Journal of Web Engineering, Vol. 11, No.4 (2012) 317-336

THE DESIGN OF E-SPERANTO – A COMPUTER LANGUAGE FOR RECORDING MULTILINGUAL TEXTS ON THE WEB

GREGA JAKUS

University of Ljubljana, Slovenia
grega.jakus@fe.uni-lj.si

JAKA SODNIK

University of Ljubljana, Slovenia
jaka.sodnik@fe.uni-lj.si

SAŠO TOMAŽIČ

University of Ljubljana, Slovenia
saso.tomazic@fe.uni-lj.si

Received July 20, 2011
Revised June 22, 2012

The present paper describes the design of E-speranto, a formal computer language for recording multilingual texts on the Web. The vocabulary and grammar of E-speranto are based on the international auxiliary language Esperanto, while its syntax is based on XML (eXtensible Markup Language). The latter is one of the key features of E-speranto, as it enables a natural integration of E-speranto documents into web pages. When a user visits such a web page, its content is interpreted and displayed in the user's preferred language. Due to the fact that E-speranto is a formal language, it is much easier for computers to comprehend documents created in this language than to comprehend texts written in natural languages. The documents in E-speranto can be created directly with the aid of tools designed especially for this purpose. For a practical application of E-speranto, each linguistic group merely needs to develop the interpreter of E-speranto for their own language. We designed a proof-of-concept implementation of the multilingual Web based on E-speranto. The testing confirmed the applicability of the concept and indicated the guidelines for further development.

Key words: E-speranto, multilingual Web, interpretation, XML
Communicated by: D. Schwabe & P. Fraternali

1 Introduction

Since its introduction in the early 1990s, the World Wide Web has been continuously evolving. If it once mostly served as the infrastructure for the exchange of scientific data, it is nowadays also indispensable for the business world, media and social networks. In the two decades of its existence, the Web also expanded globally and gained users from virtually every corner of the world.

In the early days, the access to the information on the Web was limited by the so-called *digital divide* separating the users with internet access from those without it. Nowadays, with almost one quarter of the world population using the internet, an even greater challenge is to overcome the so-called *language divide*. The information on most web pages is, namely, not readily available to the majority of potential users, because they do not understand the language of the page. Despite the fact that almost half of all internet users come from Asia [1], the majority of web pages are still written only in English.

Multilingualism on the Web is most often reflected in the fact that some of the web pages are translated into languages that – according to the owner's conviction – cover a large part of the target audience. The users that do not understand these languages can use one of the online translators, such as Babel Fish [2], Google Translate [3], PROMT [4] or SYSTRAN [5]. The drawbacks of these tools are the facts that they are optimized for responsiveness and thus do not provide the best translation quality. In addition, they often do not enable the translation of longer texts and they support only a limited set of languages. Despite these disadvantages, online translators still enable the users to grasp the basic information on the web page in a language they understand.

When automatically translating between natural languages, two issues are particularly challenging. The first issue is connected with the computer's comprehension of a natural language. The latter is complicated mainly due to the ambiguities and inconsistencies that are present in natural languages. The other issue is a scalability problem which is especially evident in systems translating between a large number of languages. In order to provide the translation between n languages, $n(n-1)$ translators are required. This actually means that 47,727,372 translators are needed in order to translate between the 6909 languages that are spoken in the world today [6].

The typical approach to solve the above-presented scalability problem is a two-phase translation that uses an intermediate language or interlingua [7]. As each linguistic group requires only two modules (i.e., one performing the transformation into an interlingua and one that does the opposite), the interlingua-based approach reduces the necessary number of modules to $2n$. However, translation via an interlingua does not solve the problem of the computer's comprehension of a natural language. As this is problematical only when translating from a natural language into an interlingua and not in the reverse process, one of the possible solutions is the introduction of a formal computer language for recording multilingual documents. Such a language would enable the author to create documents using dedicated tools and at the same time reduce the need for automatic translation from a natural language. E-speranto was designed specially for this purpose. When translators into E-speranto of sufficient quality are developed, E-speranto will also function as an interlingua in multilingual translation (Fig. 1).

As the document is created in E-speranto and is only displayed in the target language, this process is labelled as *the interpretation* and is performed by programs called *interpreters*. *Translators*, on the other hand, perform *the translation* of a document originally written in a natural language into E-speranto. The latter is analogous to translating the source code in the programming language into the executable code of the platform where the program execution takes place.

