Global Information Interoperability through Open Information Platform

Jingzhi Guo
Department of Computer and Information Science, Faculty of Science and Technology
University of Macau
Av. Padre Tomás, Pereira, S.J., Taipa, Macau
jzguo@umac.mo

Abstract—**Resolving context differences between heterogeneous systems for global information interoperability is a challenging research issue in information technology. This paper solved the problem by suggesting a new concept of Open Information Platform (OIP) that builds on the two principles of naturalization and alienation. Following these two principles, an SCI methodology has been proposed to require standardization of information, collaborative creation of atomic signs and internalization of standardized information to hide complex standards behind final users. To implement the SCI method, an OIP technical infrastructure is implemented. The short history of information interoperability research signifies that OIP concept is important and is a natural outcome towards the future technology for information exchange and use.**

Keywords-Information interoperability, context difference, sign, semantics, syntax, context, collaboration, semantic integration

I. Introduction

Information interoperability is an important task of information technology. IEEE defines interoperability as "the ability of two or more systems or components to exchange information and to use the information that has been exchanged" [21]. Obviously, two levels of requirements are explicitly proposed in this definition. First, information must be able to be exchanged between any connected systems. Second, the exchanged information must be usable in the recipient's systems. Meeting these two requirements are extremely important for many fields such as electronic commerce [42], enterprise integration [28], supply chain management [40], service composition [26], distributed systems design [39], quality of service [8], and collaborative design [14]. This is because all these fields require a global perspective on information exchange and use. For example, in electronic commerce, buyers and sellers must be able to exchange their business information and understand the exchanged information through Internet-connected e-commerce systems. Inability of exchanging and understanding business information will lead to the failure of completing business transactions.

Global information interoperability, i.e. exchanging and using information between Internet-connected systems and people, is challenging because there are two levels of interoperability problems corresponding to the two interoperability requirements. They are: *syntactic data interoperability*, which is the ability of communicating and exchanging data on certain data formats and communication protocols between connected systems; *semantic information interoperability*, which is the ability of automatically and accurately interpreting the exchanged data as meaningful information understandable by the connected systems and the end users of the connected systems. For the first problem, its task is to enable data communication and exchange. Here data is not necessary to be meaningful to human users. It only needs to be successfully sent, received and computer-processable between connected systems. For the second problem, its task is much more complex such that data must be not only computer-processable but also understandable by human users of computers as meaningful information.

Existing solutions to the above two problems are often different. To resolve the first problem, open standards are often favored for handling data formats and communication protocols, for example, HTTP for hypertext transfer, HTML for hypertext description, XML [2] for data transportation and storage, DTD and XML schema [36] for describing data schema, WSDL [6] for web service description, SVG (www.w3.org/Graphics/SVG/) for graphic description, SMIL (www.w3.org/AudioVideo/) for describing audiovisual presentation, RDF/RDFS [3] for resource description, and OWL [25][27] for describing terms and their relations. The integration of most existing open standards can enable data communication and exchange [4]. The left open issue is when connected systems use different open standards, the data communication and exchange might still not be possible. This issue asks for more researches on standards integration.

While syntactic data interoperability problem can in much sense be solved by open standards, the second problem of semantic information interoperability is somehow solved through defining the consistent information meanings of a shared domain. In particular, we can design shared data formats and information for applications, systems, enterprises, domains and communities in application-wide, system-wide, enterprise-wide, domain-wide or community-wide scopes. For example, ontology designed for various applications and systems in daml.org [9], standard product coding designed for electronic commerce in UNSPSC [38] and ecl@ss [10], and the gene ontology [35] is designed for gene domain.

Till now semantic information interoperability problem is still not well solved. The key issue is that the meaning of information is dependent on the users' environments [13][33]. Systems of different environments create and use heterogeneous information in both syntax and semantics. This causes the meaning differences when information is exchanged and used, which affects systems and their users to accurately interpret the exchanged information for further use. Currently, there are many researches that attempt to resolve this tough issue, for example, ontology mapping, alignment, integration or matching [5][22]. Many research papers can be found in ontology matching website [29]. Unfortunately, the research advancement is still experimental and limited in integrating small-scale non-interoperable vocabularies. Technical infrastructure that resolve semantic information interoperability problem in a wider scope is not available yet and needs further exploration.

This paper aims to propose a new concept for advancing the existing solutions such that information interoperability can be globally achieved. The new concept is called *open information platform* (OIP), where non-interoperable information are naturalized using a core information representation standard, called *XML Product Map* (XPM) [13], which is invisible to heterogeneous systems and intact from most information users. XPM as an information representation language represents objects in reality. It defines the information syntax on how the meanings of information across heterogeneous systems and their users can be consistently represented in a self-explainable and human-computer understandable manner. By XPM, syntactic data interoperability problem is solved by using unified information syntax, and semantic information interoperability problem is resolved through collaborative information design.

Unlike existing open standards that require information users to actively comply with the standards in use, OIP hides XPM information standard from information users and dematerializes it when information is moving out of OIP systems. This makes users to feel that they just work in their original systems, though information has already experienced transformation.

The rest of the paper is arranged as follows: Section II describes the concept of open information platform (OIP). Section III implements the OIP concept in an OIP technical infrastructure. Section IV discusses the related work on information interoperability. Finally, we conclude the paper and list some of the key contributions and future work.

