Concept-Based Readability Measurement and Adjustment for Web Services Descriptions

Pananya Sripairojthikoon, Twittie Senivongse

Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok, Thailand

pananya.s@student.chula.ac.th, twittie.s@chula.ac.th

***Abstract*—Web Services is a technology for building distributed software applications that are built upon a set of information and communication standards. Among those standards is the Web Services Description Language (WSDL) which is an XML-based language for describing service descriptions. Service providers will publish WSDL documents of their Web services so that service consumers can learn about service capability and how to interface with the services. Since WSDL documents are the primary source of service information, readability of WSDL documents is of concern to service providers, i.e., service descriptions should be understood with ease by service consumers. Providing highly readable service descriptions can then be used as a strategy to attract service consumers. However, given highly readable information in the WSDL documents, competitors are able to learn know-how and can copy the design to offer competing services. Security attacks such as information espionage, client impersonation, command injection, and denial of service are also possible since attackers can learn about exchanged data and invocation patterns from WSDL documents. While readability of service descriptions makes Web services discoverable, it contributes to service vulnerability too. Service designers therefore should consider this trade-off when designing service descriptions. Currently there is no readability measurement for WSDL documents. We propose an approach to such measurement so that service designers can determine if readability is too low or too high with regard to service discoverability, service imitation, and service attack issues, and then can consider increasing or lowering service description readability accordingly. Our readability measurement is based on the concepts or terms in service domain knowledge. Given a WSDL document as a service description, readability is defined in terms of the use of difficult words in the description and the use of words that are key concepts in the service domain. As an example, we measure readability of the WSDL document of E-commerce Web services, and experiment on redesigning of WSDL terms to adjust readability.**

***Keyword*—Concept Hierarchy, Ontology, Readability, Web Services**

Manuscript received August 28, 2013.

P. Sripairojthikoon is with the Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok 10330, Thailand. (e-mail: pananya.s@student.chula.ac.th).

T. Senivongse is with the Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok 10330, Thailand. (Corresponding author phone: +66 2 2186996; fax: +66 2 2186955; email: twittie.s@chula.ac.th).

I. INTRODUCTION

Web services is a technology for building distributed software applications. The building blocks are services which are software units that are built upon a set of information and communication standards. Among those standards is the Web Services Description Language (WSDL) which is an XML-based language for describing service descriptions. A WSDL document is defined by a service provider and used by service consumers in discovering service capability and establishing interaction between consumer-side applications and the Web service. The structure of a WSDL document (version 1.1) which describes what the service is capable of and what data are exchanged comprises the XML elements <types>, <message>, <portType>, and <documentation> [1].

Since WSDL documents are the primary source of service information, readability [2] of WSDL documents or Web services descriptions is of concern to service providers, i.e., service descriptions should be understood with ease by service consumers. Meaningful names should be given to the service interface, operations, input and output messages, and data. In addition, sufficient documentation should be provided regarding functional scope and limitation of use. Providing well-defined readable service descriptions can be used as a strategy by service providing organizations to attract service consumers.

Despite being desirable, readability of service descriptions has its downside. Other organizations can gather information from a service WSDL document to learn know-how and then copy the design to offer competing services. Also, published WSDL documents can provide security attackers with information like schemas of exchanged data, invocation patterns, and service location. Attackers may be able to guess other private operations. This leads to more serious attacks such as information espionage, client impersonation, command injection, and denial of service.

While readability of service descriptions makes Web services discoverable, it contributes to service vulnerability too. Service designers therefore should consider this trade-off when designing service descriptions. Currently there is no readability measurement for WSDL documents. We therefore aim to propose an approach to such measurement so that service designers can determine if readability is too low or too high with regard to service discoverability, service imitation, and service attack issues, and then can consider increasing or lowering service description readability accordingly.

We apply the concept-based readability measurement model proposed by Yan et al. [3] to the context of Web service descriptions. As the name implies, our readability measurement is based on the concepts or terms in service domain knowledge. Given a WSDL document as a service description, readability is defined in terms of the use of difficult words in the description and the use of words that are key concepts in the service domain. For example, if the service description contains simple words or closely-related terms within the domain, it should be easy to understand the functionality of the service from the service description. Here, service domain knowledge is described as ontology [4] that defines vocabulary of concepts and properties as well as their relationships, using an XML-based OWL language [5]. Readability assessment can be conducted by the quality assurance team or service designers who have knowledge of the service domain, and can be of several uses. The assessors can compare readability of their WSDL documents with that of the competing services and may evaluate if readability should be improved to attract more consumers. In a certain case, the assessors may consider adjusting the service descriptions if security issues are of concern. We also outline a method to increase and lower readability of service descriptions, and experiment on redesigning of WSDL terms of E-commerce Web services to adjust readability.

The rest of this paper is organized as follows. Section II discusses related work and Section III presents the concept-based document readability model. We propose a methodology to assess readability of WSDL documents and how to adjust it in Section IV. In Section V, we present an experiment on readability measurement for three Web services in the E-commerce domain for comparison, followed by an experiment to adjust readability of their WSDLs in Section VI. Section VII concludes the paper with a discussion of the approach and future outlook.

II. Related Work

By definition, readability means "the level of ease or difficulty with which text material can be understood by a particular reader who is reading that text for a specific purpose" [2]. Readability is dependent upon many characteristics of both the text and the readers, and its concept has been applied to many kinds of text material including books, technical documents, online documents, and Web pages. Many formulas for measuring text readability are available and most of them deal with only text features. That is, texts that use difficult words are more difficult to understand than those with simple words, and texts with long sentences and complex syntax are difficult to read.

Yan et al. [3] propose a different but interesting approach to measuring text readability in the context of online documents. They argue that not only domain experts but also average users are searching more and more for domain-specific information from online documents, particularly in the medical area, and these documents are of different readability level. However, traditional readability formulas are designed for general purpose texts and insufficient to deal with technical materials in a specific domain. Therefore, for the document ranking purpose, their model takes advantages of a traditional readability formula and domain knowledge to measure readability of domain-specific documents at the word level, i.e., it focuses on how the domain-specific terms in a document affect readability of the document. For example, if the document contains closely-related terms in the domain vocabulary, it should be more easily readable and comprehensible to readers of that domain. In their approach, the domain knowledge is represented as a concept hierarchy or ontology.