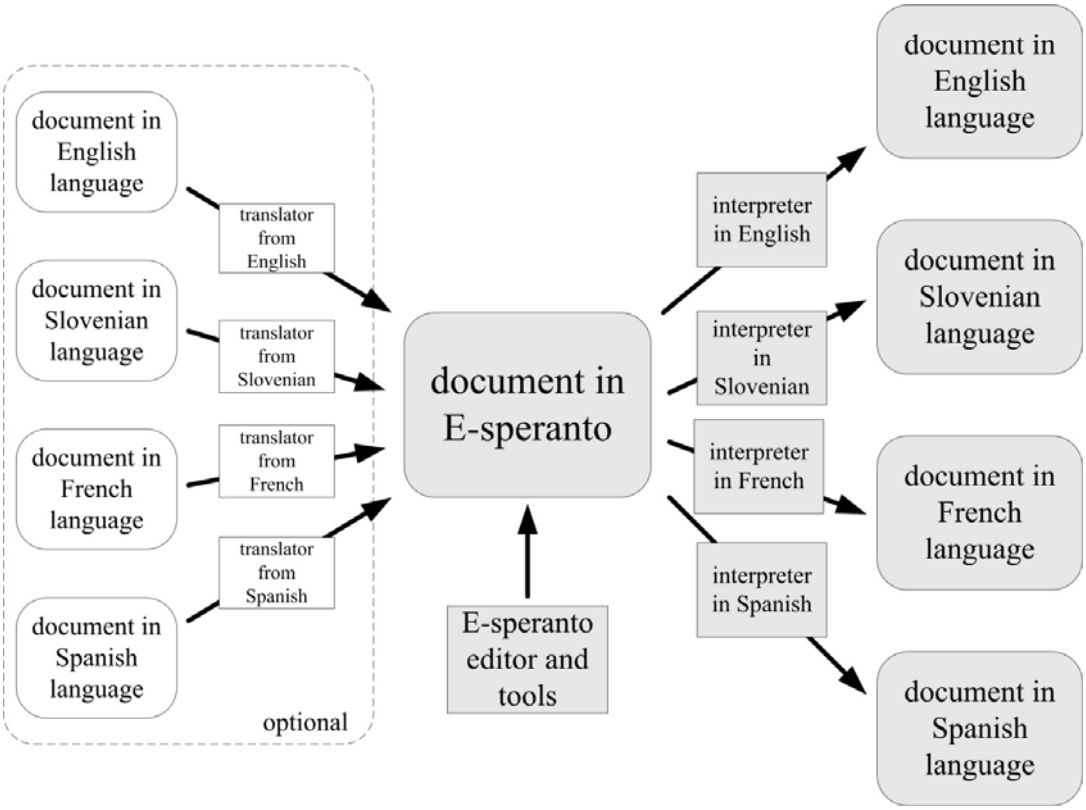

Figure 1 The process of generating documents in E-speranto. The documents are generated in a development environment with special tools, i.e., substitutes for the translators into E-speranto. When the user decides to view a document, its content is interpreted in a chosen natural language.

The remainder of the paper is organized in the following way: the next section presents the most important approaches to the development of computer languages intended for multilingual communication. The third section presents E-speranto in more detail, focusing on the abstract model of the message and its record in the concrete syntax of E-speranto. The following two sections present the proof-of-concept implementation of the multilingual Web based on E-speranto and the evaluation of the E-speranto interpretation. The paper ends with our findings and presentation of future work.

2 Related work

Computer languages that enable multilingual communication most often function as some sort of an interlingua in multilingual translation. In the ideal case, the record in an interlingua contains all the information required to generate documents in any given natural language, which inherently means that the content in the interlingua must be presented in an abstract form independent of any natural language.

When developing the interlingua, the most demanding issue proved to be the creation of a genuinely universal language that would at the same time be completely independent of all natural languages. This is due to the fact that it is virtually impossible for the vocabulary (i.e., the set of all

available concepts) of an interlingua to include all the meanings from the vocabularies of all natural languages in which we wish to interpret. That is why most interlinguae have an independent structure and a vocabulary that is not completely independent of natural languages.

One of the exceptions is DLT (Distributed Language Translation) [8], a research project that took place in the 1980s. DLT was based on a somewhat modified version of Esperanto which functioned as the interlingua. The basic idea of the project was to send a document in Esperanto over the network and interpret it in a preferred natural language on the target computer. Despite the relative consistency of its grammar, it was established that Esperanto is not appropriate for an interlingua. Esperanto, namely, has many features similar to those of natural languages, which causes lexical and structural mismatches typical for direct translations between natural languages.

Rosetta [9] uses an interlingua based on Montague grammar [10] and the principle of semantic compositionality, i.e., the argument that syntax is inherently linked with semantics. Rosetta's intermediate representation structure is defined by the isomorphic grammars of all the supported target languages, which significantly simplifies its interpretation. However, the fact that it inherently contains the features of a limited number of languages makes the interlingua non-universal and the system itself non-scalable as it would be required to modify the interlingua in order to add a new language into the system.

The interlingua KANT [11], created with the purpose of translating technical documentation, is based on English with a limited vocabulary. The system produces accurate translations; however, the interlingua cannot be used for general multilingual communication due to the limited scope of its application.

UNL (Universal Networking Language) [12], a successor of the ATLAS-II [13] and PIVOT [14] interlinguae, is a computer language for the representation and exchange of information on the internet. The content in UNL has the logical form of a graph represented with linear expressions in concrete syntax. The authors are able to write in UNL directly, but due to the fact that the language itself was not intended for such use, it is not easily intelligible.

As is the case with records in interlingua, the record in E-speranto also serves as the basis for the formation of text in natural languages. This is why E-speranto has features that are similar to those of interlingua; however, it also has the following advantages:

- E-speranto is intelligible to human users, which has in the past already proven to be an advantage when developing different internet standards. Most often only the standards that were intelligible to users remained used for a long period of time.
- The practical application does not depend on the existence of translation tools from natural languages into E-speranto, which means that the documents can be created directly with the aid of tools designed especially for this purpose and additional information resources (e.g., E-speranto grammar and vocabulary). As the users, i.e., the sources of the message, best understand the meaning that they wish to convey, the latter can be recorded in an optimum way if the users are provided with appropriate tools.
- The syntax of E-speranto is based on XML (eXtensible Markup Language), which enables the natural integration of documents generated in E-speranto into web pages.

- XML is a standardized and established language on the Web, which is also very important for the implementation of E-speranto. XML tools are widely available and can also be used within the context of E-speranto. We expect the latter to stimulate the development of E-speranto interpreters.

3 E-speranto

E-speranto was named after Esperanto [15], a language constructed at the turn of the 20th century with the aim to become the international auxiliary language. The most important features of Esperanto are the consistency of its grammar and vocabulary, unambiguousness and a suitable level of expressiveness. Although it never acquired the status of a universal second language, it is the result of a broad linguistic knowledge and years of work, which is why it was used as the basis for the development of E-speranto. The latter can thus be considered as an electronic version of Esperanto.

When creating E-speranto, we pursued the following premises [16]:

1. The purpose of our project is to define a computer language the expressivity of which can be compared to that of natural languages.
2. The language must be simple and intelligible to both human users and computers. Intelligibility to computers is important due to the fact that they will be parsing E-speranto, and intelligibility to humans is vital as the users will be the ones that will compile the documents until translation tools for E-speranto are developed.
3. There must be no exceptions to the grammar rules of E-speranto and the features of the concepts must be recorded explicitly.
4. The vocabulary of E-speranto must include lexical units that represent unique concepts. Every lexical unit must represent only one concept and every concept must only be represented by one lexical unit.
5. The record of meaning and style of the message must be kept separate.

3.1 Abstract model of the message

The message in E-speranto can be represented in an abstract manner with the help of a semantic network, similar to those that are used for knowledge representation [17]. The basic building blocks in the semantic network of E-speranto are concepts, the relations between concepts and concept attributes (Fig 2). A concept is an abstract idea or a symbol and can also be defined as a unit of meaning. The concepts in E-speranto are linked with relations that express the roles of the concepts within the message. The attributes transform the concepts into concrete objects by placing them in the world as it is perceived by the author of the message. Beside the information in the concepts and relations, the attributes include all the additional information (e.g., the quantity of the concept, the temporal frame etc.) essential for the exact transfer of a certain meaning into a natural language without any loss of details.