II. The Concept of Open Information Platform

The concept of *open information platform* (OIP) is proposed as a technical infrastructure, which serves the global information exchange and use between Internet-connected computers and human being. The design of technical infrastructure is based on an assumption of contextual difference, described as follows:

Assumption 1 (*contextual difference*). Two actors A and B from contexts X_a and X_b shall represent a same object O in reality in different ways such that:

(1) $O(A, X_a) \neq_{sem} O(B, X_b)$ and $O(A, X_a) \neq_{syn} O(B, X_b)$;

(2) $O(A, X_a) =_{sem} O(B, X_b)$ and $O(A, X_a) \neq_{syn} O(B, X_b)$;

where $=_{sem}$ notates "semantically equal", $\neq_{sem}$ notates "semantically unequal", $=_{syn}$ notates "syntactically equal", and $\neq_{syn}$ notates "syntactically unequal".

In this assumption, a *context* is a background, an environment or a situation such as a particular culture, a customs, a language, a dialect or a particular specification, which affects a person to behave on representing any object in reality. For example, a person born in English environment will represent his perceived objects in English.

This assumption is easy to be proved to exist with some experimental cases. For example, to prove Assumption 1(1), we can simply ask a mother and a baby to write the word "moon" in a paper, the mother may correctly write the word but the baby may toss the paper to the ground. This shows different contexts may result entirely different behavioral patterns (syntax) and meanings. To prove Assumption 1(2), we can simply ask two people speaking different languages to write the word of "moon", obviously they will write differently but mean the same.

Assumption 1 can immediately derive a lemma as follows:

Lemma 1 (*contextual distance*). Two actors A and B from context X_a and X_b shall represent a same object O in reality in a more similar or more dissimilar way if and only if two contexts X_a and X_b are moving closer or farther such that $[O(A, X_a) =_{sem} O(B, X_b)$ and $O(A, X_a) =_{syn} (B, X_b)] \Rightarrow [O(A, X) \neq_{sem} O(B, X)$ and $O(A, X_a) \neq_{syn} O(B, X_b)]$ if and only if $\mathit{Equal}(X_a, X_b) \Rightarrow \mathit{Unequal}(X_a, X_b)$.

This lemma can be easily proved from the simple experiments of cases such as: (1) people born and living in a same place speak same language; (2) people from a same ethic or political group shows the similar viewpoints; or (3) people from two cultures that never have interactions speak totally different languages. Lemma 1 reflects such a fact that context distance affects information interoperability.

The key concept behind Assumption 1 and Lemma 1 is context. The contextual differences between heterogeneous systems are the natural causes of information interoperability problem. The resolution of information interoperability must resolve contextual differences.

A. Principles of Resolving Contextual Differences

In this paper, we establish two principles to resolve contextual differences that cause information interoperability problem. The principles are naturalization and alienation.

Principle 1 (*naturalization*). Naturalization states that the contextually different information shall be converted into the contextually same information in order to enable information exchange and use.

This principle is the enforcement of Lemma 1 from discrete contexts to a common context to achieve commonality of information, for example, designing a common vocabulary for all information composition.

Nevertheless, contextual difference is a fact and we must admit this difference such that information must be adaptable to individual contexts when information is exchanged and used. This needs the second principle.

Principle 2 (*alienation*). Alienation states that the contextually same information shall be converted into the contextually different information in order to achieve differentiation of information.

This principle is a reverse enforcement of Lemma 1, moving from common context to many disparate contexts to achieve information personalization, for instance, mapping a vocabulary onto many language-different vocabularies of individual contexts.

Based on these two principles, the concept of OIP can be presented in Figure 1 as a guideline for designing OIP such that non-interoperable information of heterogeneous systems must first be naturalized or assimilated in certain ways and then should be alienated to adapt to different contexts.

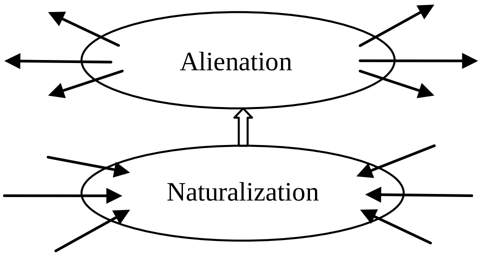

Figure 1: Principles of resolving contextual differences

B. Methodology of Resolving Contextual Differences

To resolve contextual differences for information interoperability, this paper proposes a novel methodology of Standardization, Collaboration and Internalization, shortly, an *SCI method*. This method consists of three theoretical elements as described in the following:

1) Standardization

SCI method states that to enable information interoperability between contextually different systems and users, it is necessary to develop a new common standard for representing information in both syntactic formats and semantic meanings. This standard must be able to represent most information objects in reality, such as linguistic terms, symbols, statements, logical expressions, phonetic expressions, and messages for exchanges and use. In this paper, we shall adopt XML Product Map (XPM) [13][16][17][18][19], which is an information representation language used in the entire research.

