Semantics Computation: A Problem Solving Perspective

Yucong Duan, *Member, IACSIT*

***Abstract*—Focusing on phenomena of the existence of multiple semantics for a single concept or expression, we investigated fundamental semantics bundled to concepts of type (TYPE) and instance (INS) to reveal relationships among semantics of problem expressions and expected answers. Semantics of concepts of unlimited and completeness are identified as fundamental classification characteristics for clarifying/formalizing semantics of expressions towards bridging semantics between problems and answers. An approach of semantic computation from problem solving perspectives is proposed with demonstrative cases on previously confusing or misleading issues.**

***Index Terms*—Semantics, Knowledge management, Conceptual modeling, Expression.**

I. Introduction

A. Problems on Existence of Multiple Semantics

Current semantics formalization solutions do not give enough focus on the existences of multiple semantics[1], [2], [3], [4] of concepts, so they are actually at conceptual level regardless of whether they claim to be dealing concepts or semantics at conceptual level. A general visualized formalization of the explicit interplay of "concepts vs. notations vs. semantics" in the complete sense of < human mind, interpretation > for this paper is illustrated in Fig. 1 in four related levels.

Also unification of understanding in the form of introductions of various subjective definitions [5] and enforcement hence usually only solve problems for a short term of a limited scope at the cost of hindering further unification in the long run. This situation is a main source of confusion, misunderstanding, and hindering the advancement of automatic information processing by machine in the whole scale. We explicitly reveal this situation and propose solutions towards ending this obsession. Current problems with semantics can be summarized as four kinds of fundamental issues:

1) Unconscious from unnoticed multiple semantics

Individually enforced semantic is usually not fully ensured as objective target semantics among possibly several candidate semantics. A scenario on information transfer could be as follows:

Expectation: The transfer of semantics is in the form of objective[6] transfer: human1(subjective) → expression (objective) → human2(objective)) → machine(objective). Human1 and human2 refer to different human.

Analysis: The media of transfer is the employed expression containing multiple possible semantics including the expected semantic to be transferred.

Result: The actual transfer of the expected semantic could be implemented as the transfer of one of the multiple semantics contained in the expression other than the expected semantic. The actually form of the transfer is: human1 (subjective)→expression(objective)→human2(subjective)) →machine(mismatched semantic).

2) Inconsistency among subjective enforcements

Problems of inconsistency might arise from absence of an agreed priority when different subjectively enforced semantics on the same concepts meet [5]. The relativity of subjective decisions will hinder the reaching of an agreement in a large scale since an absolute opinion is absent for referring.

3) Undetected gaps and overlaps

Individually derived semantics for parts of a whole system can contain gaps and overlaps of coverages of semantics since they are developed without a unified guide at systemic level.

4) Difficult for extension and modification

Extension and modification of existing semantic systems is hard to perform, e.g., the maintenance of the consistency and completeness for future purposes.

B. Preparation for Discussion

1) For understanding by reader

For understanding the expression of this paper, the semantics of all terms from natural language will be expected to be consistently revealed and distinguished. Only through consistently revelation and distinguishing, can efforts transferred into result semantics be seamlessly integrated as a form of computation which matches to < CLA(classification), ORD(order)> [4], [6]. Also formal semantics can be used for correctly processing at semantic level instead of at conceptual level.

2) Revelation on consistency of TYPE(type) vs. INS(instance)

a) A TYPE level already expressed targeted semantics of consistency on its INSs

::= A TYPE level expression actually bears the consistent/common/shared essence of related INSs at semantic level.

On proof: there are no necessary to prove and also impossible to prove the expression other than revelation of semantics of original expressions at TYPE level.

Alternation: All expressions will be correct/(True) if there is a single proof from corresponding INS level and vice versa.

b) Revelation of the cause of inner consistency

A case of INS::= TYPE/(shared portion) + unique difference.

A metaphor of part vs. whole: an INS is a whole while a TYPE is part of it. This is different from the most popular

Manuscript received August 22, 2011; revised November 29, 2011.

