Peer Reviewed Article

Vol.11(3) September 2009

Bootstrapping an XML Schema of syntactic functions into a skeleton ontology *

JH Kroeze
North-West University (Vaal Triangle Campus)
Vanderbijlpark South Africa
Kroeze.Jan@nwu.ac.za

Information Systems is regarded as an interdisciplinary science. Therefore, insights from the humanities are important for this discipline, although this is not always recognised or valued. One outstanding example is the current upsurge in the study and use of 'ontologies' in information systems, bridging the disciplines of philosophy and computing. The article begins with an overview of an XML Schema that was used as a thesaurus to ensure consistency in the syntactic tagging of the Hebrew text in Genesis 1. The broader syntactic taxonomy, on which the XML Schema is based, and which may be used to analyse the syntax of Biblical Hebrew texts, is discussed in detail. The research also investigates how the concept of 'ontologies' is used in computational linguistic projects. These concepts form the building blocks for suggesting an ontology of syntactic functions for Biblical Hebrew, which may be implemented and used by linguistic information systems to ensure its quality and reliability (i.e. an ontology *for* information systems). Some possibilities are also proposed of how such an ontology may be put to use.

Key words: XML Schema, thesaurus, taxonomy, ontology, Biblical Hebrew syntax

Received: 26 August 2009

Contents

1. Introduction
2. Taxonomy and XML Schema of Hebrew syntax
3. Ontologies in ICT
4. Links between XML schemas and ontologies
5. Use of IT ontologies in (computational) linguistics
6. Suggesting an ontology of syntactic functions in Biblical Hebrew
7. Possible uses of such an ontology
8. Conclusion
9. References

1 Introduction

Information Systems (IS) is regarded as an interdisciplinary science. Although it mainly focuses on social aspects regarding the development and use of software in organisations, it also deals with programming and algorithms and, therefore, contains elements of mathematical and physical sciences. In addition, insights from the humanities are as important for this discipline, although this is not always recognised or valued. Many papers, books and articles have been written on humanities computing, that is, the computer-based study of various humanities disciplines. However, not that much is available on, what the author would like to call, 'Humanities-enriched Information Systems', meaning a humanities approach and exploration of various aspects of computing subjects (Kroeze 2009).

Some Information and Communication Technology (ICT) disciplines, such as Information Science, may be regarded as the humanities branch of ICT, because they developed out of humanities disciplines associated with systems development for knowledge representation such as taxonomies and classification systems. However, humanities approaches are also present and embedded in other branches of ICT. One outstanding example is the current upsurge in the study and use of 'ontologies' in information systems. Ontology has traditionally been (and still is) a philosophical discipline that studies the nature of existence. In IS, however, ontologies refer to subsets of reality and how knowledge about these entities may be represented electronically. 'Ontologies are used to capture knowledge about some domains of interest. An ontology describes the concepts in the domain and also the relationships that hold between those concepts' (Horridge 2009:10).

This article gives an example from Biblical Hebrew (BH) grammar to illustrate a typical IS ontology. The idea for this originated in a study regarding an XML Schema of Hebrew syntax used to ensure consistent tagging of the Hebrew Bible text (Kroeze 2006). The next section gives an overview of this schema and its underlying taxonomy. Links between XML schemas and ontologies are discussed, and a brief overview of the use of ontologies in ICT and computational linguistics is then presented before suggesting an ontology of syntactic functions in BH and its possible uses. Consistency checking and visualisation of a skeleton ontology are implemented as examples of the benefits of an ontological approach.

The research is an interpretive, qualitative study. An interpretivist approach is more suitable for describing both the phenomena of syntax and ontologies because both are cultural products of society. IS ontologies capture and formalise subjective realities, while syntactical systems reflect the theoretical assumptions of various linguistic schools. The research strategy is 'design and creation' since the suggested BH taxonomy and ontology may be regarded as constructs or artifacts which form the main focus and contribution of the research (Oates 2006:291-296, 108-109).

top

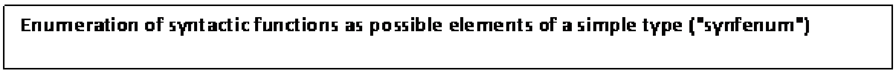
2 Taxonomy and XML Schema of Hebrew syntax

This section discusses an XML Schema used in the syntactic tagging of the Hebrew text in Genesis 1 and a more detailed underlying syntactic taxonomy that may be used to analyse the syntax of BH texts. This marked-up text of Genesis 1 was used as a databank in a thesis on the text data mining of linguistic data (Kroeze 2008). Syntactic functions refer to the formal, grammatical roles and relations in clauses. 'Syntax describes the form of clauses and sentences. The syntactic function of an element in a clause is the formal relation of that element to the other elements' (Kroeze 2000a:99). The concept of syntactic function is used as an umbrella term for elements usually called 'subject, direct object, indirect object, adjunct, copulative predicate, etc.' (Van der Merwe, Naudé and Kroeze 1999:239). According to Dik (1997a:26), syntactic functions 'specify the perspective from which a State of Affairs is presented in a linguistic expression'.

The XML Schema was created using the built-in functionality of Visual Studio.Net 2003 (VS.Net 2003). The structure of an XML document is represented by its schema. VS.Net 2003 was used because the XML functionality is not available in Visual Basic 6. VS.Net 2005 allows one to automatically create an XML Schema, but not to use it directly to validate XML databases. VS.Net 2003, however, facilitates both automatic creation and direct validation (using an option on the XML menu). Although the basic schema for the XML database of linguistic data was created automatically, a simple type and enumeration of syntactic tags were coded manually and added to the schema. A 'simple type' is a user-defined type, which enables the programmer to create custom-made types that reflect his/her exact requirements (Deitel and Deitel 2006:919-921). One may create a type to define a list (enumeration) of all possible values of syntactic functions. An enumeration is 'a set of values that a data item can select from' (Holzner 2004:213). The essential section of the schema is shown in Table 1 (for complete details, see Kroeze 2008:113-119). The schema was then used to test the XML database of Genesis 1:1-2:3, and this procedure revealed some inconsistencies in the tagging, for example with regard to the use of square brackets to indicate embedded clauses. After correcting these tagging errors the validation was successful.

Table 1 Syntactic sections from an XML Schema used to validate an XML database of Genesis 1:1 - 2:3

...