2) Collaboration

In SCI method, we think that contextual differences of information cannot be resolved through any automated systems (i.e. third-party intelligent systems), because the intelligent agents of automated systems are in the contexts that are different from the contexts in which human users are situated. Following Assumption 1, intelligent agents and human users have different information representations. This causes the understanding differences between human users and intelligent agents on same information. Collaboration is the only way of resolving understanding differences, because collaboration is a process of negotiation and arbitration. Negotiation can support the arguments of individual understanding of information representation. It can also disambiguate the understanding differences when information is represented. If some information cannot be agreed between negotiating users, arbitration can force a unified agreement between all negotiating users. Thus, collaboration can always resolve contextual differences.

3) Internalization

SCI method not only enforces common understanding of information through standardization and collaboration, it also encourages internalizing the standard that is used. The main idea of internalization is: for any information created in OIP and moved to OIP, it is standardized but only visible to OIP system and invisible to OIP users. The particular techniques we develop for achieving it are the hiding technique, separation technique and mapping technique, which will be discussed in details in Section III. Internalization allows the standard information in OIP to be personalized, customized and contextualized to adapt to different contextually different systems.

SCI method can be illustrated in Figure 2 as a triangle, where collaboration is made based on standard formats and to achieve standard information representations, which is internalized from being seen by information users. The internalization also transforms the external information to be standardized through collaboration.

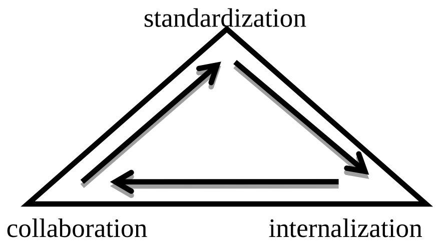

Figure 2: Methodology of resolving contextual difference

In the next section, we design the OIP technical infrastructure based on the XPM standard.

III. OIP Technical Infrastructure

In this Section, we will implement the OIP concept in an OIP technical infrastructure, shown in Figure 3. The implementation consists of three parts, namely, a set of XPM standards for building sign-based vocabularies and documents, a *vocabulary editing mechanism* (VEM) for editing vocabularies, and a *document processing mechanism* (DPM) for editing and processing documents.

The goal of OIP technical infrastructure is to enable information interoperability in both syntax and semantics through a commonly designed vocabulary, which can compose interpretable documents for exchange and use.

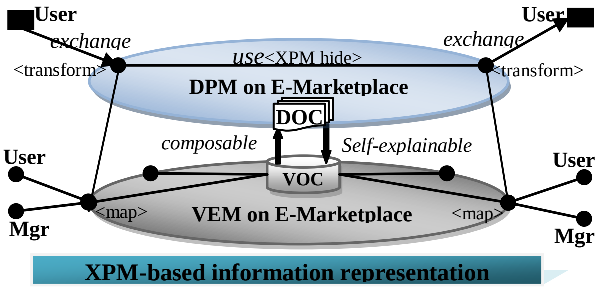

Figure 3: OIP technical infrastructure

To fulfill the goal, the OIP technical infrastructure is designed and implemented in the following use case, applying the SCI method described in Section II.B:

(1) Create three types of roles: e-marketplace designers (EMd), enterprise information manager (Mgr), and enterprise information user (User).

(2) Assign tasks to three roles such that: (a) e-marketplace designers are responsible for collaboratively designing consistent signs for composing common vocabularies (VOC in Figure 3) on vocabulary editing mechanism (VEM in Figure 3); (b) enterprise information managers (Mgr in Figure 3) is responsible for mapping local terms of local vocabulary (LOC) onto VOC; (c) enterprise information users (User in Figure 3) is responsible for using local terms of local vocabulary (LOC) to compose documents (DOC in Figure 6) for information exchange and use.

To support the above use case, OIP technical infrastructure is particularly implemented by XPM, VEM and DPM in the remaining section.

A. XPM

The XPM used in this paper is improved from an earlier version of XPM [13]. Current XPM consists of three applications: an XPM vocabulary specification (XPMV), an XPM document specification (XPMD), and an XPM mapping specification (XPMM).

XPM Vocabulary (XPMV) specification (*see* complete XML schema in [16]) is defined as a sign-based XML vocabulary language as follows:

ELEMENT ::= sign(voc(EMPTY), (term(EMPTY))*)
ATTLIST.voc ::= (iid, an, aid, author, organization, ct) (1)
ATTLIST.term ::= (iid, fc, an, aid, syn, sim, ant, hyp, hol, fcx, ct, st)

In this syntax, “sign” is a root element, “voc” element type defines a vocabulary with attribute types of “iid” (unique identify), “an” (text definition), “aid” (IID list definition), “author”, “organization” and “ct” (sign type for an extended part of speech). The “term” element type defines an atomic sign with attribute types of “iid”, “fc” (term expression), “an”, “aid”, “hyp” (hypernym), hol (holonym), “fcx” (formal context of domain category), “syn” (synonym), “sim” (similar term), “ant” (antonym), “ct” and “st” (sign processing status). The “voc” element can only appear once while “term” element can appear zero to many times.

It is worth mentioning that while “an” defines a sign in plaintext form that may consists of ambiguity, the “aid” defines a sign using a list of sign IID, which is completely unambiguous. The “aid” enables the self-explanation for a sign to look up for an exact meaning without the quest of the external helps.

An alternative XML element-based version of XPMV can be found in [17].

