Energy Conversion, Renewable Energy & Sustainable Development, Power Systems, Electronic System on Chip & Real Time Embedded Control, Signal and Image Processing & Computational Intelligence, Electrical Machines & Drives, Control Systems & Applications, Sensors, Actuators and Systems Integration, Mechatronics & Robotics, Factory Automation & Industrial Informatics, Information Processing and Communications, and related areas.

IECON is the Industrial Electronics Society's flagship conference. IECON 2013 is the biggest IECON ever, has the largest industry forum, exhibition, number of contributed papers, number of high-quality tutorials, student events, number of countries and side events. IECON 2013 had 1983 papers submitted from more than 80 countries. We will have 1350 papers for oral presentation at the conference after first going through a rigorous review process, then by second going through a quality control process conducted at the Program Committee (PC). The technical Program of IECON 2013 consists of two Keynote Talks, an Industrial Forum, a Student Forum, 311 technical sessions in 12 tracks and 66 special sessions, and 12 tutorials.

We would like to express our sincere appreciation to the IECON 2013 organizing committee members. The conference with such scale will not be possible without their strong commitment and efforts. Last but not the least, our sincere gratitude go to all the authors and invited speakers, for your participation and for providing the intellectual sharing on experiences. We hope you will enjoy Vienna experience while you find IECON2013 a fruitful, memorable conference technically and socially. Welcome and enjoy your stay in Vienna!

IECON 2013 Program Chairs
Peter Palensky, Luis Gomes, Mo-Yuen Chow

IECON 2013 General Chairs
Dietmar Dietrich, Ren C. Luo, John Y . Hung

IECON 2013 Committees

Honorary Chairs

Leopoldo G. Franquelo (Spain)
Kouhei Ohnishi (Japan)
Gerard-Andre Capolino (France)
Okyay Kaynak (Turkey)

General Chairs

Dietmar Dietrich (Austria)
John Y. Hung (USA)
Ren C. Luo (Taiwan)

Technical Program Chairs

Peter Palensky (Austria)
Luis Gomes (Portugal)
Mo-Yuen Chow (USA)

Publicity Chairs

Mariusz Malinowski (Poland)
Yoichi Hori (Japan)

Publication Chairs

Gerhard Zucker (Austria)
Andrés Meléndez Augusto Nogueiras (Spain)

Special Session Chairs

Gerhard P. Hancke (South Africa)
Dietmar Bruckner (Austria)
Friederich Kupzog (Austria)
Juan J. Rodriguez-Andina (Spain)

Tutorial Chairs

Heimo Zeilinger (Austria)
Carlo Cecati (Italy)
Seta Bogosyan (USA)

Finance Chairs

Jan Haase (Austria)
Terry Martin (USA)

Exhibit Chair

Georg Lauss (Austria)

Program Committee

Power Electronics & Energy Conversion

Chandan Chakraborty
Maria I. Valla
Babak Fahimi
Hao Ma
José Ignacio León Galván

Renewable Energy & Sustainable Development

Josep M. Guerrero
Marco Liserre
Wolfgang Hribernik

Power Systems

Le Xu
Ziang Zhang
Concettina Buccella

Electronic System on Chip & Real Time Embedded Control

Marcian Cirstea
Eric Monmasson
Marc Perron

Signal and Image Processing & Computational Intelligence

Milos Manic
Rainer Unland

Electrical Machines & Drives

Leila Parsa
Mario Pacas
Antonio Marques-Cardoso
Christian Kral

Control Systems & Applications

Xinghuo Yu
Jiming Chen
Hiroshi Fujimoto

Sensors, Actuators and Systems Integration

Antonio Luque Estepa
Aleksander Malinowski

Mechatronics & Robotics

Roberto Oboe
Kioshi Ohishi
Yousef Ibrahim
Makoto Iwasaki

Factory Automation & Industrial Informatics

Valeriy Vyatkin
Paulo Leitao

Information Processing and Communications

Thilo Sauter
Stamatis Karnouskos
Jose Fonseca

Electric and Plug-in Hybrid Electric Vehicles

Sheldon Williamson
Akshay Kumar Rathore
David Dorrell

Special Session Organizers

SS01-Matrix Converters

Marco Rivera Abarca
Jose Rodríguez
Patrick Wheeler
Haitham Abu-Rub

SS02-Power Management based on Advanced Identification and Classification Techniques

Thomas Bier
Djaffar Ould Abdeslam
Dirk Benyoucef
Jean Merckle

SS03-Induction heating systems

Óscar Lucía
Claudio Carretero

SS04-Control and Filtering For Distributed Networked Systems

Qing-Long Han
Josep M. Fuertes
Mo-Yuen Chow

SS06-Multiphase Variable Speed Drives

Emil Levi
Federico Barrero

SS07-Wind Energy Conversion Systems: Advanced Topologies and Control

Emil Levi
Mario J. Durán

SS08-Intelligent Real-time Automation and Control Systems

Thomas Strasser
Alois Zoitl
Antonio Valentini
Valeriy Vyatkin

SS09- Real-time Simulation and Hardware-in-the-Loop Validation Methods for Power and Energy Systems

Georg Lauss
Felix Lehfuß
Filip Andrén
Thomas Strasser

SS10-RFID Technology & Wireless Sensor Networks

Teresa Riesgo
Jorge Portilla
Jin-Shyan Lee
Antonio Torralba

SS11-Diagnostic of AC Machine Based Complex electromechanical systems

Humberto Henao
Shahin Hedayati Kia

SS12-Ambient intelligence of mobile robots or vehicle with human factors

Kang-Hyun Jo
Hiroshi Hashimoto
Burkhard Wuensche
Laurent Heutte

SS13-Recent applications of signal and image processing techniques and pattern recognition algorithms to condition monitoring of electrical machines and drives