Enumeration of syntactic functions as possible elements of a simple type ("synfenum")

```
<xs:simpleType name="synfenum">
<xs:restriction base="xs:string">
<xs:enumeration value="Subject"/>
<xs:enumeration value="Predicate"/>
<xs:enumeration value="Main verb"/>
<xs:enumeration value="Transitive verb"/>
<xs:enumeration value="Intransitive verb"/>
<xs:enumeration value="Preposition verb"/>
<xs:enumeration value="Copulative verb"/>
<xs:enumeration value="Copula"/>
<xs:enumeration value="Copula-predicate"/>
<xs:enumeration value="Complement"/>
<xs:enumeration value="Object"/>
<xs:enumeration value="Object clause"/>
<xs:enumeration value="Object cluster"/>
<xs:enumeration value="IndObj"/>
<xs:enumeration value="Copula-predicate"/>
<xs:enumeration value="Adjunct"/>
<xs:enumeration value="Attribute"/>
<xs:enumeration value="Disjunct"/>
<xs:enumeration value="Interjection"/>
<xs:enumeration value="Modal word"/>
<xs:enumeration value="Discourse marker"/>
<xs:enumeration value="Dislocative"/>
<xs:enumeration value="Addressee"/>
<xs:enumeration value="Conj"/>
<xs:enumeration value="Co-ordinate conjunction"/>
<xs:enumeration value="Subordinate conjunction"/>
<xs:enumeration value="Relative particle"/>
<xs:enumeration value="-"/>
</xs:restriction>
</xs:simpleType>
...
```

Using simple type *synfenum* to validate syntactic function elements

```
...
<xs:element name="level4" minOccurs="0" maxOccurs="unbounded">
<xs:complexType>
<xs:sequence>
<xs:element name="leveldesc" type="xs:string" minOccurs="0" />
<xs:element name="phrase1" type="mstns:synfenum" minOccurs="0" />
<xs:element name="phrase2" type="mstns:synfenum" minOccurs="0" />
<xs:element name="phrase3" type="mstns:synfenum" minOccurs="0" />
<xs:element name="phrase4" type="mstns:synfenum" minOccurs="0" />
<xs:element name="phrase5" type="mstns:synfenum" minOccurs="0" />
</xs:sequence>
</xs:complexType>
</xs:element>
```

This schema, which basically is merely a list of syntactic functions occurring in the tagged text, is based on a more detailed taxonomy of Hebrew syntax, shown in Table 2. The definitions contained in this taxonomy should be regarded as a reference system, built by the author on nuggets of syntactic information mined from various sources. This taxonomy may be regarded as the creation of a reality (a classification system) that occurred within the author himself, 'primarily through the use of past experience, personal knowledge, and thinking', which is typical of an agile approach in scientific endeavours (Brown, Nerur and Slinkman 2004:4141). It was used in an introductory study manual on Hebrew grammar and syntax (Kroeze 2000a:330-334), as well as in various applications, for example in Kroeze 2000b, Kroeze 2002 and Kroeze 2008. In Table 2 below, those syntactic functions that were actually used in the Genesis 1:1-2:3 XML databank are marked with an asterisk.

Table 2 Taxonomy of syntactic functions in Biblical Hebrew (Kroeze 2000a, 2000b, 2002, 2008)

	Name	**Definition**
1.	Subject*	The subject is that element in a clause which **determines** the **person**, **gender** and **number** of the **main verb**, or the **gender** and **number** of the **copula-predicate** if this is a noun, adjective or participle.[1] It is that part of the clause about which the predicate proclaims something.[2] A noun, noun phrase, verb phrase or a clause can serve as subject. Rarely, even a preposition phrase may serve as subject.[3]
2.	Predicate	The predicate is that element in a clause which is **governed by the subject** (in terms of person, gender and number) and which tells something about the subject.[4] It consists of a main verb with or without complements and adjuncts.[5] It can also consist of a copula and copula-predicate. In BH the copula is often not expressed.[6]
2.1	Main verb*	The main verb is a verb which functions as **the main element of the predicate**.[7] The following kinds of verbs can function as main verbs:[8]
2.1.1	Transitive verb	A transitive verb **takes** or supposes a noun or noun phrase as **direct object**.[9]
2.1.2	Intransitive verb	An intransitive verb **does not** (and cannot) **take a direct object**.[10]
2.1.3	Preposition verb	A preposition verb **takes a preposition phrase as complement**.[11]
2.1.4	Copulative verb*	A copulative verb (*haya*: *is*, *was*, *were*, etc.) **takes** a noun, noun phrase, adjective, adjective phrase, adverb, adverb phrase or preposition phrase as **complement**. (*haya* is often omitted in BH.)[12]
2.2	Copula	The copula is that element in a clause that **connects the subject and copula-predicate**. In BH the particles *yesh, ayin/eyn*, the independent personal pronouns and the copulative verb *haya* can serve as copula, but it is often omitted.[13]
2.3	Copula-predicate*	The copula-predicate is **the complement of the copula**. (When the copula is omitted, the copula-predicate forms the whole predicate.) A noun, noun phrase, adjective, adjective phrase, participle, participle phrase,[14] adverb, adverb phrase or preposition phrase can serve as copula-predicate.[15]
2.4	Complement*	A complement is an **obligatory, non-verbal element in the predicate** which is **selected by the verb**.[16]
2.4.1	Direct object (Object*) Object clause* Object cluster*	The direct object is the **complement of an active, transitive verb**. In a passive transformation the object of the active clause becomes the subject of the passive clause. A noun, noun phrase, verb, preposition phrase or even a clause can serve as object. An **object clause*** is a clause that functions as a direct object. An **object cluster*** is a group of clauses that functions as a direct object.
2.4.2	Indirect object (IndObj*)	The indirect object is the **third argument or second complement of the main verb**. It can occur only if there is also a direct object. In BH the indirect object cannot be transformed into the subject of a passive clause - compare, however, this phenomenon in English: '*We* were given the book'. A pronominal suffix, noun, noun phrase or preposition phrase can serve as indirect object. The indirect object follows the direct object in unmarked word order, except when the indirect object is a preposition with suffix.[17]
2.4.3	Other complements	These are **other obligatory non-verbal elements in the predicate**, for example the **complements of** certain intransitive verbs such as **verbs of being full, living, moving**. These complements cannot become the subject of a passive clause. A noun, noun phrase or preposition phrase can serve as complement.[18]
2.4.4	Copula-predicate*	See 2.3.
2.5	Adjunct*	An adjunct is an **optional, non-verbal element in the predicate**. It is added to, but not selected by the verb. It gives optional, additional information. It can be omitted without changing the classification of the verb, without making the clause ungrammatical or senseless, and without changing the meaning of the clause radically. An adverb, adverb phrase, noun, noun phrase, preposition phrase, as well as an adverbial clause, can serve as adjunct.[19]
3.	Attribute*	An attribute (or adjectival modifier) is a word, phrase