XPM Document (XPMD) specification (*see* complete XML schema in [18]) is defined as a sign-based XML document language with only one XML element “sign” such that a sign leads to a set of signs as a hierarchy following SRF denotation-connotation specification, namely, *sign*(*sign*, …, *sign*). The attributes of “sign” are described in Table 1.

Table 1: XML Attributes of Sign Element

	Name	Type & Description
	tid	xpm:signIID. Unique term identifier
	term	xpm:humanReadableTerm. Text expression
	refs	xpm:QIIDArray. IID list of term expression
	an	xs:string. Text definition of “tid”
	aid	xpm:QIIDArray. IID definition of “tid”
	gt	xpm:gt. Grammar types of document, section, paragraph, sentence, phrase, word, figure, audio, video, data structure, logical expression, axiom, assertion, function, formula
group	*xpm:childrenComputing*. Children signs	
	minOccur	xs:nonNegativeInteger. 0, 1, …,
	maxOccur	xpm:allNNI. 0, 1,…, N, or text
	sot	xpm:setOperationType. Intersection, union, complement
	ctn	xpm:containerType. Set, bag, list, choice, enumeration
	djt	xs:NMTOKEN. Disjoint set
	rch	Xpm:relationCharacteristics. Transitive, symmetric, inverse, funcational, inverseFunctional, entail, equivalent
	qfy	xpm:quantifier. Quantifier and existential relations: all, some, most, half, few, one and %1 to 99%
group	*Struct*. Define data structure	
	dimn	xs:nonNegativeInteger. Define dimension
	leng	xpm:allNNI. Define length
group	*Instruct*. Implement data structure	
	nthDimn	xs:nonNegativeInteger. n-th dimension
	nthLeng	Xs:positiveInteger. n-th length
group	*Vstruct*. Reified value structure	
	op	xpm:operand. Define operand between abstract and reified sign: types for computing, semantics, logic and assertion.
	dt	xpm:dataType. Data types of XSD and XML
	crd	xpm:allNNI. Cardinality
group	*headAttribute*. Define document head	
	lang	xs:language. Natural language
	myns	QIID or namespace. Identifier of document
any	*##any*. Any attribute for incorporating non-XPM XML standards	

The syntax of XPMD is very simple just like XPMV. It is also self-explainable such that any sign can be self-interpreted by looking up for the “refs” and “aid” down to the vocabulary terms or document terms. It is also external open standard incorporable, which is very important.

XPMD is used to represent three aspects of document: data document template as an abstract sign, reified document as a reified sign, and text document. Any abstract sign in a data document template is in the form such as:

<sign tid=“22:3” term=“quantity” refs=“hy2348” gt=“noun”>

Similarly, a reified sign in a data reified document is expressed as:

<sign tid="22:3" term="quantity"
refs="hy2348" gt="noun">Integer(25)

which means that the quantity is 25. For any reified sign, the reified sign is defined by #PCDATA of sign(#PCDATA). Any reified sign will be represented by IID directly. Those cannot exist independently by IID will be expressed by non-IIDed signs enclosed by a data type, such as Integer(25).

XPMD is highly expressive and can express nearly all objects in reality as signs.

XPM Mapping (XPMM) specification (*see* complete XML schema in [19]) is very simple. It defines a sign mapping from two different sign vocabularies, often between common vocabulary (VOC) and a local vocabulary (LOC). The defined specification is:

```
ELEMENT ::= vmap(map)
ATTLIST.vmap ::= (iid, an, aid, author, organization, ct)        (21-1)
ATTLIST.map ::= (ciid, liid, st)
```

where"vmap" is root element and "map" mapping two sign iids from a common vocabulary and a local vocabulary, in which "st" defines the mapping status.

XPM is highly expressive. It can express any linguistic terms, most types of data structure, logical expression, value structure, and document. It is used in four planes of structuration, abstraction, reification and behavior.

B. Vocabulary Editing Mechanism

Vocabulary editing mechanism (VEM) of collaborative sign design for achieving consistent meaning of signs between systems is implemented in the architecture of Figure 7.

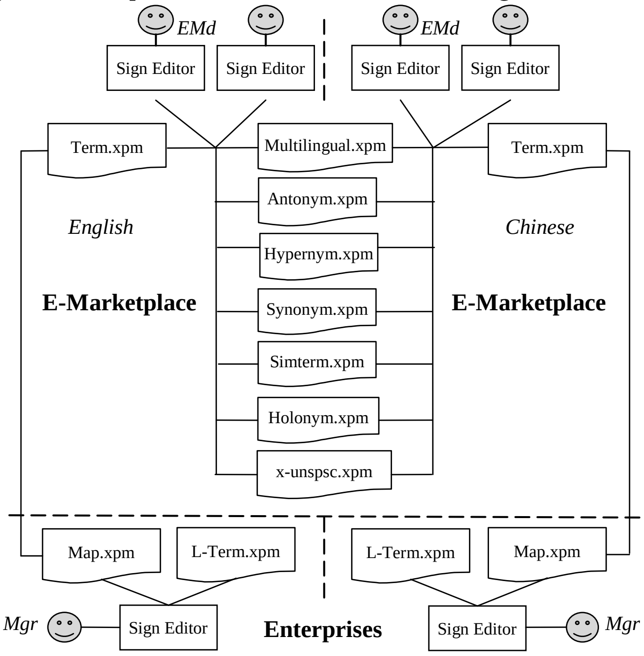

Figure 4: Architecture of vocabulary editing mechanism

In this architecture, sign designers of e-marketplace (EMd) collaboratively design sign vocabularies in language-different data sets in term.xpm, antonym.xpm, hypernym.xpm, synonym.xpm, simterm.xpm (for similar terms) and x-unspsc.xpm. The language different terms are converged in a multilingual sign file multilingual.xpm such that language-different terms of the same meanings share a same sense IID. The enterprise information managers (Mgr) located in different language regions localize (or personalize if necessary) the terms in term.xpm into local terms (L-Term.xpm) and map them in a map.xpm (following *map*(iid_1, iid_2)). By these two processes, all contextually different terms are semantically consistent.