Zhao and Kan [6] argue that the ontology-based approach such as [3] has a limitation in that it requires expert knowledge which is still expensive and not readily available in most domains. They hence present an iterative computation on a resource-concept graph based on the intuition that readability of the domain-specific resources (or documents) and difficulty of domain-specific concepts provide accurate estimations of each other. The algorithm constructs a graph that represents what concepts are contained in a particular resource and what resources contain a particular concept. Then readability computation for the resources is based on a simple mutually recursive observation as follows: (1) a domain-specific resource *A* is less readable than another domain-specific resource *B* if *A* contains more difficult domain-specific concepts than *B*, and (2) a domain-specific concept *A* is more difficult than another domain-specific concept *B* if *A* is mentioned in less readable domain-specific resources than *B*. This approach is effective and less domain-dependent but it requires multiple resources in order to determine their readability, i.e., readability of any single resource cannot be determined on its own.

The research by Jatowt et al. [7] attempts to measure readability of Web pages based on link structure. It is motivated by their study on the correlation between readability of the source pages and that of the linked pages, i.e., there is a trend that Web pages would link to other pages with generally the same level of difficulty. They are also inspired by the TrustRank algorithm in which scores are propagated from good pages in an attempt to separate useful Web pages from spam. Therefore, their algorithm utilizes the Web link structure by propagating the readability scores of the source pages to the linked pages. This approach is useful since it becomes possible to measure the readability of Web pages that have little texts and to complement traditional readability measures which rely only on textual content.

It is seen that several research attempts propose different ways to determine the readability score of a document but none of them address measurement of WSDL documents readability. Since a WSDL document is self-contained and we should be able to determine its readability using its own content, we will adopt the concept hierarchy-based model by Yan et al. [3]. This model applies at the word level, meaning that it is based on concepts or terms within a document, rather than sentences or paragraphs; this suits well with the case of a WSDL document. Using this model, we then rely on the availability of the service domain ontology on the Internet. In the case that no domain ontology is available, we argue that an approach such as [8], which builds domain ontology using terms from the WordNet database [9], can be taken.

III. Concept-Based Document Readability Model

The concept-based readability model considers difficulty of terms in a WSDL document and two features of the WSDL document, namely document scope and document cohesion, with respect to the presence of domain terms in the WSDL document. As mentioned earlier, vocabularies of the service domain is formed as a concept hierarchy. The terms in the document which have a match in the concept hierarchy are regarded as domain terms; otherwise they are non-domain terms. The detail of the model components [3] is as follows.

A. Document Scope (DS)

Document scope is defined as the coverage of the domain concepts in the document. The coverage is viewed from two angles. First, the more the document contains domain terms, the less readable the document tends to be since the document is likely to contain a larger number of specific concepts. Second, the deeper the domain terms appear in the concept hierarchy, the more difficult the document is to read. The document scope *DS* of a WSDL document d_i can be computed by (1):

$$DS(d_i) = e^{-(\sum_{i=1}^{n} depth(c_i))} \quad (1)$$

where $depth(c_i)$ = depth of domain concept c_i in the WSDL document d_i, with regard to the concept hierarchy.

B. Document Cohesion (DC)

Document cohesion refers to how focused the text is on a particular topic. It can be computed by the semantic relatedness between the domain terms in the document which is reflected by the links (or shortest path) between them with respect to the given concept hierarchy. The more cohesive the domain terms in the document are, the more readable the document is. The document cohesion *DC* of a WSDL document d_i can be computed by (2)-(4):

$$DC(d_i) = \frac{\sum_{i,j=1}^{n} Sim(c_i, c_j)}{Number\ of\ Associations}, where\ n > 1, i < j \quad (2)$$

$$Sim(c_i, c_j) = -log \frac{len(c_i, c_j)}{2D} \quad (3)$$

$$Number\ of\ Associations = \frac{n(n-1)}{2} \quad (4)$$

where $len(c_i,c_j)$ = shortest path between c_i and c_j in the concept hierarchy,

D = maximum tree depth in the concept hierarchy, and

n = total number of domain concepts in the WSDL document d_i.

C. Simplified Dale-Chall's Readability Index (DaCw)

A well-known traditional readability formula is the Dale-Chall's readability index [10]. This index sees that, the length of the sentences in a document and the difficulty of words correlate with the difficulty of reading material. Since the concept-based readability model measures readability at the word level, sentence-level complexity is not applicable and hence only word difficulty is considered. To determine the difficulty of words, words in the document are identified as either familiar or unfamiliar words. That is, they are familiar words if they can be found in the Dale list of approximately 3,000 familiar words. Otherwise, they are unfamiliar, and hence difficult, words. The simplified version of the Dale-Chall's readability index *DaCw* of a WSDL document d_i can be computed by (5):

$$DaCw(d_i) = PDW \quad (5)$$

where *PDW* = percentage of difficult words in the WSDL document d_i (i.e., number of difficult words divided by number of words and multiplied by 100).

We will use the word-level model above to measure readability of WSDL documents since contents of WSDL elements are mainly words, and not sentences.

D. Concept-Based Readability

The concept-based readability model which determines the overall readability score of a WSDL document d_i can be computed by (6):

$$CRS(d_i) = DS(d_i) + DC(d_i) + DaCw(d_i)^{-1} \quad (6)$$

IV. Concept-Based Readability Assessment Methodology

The concept-based readability of a WSDL document can be measured by an assessor who is a member of the quality assurance team or a service designer. The assessor must have knowledge of the service domain in order to choose (or construct) the concept hierarchy or domain ontology appropriately, and at the end evaluate the readability score. The assessment methodology comprises the following steps.

A. Service Information Preparation

To prepare for assessment, the assessor first does the following.

Select WSDL Document

The assessor selects a Web service and acquires its WSDL document. Note that if the assessor wants to compare readability of two Web services, both services will be assessed individually but they must be in the same domain and share the same concept hierarchy.

Obtain Concept Hierarchy of Service Domain

The concept hierarchy can be a domain ontology defined by domain experts. We consult either search engines or Web sites that publish domain-specific ontologies, e.g., [11]-[13], to discover relevant ontology for the Web service. In the case that multiple ontologies of the service domain are found, a tool like Protégé [14] can be used by the assessor to merge them into a single integrated ontology.

If relevant domain ontology is not available for the Web service, WordNet can be used to help generate the concept hierarchy [8]. The key terms in the WSDL document will be extracted and interwoven with their hypernym terms from WordNet to build up the concept hierarchy. Nevertheless, obtaining the concept hierarchy through ontology libraries is preferred to obtaining it through concept hierarchy generation. This is because concepts in domain ontology are defined by domain experts and can include specific terms of the domain which better reflect domain knowledge, whereas concepts taken from WordNet are likely to be more generic terms. Only when the Web services are built for specific domains and appropriate domain ontology is not readily available should the concept hierarchy be generated from WordNet.