Jose A Antonino-Daviu
Ioannis Tsoumas
Elias Strangas

SS14-Industrial Wireless Communication and its Applications

Johan Åkerberg
Mikael Gidlund

SS15-Network-based Control Systems and Applications

Josep M. Fuertes
Mo-Yuen Chow

SS16-Building Automation – Handling the Complexity

Jan Haase
Gerhard Zucker
Wolfgang Kastner
Yoseba Peña

SS17-Predictive Control for Power Converters and Drives

Sergio Vazquez
Jose Rodriguez
Leopoldo G. Franquelo
Hector Young

SS18-Compliant Robots

Yasutaka Fujimoto
Kiyoshi Ohishi
Naoki Oda

SS19-New Trends in Converter Topologies and Control Methods for Active Power Distribution Grids

Enrique Romero-Cadaval
Dmitri Vinnikov
Joao Martins
Marek Jasinski
Frede Blaabjerg

SS20-Lighting the Future

J. Marcos Alonso
Ricardo N. do Prado
Francisco Azcondo
Tiago B. Marchesan

SS21-Haptics for Human Support

Seiichiro Katsura
Kiyoshi Ohishi
Yasutaka Fujimoto

SS22-Network Control Systems for Interactive Power/Energy Networks

Sudip K. Mazumder
Mo-Yuen Chow
Josep M. Fuertes

SS23-Modular Multilevel Converters and other Multilevel Converter Topologies and Applications

Jose I. Leon
Leopoldo G. Franquelo
Samir Kouro
Marcelo Perez

SS24-Resilience and Security in Industrial Agents and Cyber-physical Systems

Paulo Leitão
Milos Manic
Armando Colombo

SS25-Smart Building Infrastructures for Integration of On-site Power Generation and Energy Storage

Giovanni Spagnuolo
Weidong Xiao

SS26-Biomimetics and Bionics Robotics

Maki K. Habib
Ju-Jang Lee
Keigo Watanabe
Fusaomi Nagata

SS27-Advanced Signal Processing Techniques for Power Systems Applications

Patrice Wira
Djaffar Ould Abdeslam

SS28-Advanced Motion Control for Mechatronic Systems

Hiroshi Fujimoto
Makoto Iwasaki
Roberto Oboe
Toshiaki Tsuji

SS29-Electric Traction Drives for Road Vehicles

Giuseppe Buja
Chandan Chakraborty
Ritesh Kumar Keshri

SS30-Cognitive Architectures and Multi-Agent Systems

Dietmar Bruckner
Friedrich Gelbard
Samer Schaat
Alexander Wendt

SS31-Trust in ICT Infrastructures for Smart Grids

Dominik Engel
Ulrich Hofmann

SS32-Advances in Energy Storage

Federico Baronti
Mo-Yuen Chow
Sheldon S. Williamson
Nihal Kularatna
Hubert Razik
Roberto Saletti
Walter Zamboni

SS33-Electronic System Level (ESL) Design and Virtual Prototyping (VP) for Industrial Electronics

Sumit Adhikari
Javier Moreno Molina

SS34-Engineering Tool Integration for Industrial Automation System Development (ETAS)

Dietmar Winkler
Richard Mordinyi
Leon Urbas
Vladimír Marík

SS35-V2X Communication Technology Status, Outlook and remaining Challenges

Alexander Paier
Christoph Mecklenbräuker

SS36-Processes and Tools for Mechatronical Engineering of Production Systems

Arndt Lüder
Stefan Biffl

SS37-Engineering Paradigms for Automated Facilities

Matthias Foehr
Tobias Jäger
Paulo Leitão

SS38-Photovoltaic Energy Conversion Systems

Samir Kouro
Mariusz Malinowski
Haitham Abu-Rub
Marcelo Perez
Bin Wu

SS41-Smart and Universal Grids

Wolfgang Gawlik
Georg Kienesberger
Thomas Leber
Alexander Wendt

SS42-High-performance power supplies

G. Buja
M.T. Outeiro
A. Carvalho
R. Visintini

SS43-Power Converters, Control, and Energy Management for Distributed Generation

Akshay K. Rathore
Herbert Iu
Dylan Lu

SS44-Power Electronics, Control, Motor Drives, and Energy Management in Electric and Fuel Cell Vehicles

Akshay K. Rathore
David Dorrell
Fei Gao

SS45-Aspects of Design and Manufacturing in Electrical Machine Design for Variable-Speed Drives and Generators in Automotive and Renewable Energy Applications

David Dorrell
Ke-Han Su

Jonathan Shek

SS46-Advanced Signal Processing Tools for Failures Detection and Diagnosis in Electric Machines and Drives

Mohamed Benbouzid
Demba Diallo

SS47-Industrial Agents

Paulo Leitão
Stamatis Karnouskos
Armando Colombo
Birgit Vogel-Heuser
Peter Göhner
Arndt Lüder

SS48-Advanced Control of Low Voltage Distribution Networks

M. Stifter
L. Ochoa
Benoit Bletterie

SS49-Emerging methods and technologies for Eco-Factories engineering and control

Claudio Palasciano
Paola Fantini
Gerrit Posselt
Rafal Cupek

SS50-Modeling and Simulation of Cyber-Physical Energy Systems

Edmund Widl
Sebastian Lehnhoff
M. Stifter

SS51-Intelligent information processing for the Smart Grid: innovative estimation, control and optimization methods

Gerasimos Rigatos
Pierluigi Siano
Nikolaos Zervos

SS52-Advanced Control Strategies for Wind Turbines Fault Ride-Through Capability Enhancement

Mohamed Benbouzid
Marwa Ezzat
Lennart Harnefors
S.M. Muyeen

SS53-Self-organising, robust Automation Systems

Joern Ploennigs
Dirk Pesch
Suzanne Lesecq
Antonello Monti

SS55-Special Session on Verification of Hardware Systems and Circuits

Florian Schupfer
Michael Rathmaier

SS56-High Power Factor Rectifiers

Hadi Y. Kanaan
Kamal Al-Haddad

SS57-Advanced Power Electronics for Power Quality Improvement in Distributed Generation Systems under Heavy Penetration of Renewable Energy Sources and Nonlinear Loads