		or clause that **qualifies a noun**. An adjective, adjective phrase, noun, noun phrase, adverb, adverb phrase, numeral, preposition phrase or relative clause can serve as attribute.[20]
4.	Disjunct*	These are elements which are **not connected with any of the elements in the clause**, but which are **loosely connected to the whole clause**. They may also be called *clause border adjuncts* (Afrikaans: 'sinsrandadjunkte'), or *sentence adjuncts* or *extra-clausal constituents*.[21]
4.1	Interjection/Modal word	An interjection is an **isolated element** which is not connected to any of the other elements in the clause, for example an exclamation particle.[22]
4.2	Discourse marker	A discourse marker is an isolated element which gives prominence to the sentence in the wider context.[23]
4.3	Dislocative	Any element of a clause may be marked as the topic of the clause by moving it from its usual place to precede the rest of the clause. It may even be separated from the rest of the clause by the *vav* conjunction. Its empty place in the clause may be filled by a pronoun or adverb referring to the dislocated element.[24]
4.4	Addressee	The addressee is the person to whom a clause is addressed. It is usually a common noun or proper noun. In BH the addressee is **often marked by the article *ha-*** if it is a common noun.[25]
5.	Conjunction (Conj*)	A conjunct is a particle which connects clauses or elements in clauses.[26]
5.1	Co-ordinate conjunction	A co-ordinate conjunction **connects two main clauses**. Co-ordinate conjunctions are also used to **connect elements** in a phrase or clause.[27]
5.2	Subordinate conjunction	A subordinate conjunction **connects a main clause with a subordinate clause**.[28]
5.3	Relative particle	The relative particle in BH is basically only a conjunction, but it can also fulfil a syntactic function in the relative clause. It connects the whole relative clause as an attribute to one of the nouns in the main clause. (All the other conjunctions are unconnected with regard to the subordinate clause, and they connect this whole clause as one of the syntactic elements in the main clause, for example subject, object, adjunct.) The relative particle can also be used independently, that is, without an antecedent, and such a relative clause fulfils one of the syntactic functions in the main clause, for example subject, object, etc.[29]

[1] *Cf. Gesenius et al. (1976:462-467).*

[2] *Cf. Waltke and O'Connor (1990:71).*

[3] *Cf. Waltke and O'Connor (1990:69-70).*

[4] *Cf. Waltke and O'Connor (1990:71).*

[5] *Cf. Waltke and O'Connor (1990:69, 169); Van der Merwe et al. (1999:60-62).*

[6] *Cf. Jöuon and Muraoka (1991:564-577); Waltke and O'Connor (1990:71).*

[7] *Cf. Du Plessis (1982:66-69); Van der Merwe et al. (1999:159-160, 165, 168).*

[8] *Cf. Du Plessis (1982:78).*

[9] *Cf. Van der Merwe et al. (1999:242-243, 246, 367); Waltke and O'Connor (1990:694): 'a verb that (usually) governs a (direct) object'.*

[10] *Cf. Van der Merwe et al. (1999:360), Du Plessis (1982:80), Waltke and O'Connor (1990:691): 'a verb that (usually) does not govern an object'.*

[11] *Cf. Waltke and O'Connor (1990:163, 165, 169-170, 221-222, 240, 275, 606, 690); Du Plessis (1982:82-84); Gesenius et al. (1976:378-384).*

[12] *Cf. Du Plessis (1982:85-86); Waltke and O'Connor (1990:131, 690).*

[13] *Cf. Waltke and O'Connor (1990:71-72, 131, 228, 297, 690).*

[14] *The participle is regarded as a verbal adjective (Gesenius et al., 1976:355-362; Waltke and O'Connor, 1990:612, 624). Morphologically and syntactically, it behaves primarily like an adjective.*

[15] *Cf. Du Plessis (1982: 86); Jöuon and Muraoka (1991:562); Van der Merwe et al. (1999:234, 356); Waltke and O'Connor (1990:71).*

[16] *Cf. Van der Merwe et al. (1999:241-244, 351, 355); Waltke and O'Connor (1990:163).*

[17] *Cf. Gesenius et al. (1976:369, 370); Jöüon and Muraoka (1991:442, 487, 490); Van der Merwe et al. (1999:173, 174, 240, 254, 255, 275, 359, 368); Waltke and O'Connor (1990:169, 193, 206).*

[18] *Cf. Gesenius et al. (1976:369-372); Jöüon and Muraoka (1991:455-461); Van der Merwe et al. (1999:244); Waltke and O'Connor (1990:173-177).*

[19] *Cf. Du Plessis (1982:97-103); Van der Merwe et al. (1999:241, 244-245, 351); Waltke and O'Connor (1990:163, 169-173; 689).*

[20] *Cf. Dik (1997a:151); Du Plessis (1982:48-58); Gesenius et al. (1976:414-419, 423-437); Van der Merwe et al. (1999:57, 229, 232-233, 266-270, 354); Waltke and O'Connor (1990:255-260, 689).*

[21] *Cf. Du Plessis (1982:100); Dik (1997b:379-407). 'In linguistics, a **disjunct** is a type of adjunct that expresses information that is not considered essential to the sentence it appears in, but which is considered to be the speaker's or writer's attitude towards, or descriptive statement of, the propositional content of the sentence.... More generally, the term **disjunct** can be used to refer to any sentence element that is not fully integrated into the clausal structure of the sentence. Such elements usually appear peripherally (at the beginning or end of the sentence) and are set off from the rest of the sentence by a comma (in writing) and a pause (in speech). A specific type of disjunct is the **sentence adverb** (or sentence adverbial), which modifies a sentence, or a clause within a sentence, to convey the mood, attitude or sentiments of the speaker, rather than an adverb modifying a verb, an adjective or another adverb within a sentence' [Disjunct (linguistics), 2007].*

[22] *Cf. Van der Merwe et al. (1999:309-311, 334-335); Waltke and O'Connor (1990:674-685).*

[23] *Cf. Van der Merwe et al. (1999:328-333).*

[24] *Cf. Gesenius et al. (1976:457-458); Jöüon and Muraoka (1991:586-588); Van der Merwe et al. (1999:339); Waltke and O'Connor (1990:128-129).*

[25] *Cf. Van der Merwe et al. (1999:189, 249).*

[26] *Cf. Van der Merwe et al. (1999:294-305).*

[27] *Cf. Van der Merwe et al. (1999:294); Waltke and O'Connor (1990:648-655).*

[28] *Cf. Waltke and O'Connor (1990:623-646).*

[29] *Cf. Waltke and O'Connor (1990:330-340).*

This taxonomy is one of the main building blocks for the suggested IS ontology of Hebrew syntax. The next section will give more information on the characteristics of IS ontologies, which is another important fundamental part for this study**.**

top

3 Ontologies in ICT

It is important to differentiate clearly between three closely related concepts, some of which have already been used in the discussion above, namely thesaurus, taxonomy and ontology (Gilchrist 2003). A thesaurus is a mere list or vocabulary of associated concepts that may be categorised according to their meanings. The XML Schema of syntactic functions, referred to above, may be regarded as a computerised version of an uncategorised thesaurus.