Two particular techniques are developed in implementing the above architecture, which are collaborative sign design and consistent sign identification.

Collaborative sign design is a technique that every atomic sign is collaboratively designed with semantic consistency between e-marketplace designers (EMd) such that:

(1) Assume that any e-marketplace designer is a bilingual speaker, expert at dictionary design;

(2) Given an English sign set as a vocabulary (ENG), let ENG = sign(voc, term[iid, fc_e, an_e, aid_e, fcx, hyp, syn, sim, ant, hol, ct, st]), and given a non-English sign set as a vocabulary NEN, let NEN = sign(voc, term[iid, fc_n, an_n, aid_n, fcx, hyp, syn, sim, ant, hol, ct, st]), where "fc", "an" and "aid" are language-specific and the others are the same;

(3) Initialize ENG = EMPTY and NEN = EMPTY or adopt an existing multilingual dictionary and preprocess it to ENG = NEN such that their "iid", "hyp", "fcx","hol", "syn", "sim", "ant", "ct" and "st" referring to same unique tags;

(4) For ADD any new sign into NEN, the sign must first be added in ENG and then ADD corresponding semantic consistent sign to NEN. Lock the ADDing sign until the operation is finished.

If the ENG-sign has already existed, the ADD of NEN-sign must follow the existing ENG-sign unless a modification operation is issued.

(5) For MODIFY and DELETE any existing NEN-sign, a lock must be issued to the NEN-sign and the corresponding ENG sign until the operation is finished.

This technique guarantees that any non-English signs in different languages are semantically consistent through a principal sign in English. It also guarantees that the concurrent operations of ADD, MODIFY and DELETE from concurrent e-marketplace designers will not generate any inconsistent sign creation or modification. It is worth mentioning, XPM specifies that any sign in a vocabulary is atomic and independent. This makes possible to issue a lock only to a particular sign. The sign-based lock only has minimal impact on designers' concurrent editing operations on collaborative vocabulary editing. However, the impact becomes bigger if a sign is not atomic and designed to be dependent on other signs. This is avoided in XPM implementation.

Consistent sign identification is a technique that every sign is uniquely identified in its creation and use between natural languages and during new sign evolution. This technique can be described as follows:

(1) A unique identifier consists of three parts of X:Y:Z as a 3-dimensional identifier. Part X defines the sign's context sign, Part Y defines the sign itself and Part Z defines the modification history of the sign. For example, given a vocabulary or a document as a sign, X is its sign identifier and a sign in the vocabulary or the document is identified by Y, and Z record a change of sign definition but not change the original meaning. When the original meaning has been changed, Y also changes.

(2) Any sign is sense-oriented with a unique identifier to identify its unique meaning. This is similar to some of the existing researches and practices (e.g. UNSPSC [38]). The sense-oriented identifier is tagged by Y:Z of X:Y:Z and is applied in several sign sets such as:

- X^m:Y:Z for term.xpm and multilingual.xpm;
- X^s:Y:Z for term.xpm and synonym.xpm;
- X^a :$(Y:Z)^{\%}$ for term.xpm and antonym.xpm, where $(Y:Z)^{\%}$ represents an antonym of Y:Z;
- X^r:$(Y:Z)^{\approx}$ for term.xpm and simterm.xpm, where $(Y:Z)^{\approx}$ represents a similar term;
- X^h:$(Y:Z)^+$ for term.xpm and hypernym.xpm where $(Y:Z)^+$ represents a hypernym of Y:Z;
- X^u:$(Y:Z)^{\#}$ for term.xpm and x-unspsc.xpm, where $(Y:Z)^{\#}$ represents classifier of a standard commodity classification (UNSPSC [38]) provided by United Nations, which is extended in XPM sign vocabulary implementation.

The advantages of our 3-dimensional and sense-oriented identifier (i.e. IID) are:

- An IID-ed sign is inferable. It can compare the sense of equivalence, similarity and opposition with another IID-ed sign in the same and different natural languages. It can also find whether as sign is a part-of the whole or within a specified domain.
- Any two signs will be differently identified, which avoids collision in semantic computing, because Part X is uniquely identified in OIP platform.
- No versioning technique is required when new signs are evolutionarily developed. This is because any modification of a sign will be recorded in part Z of X:Y:Z.
- Historical context of a sign has been reserved when reading a sign IID, because Part Y and Z are timestamps of real time when the IID is created. This is also a way of resolving IID collision.

Collaborative sign design and consistent sign identification are two important techniques of SCI method for resolving contextual differences in vocabulary editing mechanism.