Hadi Y. Kanaan
Kamal Al-Haddad

SS58-Current Status of Intelligent Spaces, Conversion of Robotics, Mechatronics, Control and Interfaces

Hideki Hashimoto
Peter Korondi
Géza Husi

SS59-Systems and devices for promoting energy efficiency in compressed air systems

Norma Anglani
Francesco Benzi
Carlo Cecati
Luc De Beul

SS60-Control Techniques for Efficient Management of Renewable Energy Micro-grids

Carlos Bordons
Luis Yebra

SS61-Renewable Energy Sources and their Integration to grid Power Supply

Akshay K. Rathore
Sanjib K. Panda

SS62-Demand Response integration in the Smart Grid

Sara Ghaemi
Christian Elbe

SS63-Photovoltaics: Characterization, Modeling and Simulation Methods

Stephan Abermann
Rita Ebner
Elisabeth Mrakotsky
Marcus Rennhofer

SS64-Energy and Information Technology

Peter Palensky
Hiroaki Nishi

SS65-Fault tolerant power converters for automotive applications

Arnaud Gaillard
Abdesslem Djerdir
Sheldon Williamson

SS67-Sensorless Control of Permanent Magnet Synchronous Machines

Manfred Schrödl

SS68-Human Support Technology on Human Factors

Kang-Hyun Jo
Hiroshi Hashimoto
Sho Yokota

SS69-Nonlinear Dynamics of Power Converters

Abdelali El Aroudi
Damian Giaouris

SS71-Health and Sustainable Technologies for Next Generation Home and Building Automation

Kim-Fung Tsang
Candy HY Tung
Gerhard Hancke

SS72-Advanced Controllers for High Performance AC Drives

Chandan Chakraborty
Carlo Cecati

SS73-Advanced Active Power Filters & Static VAR Compensators

Chandan Chakraborty
Kamal Al-Haddad

Tutorials

Z-Source Inverter: Basics, Modeling, Controlling, topology modifications and applications

Ellabban, Omar (Texas A&M University at Qatar)
Nov 11 Afternoon

Industry 4.0 - Utilizing Wearable & Mobile Systems for Improved Service Delivery

Markus Aleksy (ABB Corporate Research, Germany)
Nov 11 Afternoon

Tools, Services and Engineering methodologies for Robust, Adaptive, Self-organising and Cooperating Monitoring and Control Systems

Suzanne Lesecq (Campus, Grenoble, France)
Nov 11 Afternoon

Electric Vehicle Charging Integration in Distribution Grids

Johan Driesen (ESAT--ELECTA, Belgium)
Nov 12 Morning

PHM of fuel cell system – a state of the art

Daniel Hissel (FCLAB Research Federation (CNRS))
Nov 12 Morning

Industrial Ethernet – Technologies, Comparisons, Practical Considerations

Thilo Sauter (Vienna University of Technology)
Nov 12 Morning

Frequency Control and Inertia Response Schemes for the Future Power Networks

Francisco M. Gonzalez-Longatt (Coventry University)
Nov 12 Afternoon

Modern Design Process of Electric Motors

David A. Staton (Motor Design Ltd, United Kingdom)
Nov 12 Afternoon

Xilinx - Enabling New Product Innovations Across Markets with Zynq-7000 All Programmable SoC, Vivado HLS and IP Integrator

Olivier Tremois (Xilinx: Le Val Saint Quentin – Bat B)
Nov 12 Afternoon

Energy Harvesting from Motion: Fundamentals and Recent Advances

Andrew S. Holmes (Imperial College London)
Nov 13 Morning

Solid-State Transformer Concepts in Traction and Smart Grid Applications

Johann Walter Kolar (Power Electronics - ETH Zurich)
Nov 13 Morning

New Emerging Technologies in Motion Control Systems

Toshiaki Tsuji (University of Saitama, Japan)
Nov 13 Morning

Extracting and Integrating Structured Information from Web Databases Using Rule-Based Semantic Annotations

Benjamin Dönz and Dietmar Bruckner
Institute of Computer Technology
University of Technology
Vienna, Austria
{doenz, bruckner}@ict.tuwien.ac.at

***Abstract*—The Semantic Web is envisioned to be the next evolutionary step of the Word Wide Web. It will allow programs to access information and interact with services similar to the way people use the Internet today. The required standards have already been developed, and also large databases containing factual information have been published in the appropriate format. However, more casual, every-day information like that published by used car dealerships, resellers, and other data contained in web databases is not available yet. We believe that bringing this type of information to the Semantic Web could greatly increase its use and help bring it into people's everyday lives. In this paper we present a model for such web databases and show how to integrate these sites into the landscape of the Semantic Web by transparently converting and forwarding queries to conventional web databases.**

***Keywords*—Semantic Web; Semantic Annotation; Semantic Information Extraction; On-Demand Information Extraction; Information Integration;**

I. INTRODUCTION

The Word Wide Web offers an immense amount of information on a plethora of topics. Even if state of the art information retrieval technology helps users to find relevant documents with astonishing precision, considering the size of the Internet, thoroughly researching a specific topic still remains a tedious task. As an example, let us consider a user, who intends to buy a car. Let us also assume that he or she is not focused on a specific model, but does prefer specific brands and is looking for a car that meets his requirements under the constraints of a given budget. An example could be a budget of 15.000€ for a Volkswagen or Kia which should have 4 doors (so the kids can get in and out more easily), either the mileage should be under 50.000 or it should be less than 4 years old, and it should have decent fuel efficiency. Besides the hard facts, the user might also be interested in other people's experiences or recommendations and tests in car magazines. It can be assumed, that all the required information for this research task can be found on the Internet. However, it will be distributed over several sites, such as used car dealerships' databases, manufacturer homepages, discussion forums, and others.