A taxonomy is a thesaurus that has been classified according to a scientific theory (Gilchrist 2003:7, 10). It may be viewed as a controlled vocabulary that involves relationships between the concepts contained in it (Lambe 2007:6). The system of BH syntax underlying the XML Schema, referred to above, may be considered as a taxonomy since it also defines the relationships between the syntagms, for example that a subject governs a verb, that a direct object is the complement of a transitive verb, etc.

An ontology refers to the knowledge of a subset of reality. It is a knowledge representation of 'a particular domain of knowledge' (Zúñiga 2001:187) that is shared and agreed upon by a certain community of scientists (Gilchrist 2003:7, 13). Since an ontology is a representation of the knowledge of its builder it is also limited to his/her particular view of the subset of reality (Buchholz 2006:694-695). As such, there is not much of a difference between a taxonomy and an ontology. However, a fourth concept is that of *formal* ontology, which is an ontology that has been enriched with inference rules and axioms, using description logics and artificial intelligence. It is machine readable and can be used to reason about the ontology. Ontologies in the IS realm usually refer to formal ontologies. Zúñiga (2001:187) defines an information systems ontology as a 'formal language designed to represent a particular domain of knowledge'.

The ontology suggested below is indeed a formal ontology; the concepts and relationships have been extracted from standard BH grammars (such as Gesenius, Kautzsch and Cowley 1976; Jöüon and Muraoka 1991; Van der Merwe *et al.* 1999; Waltke and O'Connor 1990), implying that a large community of BH grammarians should agree on the taxonomy (if not, adjustments could of course be made in another cycle of this research endeavour). References to these textbooks are provided in order to facilitate confirmability of the interprevist work, but it still is necessary that members of the BH linguistic community check the proposed system to ensure a jointly constructed reality (Oates 2006:294-295). The rules and axioms are deduced from the taxonomy and can be built into the ontology by the use of Protégé 4, a software tool that allows semi-automatic creation of IS ontologies. The ontology is machine readable and should allow computerised reasoning about itself and areas of application. The ontology is also the creation of an artifact that formalises a subset of reality: 'In information science, an ontology refers to an engineering artifact, constituted by a specific vocabulary used to describe a certain reality' (Fonseca 2007).

Fonseca (2007) differentiates between ontologies *for* and *of* information systems. While ontologies for information systems are part and parcel of the systems themselves, ontologies of information systems are used on a higher conceptual level to 'support the creation of modeling tools'. Ontologies of IS are 'the creation of ontologies that study the information system as an object per se with the objective of creating better modeling tools'. The ontology of BH syntax, suggested in this article, is, however, an ontology *for* information systems. The possible uses of this ontology will be discussed later on. The next section will discuss the links between XML schemas and IS ontologies in more depth.

top

4 Links between XML schemas and ontologies

IS ontologies may be used to validate conceptual schemas. Ontologies for IS 'are useful to ensure that the conceptual schemas (conceptual-modeling scripts) we create using the grammar are correct' (Fonseca 2007). Therefore, they may also be used to check the consistency of an existing XML Schema. While conceptual schemas focus on a specific application, an ontology is more general and could be shared by various applications; schemas ensure correct data, ontologies check logical consistency of underlying theories (Fonseca and Martin 2007:137).

An ontology Web language (OWL), such as Protégé 4, is an ontology language that facilitates the semi-automatic creation of a hierarchy of concepts, the definition of relations between these concepts and the checking of the ontology's consistency (Horridge 2009:10). It also allows logical reasoning using its built-in description logics.

OWLs use a resource description framework (RDF) to express sets of rules, referred to as ontologies, regarding domains of knowledge, to ensure its validity and correctness. RDF is a semantic Web technology, the main strengths of which are 'its simplicity, rigour and the use of URIs' (uniform resource identifiers) (Tummarello, Morbidoni, Puliti and Piazza 2008:469). Every element (class, property and instance) in the ontology is regarded as a resource which is identified by its unique URI and data value (literal). The resources are regarded as nodes connected by labelled edges to form a semantic network. RDF offers improved knowledge representation over hierarchical XML schemas, because 'RDF is a graph-based data model, ... a super-case of the tree-based XML model' (Tummarello *et al.* 2008:469). Ontologies are a further improvement in terms of information and knowledge management, since they contain additional semantics. 'While the graph structure of RDF provides a very suitable data model for annotations, it is the use of annotation vocabularies with well specified semantic meanings (ontologies) that possibly enables the most interesting results' (Tummarello *et al*. 2008:475). The additional built-in semantics allows more advanced reasoning and automated processing.

According to Ferdinand, Zirpins and Trastour (2004:354), existing XML schemas may be used as a basis for formal ontologies, which are needed to create the semantic Web, allowing 'software agents to understand, share and reason about data'. The idea behind the semantic Web is to enhance the current Web into a form that can be processed intelligently by machines (Antoniou and Van Harmelen 2004:3-4). This article is such an attempt to semi-automatically upgrade or 'bootstrap' an existing schema into a skeleton information systems ontology of BH syntax (compare Ferdinand *et al*. 2004, who propose an automated process of bootstrapping). This may be regarded as a small step towards the integration of and improved access to linguistic information systems.

top

5 Use of IT ontologies in (computational) linguistics

Ontologies are used in knowledge management endeavours to enhance knowledge representation, and its storage, search and communication (Buchholz 2006:694). Although linguistic concepts are often used in the building of IS ontologies, not that much has been done on the creation of ontologies for the purpose of linguistics itself, especially when one starts searching for ontology literature on syntax and semantics. According to Buchholz (2006:700), WordNet is a lexicon or terminological ontology and not a formal ontology because it is not axiomatised.

Tummarello *et al*. (2008), however, do propose a new textual encoding system using ontologies in order to overcome the limitations of XML tagging, such as the unsatisfactory handling of overlapping hierarchies and embedded text. Elements of various levels of language, such as structure and grammar, are encoded as instances of classes. This approach allows improved cooperative encoding, consistency checks, and enhanced searching facilities.

The following benefits of an ontological approach are indicated by Tummarello *et al.* (2008:467-468, 474-476):

- Enabling 'collaborative and distributive textual encoding'
- Allowing 'ontology-based reasoning in text processing and querying'
- Solving the encoding problem of overlapping hierarchies and 'cross concern annotations'
- Facilitating enhanced searching and querying, even across divergent levels of annotation
- Enhancing filtering and formatting
- 'Lowering the complexity of advanced textual document encoding'
- 'Increasing interoperability and reusability'
- Merging or joint processing of distributed documents
- Validity checking and inference of new information
- Reuse of previous encoded results.