B. Readability Measurement

Generally the assessor measures readability of the whole WSDL document, but in some case it might be useful to measure readability of particular WSDL elements for a more detailed analysis. The assessor can first select to measure readability of either the whole document or certain elements, and then apply the concept-based readability model (6). In the case that the measurement targets a particular WSDL element, the content of the element corresponds to a textual document of the model.

To support WSDL readability measurement, we develop a tool called WSDL Readability Calculator which is shown in Fig. 1. The tool is developed by using Eclipse Java EE IDE [15]. Protégé Java library [14] is used to read the domain ontology file. The tool requires the following input from the assessor:

- Web service description URL or a WSDL file;
- Service domain ontology URL or an OWL file;
- Calculation method, i.e., whole document or specific WSDL element.

Once the assessor inputs all service details and starts the calculation, the tool will perform the seven steps below:

Step 1: Extract terms from WSDL elements. WSDL syntax (such as tag names, types, cardinality) is excluded; only their contents will be considered. Duplicate terms are also removed. The following is an example of an input message element of a WSDL operation and a term extracted from this element.

- Input message: <wsdl:input message="services:CreateInvoiceRequest" name="CreateInvoiceRequest" />

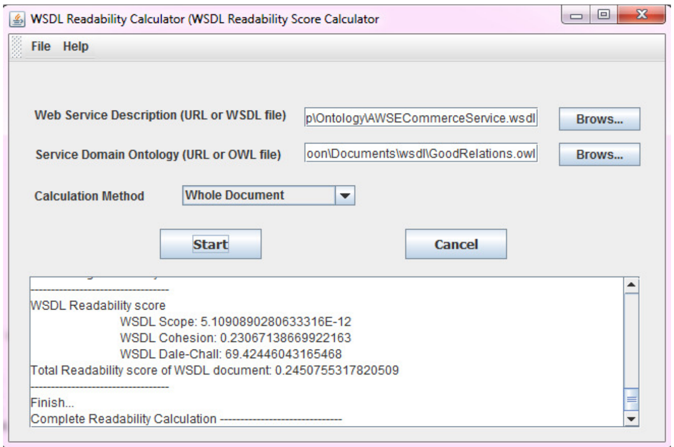

Fig. 1. WSDL readability calculator.

- Extracted term: CreateInvoiceRequest

Step 2: Extract key individual terms from the list of extracted terms in step 1. The terms will be changed to their singular form. Duplicate terms and single character will be removed. For example, the term "CartItems" will be extracted into two individual terms: "Cart" and "Item".

Step 3: Map key individual terms to the concepts in the concept hierarchy. To measure readability score of a WSDL document, only the WSDL terms that are present in the concept hierarchy will be used for measuring the document scope and document cohesion. On the other hand, all terms in a WSDL document will be used for measuring the simplified Dale-Chall's readability index.

Step 4: Calculate the document scope using (1). To calculate the document scope, the depth of each domain concept found in the WSDL document is determined with regard to the concept hierarchy.

Step 5: Calculate the document cohesion using (2)-(4). To calculate the document cohesion, the shortest path between each pair of domain concepts that are found in the WSDL document is determined, with respect to the concept hierarchy.

Step 6: Calculate the simplified Dale-Chall's readability index using (5). To calculate the simplified Dale-Chall's readability index, the percentage of difficult words not found in the Dale list of familiar words is determined.

Step 7: Calculate the concept-based readability score using (6). The readability score of a WSDL document is calculated using the concept-based readability model.

C. Evaluation

Once the readability score is obtained, the assessor evaluates if readability of the whole WSDL document (or a particular WSDL element) is appropriate. The assessor may be concerned about attracting the consumers and at the same time being vulnerable to attackers and competitors. The assessor can use the readability score for comparison purpose, e.g., comparing with the scores of competitors' services. The scores can be adjusted if the assessor sees fit.

D. Readability Adjustment

The concept-based readability score is dependent on the design and naming of terms within the WSDL document and the quality of the concept hierarchy. That is, the assessor should choose service domain ontology from a reliable source.

Measuring readability of specific WSDL elements can help the assessor to pinpoint which parts of the WSDL document should be redesigned and how to adjust their contents in order to increase or lower the score appropriately. According to (6), it is apparent that we can adjust the readability score by redesigning or renaming of terms within the WSDL document since change of terms can affect the document scope, document cohesion, and simplified Dale-Chall's readability index in the following ways.

- Document Scope: For a WSDL document, adding terms or changing to deeper terms in the concept hierarchy will decrease its readability score.
- Document Cohesion: For a WSDL document, adding terms or changing to terms that are more cohesive, i.e., more closely associated, with respect to the concept hierarchy will decrease the shortest path and hence increase its readability score.
- Simplified Dale-Chall's readability index: For a WSDL document, adding or changing to terms in the Dale list of familiar words will increase its readability score.

Therefore, to increase the readability score to improve service discoverability, the assessor may, where appropriate,

- Change domain terms in the WSDL document to non-domain terms;
- Change domain terms in the WSDL document to those that are shallower in the concept hierarchy;
- Change or add domain terms so that the WSDL document contains domain terms that are more cohesive with respect to the concept hierarchy;
- Change or add terms so that the WSDL document contains more of Dale's familiar words.

On the other hand, to lower the readability score to make the service description less comprehensible to competitors and attackers, the assessor may, where appropriate,

- Change non-domain terms in the WSDL document to domain terms;
- Change domain terms in the WSDL document to those that are deeper in the concept hierarchy;
- Change or add domain terms so that the WSDL document contains domain terms that are less cohesive with respect to the concept hierarchy;
- Change or add terms so that the WSDL document contains less of Dale's familiar words.

After redesigning or renaming of terms within the WSDL document, the assessor will repeat the measurement process to obtain the adjusted score. The adjustment and measurement can be repeated as necessary until the assessor is satisfied with the score.

V. Experiment on Readability Measurement

As an experiment, this section describes the application of the methodology in Section IV to three Web services in the E-commerce domain, i.e., Amazon [16], PayPal [17], and eBay [18]. The WSDLs of these services are processed by the WSDL readability calculator to determine readability of the whole WSDL documents. Note that, to measure readability of a specific type of WSDL elements, the methodology can be applied in a similar manner.

For demonstration purpose, we show step-by-step application of the methodology to measure readability of the whole WSDL of Amazon Web service.

A. Service Information Preparation

Amazon Web service delivers a set of E-commerce services, i.e., payment management, stock management, and package and shipping management. Its WSDL can be found at [16]. The GoodRelations ontology [19] is selected as the concept hierarchy for the E-commerce domain. Part of it is depicted in Fig. 2. A number of concepts such as account, contact information, payment method, etc. are listed in this ontology.

B. Readability Measurement

The process to measure readability of Amazon's WSDL is as follows.