Yucong Duan is currently with the DISCO, University of Milano Bicocca, Milano, Italy, 20126 (e-mail: duanyucong@hotmail.com).

view where the confirming/identification relationship extended from INSs to a target TYPE is taken as a proof of a converse part vs. whole match.

c) Limitless vs. directional/order for TYPE

The same portion of INSs which composes a TYPE is actually the complete/CWA content of a TYPE. An added direction/tendency of the development of related INSs in subsequent discussion will only lead to more difficulty to understand the phenomena of INSs at TYPE expression level if the direction of extension itself is not conceptualized and explicitly managed as a CPT. If a direction is introduced as a new thing which is not able to be classified as either INS or TYPE in an expression system, it is actually introduced as an unknown thing unconsciously which will result in the failure of any efforts towards explain the existing things in a consistent manner if it does not end with a reduction to a "unknown".

3) Boundary of language based expressions/thoughts with correct vs. wrong

Subjectively giving decisions on correct vs. wrong is all what we can do/conclude consciously with our expression languages in the strict sense of both completeness and consistency: <completeness, consistency>. All we can say that it is the boundary of the capacity of language based expression/thought.

4) The scope of the target

The problem is how to recognize and identify the concepts of TYPE and INS. Actually there are only three kinds of expressions which we can derive based on above hypotheses:

Expression::=<at TYPE level, at INS level, mixed level>

a) Expression at TYPE level

Semantic revelation:

TYPE::= <INS>;

Entity(ENT) of expression ::= <TYPE, INS>;

Relationship of expression ::=<ORD>;

Behavior::= <ORD>

(or behavior*::= <(ORD-(ENT-ORD)*-ORD>)*

Identification of TYPE:

Characteristics of TYPE level::= <unlimited completeness of consistency>

b) Expression at INS level

There is no expression at INS level directly since that it is at the bottom position to initiate an original cognition.

c) Mixed expression of TYPE and INS level

This kind of expression should be an expression with no conscious semantics or as a mistake, since that it cannot match to any reasonable semantics which can be derived from semantics of either at TYPE level or at INS level.

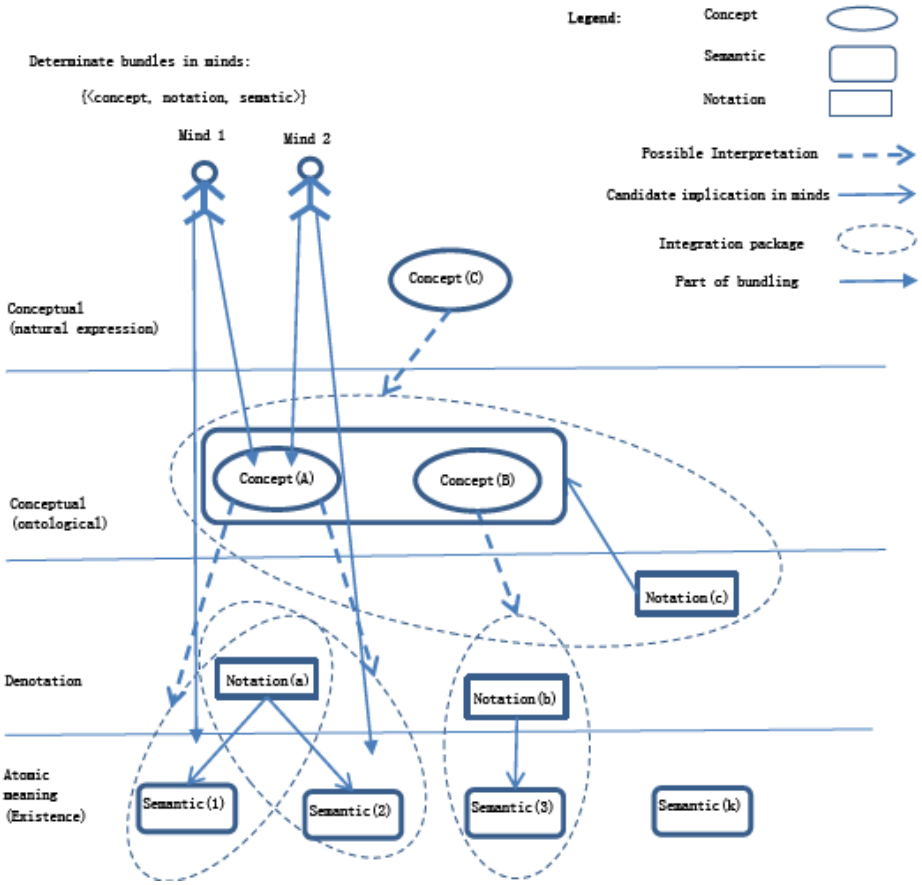

Fig. 1. Visualized formalization of "concepts vs. notations vs. semantics"[3].