Tummarello *et al*. (2008:470-472) provide an example of an RDF model of text. Words, punctuation and other printable elements constitute the nodes or resources in the network. Using further annotations the natural word order is encoded using a linked list - each element has a property that points to the next element. In addition, clauses and sentences are encoded by pointing to their first and last symbols. Even non-contiguous and interleaving elements, such as dependent and independent clauses, can be annotated using the relevant URIs. The flexibility of the model is shown by a complex annotation bundle in which 'different overlapping hierarchies and cross hierarchy (concern) annotations coexist and interrelate'. A formal ontology is used to regulate and validate the interconnection of resources.

Some other examples of the application of ontologies in linguistics are Caracciolo (2006) who proposes the use of an ontology to access the content of a textbook; Oltramari and Vetere (2008) on an Italian machine-readable dictionary; Oltramari and Stellato

(2008) discussing sharing and integration of vocabularies from different computational ontologies; Guarino's (1998) discussion of ISA overloading in upper-level lexical ontologies; Dahlgren (1995) who uses linguistic constraints for syntactic disambiguation, but does not provide a taxonomy of syntactic functions; and Farrar (2005) who differentiates between the various layers of linguistic analysis but does not describe syntactic functions of clauses in detail, and invites communities of practice extensions (COPEs) to complement their proposed ontology with information in language-specific domains. This article could be a step in the direction of creating a sub-ontology for BH syntax.

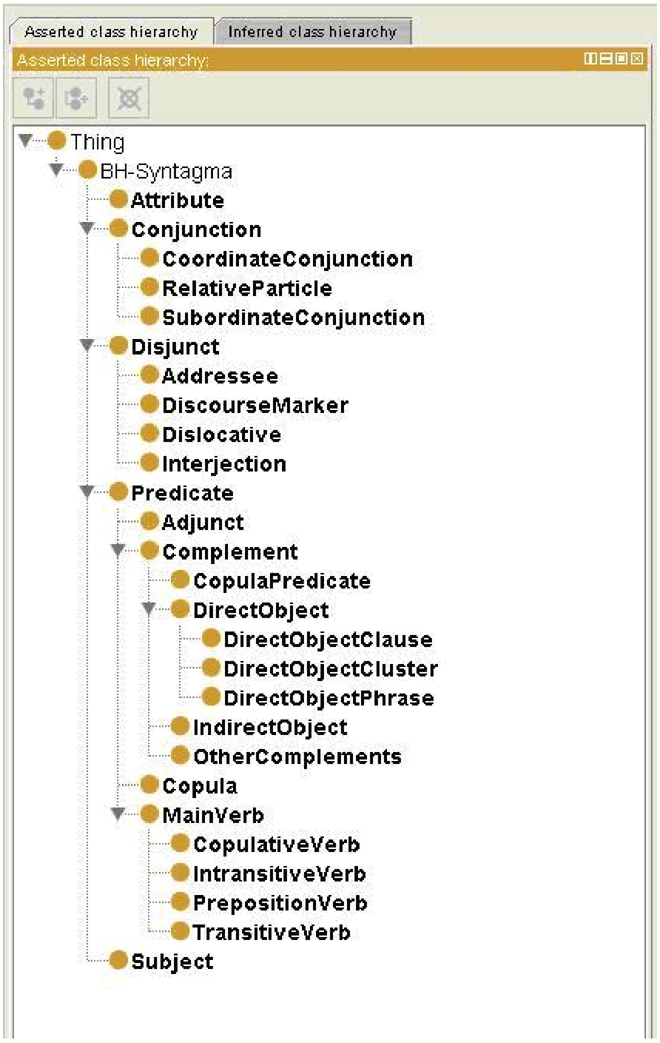
top

6 Suggesting an ontology of syntactic functions in Biblical Hebrew

Protégé 4 is free and open source software that facilitates the semi-automatic building of IS ontologies. This software may be used to semi-automatically create an ontology of BH syntax. Such an ontology is a form of knowledge representation since it captures and organises existing information in a machine-readable, logical and coherent system. It describes the syntactic entities of BH and the structural relationships between them. Although the concepts are organised hierarchically, other connections are allowed using OWL's description logics. The BH syntax ontology will, therefore, allow consistency checking and logical reasoning about the system itself, and may be implemented in linguistic information systems, for example to ensure correct tagging of syntagms. An ontology of syntax will consist, like any other ontology, of individuals, properties and classes. Individuals may eventually be used to identify specific instances of syntactic classes in text. A class (also called a concept) is a set of individuals, for example, *DirectObject* is the collection of all instances of direct objects. Properties describe the relations between two individuals, linking them together, for example 'Subject *governs* Verb'. The inverse property would be 'Verb *isGovernedBy* Subject'. A property that has a single value is functional. In description logics properties are referred to as roles. They are also sometimes called attributes. If the classes are organised into a hierarchy of super- and subclasses (e.g. Predicate - MainVerb - TransitiveVerb), it becomes a taxonomy. Subclasses are finer specialisations of a superclass. The built-in reasoner of Protégé 4 can compile these subsumption relationships automatically. Descriptions 'specify the conditions that must be satisfied by an individual for it to be a member of a class' (Horridge 2009:9-12).

Building the BH syntax ontology starts by defining classes of syntagms (classes are indeed the main building blocks of OWL ontologies; see Figure 1). The class hierarchy cannot yet be considered as an ontology - it is still merely a taxonomy or skeleton ontology. All the sub-classes on the same level of the hierarchy are declared as disjoint, meaning that, for example, a subject cannot be a predicate, neither can a predicate be a subject. An individual of a class can only be an instance of that class. In language, of course, a word may often be used in various ways, but a specific occurrence of a word (e.g. the third word in the fourth sentence of the first paragraph of a specific book) can only be an instance of one specific class. OWL and Protégé assume an open world and therefore one cannot assume that an individual is not a member of class B if it has been declared as a member of class A. All instances of a subclass are per implication also instances of the superclass, for example, all verbs that are classified as transitive verbs are per implication main verbs, and predicates, and syntagms (Horridge 2009:15-23).