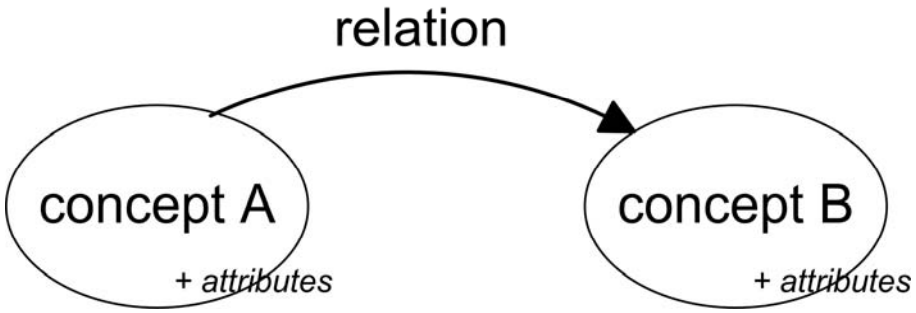

Figure 2 The communication model in E-speranto: the meaning is presented with concepts which are linked with relations.

3.1.1 The set of concepts

One of the biggest challenges when creating a language intended for multilingual communication is the choice of its vocabulary (i.e., the set of all available concepts). The most important question is how to choose the meaning primitives that define the accuracy of expression and the expressiveness of the language. The meaning that the concepts stand for is, namely, far more important than the manner (i.e., the form) in which the concepts are presented. When developing the basic vocabulary of E-speranto, we used the vocabulary of Esperanto together with the meanings that are conveyed by the individual items in the vocabulary. A part of the E-speranto vocabulary is presented in Table 1.

E-speranto	**English**	**description**[a]
linago	linen	*strong cloth that is woven from the fibres of the flax plant*
lingva	language	*adjective "language"*
lingvistika	linguistic	*connected with language or the study of language*
lingvistiko	linguistics	*the systematic study of the structure and development of language in general or of particular languages*
lingvisto	linguist	*someone who studies foreign languages or can speak them very well, or someone who teaches or studies linguistics*
lingvo	language	*a system of communication consisting of sounds, words and grammar, or the system of communication used by the people of a particular country or profession*

Table 1 A part of the E-speranto vocabulary (left column). In addition to the lexical units from E-speranto, the equivalent English lexical units and their respective description are also presented.

[a] The description of the concepts is taken from the Cambridge Dictionary.

3.1.2 The relations between the concepts

In addition to the meaning conveyed by the concepts, the relations between the concepts are another vital component that has an important influence on the meaning of the message. The relations in E-speranto can be divided into logical and semantic. Semantic relations are used for describing the meaning relationship between the concepts, while logical relations describe the logical relationships.

The use of relations between the concepts at the semantic level is vital, as this is the only way in which we can establish the independence of the record in E-speranto from the record in a given natural language. Individual relations are interpreted into different languages in different ways, for example by using different prepositions (Table 2).

meaning	**record in E-speranto**	**example in Slovenian**	**example in English**
expressing the means, tool or instrument for performing an activity	tool-instrument	*pokriti streho* ***s*** *strešniki*	*to cover the roof* ***with*** *tiles*
expressing company or attendance	company-attendance	*na obisk je prišel moj sin* ***z*** *družino*	*my son* ***with*** *his family came for a visit*
expressing the characteristic property or feature	characteristic-property	*konj* ***z*** *dolgo grivo*	*a horse* ***with*** *a long mane*
expressing the separation or removal	remove_from_sth	*s težavo so ga odstranili* ***s*** *stola*	*they removed him* ***from*** *the chair*
expressing the separation or removal	remove_from_sth	*s težavo so ga odstranili* ***z*** *mize*	*they removed him* ***from*** *the desk*

Table 2 Examples of semantic relations in E-speranto. All the relations presented are interpreted into Slovenian with the aid of the prepositions "s" or "z" [b], while they are interpreted into English with the aid of the preposition "with" in most, but not all cases.

In theoretical linguistics, there is no consensus on the number and set of semantic relations. Levi [18] for example argues that the number of semantic relations between noun compounds is limited, whereas Downing [19] claims that the number of these relations is infinite. The set of semantic relations in E-speranto has not been defined yet; however, the framework for adding the relations has already been created.

The basis for determining the set of relations is Esperanto, in which the relations are recorded implicitly (for example with affixes). The set of relations from Esperanto is continuously supplemented by new entries, as it turned out that the relations from Esperanto grammar do not suffice for a suitable semantic description of the content.

[b] In Slovenian, the prepositions "s" and "z" are merely two different physical representations of the same preposition. The preposition "s" is used in front of the words beginning with a voiceless consonant (c, č, f, h, k, p, s, š or t); in all other cases the preposition "z" is used.

3.1.3 Concept attributes

Concept attributes contain the pieces of information that are not inherently present in the concepts and relations themselves, but are of vital importance for interpreting the meaning into a natural language. The attributes mostly present the relationship of the author to the message and can for example refer to the time of the occurrence in relation to the time when the message was created (past, present, future, not stated etc.), verbal aspect (finite, non-finite), mood (conditional, imperative, indicative) etc. Whereas some attributes are vital for the correct interpretation of meaning into a natural language, some only indicate the style with which the author has formed his message.

3.2 Concrete syntax

The concrete syntax of E-speranto is based on the syntax of XML [20]. The latter is an established language on the Web and is also intelligible to human users. The prevalence of XML opens a wide field of possibilities of using already existing and established technologies, such as for example XML parsers, the interfaces for the manipulation of documents (e.g., DOM, Document Object Model), document validators and scheme languages. An example of a scheme language is XML Schema [21], which was used when defining the exact grammatical and syntactical rules of E-speranto (Fig. 3).

```
<xs:element name="subject">
   <xs:complexType>
      <xs:all>
         <xs:element minOccurs="0" ref="word"/>
         <xs:element minOccurs="0" ref="sequence"/>
         <xs:element minOccurs="0" ref="sentence"/>
         <xs:element minOccurs="0" ref="subordinate"/>
      </xs:all>
      <xs:attributeGroup ref="subject_att"/>
   </xs:complexType>
</xs:element>
<xs:attributeGroup name="subject_att">
   <xs:attribute ref="comment"/>
   <xs:attribute ref="number"/>
   <xs:attribute ref="detail"/>
   <xs:attribute ref="gender"/>
   <xs:attribute ref="proper_name"/>
</xs:attributeGroup>
```

Figure 3 A part of the XML Schema, which defines the grammar and concrete syntax of E-speranto.