C. Document Processing Mechanism

Document processing mechanism (DPM) achieves document creation, exchange and use in a local manner, namely, adapting to local system environments or contexts. The architecture of DPM can be illustrated in Figure 5.

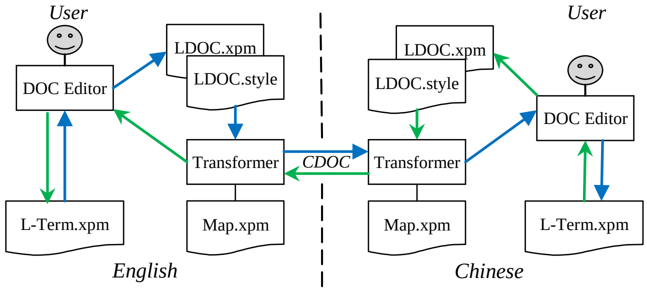

Figure 5: Architecture of document processing mechanism

In this architecture, any two contextual different enterprises are Internet-connected. Enterprise information users (User) do not need to know anything about XPM standards for creating, exchanging and using information but simply to work on a DOC editor as client-side software. DPM will assist sending users to automatically exchange and map the document they create onto the receiver's local signs.

Techniques for implementing DPM are: a *hiding technique* such that XPM specification is invisible to Users, a *separation technique* such that XPM data is separated from its presentation style, and a *mapping technique* such that every local sign has a mapping onto a common sign for transformation through IID. These techniques work together to make the contextual different information exchangeable and usable, as described in the following:

(1) A locally understandable document (LDOC) is created using local sign set (local vocabulary L-Term.xpm) at the sender's side on a DOC editor (like Microsoft Word or Excel) through a sign-based input method (note: this input method is similar to a plug-in Chinese input method. The difference is that its terms are from L-Term.xpm);

(2) LDOC is divided into two parts: an LDOC.xpm document to be a data file for computer understanding and an LDOC.style document to be a style file for displaying LDOC.xpm as a human-understandable file on DOC editor;

(3) LDOC.xpm is transformed into a commonly understandable document (CDOC.xpm) by swapping local IID (LIID) to common IID (CIID) through the sender's local-common mapping vocabulary (Map.xpm);

(4) At the receiver's side, CDOC.xpm is transformed into another LDOC.xpm by swapping common IID to local IID (LIID) using Map.xpm local vocabulary.

(5) The receiver displays the received LDOC.xpm on DOC editor based on the received LDOC.style file.

The key benefits of hiding, separation and mapping techniques for document processing mechanism is that internalization of XPM specifications is achieved. Users do not need to care about the contextual differences and they can personalize and customize their documents in their own ways yet semantic consistency is still maintained.

IV. Related Work

The information interoperability problem described in this paper is a challenging problem in the research area of computer science and information systems. It has long been studied for many years. The research activities can be roughly divided into four stages [33] (we extend a fourth stage from [33]): Stage I covers the period roughly to 1985; Stage II which covers the period through 1995; Stage III covers a period through 2005; Stage IV covers a period yet to be bounded since 2005.

Solutions to information interoperability problems on syntax, structure and semantics in Stage 1 mainly focus on heterogeneous databases. Multidatabases [24] or federated database systems [34] are designed and implemented to resolve the differences in data constraints, structure, query languages and system-level heterogeneity [23][32]. Efforts made in this period are in the aspects of dealing with different data models, schema integration and understanding the differences of schematic and semantic issues. Two federation architectures emerged in this period: a loosely-coupled architecture that provided for a more dynamic or flexible federation and a tightly-coupled architecture that provided for more stable federation. These two architectures provide different supports to data management requirements [33].

In Stage II, two important trends brought more opportunities and changelings in information interoperability, which are (a) proliferation of a variety of data - from structured database, and semi-structured data, to digital media, including visual media [20], and (b) spread of the Internet and emergence of the Web [30]. Applications such as digital libraries [30] and electronic commerce [42] provided the context of interoperability. Issues are found to represent and support broader varieties of data that are not only structured data, but also semi-structured, text, semi-structured and unstructured information. Heterogeneity becomes even higher and the conflicts between various types of information that needs to be resolved. Mediator and metadata were solutions to the problems although metadata still has not ability to bridge heterogeneous systems with different contexts.

Stage III is a direct response on metadata for achieving computer-readable information such that computer can understand with each other through metadata. This brought two aspects of development: standard development to use metadata (e.g. XML 1.0 [2], XHTML [31], XSLT [7], XML Schema [36], SOAP 1.1 [1] and WSDL 1.1 [6]), and domain-specific ontology development (e.g. Gene ontology [35]). Ontology [11] as a specification of conceptualization is a direct extension of domain-specific metadata. Compared with various other classification schemes and structures, including keywords, thesauri, and taxonomies, ontologies are often viewed as allowing more complete and precise domain models. One challenging issue involved in ontology design and use to support semantic interoperability is: how to allow both ontology users and providers to subscribe to existing domain-specific ontologies of their choice or create a new one [33]. An information system may subscribe to multiple independent ontologies and how we can guarantee these subscribed ontologies are semantically consistent since semantic relations such as synonyms, homonyms and hypernyms may be different when inter-ontological relationships are presented. The problem behind this issue is the contextual difference between domain-specific ontologies. The massive research papers on ontology mapping, alignment and integration are presented and can be found in the website of ontology matching [29].