Confronted with such a task, current search engines can only provide little help. The first problem is that search engines have a hard time indexing information that is "hidden" behind search forms, such as those typically found on sites like used car dealerships. This so called "Deep Web" or "Hidden Web" is estimated to contain 500 times more information than the common web [1]. Only in 2008 employees of Google published their approach for surfacing this information and including it in their search index [2]. By first estimating the domain of the database and then submitting relevant keywords, they managed to access portions of the content, with a coverage as low as 20% in some cases. So the user cannot really use the search engine to find the thing he is actually interested in (the cars), but only to find web databases that may hold this information. One then has to navigate there and use whatever interface that site provides to access the actual data. Since each site offers a different interface, one also has to spend some time to get to know how to use the site before one can actually extract information from it. After submitting the query to several sites one has to aggregate the results in an integrated list containing the relevant facts and the page where the offer was found. This task will be referred to vertical information integration for the remainder of this paper.

The second problem a user might have is that a typical ad might not contain all the necessary facts. It might state the age, mileage and make of a car, but not the fuel efficiency or the number of doors. So the user will also have to cross-check with the manufacturer's homepage or other sites to figure out if an offer actually meets the requirements. If the information can be found, the user can add it to the compiled list together with recommendations from car magazines and customer experiences. This task does not produce new records, but extends existing ones, so in contrast to vertical information integration, this will be referred to as horizontal information integration for this paper.

So after having done all this work – looking for sites, cross checking with other sites and compiling the final list – the user can do what he actually intended to do, which is to select the top few offers and go through them in detail to form a decision. For some similar tasks, specialized sites have emerged such as sites for finding the lowest price for a given product (e.g.

pricegrabber.com) or for finding plane tickets (e.g. check-felix.com). However, for most topics there is no such service, and interested users will have to perform the steps listed in the example themselves.

The Semantic Web, as envisioned by Tim Berners-Lee [3], could provide the means for such tasks by default. The community has already developed most necessary standards and languages to create, populate and access the web of data, following the semantic web stack [4] roadmap. Using SPARQL, a query can be formulated manually or with tool support that returns an integrated list of exactly the facts our example user had to compile manually. However, the problem here is that even if there are billions of facts in semantic databases included in the linked data project alone [5], every-day information contained in arbitrary databases such as those of car dealerships are not available yet. It would be necessary to publish the information in the appropriate format (RDF) to allow programs to interpret the data, and as long as the extra effort doesn't yield a return of investment, the site owners will not take the trouble. On the other hand, people will not use the Semantic Web and thereby make it investible, as long as the information they are looking for is not available. So due to this chicken and egg problem [6], mainly research projects have contributed to the Semantic Web by publishing their results or converting large databases such as Wikipedia (http://dbpedia.org).

We believe that if an efficient method can be found to bring existing information from the Deep Web to the Semantic Web, not only would it make data that is currently not really accessible by search engines more apparent and useable, it could also help bootstrap the Semantic Web. We propose to do so, not by converting the data directly to RDF, but by annotating existing web databases in a way that allows a software agent to access and use it in the same way a human user would.

II. Related Work

Our proposal can be viewed from three perspectives that we want discuss separately first, before presenting our concept: semantic annotation, information extraction and information integration.

A. Semantic Annotations

Annotations have shown to be an effective way to add machine usable information to documents that are mainly intended for human use. The main idea is to link a part of a document to a concept of known semantics, giving the annotated segment the semantics of that concept. The reference to the document segment can be either done implicitly (embedded), by including the reference at the exact place in the document that should be annotated, or explicitly by referencing the segment of the document using a pointer of some sorts, e.g. the tag-path in an html document. The concept that is referred to can also be either implicit, by using a pre-defined symbol, e.g. a special tag name or attribute, or explicit, e.g. referring to a concept using some sort of pointer to an external resource.

The first commonly used semantic annotations on the World Wide Web were Microformats [7]. The basic concept is to add special class-attributes to tags in the document that contain the semantic information. The names of these classes are predefined and several vocabularies, e.g. hCard, hRecipie or hReview have been published on microformats.org. Following the definitions above, Microformats are embedded, implicit annotations, and both the publisher and consumer have to use a predefined vocabulary, which limits the information that can be published this way. If a site would choose to add proprietary terms in order to publish additional information, no one can make use of them as long as they are not added to the common vocabulary.

The W3C recommendation RDFa (Resource Description Framework – in – Annotations) is a more open annotation scheme: it is also embedded, but uses URI references instead of a pre-defined vocabulary. This way a publisher can add new concepts and reference them. If someone stumbles on such a proprietary reference, and if the publisher followed the best-practices [8], the meaning of the concept can be looked up by dereferencing the URI and either reading the description or follow links to other concepts (see also Section C).

Embedded annotations have to be included by the publisher. To allow including web sites without having to rely on contributions by their publishers, who might not be willing to participate without any foreseeable return of investment, as discussed in the introduction, external annotations should be favored. In this case, a reference to the segment of the document is needed, which can be given using a language like XPath or XPointer as was done, for example, in the Annotea project [9] (another approach will be discussed in Section B).

Besides the annotation style, the method used to create the annotations might also be relevant. It is of course possible to create annotations manually or with a tool that helps a human user to create the annotations more easily, but some publications argue that this approach is not scalable [10], and autonomous annotation is therefore necessary. Regarding this issue, we feel that the primary objective is to obtain accurate information rather than more, but possibly unreliable data. We too would prefer a fully automatic annotator and encourage work in this field, but actually, as soon as that is possible we would not need annotations nor the Semantic Web anyway – since if the page can be interpreted automatically and allow programs to use them, the Semantic Web is obsolete. Favoring precision over scalability, we currently propose to use assisted semi-automatic annotation methods that can produce reliable results. For this paper however, we will not focus on the creation of the annotations, but proceed on the assumption that it is already provided.