II. Expectations and Initiatives

A. Idelogy: Approaching Realization Through Becoming Real

We aim to deal with semantics of NL (natural language)[1], [2] expressions in an ultimate manner. We expect that readers can gain a clearer view on semantics vs. concepts from our revelation. Implementation of the proposed method will help them to conquer the limitation of unconsciousness at conceptual level and reach consciousness for understanding and creating consistent semantics of expressions. Qualities such as efficiency and simplicity in contrast to existing NL

semantics processing approaches can be expected as extensions of our clarification work.

From our approach, a consistency can be maintained during the extension of our approach at semantic level. This will be a breakthrough in contrast to current situations where most people just stay at the conceptual level during their communication which was limited by that they are thinking within the expression capability of languages concepts at conceptual level instead of the semantics at semantic level. They are unconscious with this situation, so they will raise expressions of "questions" which actually lay solely at conceptual level. And the pursuing of answers to these questions will only reach interrelationships (among these concepts) of semantics of relativities, incompleteness/gaps or seemingly limitless.

B. Initiatives: Raising Open Problems

1) Can we approach something like Y(yes)/N(no) or T(true)/F(false)?

Both Y/N and T/F are created based on the dualism of human vs. machine [1], [2], [8]. All what can be gained is nothing more than consistency. Also consistency is the ultimate relationship which will compose all what is required between an integrated system of semantics and discrete individual semantics. Based on the dualism [6], we identified the Y/N and T/F as follows:

Y/N::=semantics from individual human beings.

It is subjective as that it is only limited to the speaker itself.

T/F::=semantics which are shared by more than one individual.

It is objective as it is shared by individuals. Objective embodies the semantic of deny/negation if one human is changed while the other is not changed simultaneously.

2) What kind of theory is demanded?

Actually a theory of all theories is demanded as the solution. Even if the theory is identified, a feasible expression will be another challenge.

3) What is our reaction for this challenge?

We have only the courage to restart or to face empty again. We believe that it is a shortcut to get to the bottom since that every beginning include fundamental level of semantics starts from nothing or empty[1], [2].

4) Obsessions on extending from existing work

If we base our solutions on existing subjective or relative definitions, the result will inherit the limitation of the subjective of existing work or staying at the same relative semantic level or at conceptual level.

5) Conceptualization from existence

From conceptualization: Individual semantics and their relationships can be clarified through referring semantics and relationships derived through conceptualization [2]. Semantics extended from conceptualization will be objective. Initial cases are shown subsequently to aid the understanding of this approach and potential applications.

III. Core Concepts Discussion

A. Multiple Semantics of an Expression vs. Concrete Semantic

The source of unconsciousness and absurdness: concrete semantics are identified individual/atomic semantics from a semantic integration. Most expressions such as in the form of NL are integrations of multiple semantics [2], [3],[4]. The coexistence of multiple semantics is the source of confusing and mistakes unconsciously. Unconsciousness can be created when accepted semantics through reading the same expression containing multiple semantics are assumed as the same while turning out to be actually different. Similarly absurdness can be incurred when an expression contains an intended semantic while there are other semantics which are not clearly excluded in the expression by an author for audiences.

B. Views from TYPE Level

Computation actually functions as constructing a window/perspective to view the data.

Existence::= pure unlimited.

Identification::=completeness

TYPE::= completeness and unlimited.

INS::= it is relative to TYPE. The focused semantic is its incompleteness in comparison with the completeness of the corresponding TYPE.

CPT(concept)::=The first semantic of a CPT is TYPE. Or a CPT equals to TYPE at the beginning of a conceptualization.