The difficulty of ontology mapping [5][22] lies in the fact that the contexts of independently designed ontologies are different. Context-oriented thinking [12] leads to the current Stage IV of information interoperability research. In this stage, researchers start to think whether it is possible to resolve semantic interoperability problem by collaboratively designing concepts, so that concepts of different contexts could be agreed in meaning at the design stage. This paper inherits the collaborative concept design tradition recently developed. It considers not only the contextual differences between systems but also the concept designers' differences that affect the semantic meaning of the created concepts. It also concerns the users' interpretation on using the designed concepts. By these, this paper provides an open information platform based on collaborative concept design and use by hiding XPM specifications.

V. Conclusion

This paper introduced a new concept of Open Information Platform (OIP) to resolve information interoperability problem that prevents global information exchange and use. It states that contextual differences between Internet-connected systems are the main cause of information interoperability and lead to heterogeneity in syntax, structure and semantics. To resolve contextual differences, an SCI method was proposed based on two principles of naturalization and alienation. The method suggests that a feasible solution must consider standardization, collaboration and internalization. The SCI method is implemented as a technical infrastructure of OIP.

OIP concept proposed in this paper has two contributions: (1) It provided a rather complete understanding of information interoperability that is very much concerned by the research field; (2) It suggested the principles, methodology and realization approaches to resolve information interoperability problems.

OIP concept presented in this paper is a natural step when looking back to the short history of information interoperability research. Nevertheless, this step is significant and could lead the relevant researches to a new era of cross-domain semantic consistency. This is utmost important for the future researches on semantic Web, Web 3.0, electronic commerce, enterprise information systems, social networking and virtual worlds.

OIP concept is new. Much of its implementation needs further exploration, especially on editors' development for usability and performance.

ACKNOWLEDGEMENT

This research is partially supported by University of Macau Research Grant No. MYRG069(Y1-L2)-FST12-GJZ.

REFERENCES

[1] D. Box, D. Ehnebuske, G. Kakivaya, A. Layman, N. Mendelsohn, H. Nielsen, S. Thatte and D. Winter, *Simple Object Access Protocol (SOAP) 1.1*, W3C Note 08 May 2000. [Online]. Available: http://www.w3.org/TR/2000/NOTE-SOAP-20000508/.

[2] T. Bray, J. Paoli and C.M. Sperberg-McQueen, *Extensible Markup Language (XML) 1.0*, W3C Recommendation 10-February-1998. [Online]. Available: http://www.w3.org/TR/1998/REC-xml-19980210.

[3] D. Brickley, R.V. Guha and B. McBride, *RDF Vocabulary Description Language 1.0: RDF Schema*, W3C Recommendation 10 February 2004. [Online]. Available: http://www.w3.org/TR/2004/REC-rdf-schema-20040210/.

[4] G. Cengic and K. Åkesson, "On Formal Analysis of IEC 61499 Applications, Part A: Modeling", *IEEE Transactions on Industrial Informatics*, Vol. 6, No. 2, pp. 136-144, May 2010.

[5] N. Choi, I-Y. Song and H. Han, "A Survey on Ontology Mapping", *SIGMODRecord*, Vol. 35, No. 3, pp. 34-41, Sep. 2006.

[6] E. Christensen, F. Curbera, G. Meredith and S. Weerawarana, *Web Services Description Language (WSDL) 1.1*, W3C Note 15 March 2001. [Online]. Available: http://www.w3.org/TR/2001/NOTE-wsdl-20010315.

[7] J. Clark (ed), *XSL Transformations (XSLT): Version 1.0*, W3C Recommendation 16 November 1999. [Online]. Available: http://www.w3.org/TR/xslt.

[8] T. Cucinotta, L. Palopoli, L. Abeni, D. Faggioli, G. Lipari, "On the Integration of Application Level and Resource Level QoS Control for Real-Time Applications", *IEEE Transactions on Industrial Informatics*, Vol. 6, No. 4, pp. 479-491, Nov 2010.

[9] DAML Ontologies. [Online]. Available: http://www.daml.org/ontologies/keyword.html.

[10] eCl@ss. [Online]. Available: http://www.eclass-serviceportal.com/.

[11] T. Gruber, "A Translation Approach to Portable Ontologies", *Knowledge Acquisition*, Vol. 5, No. 2, pp. 199-220, 1993.

[12] J. Guo and C. Sun, "Context Representation, Transformation and Comparison for Ad Hoc Product Data Exchange", in: *Proc. DocEng'03*, ACM Press, pp.121-130, 2003.

[13] J. Guo, *Collaborative Concept Exchange*, VDM Verlag, Germany, 2008.

[14] J. Guo, "Collaborative conceptualization: towards a conceptual foundation of interoperable electronic product catalogue system design", *Enterprise Information Systems*, Vol.3, No. 1, pp. 59-94, Feb. 2009.

[15] J. Guo, I. Lam, C. Chan and G. Xiao, "Collaboratively Maintaining Semantic Consistency of Heterogeneous Concepts towards a Common Concept Set", in: *Proc. EICS 2010*, ACM Press, pp. 213-218, 2010.