Step 1: Extract Terms from WSDL Elements

Element names are extracted from WSDL syntax. Duplicate terms are also removed. In total, there are 373 extracted terms. Table I shows some of the terms that are extracted from a WSDL snippet in Fig. 3.

Step 2: Extract Key Individual Terms

The extracted terms are further extracted to obtain key individual terms and are changed to a singular form. Again, duplicate terms and any single character are removed. For example, the term "SimilarProducts" will be extracted into two individual words "Similar" and "Product". In total, there are 275 individual terms, and Table II shows some of them.

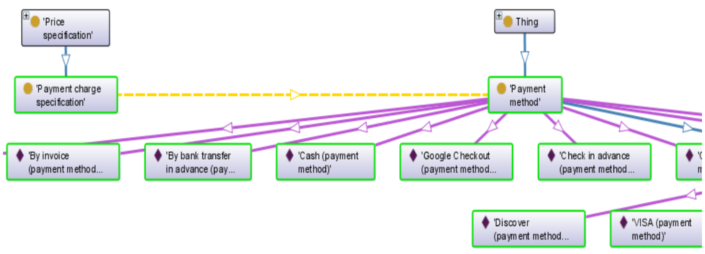

Fig. 2. Part of GoodRelations ontology.

TABLE I
EXAMPLE OF EXTRACTED TERMS FROM AMAZON'S WSDL

PurchaseURL	SimilarProducts
MobileCartURL	TopSellers
SubTotal	TradeInValue
CartItems	SimilarViewedProducts
SavedForLaterItems	LowestNewPrice

```
<xs:element ref="tns:Request" minOccurs="0" />
<xs:element name="CartId" type="xs:string" />
<xs:element name="HMAC" type="xs:string" />
<xs:element name="URLEncodedHMAC" type="xs:string" />
<xs:element name="PurchaseURL" type="xs:string" minOccurs="0" />
<xs:element name="MobileCartURL" type="xs:string" minOccurs="0" />
<xs:element name="SubTotal" type="tns:Price" minOccurs="0" />
<xs:element ref="tns:CartItems" minOccurs="0" />
<xs:element ref="tns:SavedForLaterItems" minOccurs="0" />
<xs:element ref="tns:SimilarProducts" minOccurs="0" />
<xs:element ref="tns:TopSellers" minOccurs="0" />
<xs:element ref="tns:NewReleases" minOccurs="0" />
<xs:element ref="tns:SimilarViewedProducts" minOccurs="0" />
<xs:element ref="tns:OtherCategoriesSimilarProducts" minOccurs="0" />
...
<xs:element name="TrackSequence" type="xs:string" minOccurs="0" />
<xs:element name="TradeInValue" type="tns:Price" minOccurs="0" />
<xs:element name="UPC" type="xs:string" minOccurs="0" />
<xs:element name="LowestNewPrice" type="tns:Price" minOccurs="0" />
<xs:element name="LowestUsedPrice" type="tns:Price" minOccurs="0" />
```

Fig. 3. Snippet of Amazon's WSDL.

TABLE II
EXAMPLE OF INDIVIDUAL TERMS FROM AMAZON'S WSDL

Purchase	Url	Mobile	Cart
Sub	Total	New	Price
Saved	For	Later	Viewed
Similar	Product	Top	Seller
Item	Trade	In	Value

Step 3: Map Key Individual Terms to Concepts in Concept Hierarchy

Individual terms will be mapped to the concepts in the GoodRelations concept hierarchy. In total, there are 21 individual terms that can find a match in the concept hierarchy. Table III shows some of these domain terms.

Step 4: Calculate Document Scope

The depth of each domain term found in step 3 is determined with regard to the concept hierarchy. Table IV shows the depth of some domain terms. Given (1), the document scope of Amazon's WSDL document is 5.1091E-12.

Step 5: Calculate Document Cohesion

The shortest path between each pair of 21 domain terms is determined with respect to the concept hierarchy of depth 5. Table V shows the shortest path between some of them. Given (2)-(4), the document cohesion of Amazon's WSDL document is 0.2307.

TABLE III
EXAMPLE OF DOMAIN TERMS FOUND IN AMAZON'S WSDL

Item	Product
Value	Price

TABLE IV
EXAMPLE OF DEPTH OF DOMAIN TERMS IN AMAZON'S WSDL

Concept c_i	Depth
Item	3
Product	2
Value	1
Price	1

TABLE V
EXAMPLE OF SHORTEST PATHS BETWEEN PAIRS OF DOMAIN TERMS IN AMAZON'S WSDL

c_i	c_j	$len(c_i, c_j)$
Item	Product	6
Item	Value	6
Item	Price	5
Product	Value	5
Product	Price	4

Step 6: Calculate Simplified Dale-Chall's Readability Index

There are 86 out of 275 individual terms which are found in the Dale list of familiar words. Table VI shows some of them. Therefore there are 189 terms that are considered difficult. Given (5), the simplified Dale-Chall's readability index of Amazon's WSDL document is 68.7273.

Step 7: Calculate Concept-Based Readability Score

Given (6), the concept-based readability score of Amazon's WSDL document is 0.2452.

C. *Readability Comparison*

Similarly to Amazon's WSDL, the WSDLs of PayPal and eBay can be processed to determine readability based on GoodRelations E-commerce ontology. Table VII shows the measurements of all three services in comparison. PayPal's Web service provides the most readable WSDL document whereas readability of Amazon's WSDL is the lowest. Readable and comprehensible service description can be one factor in the popularity of PayPal Web service, while WSDL readability of eBay, and Amazon follows PayPal's very closely, indicating that they can be competing Web services providing similarly readable service descriptions. On the other hand, Amazon's WSDL document is likely to expose less information to competitors and attackers.

VI. EXPERIMENT ON READABILITY ADJUSTMENT

After the readability score is obtained, the assessor evaluates if readability of the whole WSDL document (or a particular WSDL element) is appropriate. If not, the assessor determines whether readability score of WSDL document should be increased or lowered compared to the current score.

TABLE VI
EXAMPLE OF FAMILIAR WORDS FOUND IN AMAZON'S WSDL

For	Cart	Top
New	Value	Price

TABLE VII
READABILITY MEASUREMENTS

	Amazon	PayPal	eBay
Document Scope	5.1091E-12	1.0262E-10	5.7495E-19
Document Cohesion	0.2307	0.2989	0.2875
Simplified Dale-Chall's Readability Index	68.7273	59.659	79.5482
Concept-Based Readability Score	0.2452	0.3156	0.3001

Adjustment of WSDL readability score can be done according to the criteria that have been described in Section IV.D. With the adjustment process, readability score calculated from the document scope, document cohesion, and simplified Dale-Chall's readability index will be affected.