C. For Type Identification

We proposed that behaviors/operators such as "+" and "unlimited" can be in cooperated in the form of composed operations in TYPE expressions without any side effects on maintaining the expression at TYPE level.

D. Conservation of Existences of Semantics for Semantics Computation

The existence of individual semantics cannot be created and destroyed [1], [2] during reasoning processes of an expression at existence/ontological level. Subsequently the amount of existences of individual semantics will maintain the same for the whole process of semantics analysis of an expression.

E. Existence(E) of Unlimited vs. Limited

A rule on existence of semantics: an individual/concrete semantic of "unlimited" matches to a single E=1 of semantic "unlimited" which can neither be reduced to none existence nor be created from nothing.

Therefore there will be always limited amount of "live cycle" for "O_1x+O_2" extended from "3x+1" once the O_1 and O_2 is decided. If reasoning, from this rule, a semantic computation at the level of the amount of existences of semantics of "unlimited" and "limited", there will be no semantic of "unlimited" which can be created for the semantic of the expression of a result.

F. Conceptualization vs. Virtualization for Cloud Computing

At TYPE level, usually a target expression is already an expression at TYPE level which however is not the most efficient one since that it is a mix of multiple levels of semantics if it is not a wrong expression from the view of an integrated system of semantics. So it is less efficient in the sense of representing a more general theory at a higher level of semantic through a case by case manner. Our system will support to reveal these phenomena and supply absent

theories to improve the efficiency.

G. Integrating Knowledge Expressions Through Semantics Conceptualization

Motivation: current situation is that engineering knowledge in computing area is developed rapidly to satisfy the request of new problems. To attain and maintain the controllability of the growing body of knowledge, a systematization of the knowledge body which is shared by all is desired. We believe that knowledge expressions can be organized or integrated by ways of semantics organization of their composing concepts. Starting from exploring language expressions of related knowledge, we identify that relativity and multiple existences of semantics of expressions are the source of confusion and subjectivity of existing solutions. To cope with these deficiencies, we propose to start from conceptualization to reconstruct the semantics and their relationships of concepts. Examples are demonstrated to show that concepts can be reorganized by ways of semantic reconstruction.

IV. Framework of our Approach

A. Semantics Modeling and Computation vs. Conceptual Modeling

Most existing presentations are done at conceptual level and represented with concepts instead of semantics consciously or unconsciously regardless of whether they are expected to be at semantic level or not. However usually behind representations in the form of CPTs, there might be more possibilities of semantics compositions than the expected one. If the expected semantic does not exist behind a representation in the form of CPTs, it is an unexpected situation or a mistake of presentation. We do not consider this situation in the subsequent discussion since that it is a situation that the representation is meaningless to be discussed more as that the content contains no expected result.

What are we capable of with conceptualization and semantics modeling and computation? We are capable of locate and represent semantics instead of concepts. We are capable of evaluating, relieving and resolving the vagueness, overlap, inconsistency, gaps and extension of semantics of results of conceptual modeling. Current computation: For mathematics, some computations are at conceptual level. Or they are mixed expressions crossing several semantics levels. With the aid of computer, they can speed the completion of lower level instances of computation in a manner of case by case. However this cannot help the situation where unlimited amount of target is a target. Unlimited actually implies the existence of gaps among different semantics represented by TYPTs at conceptual level. Without the introduction of semantics computation, it is an unconquerable obstacle for reasoning at conceptual level.

Obsession for progress from conceptual level: It is impossible to realize the absence and necessary of a form of computation other than at conceptual level within the reasoning capability of conceptual level.

Semantic computation: Semantic computation will construct formulas at a higher semantic level which will efficiently solve the problem of representing unlimited at a lower semantic level.

Core/essence of semantic computation and problem solving: Actually the computation is to find the formula which will match to the representation of certain facts. The formula is a representation at TYPE level. So the creation of the formula is to find the right representation of intended problems at semantic level which are often implicit while are taken as explicit at conceptual level unconsciously. Or in another word, based on the analyzing and locating of semantics instead concepts of unconscious authors, rewrite or construct a proper representation of the content of intended semantics in a description level at the highest semantic level of involved semantics at original problem descriptions. The ideology in this process confirms to our previous proposal that the presentation of answers lies in the proper presentations of problems [1], [2]. The demanded work is to locate and represent the intended semantic from the multiple semantics embodied in the original representations of CPTs.