The basic principle of generating a document in E-speranto is quite straightforward: the concepts are first recorded with lexical units from the E-speranto vocabulary, are assigned their individual properties by using XML elements and attributes, and are then linked with relations (Fig. 4).

```
<sentence feelings="declarative" organization="simple">
   <subject proper_name number="singular">
      <word>e-speranto</word>
   </subject>
   <predicate detail_predicate="main" mood="indicative" voice="active"
              tense="present" person="third">
      <word>esti</word>
      <predicate detail_predicate="predicate_noun" number="singular">
         <word>dezajno</word>
         <object relation="composition_element" number="singular">
            <word>lingvo</word>
            <attribute detail_attribute="relativity">
               <word>komputero</word>
            </attribute>
         </object>
      </predicate>
   </predicate>
</sentence>
```

Figure 4 A shortened version of the sentence “E-speranto is a design of a computer language” in E-speranto (“dezajno”=“design”, “lingvo”=“language”, “esti”=“to be (an instance of)”, “komputero”=“computer”).

The basic building block in the concrete syntax is the element *sentence*, a semantic unit that roughly corresponds to a sentence in a natural language. A sentence in E-speranto comprises concepts arranged into classes which are defined in the Schema. Classes have a vital role, both from the semantic and grammatical perspective:

- Assigning concepts to appropriate classes enables the distinction between the concepts as regards their role in the meaning of the message, while it at the same time establishes the frame of their interpretation (Table 3).
- Each class has its own “subgrammar” defining the syntactic and semantic restrictions of the class, for example:
    - the available subordinate classes;
    - the set of attributes used to describe the concept in more detail;
    - the set of relations used to link the concept with other concepts.
- Classes are also important when interpreting E-speranto. In the abstract syntax tree (AST), which functions as the data structure in the interpretation process, the classes match the non-terminal nodes, the basis for defining the transformational rules.

class	interpretation	features
predicate	The concept that represents an action or state.	tense, person, mood, etc.
subject	The concept with the semantic role of the doer of the action (the agent, deep subject).	number, gender, etc.
object	The concept that is involved in the action or state, but is not its doer (the recipient).	number, gender, semantic relation, etc.
adverbial	The concept describes the circumstances of the action or state.	number, semantic relation, etc.
attribute	The concept describes the features of the other concepts.	type, etc.

Table 3 The classes are used for distinguishing between the concepts in E-speranto as regards their roles in the meaning of the message, while at the same time defining their syntactic and semantic limitations.

3.3 Recording the concepts

The concepts in E-speranto are recorded using lexical units from the vocabulary of Esperanto and are then placed in the element *word*. The concepts must subsequently be assigned to an appropriate class with the aid of the parent element of the element *word* (Fig. 4). The element *word* must not have any subordinate elements. In the abstract syntax tree created when parsing the document in E-speranto, the content of the element *word* thus represents a terminal node.

3.4 Recording the relations between the concepts

A semantic relation is recorded as the value of the XML attribute *relation*. The direction of a relation is implied with the nesting of elements. The relation within a specific element implies the semantic relation of the concept in this element to the concept in a parent element, as is evident from Table 4.

The available set of semantic relations depends on which class the concept is assigned to and consequently its role in the meaning of the message. Different concept classes use different sets of semantic relations in accordance with the formal definition in the scheme. The concepts in the class *subject* thus have a pre-set role of the agent or the doer of the action represented by the class *predicate*. The class *object* allows only the usage of semantic relations linking several nominal concepts. Other relations (temporal, locative, causal etc.) can only be used in concepts belonging to the class *adverbial*.

Semantic relations can be combined with logical relations by using the element *sequence* (Table 5). As opposed to semantic relations, logical relations do not depend on the class of the concept.

semantic relation	**record in Esperanto**	**record in E-speranto**

"A is a part of B"	/	`<object> B`
		`  <object relation="partOf"> A </object>`
		`</object>`

"A is the receiver of B's activity"	-n	`<predicate> B`
		`  <object relation="direct"> A </object>`
		`</predicate>`

"A is an instrument for B"	/	`<object> B`
		`  <object relation="instrument"> A </object>`
		`</object>`

Table 4 Simplified examples of the semantic relations between the concepts and their record in E-speranto. The concept in an internal element (A) is in a meaning relationship with the concept in the external element (B). In the first case, the internal concept is a part of the external concept (relation "partOf"); in the second case, the internal concept is the receiver of the external concept's activity (relation "direct"); whereas, in the third case, the internal concept represents an instrument for the external concept (relation "instrument").

logical relation	**record in Esperanto**	**record in E-speranto**

"A and B"	the conjunction "kaj" is used: "A kaj B"	`<sequence relation="and">`
		`  <x> A </x>`
		`  <x> B </x>`
		`</sequence>`

"A or B"	the conjunction "aŭ" is used: "A aŭ B"	`<sequence relation="or">`
		`  <x> A </x>`
		`  <x> B </x>`
		`</sequence>`

Table 5 Simplified examples of the logical relations between the concepts and their record in E-speranto. The element x stands for any of the existing concept classes.

3.5 Recording concept attributes

The concept attributes are recorded in the form of XML attributes (Table 6). The base for defining the attributes and the set of their values is the grammar of Esperanto; however, new attributes and their values can be added if required. As is the case with semantic relations, concept attributes also depend on the class to which the concept has been assigned. For example, tense or person can be assigned only to the concept belonging to the class *predicate*.

feature	**record in Esperanto**	**record in E-speranto**
plural	-j	`<subject number="plural">`
activity in the present	-as	`<predicate tense="present">`
conditionality of the activity	-us	`<predicate mood="conditional">`
negation	ne	`<predicate negation>`
proper name	capital letter	`<subject proper_name>`

Table 6 Examples of concept attributes.

3.6 Style and translation hints

A record in E-speranto has an abstract canonical form which is primarily intended for the record of meaning and does not inherently define the style of the message. The style that should be used by the interpreter can be implied by the author with the aid of the so-called translation hints (Table 7). The interpreter decides whether or not it will comply with the hint based on the grammar and vocabulary of the target language. In addition, different interpreters interpret style in different ways. The separation of the meaning of the message from its style is analogous to the separation of the content of the web document in HTML (HyperText Markup Language) from its style in CSS (Cascading Style Sheets).

translation hint	**record in Esperanto**	**record in E-speranto**
passive voice	-t	`<predicate voice="passive">`
exclamation	/	`<sentence feelings="exclamation">`
irony	/	`<sentence feelings="irony">`

Table 7 Examples of translation hints. The hints are intended as an additional piece of information that improves the style of interpretation.