B. Information Extraction

In order to answer a query such as the one presented in the introduction, two possible approaches regarding the temporal aspect of the information extraction are be possible: the first is to extract all data beforehand and store it in a combined database which is then used to answer any query that may come up. This approach is similar to the way search engines work, where crawlers constantly update an index that is then used to answer queries. Examples for this approach include KnowItAll [11] or dbPedia [12]. The latter leverages statistics

about page-updates to determine topics that need to be converted. However, since this kind of information is not available on most sites, such an approach is not possible for most web databases. Other current methods for extracting data from arbitrary databases rely on submitting a multitude of queries with overlapping results that might not even by able to extract all available information while producing high traffic to the site [2]. We therefore follow the conclusion stated in [13] and propose to implement a discover-and-forward model, where a user's query is forwarded to a site to obtain results on-demand.

From the viewpoint of information extraction, the semantic annotations that allow a program to interact with the site, i.e. forward the query and obtain the results, can be regarded a wrapper in the sense of the definition in [14]. The main task of a wrapper, equal to that of the semantic annotation, is to create a link between segments on a web page and a known concept. The main problem hereby lies in referencing the document segments. If the page is static, this is simple and can be done by using the tag-path, but in the case of web databases, the result pages may differ depending on the query and the records. Current approaches therefore try to reference the segment by describing some properties it may have. These properties will be referred to as features in this paper, and examples may include:

- Tag paths or parts of it, often evaluated using regular expressions
- Adjacency, e.g. the tag next to the label "name:"
- Data format, e.g. number sequences separated with spaces or hyphens are interpreted as phone numbers
- Visual aspects in the layout, e.g. left or right of/below/above

More complex approaches also exploit several features to create more robust solutions, e.g. [10].

C. Information Integration

The data structure of the semantic web corresponds to that of the World Wide Web: a directed graph. While the traditional web uses links pointing from one document to another, the semantic web uses triples that resemble a qualified link (i.e. it contains a semantic relevance) between a subject and an object. Since this graph can be defined solely by listing the facts that correspond to the edges of the graph, it is trivial to integrate information, since every new statement is just added as a new edge to the graph. Real conceptual integration however also requires that the URIs used to reference a certain concept must be equally used by all databases that should be integrated. Since the vocabulary is not fixed, but can be extended and defined by anyone, this will not necessarily be the case. Ontology matching aims at figuring out alignments between these concepts, and is an active research area [15]. However, since the ontologies on the Semantic Web are part of the same network (the Internet), relations between concepts can also be published as facts, which makes ontology matching obsolete. By using constructs such as "same as", semantic bridges [16] also called RDF links [5] can be created that join different ontologies. Currently the largest project of this type is the Linked Open Data project, which according to [5] contained 4.7 billion triples in 2008 and has surely grown since.

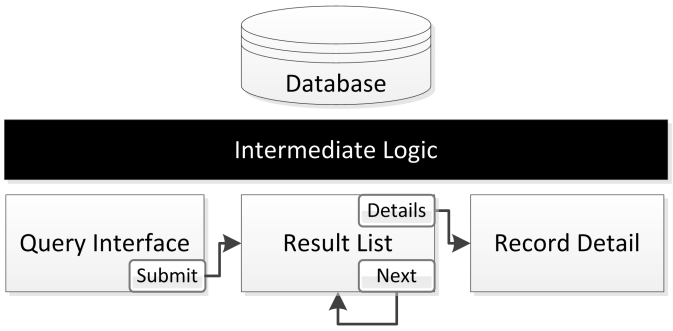

Fig. 1 Architectural and Navigational View

Regarding the context of this paper, both types of information integration (vertical and horizontal) can be accomplished by leveraging this as will be shown in the next Chapter.

III. Concept

In this Chapter we will introduce a model for web databases and show how the user's query must be transformed to be compatible with the model. From that we then infer the annotations needed to allow extracting data from a site and how information from heterogeneous web databases can be integrated.

A. A Web Database Model

To create a model, we analyzed several websites in example domains of used car dealerships and realtors available in Austria and Germany. The final model is simplistic, but fits almost all the sites we encountered and is also easily extendible if needed.

As shown in Fig. 1, a web database following this model consists of a database that holds the records we are interested in, some unknown intermediate logic and three types of views: a query submission form, a result list and a result detail view. Regarding navigation, a user starts with the query form, enters parameters to get a result list which may only contain a limited amount of results, but can offer a "get more results" function, and it may also offer links to detail pages that contain additional values of a record.

The database in our model consists of exactly one table T that holds records $T = \{R_1, R_2, R_3, \ldots\}$. All records are instances of the same `rdf:type` t_T, and each record R_i contains exactly $|R_i|=N$ distinct values $R_i=\{v_{i1},v_{i2},v_{i3},\ldots v_{iN}\}$. The semantic concept associated with a value of a given index is the same for all records and corresponds to an `rdf:property` p_x, so that $p_x(R_i)=v_{ix}$. This allows representing the same information as a row in the table and also as an RDF graph, where each R_i can be defined as a blank node that is an instance of type t_T and each value contributes a triple to the RDF graph in the form R_i p_x v_{ix} as shown in Fig. 2. It should be noted that each record is resembled by a separate graph and no relations between the records exist in this model.