A. *Increase Readability Score of WSDL Document*

To provide an example of how to increase the readability score of a WSDL document, we redesign terms in the WSDL document of Amazon's Web service.

Change Domain Terms in WSDL to Non-domain Terms

Domain terms found in WSDL elements may be redesigned to non-domain terms as shown in Table VIII. After redesigning terms, the readability score is recalculated and new relevant scores are obtained in Fig. 4. The new readability score is increased.

Change Domain Terms in WSDL to Those That Are Shallower in Concept Hierarchy

Domain terms found in WSDL elements may be redesigned to those that are shallower in the concept hierarchy as shown in Table IX. After redesigning terms, the readability score is recalculated and new relevant scores are obtained in Fig. 5. The new readability score is increased.

TABLE VIII
CHANGE OF DOMAIN TERMS TO NON-DOMAIN TERMS IN AMAZON'S WSDL

WSDL Element	Original Domain Term	Redesigned Term
AWSECommerceService	Service	Work
SimilarProducts	Product	Goods
Model	Model	Form
NonNegativeIntegerWithUnits	Integer	Number
BrowseNodes	Node	Object

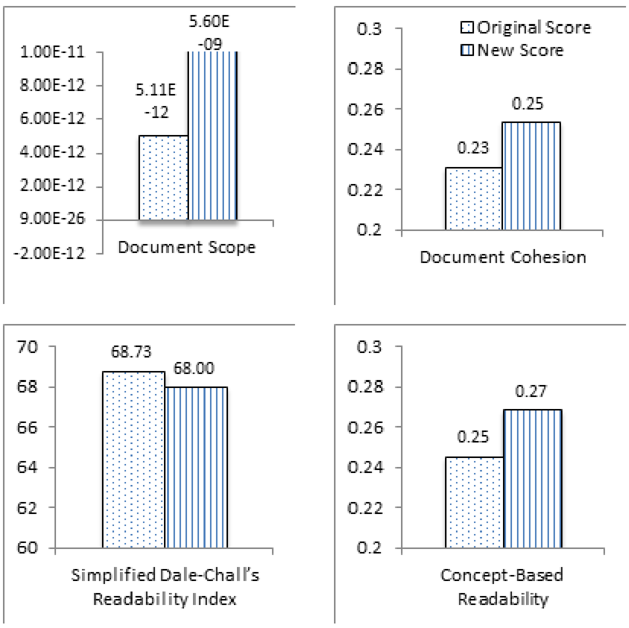

Fig. 4. Adjustment result – Change domain terms to non-domain terms.

TABLE IX
CHANGE TO SHALLOWER DOMAIN TERMS IN AMAZON'S WSDL

WSDL Element	Original Domain Term	Original Depth	Redesigned Domain Term	New Depth
ItemSearch	Item	3	Product	1
Model	Model	2	Type	1
AWSECommerce Service	Service	2	Product	1

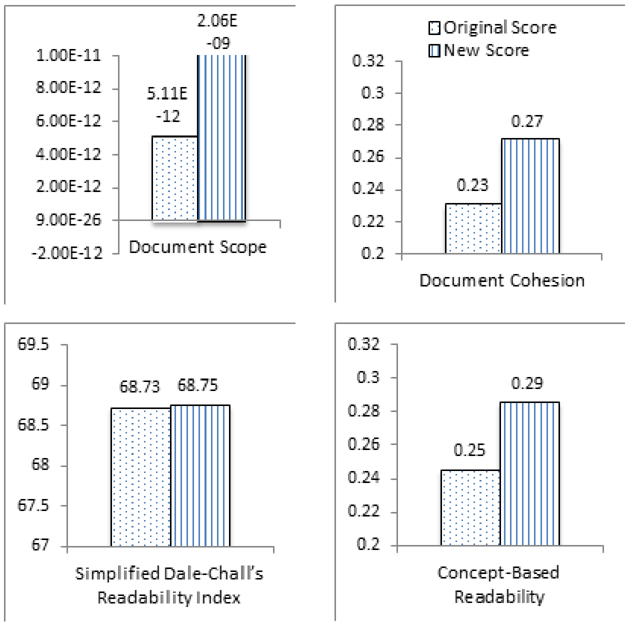

Fig. 5. Adjustment result – Change domain terms to shallower domain terms.

Change or Add Domain Terms So That WSDL Contains Domain Terms That Are More Cohesive with Respect to Concept Hierarchy

Domain terms found in WSDL elements may be redesigned so that they are more closely associated with each other with respect to the concept hierarchy, i.e., the shortest paths between them are shorter. The redesigned terms are shown in Table X and new associations in Table XI. After redesigning terms, the readability score is recalculated and new relevant scores are obtained in Fig. 6. The new readability score is increased.

Change or Add Terms So That WSDL Contains More of Dale's Familiar Words

Domain terms found in WSDL elements may be redesigned to terms that are in the list of Dale's familiar words as shown in Table XII. After redesigning terms, the readability score is recalculated and new relevant scores are obtained in Fig. 7. The new readability score is increased.

Summary

In Fig. 8, we summarize the new readability scores obtained from the adjustment approaches to increase readability. It is likely that change to shallower and more cohesive domain terms has a strong positive impact on readability.

TABLE X
CHANGE TO MORE COHESIVE DOMAIN TERMS IN AMAZON'S WSDL

WSDL Element	Original Domain Term	Redesigned Domain Term
ItemSearch	Item	Product
Model	Model	Type
AWSECommerceService	Service	Product

TABLE XI
SHORTER SHORTEST PATHS BETWEEN PAIRS OF DOMAIN TERMS

Original Pair	Original Length	Redesigned Pair	New Length
Item-Sale	7	Product-Sale	6
Item-Value	6	Product-Value	5
Item-Unit	6	Product-Value	6
Model-Sale	6	Type-Sale	4
Model-Value	5	Type-Value	3

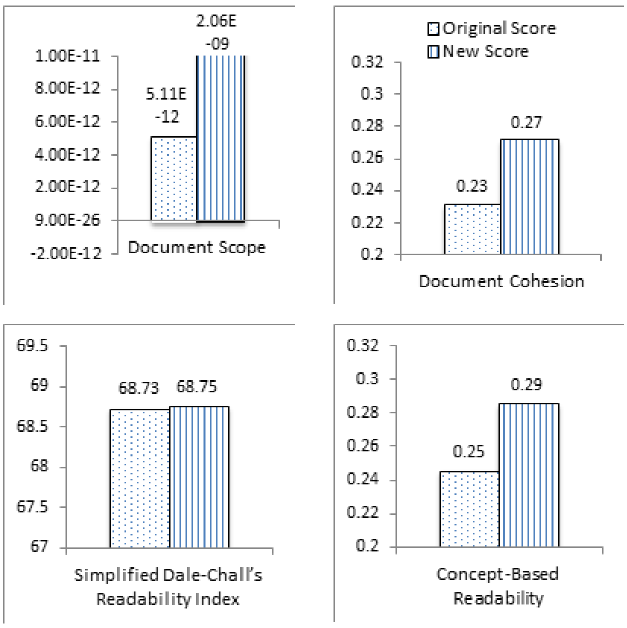

Fig. 6. Adjustment result – Change domain terms to more cohesive domain terms.