Figure 1 Taxonomy of BH syntax implemented as a skeleton ontology using Protégé 4



- Thing
  - BH-Syntagma
    - Attribute
    - Conjunction
      - CoordinateConjunction
      - RelativeParticle
      - SubordinateConjunction
    - Disjunct
      - Addressee
      - DiscourseMarker
      - Dislocative
      - Interjection
    - Predicate
      - Adjunct
      - Complement
        - CopulaPredicate
        - DirectObject
          - DirectObjectClause
          - DirectObjectCluster
          - DirectObjectPhrase
        - IndirectObject
        - OtherComplements
      - Copula
      - MainVerb
        - CopulativeVerb
        - IntransitiveVerb
        - PrepositionVerb
        - TransitiveVerb
    - Subject

Although more advanced functions are available in Protégé 4, their implementation fall outside the scope of this article. A more complete form of the ontology should be created and discussed in follow-up work. The next section explores possible uses of such an ontology.

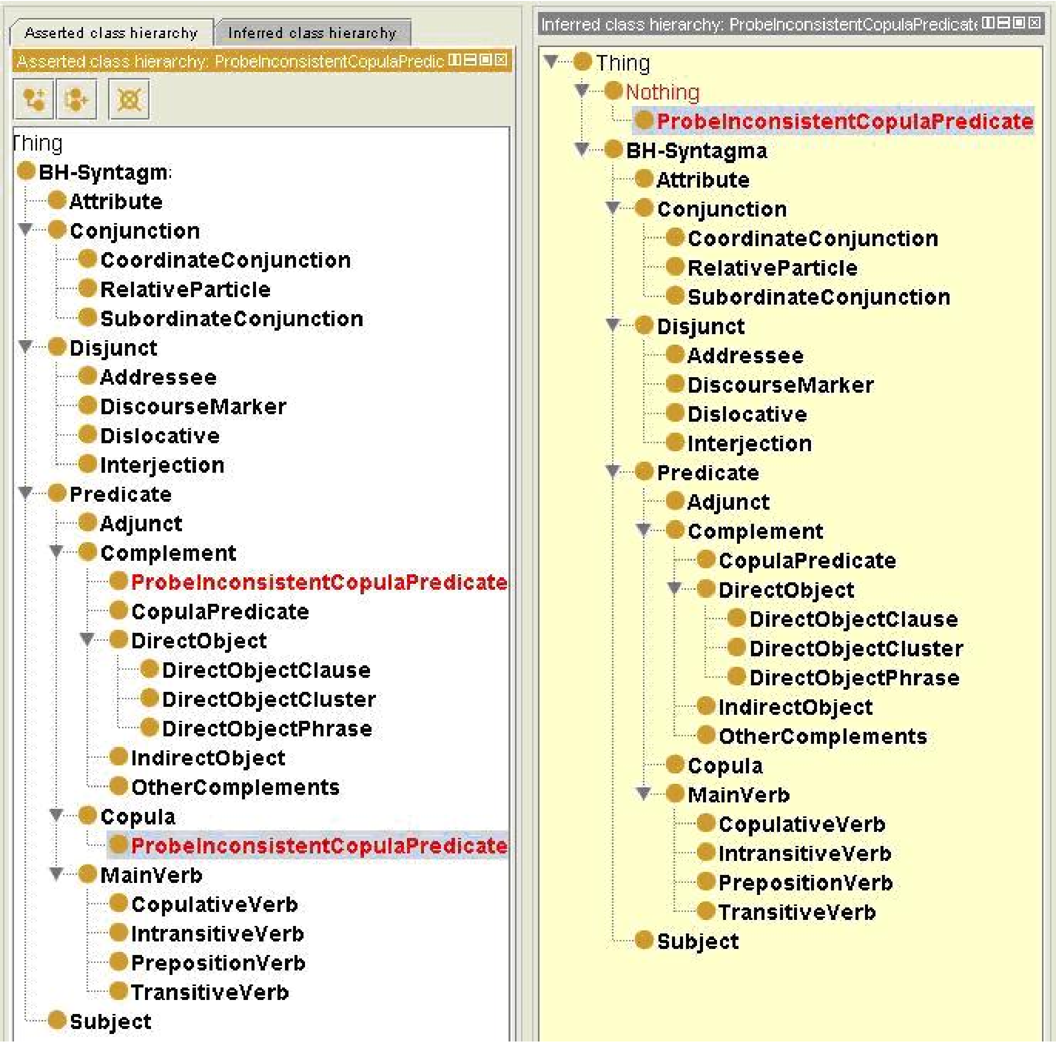
top

7 Possible uses of such an ontology

According to Buchholz (2006:695) ontologies are used by computerised agents to simulate human thinking to some extent. In information systems, ontologies replace the function of the human brain by creating the illusion of storing, analysing and understanding information. 'While never pretending to duplicate exactly the workings of the human imagination or experience, ontologies attempt to capture conceptually the rational building blocks of the mind by modeling our knowledge of reality' (Buchholz 2006:695). Domain-specific ontologies are used by software solutions in various ways, for example in efficient information retrieval, to do automatic translations, to model enterprises and to tag information on the semantic Web (Gilchrist 2003:13-14). A BH syntax ontology could be used to intelligently retrieve formally related constituents in a text. This information could be used to study the structures of typical syntactic frameworks. An intelligent tagging program could use its knowledge, that a subject governs the person, number and gender of a finite verb, to suggest all possible subjects of a main verb in a clause. If enough knowledge could be built into the ontology using description logics, it could even be used to assist translators by suggesting syntactically correct alternative renderings which they might not have thought about.