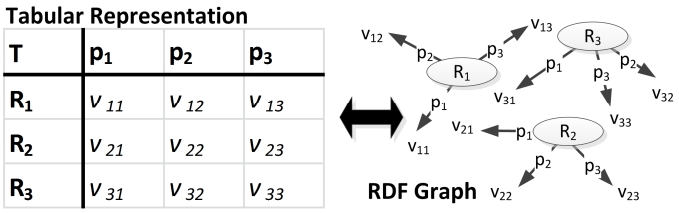

Fig. 2 Data Model

The query page P_Q consists of a set of fields for query parameters $P_Q=\{q_1,q_2,q_3,\ldots\}$, and is associated with a logical operation that imposes restrictions on the records that are returned. For our model, we will consider only conjunctions of binary operators of the form $f_Q(p,q)$ that compare a property or an instance to a value given as an operand in a field on the query page. As a practical restriction we will assume that if q is an empty value, $f_Q(p,q)$ returns true and does not restrict the results. This behavior also reflects our experiences with the investigated databases. Returning to the initial example, these operations include filters like mileage<50000 to be expressed as lessThan(mileage,50000) or doors=4 as equal(doors,4). The result T_R of a query is the restricted set of records after applying the conjunction of all available filters to each record $T_R(Q) = \{R_i | R_i \in T, f_{Q1}(p_1(R_i),q_1) \wedge f_{Q2}(p_2(R_i),q_2) \wedge f_{Q3}(p_3(R_i),q_3) \wedge \ldots\}$. The result list presents some values v_{ix} from any of these records, whereas the detail view contains only values of one specific record. Fig. 3 shows these relationships between the values in the database and the views.

B. *Query Transformation*

As stated in Chapter II, we propose to use a discover-and-forward model: the initial input is a user's query that is then forwarded to the sites. SPARQL is the designated query language of the Semantic Web, so the input query will be formulated in this language. For this paper we will only discuss how to integrate `SELECT` style queries: these consist of patterns that must be matched by the resulting data and an optional additional logical filter. A set of patterns corresponds to a conjunctive condition, and the keyword `UNION` allows to create disjunctions of such sets. Since the web database model only allows conjunctive queries, each set must be queried

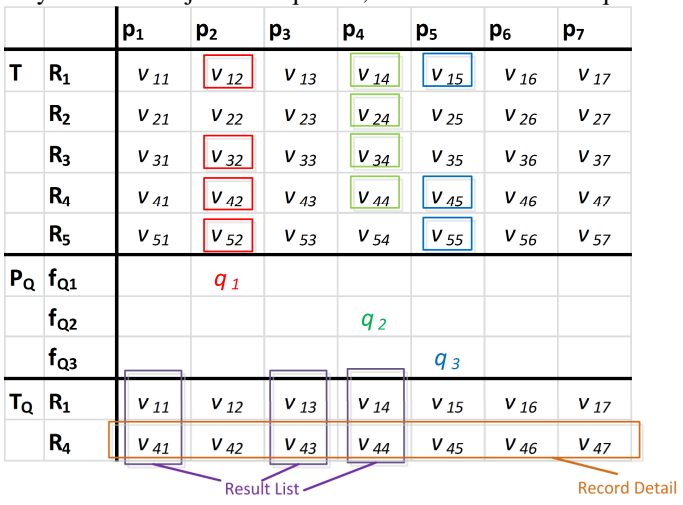

Fig. 3 Query Operation

separately and then joined in the end, which corresponds to vertical integration. The optional keyword `FILTER` allows posing further constraints on the results, thereby restricting the solution by providing a conditional expression consisting of conjunctions or disjunctions of logical operations. This filter could also be applied after extracting all the data that matches the pattern, but if the query interface supports these expressions, they should be submitted as part of the query, so only relevant records are accessed. Since the expression given as the filter can be an arbitrary combination of conjunctions and disjunctions, and our model for web databases only supports conjunctive queries the expression has to be transformed into its disjunctive normal form, and then split into an equivalent union of sub queries which then only contain conjunctions in their filter expressions.

The query interface in our model also only allows binary operations, with one operand being a value of the record and the second operand being a value that can be provided in the interface. If we examine the filter-section first, SPARQL supports unary, binary, and one trinary operator. The unary operators such as `isBlank(A)` or `isLiteral(A)` can simply be represented as binary operators with a second irrelevant operand. The only trinary operator is `REGEX(STRING,PATTERN,FLAGS)`. This function could be represented as a binary function by concatenating the parameters `PATTERN` and `FLAGS` into one parameter. However, since the parameter `FLAGS` is optional, and since none of the sites we came across actually allowed to submit regular expressions, the only practical application will be a regular expression for filters of the form property-contains-substring, so the flag will be omitted. Regarding the operands, the model of the query page only supports operations where one operand is a value of the record, and the second one is a given as a parameter on the page. Therefore only operands of the query can be used that follow this pattern, i.e. operations that contain one variable and one constant. Additionally, the variable that is used has to be associated with a property, i.e. there must be a statement of the form object-property-variable in the pattern part of the query. All others have to be omitted for the extraction process and can only be evaluated afterwards.

To convert the patterns to conjunctions of binary operators, we first make the following observations: the data model described in Section A is resembled by a separate graph for each record, and each graph contains exactly one entity, represented as a blank node, and properties containing literal values are assigned to that entity. Since the pattern in the graph must match the pattern in the query, only queries that follow the same pattern will return any results. Query patterns in SPARQL are a list of subject-predicate-object triples, where the subject and predicate must be either a variable or URI, and the object can be a variable, URI or literal value. Since the records are represented as blank nodes and cannot be referenced in the query, only query patterns of the form variable-predicate-object are relevant. If the query contains patterns with a fixed reference as a subject, the query process must be aborted.

As a second observation, since each record is represented as a separate graph, and since the query only evaluates conditions

record by record, the whole pattern set can only be matched against a single record. And since there is only one entity involved, as already noted in the previous abstract, all variables used as a subject must reference the same entity and can be replaced with the same variable representing the record.

The result is a query that consists only of statements of the form recordvariable-predicate-object. If the predicate or object are variables, then the statement cannot be included in the filter and only be evaluated after the information extraction process is completed. The only type of statement relevant for the extraction process is therefore a statement of the form recordvariable-property-literal. And these can be represented with a binary equality operator.