TABLE XII
CHANGE TO TERMS AMONG DALE'S FAMILIAR WORDS IN AMAZON'S WSDL

WSDL Element	Original Term	Redesigned Term
ItemDimensions	Item	Whole
BinParameter	Parameter	Writing
Request	Request	Call
Brand	Brand	Name
BrowseNode	Browse	Search

B. *Lower Readability Score of WSDL Document*

To provide an example of how to lower the readability score of a WSDL document, again we redesign terms in the WSDL document of Amazon's Web service.

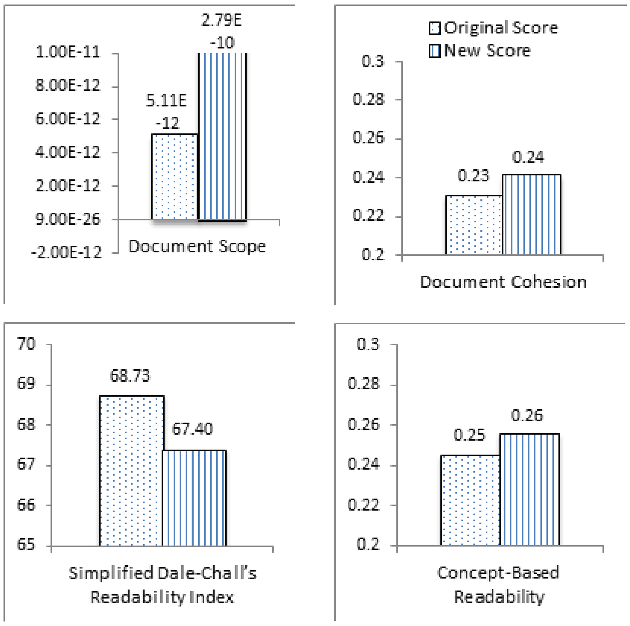

Fig. 7. Adjustment result – Change to terms among Dale's familiar words.

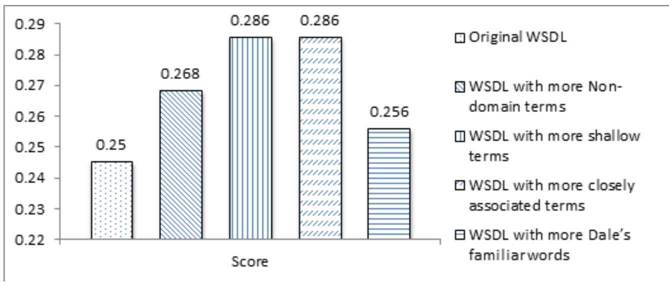

Fig. 8. Comparison of adjustment approaches to increase readability score.

Change Non-domain Terms in WSDL to Domain Terms

Non-domain terms found in WSDL elements may be redesigned to domain terms as shown in Table XIII. After redesigning terms, the readability score is recalculated and new relevant scores are obtained in Fig. 9. Note that, in this experiment, the change of terms can decrease the document scope but at the same time it increases the document cohesion and simplified Dale-Chall's readability index, resulting in higher overall readability.

TABLE XIII
CHANGE OF NON-DOMAIN TERMS TO DOMAIN TERMS IN AMAZON'S WSDL

WSDL Element	Original Non-domain Term	Redesigned Domain Term
Detail	Detail	Specification
Shipping	Shipping	Delivery
Feature	Feature	Function
PackageDimensions	Package	Parcel
IsCategoryRoot	Category	Entity

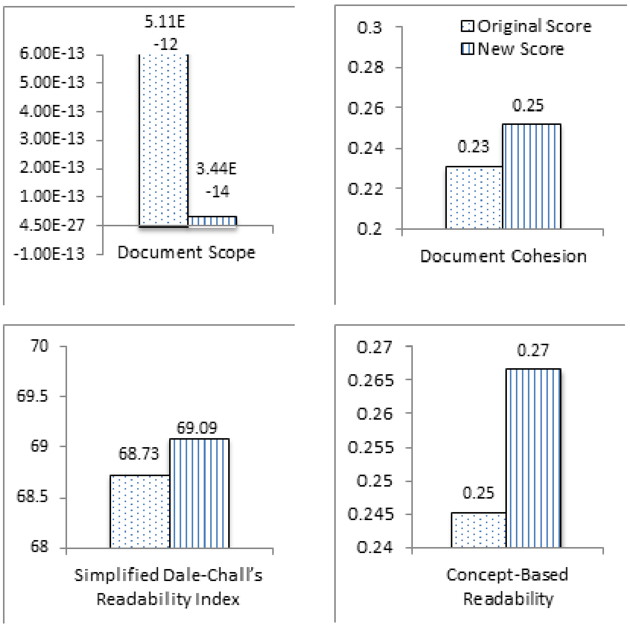

Fig. 9. Adjustment result – Change non-domain terms to domain terms.

Change Domain Terms in WSDL to Those That Are Deeper in Concept Hierarchy

Domain terms found in WSDL elements may be redesigned to those that are deeper in the concept hierarchy as shown in Table XIV. After redesigning terms, the readability score is recalculated and new relevant scores are obtained in Fig. 10. The new readability score is lower.

Change or Add Domain Terms So That WSDL Contains Domain Terms That Are Less Cohesive with Respect to Concept Hierarchy

Domain terms found in WSDL elements may be redesigned so that they are more loosely associated with each other with respect to the concept hierarchy, i.e., the shortest paths between them are longer. The redesigned terms are shown in Table XV and new associations in Table XVI. After redesigning terms, the readability score is recalculated and new relevant scores are obtained in Fig. 11. The new readability score is lower.

Change or Add Terms So That WSDL Contains Less of Dale's Familiar Words

Domain terms found in WSDL elements may be redesigned to terms that are not in the list of Dale's familiar words as shown in Table XVII. After redesigning terms, the readability score is recalculated and new relevant scores are obtained in Fig. 12. The new readability score is lower.