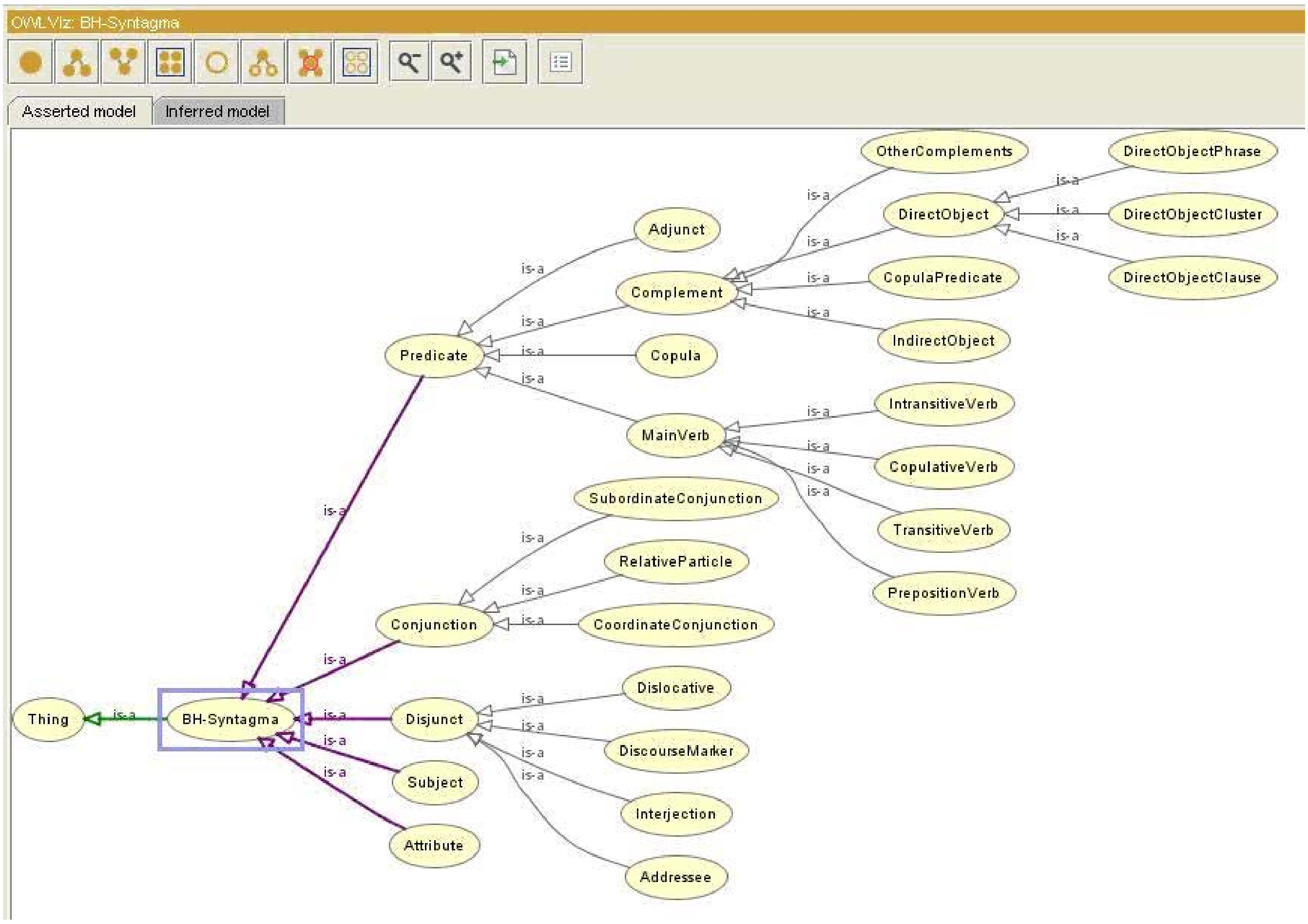
Another functionality of a formal ontology is that one could use it to check 'whether or not one class is a subclass of another class' (subsumption testing) (Ferdinand *et al*. 2004:357; Horridge 2009:49-54). This is done by using a reasoner or classifier that computes the inferred ontology class hierarchy. The reasoner can also check the consistency of the ontology (if a class cannot have any instances, it is inconsistent). This facility has been tested on the skeleton BH syntax ontology. Classifying the ontology that has been asserted so far results in the inferred hierarchy, implying that all definitions so far have been consistent. Any inconsistent classes would have been marked in red (for example, if one would have tried to declare Copula-Predicate both as a subclass of Copula and Complement). This is demonstrated in the tool by *ProbeInconsistentCopulaPredicate* (see Figure 2 below). However, making ProbeInconsistentCopulaPredicate a subclass both of Predicate and Complement (as is indeed suggested by the original taxonomy underlying the XML Schema and OWL ontology) seems not to give a problem, because Predicate and Complement are not marked as disjoint. Yet, in the inferred class ProbeInconsistentCopulaPredicate is shown only in one place (subclass of Complement).

Figure 2 Inconsistent classes discovered and highlighted by Protégé 4



Protégé 4 also has a built-in visualisation tool called OWL Viz, which is a very useful facility that may be used to render a graphic overview of an implemented ontology. This facility has indeed been tested for the proposed skeleton ontology and the resulting visualisation provides a compact, graphical synopsis of the five hierarchical layers of syntagms (Figure 3).

Figure 3 Skeleton ontology of BH syntax visualised using Protégé 4's built in OWL Viz tool



top

8 Conclusion

Since an XML Schema does not facilitate the checking of the logic and consistency of a syntactic taxonomy of BH, the use of an IS ontology to fulfil this need was suggested. Some of the basic constructs of a formal ontology were implemented in Protégé 4 to serve as an example of how this could be done. This may be regarded as a 'skeleton ontology' which should be extended using description logics and OWL expressions (Ferdinand *et al*. 2004:357), some examples of which have been provided. The research made a contribution by differentiating the roles of IS ontologies, taxonomies and XML schemas in the information and knowledge management subject field. It also indicated the value of ontologies in terms of ensuring the quality and reliability of linguistic information systems. The suggested ontology may, therefore, be regarded as an example of a domain specific ontology (Buchholz 2006:699), the domain being linguistics and, more specifically, BH syntax. Once the syntax ontology is completed it may be used to ensure the quality and reliability of the system itself. It may then be implemented in a myriad of applications, for example to ensure correct tagging of Hebrew texts or to integrate various existing marked-up products. The author trusts that this research has been a small step towards Buchholz's (2006:700) dream: 'Ontologies will continue to play an important role in the development of large-scale, computer mediated, and global knowledge-management projects. Communicating knowledge within an organization, and among organizations worldwide, will be facilitated by ontologies, as they create a knowledge layer critical to the automated sharing and reuse of essential explicit knowledge.'

top

9 References

Antoniou, G. and Van Harmelen, F. 2004. *A semantic Web primer.* Cambridge, MA: MIT Press.

Brown, R., Nerur, S. and Slinkman, C. 2004. Philosophical shifts in software development. In: *Proceedings of the Tenth Americas Conference on Information Systems (AMCIS),* August 2004, New York, NY:4136-4143. [Online]. Available WWW: **http://aisel.aisnet.org/amcis2004/516** (Accessed 23 August 2009).

Buchholz, W. 2006. Ontology. In: Schwartz, D. (ed). *Encyclopedia of Knowledge Management.* Hershey, PA: IGI (Idea Group):694-702.

Caracciolo, C. 2006. Designing and implementing an ontology for logic and linguistics. *Literary and Linguistic Computing* 21:29-39.

Dahlgren, K. 1995. A linguistic ontology. *International Journal of Human-Computer Studies* 43:809-818.

Deitel, H.M. and Deitel, P.J. 2006. *Visual Basic® 2005: how to program,* 3rd edition. Upper Saddle River, NJ: Pearson.

Dik, S.C. 1997a. *The theory of functional grammar. Part 1. The structure of the clause* (edited by Kees Hengeveld), 2nd edition. Berlin: Mouton de Gruyter.