After this transformation, the original query Q is represented as a disjunction of sub queries $Q=(Q_{S1} \vee Q_{S2} \vee Q_{S3} \vee \ldots)$, each of which only contains conjunctions, either derived from the patterns or the filter as described above $QS1=(f_{S1}(p_1,c_1) \wedge f_{S2}(p_2,c_2) \wedge f_{S3}(p_3,c_3) \wedge \ldots)$. Fig. 4 shows the SPARQL version of the main part of the query used in the introductorily example, and the resulting representation. To include possibly omitted parts of the query, it is suggested to collect the results in an intermediate file or database, and execute the original query after the extraction process has been completed.

C. Annotations

To allow a program to interact with the web database, we propose to use annotations as they have been defined in Chapter II, and we propose to use external annotations rather than embedded ones, so the sites themselves do not have to be changed. As defined in Chapter II, an annotation links a document segment with a semantic concept. Since the annotations are external, a set of concepts that allow interacting with the site and extracting the data is needed, and also a solution that allows referencing applicable document segments. We propose to base these references on features, i.e. an element is identified because it has a certain combination of features, which can include a tag path, some adjacent value or a specific attribute. Annotations of this form can be expressed very naturally as rules: if a document segment has a distinctive set of features, then assert a certain annotation. The annotation itself would then consist of an URI for the property and one for the concept as shown in Fig. 5.

SPARQL Query Q

```
SELECT ?brand ?model ?mileage ?age ?price
WHERE {
{[car:brand ?brand, "Volkswagen";
  car:model ?model;
  car:mileage ?mileage;
  car:age ?age;
  car:doors ?doors;
  offer:price ?price;]
FILTER ((?mileage<50000 || ?age<4)
       && ?doors=4 && ?price<15000)}
UNION
{[car:brand ?brand, "Kia";
  car:model ?model;
  car:mileage ?mileage;
  car:age ?age;
  car:doors ?doors;
  offer:price ?price;]
FILTER ((?mileage<50000 || ?age<4)
       && ?doors=4 && ?price<15000)}
}
```

Transformation Result

Q_{S1}: equals(car:brand,"Volkswagen") ∧ lessThan(car:mileage,50000) ∧ equals(car:doors,4) ∧ lessThan(offer:price,15000)

Q_{S2}: equals(car:brand,"Volkswagen") ∧ lessThan(car:age,4) ∧ equals(car:doors,4) ∧ lessThan(offer:price,15000)

Q_{S3}: equals(car:brand,"Kia") ∧ lessThan(car:mileage,50000) ∧ equals(car:doors,4) ∧ lessThan(offer:price,15000)

Q_{S4}: equals(car:brand,"Kia") ∧ lessThan(car:age,4) ∧ equals(car:doors,4) ∧ lessThan(offer:price,15000)

Fig. 4. Query Transformation

The list of features can include those already referenced in Chapter II, e.g. tag-path, adjacency, etc., but should be extendible, since we believe that only practical experimentation will show which features perform best. A two-stage extraction process – one that identifies the features, and a second that applies the annotation rules – could be an approach that allows adding new discoverable features over time, while still keeping existing annotation rules working.

The annotations necessary for the extraction process correspond to the elements of the model. We propose the following concepts as an upper ontology that allows implementing a concrete annotation vocabulary compatible to the web database model:

- The class `PageElement` is used to represent an individual element on a page that can be annotated.
- The property `hasFeature` allows assigning features to an element. Examples for sub properties could include `ReferencePath` or `isAdjacentTo`.
- The property `executeFunctionWith` is used to define an event that triggers a certain function, e.g. “click”. Sub properties could be `submitQueryWith` or `getNextWith`.
- `restrictsProperty` is the base-property for defining restrictions that the value of associated `PageElement` imposes, such as `lessThan` or `equalTo`. The property it restricts is given as the object.
- The property `containsProperty` allows assigning the value of the PageElement to a concept given as the object.
- The property `belongsToRecord` allows assigning a `PageElement` to a record.

An example for annotation rules and how they apply to an example page is shown in Fig. 5. Our previous proposal for an annotation vocabulary [17] can be seen as an application of this more abstract upper ontology.

D. Information Integration

Since our model, and also most real life web databases, do not support disjunctive queries, it is necessary to aggregate the results of the various sub queries. This can of course also be done when submitting the query to different sites to vertically integrate the results. However, to conceptually integrate the data, it is necessary that the concepts associated with the values in the individual databases and in the query are used in a consistent way. So if p_{1x} is the concept associated with the values with index x in the first web database, and p_{2y} is the concept associated with the values with index y in the second