An example of a translation hint used when interpreting a sentence is the author's preference for the passive voice (Figure 5). In this case, the concept assigned to the class *subject* (the so-called "deep subject") in E-speranto, turns into a "surface" object and the verb changes into the passive form.

```
<subject proper_name number="singular">
   <word>John</word>
</subject>
<predicate tense="present" aspect="ongoing" voice="passive">
   <word>skribi</word>
   <object relation="direct" number="singular">
      <word>letero</word>
   </object>
</predicate>

A letter is being written by John.
John is writing a letter.
```

Figure 5 Using a translation hint to interpret a predicate in the passive voice. The two sentences below the record in E-speranto demonstrate the results of the interpretation with and without the use of the translation hint.

4 Proof-of-concept

The existing version of E-speranto was used to design a proof-of-concept implementation of the multilingual Web based on E-speranto[c]. The system combines the tools for the generation of documents in E-speranto and their interpretation, and is based on the client-server architecture. Figures 6 and 7 show the structure and architecture of the system.

[c] The system can be tested on the web site http://www.e-speranto.org/. The website also contains more information on E-speranto and related resources.

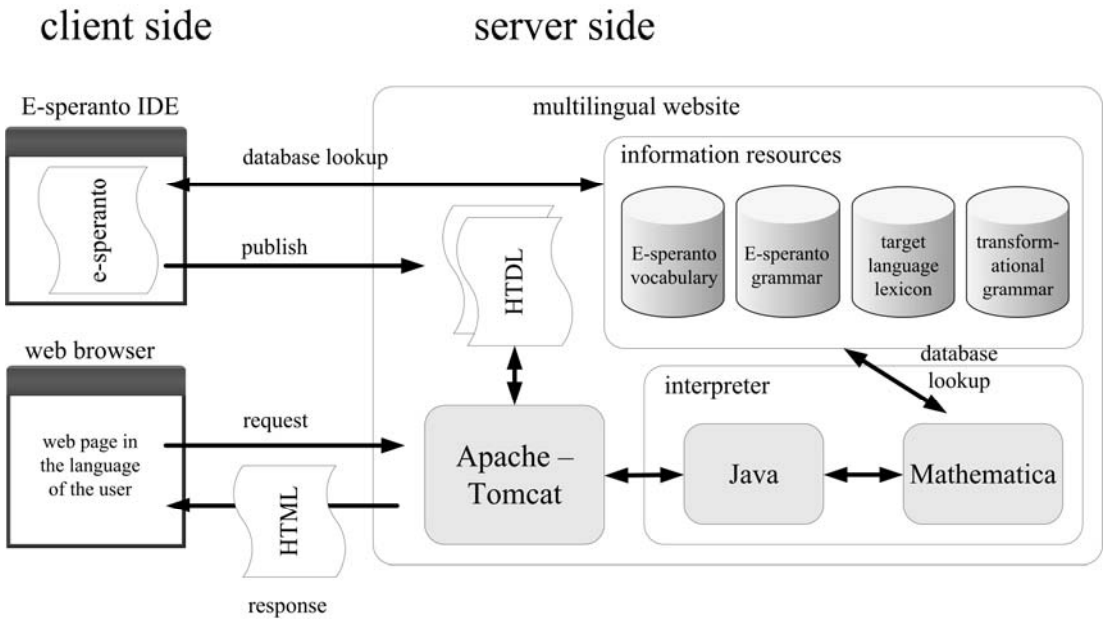

Figure 6 The proof-of-concept implementation of the multilingual web based on E-speranto.

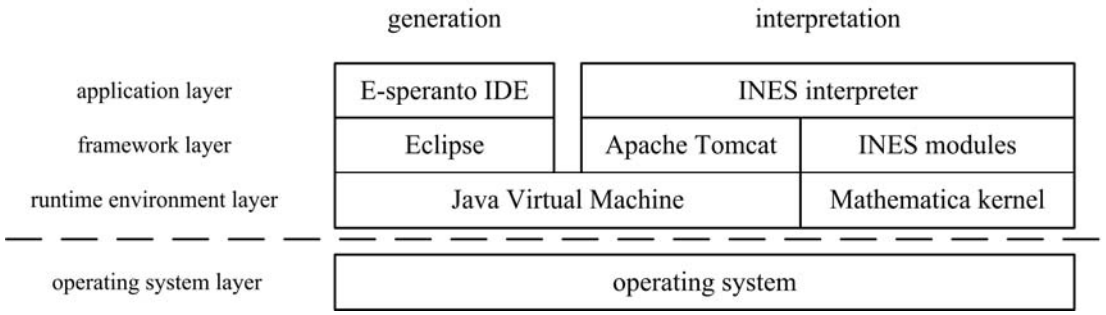

Figure 7 The architecture of the proof-of-concept. The system it is built on two operating system-independent runtime environments: the kernel of the interpreter is based on Wolfram Mathematica and the rest of the system is based on Java platform.

The basic elements required for interpreting E-speranto are the interpreters and information resources. The interpreters contain all the procedures (algorithms) necessary for the interpretation, whereas the information resources "parameterize" the above-mentioned procedures with additional data (the rules of transformational grammar, the data on the possible ways of using specific lexical units etc.). The system for interpreting E-speranto is presented in more detail in [22].

In order to create documents in E-speranto, an environment based on the open source platform Eclipse [23] was developed. The basic tool of the development environment is a text editor adapted for writing in E-speranto. The most important functionalities of the environment are:

- compliance testing of the document with the grammar of E-speranto;
- suggestions of available content in accordance with the E-speranto grammar;
- survey of the E-speranto vocabulary, descriptions of the concepts and examples of their use in the language of the user;
- instantaneous interpretation in the chosen language (a "preview" of the interpretation).