[16] J. Guo, XPM Vocabulary Schema, std v. 09.65. [Online]. Available: http://www.cis.umac.mo/~jzguo/schema/XPMStdVoc_V09R65.xsd.

[17] J. Guo, XPM Vocabulary Schema, alt v. 09.65. [Online]. Available: http://www.cis.umac.mo/~jzguo/schema/XPMVoc_V09R65.xsd.

[18] J. Guo, XPM Document Schema, std v. 09.65. [Online]. Avaliable: http://www.cis.umac.mo/~jzguo/schema/XPMDocSchema0965.xsd.

[19] J. Guo, XPM Mapping Schema, std. v. 09.65. [Online]. Available: http://www.cis.umac.mo/~jzguo/schema/XPMMap_V09R65.xsd.

[20] A. Gupta and R. Jain, "Visual information retrieval",*Communications of the ACM*, Vol. 40, No. 5, pp. 70–79, 1997.

[21] Institute of Electrical and Electronics Engineers,*IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries*,New York, NY, 1990.

[22] Y.Kalfoglou and M. Schorlemmer, "Ontology mapping: the state of the art", *The Knowledge Engineering Review*, Vol. 18, No. 1, pp. 1-31.

[23] W. Kim, I. Choi, S. Gala and M. Scheevel, "On resolving schematic heterogeneity in multidatabase systems", *Distributed Parallel Databases International Journal*, Vol. 1, pp.251-279, 1993.

[24] W. Litwin, J. Boudenant, C. Esculier, A. Ferrier, A. Glorieux, J. La Chimia, K. Kabbaj, C. Moulinoux, P. Rolin, C. Stangret, "SIRIUS: systems for distributed data management",in: Schneider H-J (ed),*Distributed Data Bases*, North-Holland, Netherlands, pp. 311–66, 1982.

[25] D. McGuinness and F. van Harmelen, *OWL Web Ontology Language: Overview*, W3C Recommendation 10 February 2004. [Online]. Available: http://www.w3.org/TR/owl-features/.

[26] R. Mietzner, F. Leymann and T. Unger, "Horizontal and vertical combination of multi-tenancy patterns in service-oriented applications", *Enterprise Information Systems*, Vol. 5, No. 1, pp. 59-77, 2011.

[27] B. Motik, P. Patel-Schneider and B. Parsia (eds), *OWL 2 Web Ontology Language: Structural Specification and Functional-Style Syntax*, W3C Recommendation 27 October 2009. [Online]. Available: http://www.w3.org/TR/2009/REC-owl2-syntax-20091027/.

[28] P. Närman, H. Holm, P. Johnson, J. König, M. Chenine and M. Ekstedt, "Data accuracy assessment using enterprise architecture", *Enterprise Information Systems*, Vol. 5, No. 1, pp. 37-58, 2011.

[29] Ontology Matching. [Online]. Available: www.ontologymatching.org.

[30] A. Paepcke, C. Chang, H. Garcia-Molina and T. Winograd, Interoperability for digital libraries worldwide, *Communications of the ACM*, No. 41, No. 4: 33-43, 1998.

[31] S. Pemberton, et al., *XHTML™ 1.0: The Extensible HyperText Markup Language: A Reformulation of HTML 4 in XML 1.0*, W3C Recommendation 26 January 2000. [Online]. Available: http://www.w3.org/TR/2000/REC-xhtml1-20000126/.

[32] S. Ram (ed), *Heterogeneous distributed database systems*, Special Issue, IEEE Computer, Vol. 24, No.12, 1991.

[33] A. Sheth, "Changing Focus on Interoperability in Information Systems: From System, Syntax, Structure to Semantics", in: Goodchild MF, Egenhofer MJ, Fegeas R and Kottman CA, Eds., *Interoperating Geographic Information Systems*, Norwell, MA: Kluwer, 1999.

[34] A. Sheth and J. Larson, "Federated database systems for managing distributed, heterogeneous, and autonomous database", *ACM Computing Surveys*, Vol. 22, No. 3, pp.183–236, Sep. 1990.

[35] The gene ontology. [Online]. Available: http://www.geneontology.org/.

[36] H. Thompson, D. Beech, M. Maloney and N. Mendelsohn, *XML Schema Part 1: Structures Second Edition*, W3C Recommendation 28 October 2004. [Online]. Available: http://www.w3.org/TR/xmlschema-1/.

[37] Unicode, Unicode Consortium. [Online]. Available : http://unicode.org/.

[38] UNSPSC. [Online]. Available: http://www.unspsc.org/.

[39] M. Vallee, M. Merdan, W. Lepuschitz, G. Koppensteiner, "Decentralized Reconfiguration of a Flexible Transportation System", *IEEE Transactions on Industrial Informatics*, Vol. 7, No. 3, pp. 505-516 , Aug 2011.

[40] E. Xu, M. Wermus and D. Bauman, "Development of an integrated medical supply information system", *Enterprise Information Systems*, Vol. 5, No. 3, pp. 385-399, 2011.

[41] L. Xu, " Enterprise systems: state of the art and future trends", *IEEE Transactions on Industrial Informatics*, Vol.7, No.4, pp. -, Nov 2011.

[42] V. Zwass, "Electronic commerce: structures and issues", *International Journal of Electronic Commerce*, Vol. 1, No. 1, pp. 3-23, Sep. 1996.