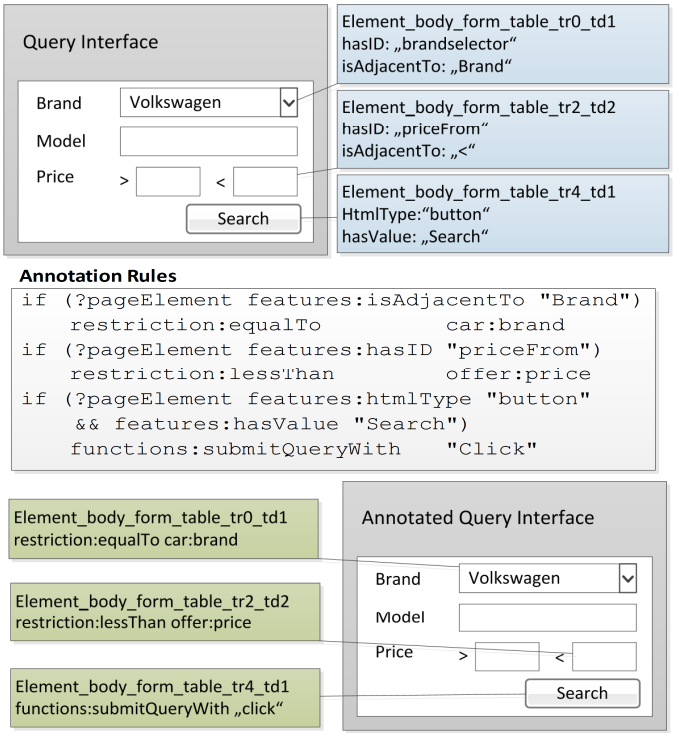

Fig. 5. Annotation Example

database, and both have the same meaning, they should be associated with the same concept – hence $p_{1x}=p_{2y}$. However, since the Semantic Web allows anyone to define any vocabulary, it cannot be assumed that two authors use the same reference for the same concept. A semantic bridge in the form of a chain of `owl:sameas` statements however would allow linking these references, e.g. $p_{1x}\leftrightarrow\ldots\leftrightarrow p_{2y}$. This is also the concept used in the Linked Open Data project that allows linking concepts in ontologies [5]. The best chances for consistent vertical integration of databases can therefore be achieved by referring to concepts that have many semantic bridges to other ontologies.

In our model, each record is represented as a blank node, so horizontal integration over heterogeneous databases is not possible directly, because there is no URI for a record that allows another record to reference it. The new OWL 2 recommendation introduces the language feature `owl:key` that allows assigning a set of properties to a class, which identify a unique individual. For books this could be the ISBN number, for car models it could be brand, model and construction year. To allow horizontal integration, the definition of keys and the existence of these properties in the database allow a semantic reasoner to integrate the properties of the different blank nodes into one entity.

IV. Conclusion

In this paper we presented a model for web databases that is simple, but still applies to many existing real world implementations. We devised a method for transforming SPARQL queries, so they can be forwarded to query interfaces that correspond to this model and suggested to use external rule-based semantic annotations that allow a program to interact with these sites. The annotations require a method for referencing document segments in order to link them to concepts that are then used to promote the extraction process. We presented both a corresponding upper ontology that allows developing compatible annotation vocabularies, as well as a feature based referencing mechanism to achieve that.

Coming back to our initial example, the methods proposed in this paper allow a user to create a SPARQL query manually or with tool support that can be used to retrieve the required data as if it were already available on the Semantic Web. We also showed how the proper use of annotations and readily available semantic constructs such as semantic links and keys allow integrating heterogeneous databases. The result presented to the user can therefore include information from several car dealerships, facts from manufacturer's sites and possibly also customer feedback.

References

[1] M. Bergmann, "The Deep Web: Surfacing HiddenValue." *Journal of Electronic Publishing*, Vol. 7, Issue 1. Retrieved April 30, 2013 from http://hdl.handle.net/2027/spo.3336451.0007.104, 2001

[2] J. Madhayan et al, "Google's Deep-Web Crawl." *Proceedings of the VLDB Endowment*, Vol. 1, Issue 2, pp. 1241-1252, 2008

[3] T. Berners-Lee, *Weaving the Web*. HarperCollins Publishers, pp. 177-198, 1999

[4] T. Berners-Lee, Talk on Semantic Web Architecture. Retrieved April 30, 2013 from http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html, 2000

[5] C. Bizer et al, "Linked Data – The Story So Far". *International Journal on Semantic Web and Information Systems*, Vol. 5, Issue 3, pp. 1-22, 2009

[6] J. Hendler, "Web 3.0: Chicken Farms on the Semantic Web." *Computer*, Vol. 41, Issue 1, pp. 106-108, 2008

[7] R. Khare, "Microformats: The Next (Small) Thing on the Semantic Web?" *IEEE Internet Computing*, Vol. 10, No. 1, pp. 68-75, 2006

[8] T. Berners-Lee, "Linked Data – Design Issues", Retrieved April 30, 2013 from www.w3.org/DesignIssues/LinkedData.html, 2009

[9] J. Kahan, M.-R. Koivunen, E. Prud'Hommeaux, and R.R. Swick, "Annotea: an Open RDF Infrastructure for Shared Web Annotations", *Computer Networks*, Vol 39, No 5, pp. 589-608, 2002

[10] Yiyao Lu, Hai He, Hongkun Zhao, Weiyi Meng, Clement Yu, "Annotating Search Results from Web Databases", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 25, No. 3, pp. 514-527, 2013

[11] O. Etzioni, M. Cafarella, D. Downey, S. Kok, A.-M. Popescu, T. Shaked, S. Soderland, Daniel S. Weld, A. Yates, "Web-Scale Information Extraction in KnowItAll", *Proceedings of the 13th International Conference on the World Wide Web*, pp. 100-110, 2004.

[12] F. Wu, D.-S. Weld, "Autonomously Semantifying Wikipedia", *Proceedings of the 16th ACM Conference on Information Knowledge Management (CIKM'07)*, pp. 41-50, 2007

[13] B. He, M. Patel, Z. Zhang, K. Chang, "Accessing the Deep Web: A Survey", *Communications of the ACM*, Vol. 5, pp. 94-101, 2007

[14] A. Laender, A. Berthier, A. da Silva, J. Teixeira, „A Brief Survey of Web Data Extraction Tools", *SIGMOD Record*, Vol. 31, No. 2, pp. 84-93, 2002

[15] P. Shvaiko and J. Euzenat, "Ten Challenges for Ontology Matching." *Lecture Notes in Computer Science*, Vol. 5332, pp. 1164-1182, 2008

[16] A. Maedche et al, "MAFRA – A Mapping FRAmework for Distributed Ontologies." *Lecture Notes in Computer Science*, Vol. 2473, pp. 69-75, 2002

[17] B. Dönz, D. Bruckner, "External Semantic Annotation of Web-Databases", 2012 *IEEE International Symposium on Industrial Electronics*, pp. 841-845, 2012