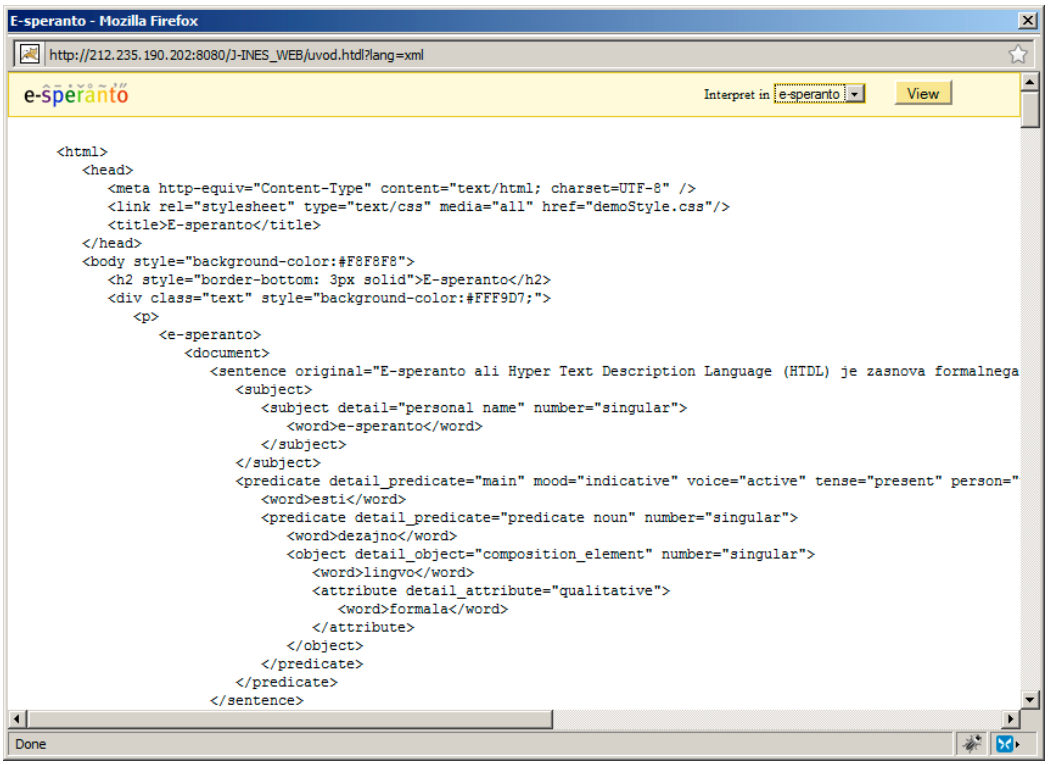

Figure 8 The source code of the web page in HTDL, a combination of the HTML used to record the structure of a web page and E-speranto which records the multilingual content.

When the document in E-speranto is created, the author can incorporate it into the existing HTML. Due to this fact, the newly created format was named HTDL (HyperText Description Language) (Fig. 8). The author can publish the document on a "multilingual web site". When the web server that hosts the multilingual web site receives a request for the HTDL document, it forwards the request to the interpreter INES (INterpreter of E-Speranto). The interpreter parses the document and separates the records in E-speranto from the ones in HTML. It constructs an abstract syntax tree from the content in E-speranto. The abstract syntax tree is then processed by the kernel of the interpreter. As the resulting text substitutes the E-speranto record in the HTDL document, the server responds with a plain HTML document with the content in the language of the user (Fig. 9).

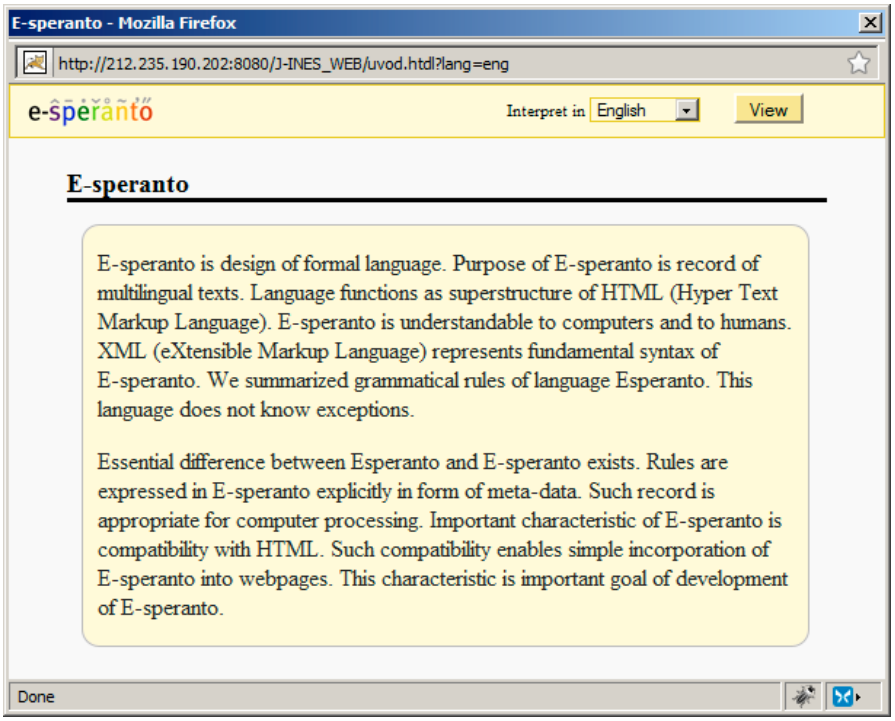

Figure 9 The result of the interpretation of a web page in E-speranto.

In light of an easier implementation, we decided to use the interpreters on the server side, as this does not require the standardization of E-speranto and its support by the web browsers. If E-speranto becomes standardized, the interpreters will be moved to the client side in the form of browser plug-ins.

5 Evaluation

We evaluated the E-speranto interpretation with the use of the proof-of-concept system. We chose four English texts (three recipes and one message from a mobile operator to the roaming user) that altogether consist of 50 simple sentences. Due to the limited functionality of the interpreters, the chosen texts were somewhat adapted when recorded in E-speranto. We added the records required for the interpretation to the information resources; however, the functionalities of the interpreters were not changed in any way. The record in E-speranto was then interpreted into Slovenian with the aid of the INES interpreter, while the (adapted) original text was at the same time translated with the online translation tool Google Translate.

The evaluation of the quality of the translation and the interpretation was performed in two stages. In the first stage, nine evaluators, all Slovenian native speakers and potential users of the system, were selected. They were given both the translated and the interpreted texts, whereby the manner in which the text was created was not revealed. They were also not shown the original text. The evaluators marked each sentence in the texts with marks from 1 to 5 (with 1 being the worst and 5 being the best possible translation).