Summary

In Fig. 13, we summarize the new readability scores obtained from the adjustment approaches to lower readability. Compared to the original readability score, the readability scores obtained from three approaches decrease as expected. However, the readability score of the approach in which more domain terms are added increases. This is possible because change of domain terms always affect all three components of the readability model. In this example as seen previously in Fig. 9, the added domain terms that help decrease the document scope also have positive effects on both document cohesion and simplified Dale-Chall's readability index. This hence results in an increase in readability. It is seen that selecting the right terms to remove from or add to the WSDL document may not be easy and a single adjustment approach may not achieve the desired effect.

TABLE XIV
CHANGE TO DEEPER DOMAIN TERMS IN AMAZON'S WSDL

WSDL Element	Original Domain Term	Original Depth	Redesigned Domain Term	New Depth
Brand	Brand	1	Entity	2
BrowseNode	Node	1	Location	2
IssuesPerYear	Per	1	Individual	3
Offers	Offer	1	Provisioning	2

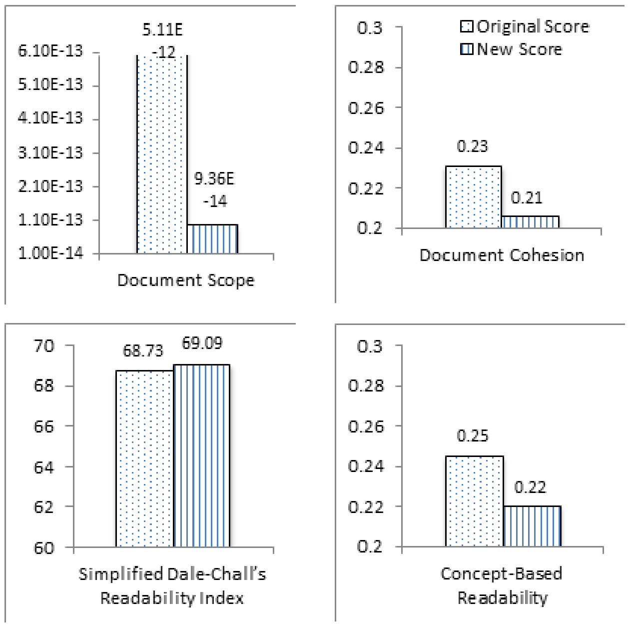

Fig. 10. Adjustment result – Change domain terms to deeper domain terms.

TABLE XV
CHANGE TO LESS COHESIVE DOMAIN TERMS IN AMAZON'S WSDL

WSDL Element	Original Domain Term	Redesigned Domain Term
Brand	Brand	Entity
BrowseNode	Node	Location
IssuesPerYear	Per	Individual
Offers	Offer	Provisioning

TABLE XVI
LONGER SHORTEST PATHS BETWEEN PAIRS OF DOMAIN TERMS

Original Pair	Original Length	Redesigned Pair	New Length
Brand-Offer	3	Entity-Provisioning	4
Node-Value	3	Location-Value	5
Per-HTTP	2	Individual-HTTP	6
Offer-Hour	3	Provisioning-Hour	4
Offer-Warranty	3	Provisioning-Warranty	4

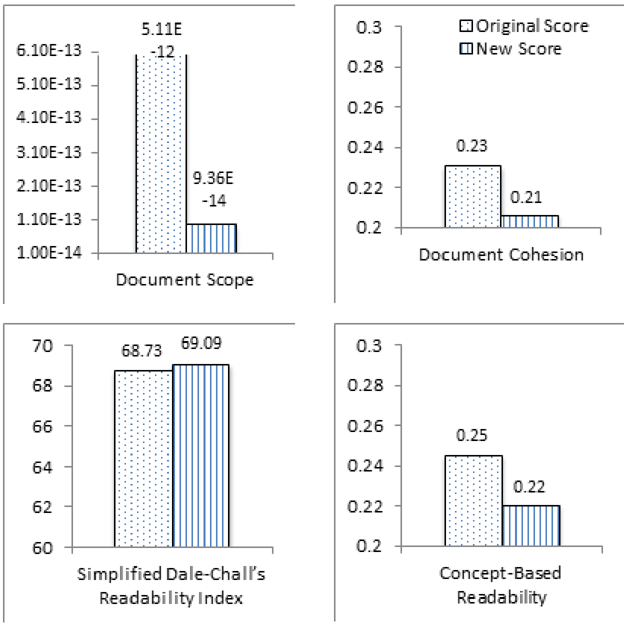

Fig. 11. Adjustment result – Change domain terms to less cohesive domain terms.

TABLE XVII
CHANGE TO TERMS NOT AMONG DALE'S FAMILIAR WORDS IN AMAZON'S WSDL

WSDL Element	Original Term	Redesigned Term
Name	Name	Brand
SearchBinSet	Search	Browse
NarrowBy	Narrow	Specify
CartGet	Get	Acquire
Length	Length	Duration

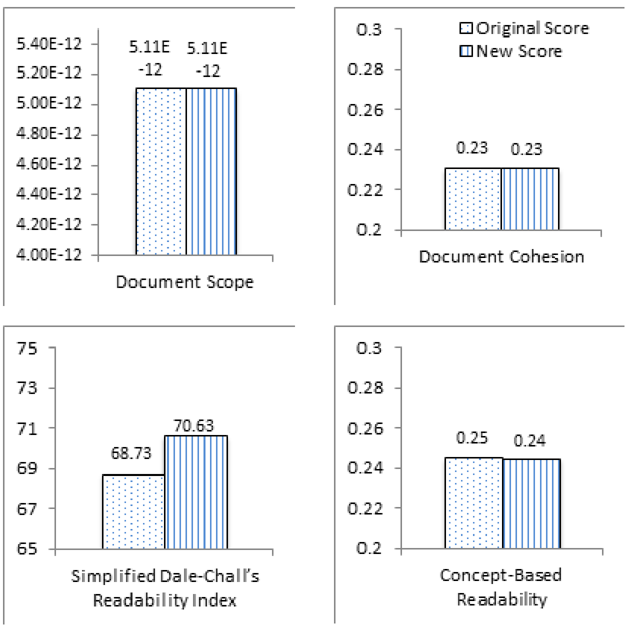

Fig. 12. Adjustment result – Change to terms not among Dale's familiar words.

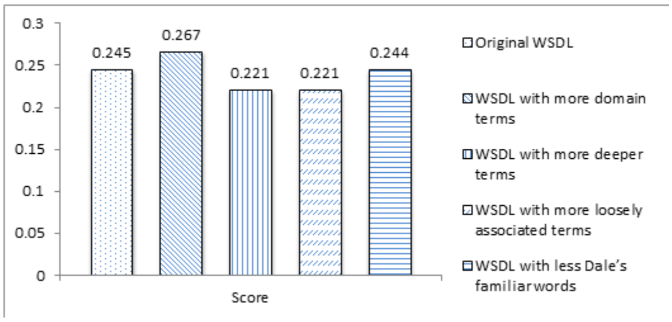

Fig. 13. Comparison of adjustment approaches to lower readability score.