Dik, S.C. 1997b. *The theory of functional grammar. Part 2. Complex and derived constructions* (edited by Kees Hengeveld). Berlin: Mouton de Gruyter.

Disjunct (linguistics). 2007. *Wikipedia.* [Online]. Available WWW: **http://en.wikipedia.org/wiki/Disjunct_%28linguistics%29** (Accessed 24 November 2007).

Du Plessis, H. 1982. *Sintaksis vir eerstejaars.* Pretoria: Academica.

Farrar, S. 2005. *Modularizing GOLD.* [Online]. Available WWW: **http://linguistlist.org/gold/linguistics-ontology.org/documents/gold-modules.pdf** (Accessed 26 November 2009).

Ferdinand, M., Zirpins, C. and Trastour, D. 2004. Lifting XML Schema to OWL. In: *Proceedings of the 4th International Web Engineering Conference,* 26-30 July 2004, Munich, Germany: Springer. (Lecture Notes in Computer Science, 3140:354-358.)

Fonseca, F. 2007. The double role of ontologies in Information Science research. *Journal of the American Society for Information Science and Technology* 58(6):786-793. (Preprint.) [Online]. Available WWW: **http://www.personal.psu.edu/faculty/f/u/fuf1/publications/Fonseca_Ontologies_double_role_JASIST_2006.pdf** (Accessed 23 March 2009).

Fonseca, F. and Martin, J. 2007. Learning the differences between ontologies and conceptual schemas through ontology-driven information systems. *Journal of the Association for Information Systems,* 8(2):129-142 (Article 3). [Online]. Available WWW: **http://aisel.aisnet.org/jais/vol8/iss2/4** (Accessed 24 November 2009).

Gesenius, F.H.W., Kautzsch, E. and Cowley, A.E. 1976. *Gesenius' Hebrew Grammar,* 2nd English edition. Oxford: Clarendon.

Gilchrist, A. 2003. Thesauri, taxonomies and ontologies - an etymological note. *Journal of Documentation* 59(1):7-18.

Guarino, N. 1998. Some ontological principles for designing upper level lexical resources. In: *Proceedings of First International Conference on Language Resources and Evaluation,* ELRA - European Language Resources Association, May 28-30, 1998, Granada, Spain:527-534.

Holzner, S. 2004. *SAMS teach yourself XML in 21 days,* 3rd edition. Indianapolis, Indiana: SAMS.

Horridge, M. (ed.) 2009. *A practical guide to building OWL ontologies using Protégé 4 and CO-ODE tools, edition 1.2.* The University of Manchester. [Online]. Available WWW: **http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/resources/ProtegeOWLTutorialP4_v1_2.pdf** (Accessed 23 August 2009).

Joüon, P. and Muraoka, T. 1991. *A grammar of Biblical Hebrew. Vol. II, Part Three: syntax; paradigms and indices.* Roma: Editrice Pontificio Istituto Biblico. (Subsidia biblica 14/II.)

Kroeze, J.H. 2000a. *Semitic Languages 115: Study guide for SET115.* Potchefstroom: PU for CHE. (Study manual - Telematic Learning Systems: BA Theol.)

Kroeze, J.H. 2000b. *Semitic Languages 125: Study guide for SET125.* Potchefstroom: PU for CHE. (Study manual - Telematic Learning Systems: BA Theol.)

Kroeze, J.H. 2002. Developing a multi-level analysis of Jonah using *html.* In: Cooke, J. (ed). *Bible and computer: The Stellenbosch AIBI-6 conference. Proceedings of the Association Internationale Bible et Informatique 'From alpha to byte',*17-21 July 2000, University of Stellenbosch. Leiden: Brill:653-662.)

Kroeze, J.H. 2006. Building and displaying a Biblical Hebrew linguistics data cube using XML, *Israeli Seminar on Computational Linguistics (ISCOL),* Haifa, Israel, 29 June 2006. [Online]. Available WWW (seminar presentations): **http://mila.cs.technion.ac.il/english/events/ISCOL/2006/index.html** (Accessed 27 November 2009).

Kroeze, J.H. 2008. *Developing an XML-based, exploitable linguistic database of the Hebrew text of Gen. 1:1-2:3.* PhD(IT) thesis. University of Pretoria. (Unpublished.) [Online]. Available WWW: **http://upetd.up.ac.za/thesis/available/etd-07282008-121520/** (Accessed 23 August 2009).

Kroeze, J.H. 2009. *Information Systems and the humanities: a symbiotic relationship?* (Inaugural lecture, 13 Nov. 2009.) Vanderbijlpark: North-West University, Vaal Triangle Campus. (Vaal Triangle Occasional Papers: Inaugural lecture 5/2009. ISBN 978-1-86822-583-5.) (Scientific Contribution Series H.)

Lambe, P. 2007. *Organising knowledge: taxonomies, knowledge and organisational effectiveness.* Oxford: Chandos.

Oates, B.J. 2006. *Researching information systems and computing.* Los Angeles: Sage.

Oltramari, A. and Stellato, A. 2008. Enriching ontologies with linguistic content: an evaluation framework. In: *Proceedings of OntoLex 2008* (hosted by **Sixth international conference on Language Resources and Evaluation)**, Marrakech (Morocco). [Online]. Available WWW: **http://www.loa-cnr.it/Papers/enriching_onto_oltramari-stellato.pdf** (Accessed 25 November 2009).

Oltramari, A. and Vetere, G. 2008. Lexicon and ontology interplay in Senso Comune. In: *Proceedings of OntoLex 2008* (hosted by Sixth international conference on Language Resources and Evaluation), Marrakech (Morocco). [Online]. Available WWW: **http://www.loa-cnr.it/Papers/lexicon_oltramari-vetere.pdf** (Accessed 25 November 2009).

Tummarello, G., Morbidoni, C., Puliti, P. and Piazza, F. 2008. A proposal for textual encoding based on semantic Web tools. *Online Information Review* 32(4):467-477. [Online]. Available WWW: **http://www.emeraldinsight.com/1468-4527.htm** (Accessed 23 August 2009).

Van der Merwe, C.H.J., Naudé, J.A. and Kroeze, J.H. 1999. *A Biblical Hebrew reference grammar.* Sheffield: Sheffield Academic Press.

Waltke, B.K. and O'Connor, M. 1990. *An introduction to Biblical Hebrew syntax.* Winona Lake, Indiana: Eisenbrauns.

Zúñiga, G.L. 2001. Ontology: its transformation from philosophy to information systems. In: *Proceedings of the International Conference on Formal Ontology in Information Systems,* Ogunquit, Maine, USA, Volume 2001:187-197.

ISSN 1560-683X

Published by **InterWord Communications** for Department of Information and Knowledge Management,
University of Johannesburg