In the first stage, the evaluation was based on two aspects. The first one was the clarity of the expressed meaning in the translated and the interpreted texts, whereby the evaluators were asked the following question: "*Does the sentence express the meaning clearly; that is, do you understand what it says?*" The second aspect incorporated the grammatical correctness of the expressed meaning. The evaluators were asked the following questions: "*Is the sentence grammatically correct? Does it express the meaning in a natural way?*"

The second stage of the evaluation was conducted in a similar way, the only difference being that the evaluation was conducted by three professional translators. They were asked to compare the translated and the interpreted texts with the adapted original text and were asked the following question: "*Does the sentence contain the same information (i.e. the same meaning) as the original sentence?*".

The average evaluation marks of the translation and the interpretation are given in Table 8.

	translation			interpretation		
	clarity of expression	grammatical correctness	retention of meaning	clarity of expression	grammatical correctness	retention of meaning
text 1 – "mashed potatoes"	3,83	3,42	3,86	4,63	4,60	4,45
text 2 – "Mojito cocktail"	3,94	3,38	3,92	4,93	4,95	4,56
text 3 – "pancakes"	3,59	3,19	3,43	4,90	4,81	4,83
text 4 – "operator's message"	4,83	4,94	4,72	4,70	4,22	4,72
all texts	3,91	3,53	3,86	4,80	4,71	4,63

Table 8 The results of the evaluation of the translation with an online translation tool and the interpretation with an E-speranto interpreter

6 Discussion

6.1 The results of the evaluation

The results of the evaluation show that the E-speranto interpretation proves to be more accurate than the translation with the online translation tool Google Translate in all three aspects. In fact, the average

mark of the E-speranto interpretation is 0,8 mark higher for the retention of meaning and 1,2 mark higher for grammatical correctness.

When interpreting the evaluation results, we need to point out that the above-mentioned results should not be understood as a comparison between two implementations of machine translation systems, but rather as a comparison between two translation approaches. If we were to select a random text and repeat the evaluation, the online translation tool Google Translate would most likely get the same marks, whereas the E-speranto interpreter would most probably get significantly lower marks. Due to the limited information resources is it highly unlikely that they would contain the exact content required for the interpretation of the selected text. The presented results should thus be understood rather as the potential of multilingual web based on E-speranto and not as a comparison between the prototype of the E-speranto interpreter and existing translation tools.

The evaluation of the E-speranto interpretation reveals the advantages of the language and also its weaknesses. The main advantage of E-speranto lies in the fact that the interpretations can be very precise. When interpreting E-speranto, very few words turn out to be wrongly translated, not fully translated or even not translated at all, the reason being that the dictionary of the target language, the E-speranto vocabulary and the set of semantic relations can easily be supplemented.

In fact, it is the limited vocabulary and set or semantic relations that currently represent the biggest shortcomings of E-speranto. In order to accurately interpret the language, one needs to precisely define the meaning of concepts and semantic relations, both of which also need to be studied in detail and included in the information resources. This shortcoming will be eliminated with further development of E-speranto and its interpreters, as both the vocabulary and the set of semantic relations are subject to constant supplementation.

An additional shortcoming of E-speranto is also the fact that the language itself is primarily intended for recording the meaning, whereas its grammar contains very few means for expressing the style of the message. The latter was also reflected in the comments of the translators when evaluating the interpretation, as almost half of them found the interpretations somewhat “artificial” despite the fact that they were grammatically correct.

Until significant advances have been made in the field of natural language understanding, a part of the burden of precise interpretation will fall on the user. The tools for automatic recording of the texts in E-speranto play an important role in alleviating this problem. Although the existing development environment proved as appropriate for the records in E-speranto, it will need to be improved and upgraded especially with graphic editors. The latter would make the recording in E-speranto more intuitive and the users would no longer be required to use the concrete E-speranto syntax. Recording texts in E-speranto can in a way be compared to creating web pages, where many existing tools facilitate the work of the developers who are therefore not required to know a single HTML tag.

6.2 The development of E-speranto

When developing E-speranto, we decided to avoid Semantic Web languages such as RDF (Resource Description Framework) and OWL (Web Ontology Language). There are numerous reasons for this decision. Semantic Web technologies were introduced above all to model conceptual relationships among resources on the Web and are therefore less appropriate for denoting syntactic relationships

(such as constituency) that appear in statements in human languages. Furthermore, the primary aim of Semantic Web was to structure the Web information with the help of metadata which are primarily intended to be readable by the computers. During the development of Semantic Web, this resulted in an extensive language stack with complex syntaxes which are not easily comprehensible to human authors and are thus less appropriate for manual creation of "interlingual" documents.

Nevertheless, Semantic Web technologies are suitable for describing (static) facts (or world knowledge). In the context of E-speranto, for example, world knowledge in the form of existent ontologies (e.g., [24]) can be used to link the concepts from the vocabulary with the knowledge about the concepts. Because such knowledge is mandatory for both semantic analysis when creating a document in E-speranto and reasoning when interpreting the document in a natural language, we plan to incorporate it into the system in the future.

The development of E-speranto is closely related to the development of the interpreters. The results of the interpretation process within the proof-of-concept system have given us valuable feedback that can be used when further developing E-speranto. Based on the proof-of-concept we, for example, established that the interpretation needs to take place on a much more abstract level that was first intended. Apparently, if E-speranto mirrors the grammar and syntax of natural languages, the authors often make the mistake of modelling E-speranto records on the records in their mother tongue.

An additional case is the separation of meaning from style, as is for example the case when the subject is repeated in two successive sentences. In E-speranto, text (1) is recorded by repeating the subject of the first sentence as in text (2). The task of the interpreter is to establish whether both sentences are referring to the same subject and to provide a stylistically more appropriate interpretation, such as for example text (1).

(1) "*John lives in London. He works at the University.*"

(2) "*John lives in London. John works at the University.*"

These and many other examples open additional issues that must be addressed, both about the content in E-speranto and the content that must be created in the interpretation process. Currently, four groups are working on the development of E-speranto interpreters – three on the interpreters into Slovenian, Serbian and Russian, the representatives of the Slavic languages, and one on the interpreter into English, which is mostly used for demonstrative purposes. Each of the afore-mentioned groups regularly provides feedback that is used to further improve the design of E-speranto.

6.3 Simplifications

In the practical application of the system we used several simplifications:

- we used the vocabulary of Esperanto and thus also reduced the precision in expressing the meaning,
- we limited ourselves to the interpretation of E-speranto in Slavic languages,
- we limited ourselves to the interpretation of simple sentences.