VII. Discussion and Conclusion

This paper presents an application of a concept-based document readability model to measure readability of Web services descriptions. The approach is motivated by a design principle to expose readable service information and by potential risks from service imitation and attacks which may follow such information exposure. The main components of the model are the scope and cohesion of a WSDL document with regard to the vocabulary of its service domain, as well as the difficulty of terms that are used in the document.

Readability can be one factor to service popularity and service imitation and attacks, but it does not say that a highly readable service description will make the service more popular among service consumers, or become a target of competitors and attackers. A WSDL document with high readability score may not attract service consumers if other kinds of service quality are not well-maintained. Likewise, the service provider may also take additional measures to handle vulnerability from exposure of highly readable information.

It is important to note that the readability score of a WSDL document can change if the service domain ontology and the list of familiar words are changed. Several ontologies of different quality may be published on the Internet for a particular service domain, and thus the assessor should pick the domain ontology that is most complete and from a reliable source. We also see that the use of the familiar word list is a useful intuitive approach. Despite the fact that the Dale's list contains familiar words that are known in reading by at least 80 percent of the children in Grade 4, the terms familiar and unfamiliar describing the words here are used in a statistical sense. The percentage of words outside this list is a very good index of the difficulty of reading materials, and has so far been used to determine readability in various contexts including that of technical materials. Therefore we still use the Dale's list in the model even though we see the possibility that an extended list, constructed for readers at higher grade levels, should reflect better the readability scores of technical WSDL documents whose readers are programmers.

Readability adjustment requires that the assessor or service designer give careful consideration to the context of WSDL elements before redesigning them with new terms. It may not be easy to select new terms that are appropriate and give the expected change to the readability level.

Our future work is to experiment with a number of service providing organizations in Thailand to see if the approach can help them with the design of service descriptions and how they use the adjustment method to adjust readability of their WSDL documents. The result from this experiment can be used to determine the appropriate WSDL design which gives the clients a satisfactory readability level. In addition, the readability score and readability adjustment method should be validated. To do so, a number of service clients will be asked to answer a questionnaire to validate if the score truly reflects WSDL readability and the adjustment method can truly increase or lower readability.

REFERENCES

[1] World Wide Web Consortium (W3C). (2001, Mar. 15). *Web Services Description Language (WSDL) 1.1* [Online]. Available: http://www.w3.org/TR/wsdl

[2] J. J. Pikulski. (2013, Aug.). *Readability*. Houghton Mifflin Harcourt. [Online]. Available: http://www.eduplace.com/state/author/pikulski.pdf

[3] X. Yan, D. Song, and X. Li, "Concept-based document readability in domain specific information retrieval," in *Proc. 15th ACM Int. Conf. Information and Knowledge Management (CIKM 2006),* 2006, pp. 540-549.

[4] T. Gruber. (2013, Aug.). *Ontology* [Online]. Available: http://tomgruber.org/writing/ontology-definition-2007.htm

[5] World Wide Web Consortium (W3C). (2004, Feb. 10). *OWL Web Ontology Language Overview* [Online]. Available: http://www.w3.org/TR/owl-features/

[6] J. Zhao and M. Y. Kan, "Domain-specific iterative readability computation," in *Proc. 10th Annual Joint Conf. Digital Libraries (JDCL 2010)*, 2010, pp. 205-214.

[7] A. Jatowt, K. Akamatsu, N. Pattanasri, and K. Tanaka, "Towards more readable web: measuring readability of web pages based on link structure," *SIGWEB Newsletter*, Winter 2012, pp. 1-7.

[8] Y. J. An, J. Geller, Y. T. Wu, and S. A. Chun, "Automatic generation of ontology from the deep web," in *Proc. 18th Int. Workshop Database and Expert Systems Applications (DEXA 2007)*, 2007, pp. 470-474.

[9] Princeton University. (2013, Aug.). *WordNet A Lexical Database for English* [Online]. Available: http://wordnet.princeton.edu/

[10] J. S. Chall and E. Dale, *Readability Revisited: The New Dale-Chall Readability Formula*. Brookline Books/Lumen Editions, 1995.

[11] Stanford Center for Biomedical Informatics Research. (2013, Aug.). *Protégé Ontology Library* [Online]. Available: http://protegewiki.stanford.edu/wiki/Protege_Ontology_Library

[12] DARPA. (2013, Aug.). *DAML Ontology Library* [Online]. Available: http://www.daml.org/ontologies/

[13] Open Knowledge Foundation (OKFN). (2013, Aug.). *Linked Open Vocabularies (LOV)* [Online]. Available: http://lov.okfn.org/dataset/lov/index.html

[14] Stanford Center for Biomedical Informatics Research. (2013, Aug.). *The Protégé Editor and Knowledge Acquisition System* [Online]. Available: http://protege.stanford.edu

[15] Eclipse Web Tools Platform Project. (2013, Aug.). *Eclipse Java EE IDE for Web Developers* [Online]. Available: http://www.eclipse.org/webtools

[16] Amazon Web Services LLC. (2013, Aug.). *eCommerce Web Service* [Online]. Available: http://webservices.amazon.com/AWSECommerceService/AWSECommerceService.wsdl

[17] PayPal. (2013, Aug.). *Invoicing Service API* [Online]. Available: https://svcs.paypal.com/Invoice/CreateAndSendInvoice?wsdl

[18] eBay Inc. (2013, Aug.). *eBay Shopping API* [Online]. Available: http://developer.ebay.com/webservices/latest/ShoppingService.wsdl

[19] E-Business and Web Science Research Group. (2011, Oct. 1). *GoodRelations Language Reference Version 1.0* [Online], Available: http://www.heppnetz.de/ontologies/goodrelations/v1.html

Pananya Sripairojthikoon has been working as a software engineer for Accenture Thailand since June 2012. She received B.Eng. in Computer Engineering from King Mongkut University of Technology Thonburi, Thailand in 2010, and M.Sc. in Software Engineering from Chulalongkorn University, Thailand in 2013. Her research interest includes service computing and application of semantic technology. This research was conducted while she pursued a Master degree.

Twittie Senivongse is an associate professor at the Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Thailand. She received B.Sc. in Statistics from Chulalongkorn University in 1989, M.Sc. in Computing Science from Imperial College, UK in 1992, and Ph.D. in Computer Science from University of Kent, UK in 1997. Her research interest includes service computing and application of semantic technology.