Ideally, E-speranto should be free of any lexical or structural ambiguities and should enable the interpretation in any natural language. Such a language would, however, be complex and impractical

for everyday use. In addition, different languages enable different ways of describing the world that surrounds us, as their means of expression have developed in different ways through the centuries. The way in which a person perceives reality strongly depends on the way his culture experiences the world, as this affects the way in which people belonging to a specific culture express themselves and how they name certain concepts (i.e., the vocabulary)[d]. The E-speranto vocabulary should be able to grasp all the meanings in all the natural languages in which we wish to interpret E-speranto, even though a specific meaning only appears in one of the above mentioned languages.

In reality, however, a compromise between complexity and expressiveness should be considered. In our case, the basic E-speranto vocabulary was formed on the vocabulary of Esperanto, although this means that it does not match the vocabularies of all the target languages. A concept in E-speranto might not have a related representation in the target language and therefore has to be expressed in a different way (e.g., descriptively[e]). In later stages of development we intend to supplement the E-speranto vocabulary with entries from existing dictionaries (for example [25]).

In the initial development stages, we settled for a record that can be interpreted into Slavic languages, especially Slovenian, Serbian and Russian - the mother tongues of the researchers working on the project. For this reason, we focused merely on the sets of attributes and relations that are needed for the interpretation into the aforementioned languages without compromising the expressed meaning. Nevertheless, E-speranto can, with somewhat lower precision, also be interpreted in other, especially Indo-European languages.

In addition, we also limited ourselves to the interpretation of simple sentences at this stage. In this way, the meaning mostly stays intact due to the fact that simple sentences can be used to express almost any meaning. This simplification, however, requires that the formulation of the intended message be properly adapted.

7 Conclusion

The paper describes the design of E-speranto, a formal computer language for recording multilingual texts on the Web. The main purpose of E-speranto is the interpretation of web content in the language of the user. This goal will be achieved when sufficient E-speranto interpreters are developed. In addition to its use in multilingual communication, E-speranto can also be used in other applications, such as for example semantic searching and automatic reasoning.

In order for E-speranto to become an established computer language, many issues still have to be addressed. As a part of our future work on the project, we intend to research the possibility of supplementing the E-speranto vocabulary with entries from existing dictionaries and also define the semantic relations between the concepts in more detail. In addition, we intend to include the prototypes of E-speranto interpreter for other Slavic languages into the proof-of-concept system and evaluate the interpretation results based on a much larger number of texts.

---

[d] The language of the Eskimos contains more words for snow than any other language, while E-speranto only has one – the one from Esperanto. In Hindi, there are several words for love, depending on the way it is expressed.

[e] In E-speranto, the concept *snow* can be further defined with the use of adjectives: packing, heavy, driving etc. The interpreter substitutes such a "complex concept" for the suitable expression in the target language, if the latter contains such an expression.

References

1. Internet World Stats, http://www.internetworldstats.com. Accessed 30 June 2011.
2. Yahoo! Babel Fish - Text Translation and Web Page Translation, http://babelfish.yahoo.com. Accessed 30 June 2011.
3. Google Translate, http://translate.google.com. Accessed 30 June 2011.
4. PROMT Translation Software and Dictionaries, http://www.promt.com. Accessed 30 June 2011.
5. SYSTRAN - Online translation, translation software and tools, http://www.systranet.com. Accessed 30 June 2011.
6. Paul, L. M. (ed.), Ethnologue: Languages of the World, Sixteenth edition. SIL International, (Dallas, 2009).
7. Hutchins, W. and Somers, H., An Introduction to Machine Translation. Academic Press, (New York, 1992).
8. Schubert, K., The Architecture of DLT – interlingual or double-dialect. In: Maxwell, D., Schubert, K. and Witkam, T. (eds.), New Directions in Machine Translation. Floris Publications, (Holland, 1988).
9. Rosetta, M. T. (pseud.), Compositional Translation. Kluwer Academic Publishers, (Dordrecht, 1994).
10. Thomason, R. (ed.), Formal Philosophy, Selected Papers of Richard Montague. Yale University Press, (New Haven, 1974).
11. Nyberg, E. and Mitamura, T., The KANT system: Fast, accurate, high-quality translation in practical domains. In: Proceedings of the 14th conference on Computational linguistics (1992).
12. Uchida, H., Zhu, M. and Della Senta, T., Universal Networking Language: A gift for a millennium. The United Nations University, (Tokyo, 1999).
13. Uchida, H., ATLAS II: A Machine Translation System Using Conceptual Structure as an Interlingua. In: Proceedings of the Second Machine Translation Summit, (Tokyo, 1989).
14. Muraki, K., PIVOT: Two-Phase Machine Translation System. In: Machine Translation Summit: Summit Manuscripts and Program, (Japan, 1987).
15. Amerio, F., Bonvecchiato, G. and Fighiera, G. C., Esperanto: Data and Facts, 2nd edition. FEI, (Milan, 2002).
16. Tomažič, S., Multilingual Web with E-speranto. IPSI BgD Transactions on Internet Research, 3(2), 13-15.
17. Sowa, J. F., Knowledge Representation: Logical, Philosophical, and Computational Foundations. Brooks Cole Publishing Co., (Pacific Grove, 2000).
18. Levi, J. N., The Syntax and Semantics of Complex Nominals. Academic Press, (New York, 1978).
19. Downing, P., On the creation and use of English compound nouns. Language, 53(4), 810-842.
20. Bray, T., Paoli, J. and Sperberg-McQueen, C. M. (eds.), Extensible Markup Language (XML) 1.0 (Fifth Edition), W3C Recommendation 26 November 2008, http://www.w3.org/TR/REC-xml/. Accessed 30 June 2011.
21. Thompson, H. S., Beech, D., Maloney, M. and Mendelsohn, N. (eds.), XML Schema Part 1: Structures, W3C Recommendation 28 October 2004, http://www.w3.org/TR/xmlschema-1/. Accessed 30 June 2011.
22. Jakus, G., Sodnik, J. and Tomažič, S., The Architectural Design of a System for Interpreting Multilingual Web Documents in E-speranto. Journal of Universal Computer Science, 17(3), 377-398.
23. Eclipse web page, http://www.eclipse.org. Accessed 30 June 2011.
24. DBpedia web page, http://dbpedia.org. Accessed 30 June 2011.
25. Cambridge Advanced Learner's Dictionary Online, http://dictionary.cambridge.org/. Accessed 30 June 2011.