Absolute conclusions on semantic computation: It is the only form of computation which is expected and can be actually implemented.

Semantic::=a visible relationship(REL) by an observer or existing REL.

Existence::=confirmed meaningful on any entity/CPT.

Meaningful can only be meaningful to human being::= REL(human→CPT)

Conceptualization::=develop CPTs from the outside matter through the mind of a human. The result of conceptualization will be meaningful. It exists in the system of the mind of a human after the state of conceptualization.

Not exist::= not shared “CPTs” or “CPTs” which “exist” exclusively in other “people”s’ minds.

B. Help Readers to Understand what They Really Mean in the Manner of Thought with Languages

We are not always clear with our thoughts when we think with our languages inside our brains if we are doing it in the manner of conceptual modeling instead of semantic modeling. Also we might find problems of transitions among two systems, e.g., we might have difficult to express what we really want to express, share or communication. Our work aims at revealing these mechanisms and aiding people to understand in a clearer manner, get near to the essence and keep in accordance with their truth/false.

C. Semantics vs. Concepts

Concepts::= identified semantics by individuals and expressed with notations to communicate ideas.

Semantics::= intentions by individuals.

They are at different levels of reasoning. The semantics of concepts could be implicit among several explanations. The relationships among them could be in the form of cardinalities of “1:1”, “1:n” and “n:1”.

D. Principle of Semantics Computation

Semantic computation::=<complete trace, consistent use>

Complete trace::= trace all related semantics instead of independent concepts for a computation. All related semantics actually identify/form a semantic state.

Consistent use::=use all semantics which are brought/changed by an operation.

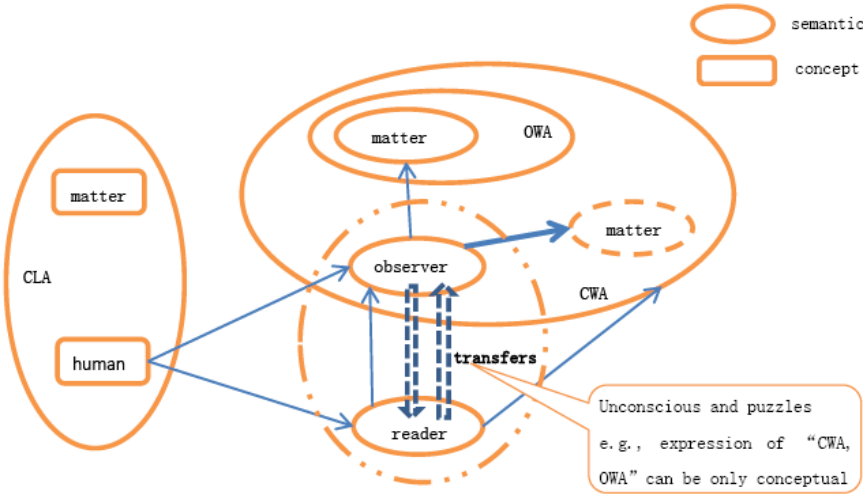

Fig. 2. Original evolution of semantics and concepts from dualism.

Computation::=<CLA>
Consistent::=<ORD>

E. From the Dualism: Semantics, Concepts, CLA, OWA, CWA[3]

CWA(closed world assumption)::= complete
OWA(open world assumption)::= partial$|_{reader}$
ORD::= directional and consistent

Examples for unconscious and puzzles which corresponding to Fig. 2: e.g., expression of "CWA, OWA" can be only pure conceptual. It will not have a valid semantic as that it will never form a valid state. It is similar for "True and false at the same time" or "Yes and No at the same time". These expressions will stay at the notation form of conceptual expressions and bear no valid semantic.

F. All meaningful Computation/Change

Computation/change: it is for expressions and semantics which can be reflected inside minds

There is only one meaningful behavior/change/verb/action: CLA

Complete result: There is only one result: theories/TYPE vs. INS

Consistency: If a INS can be found, it means that for a set of similar INS or expression in the form of TYPE it will found. (This is a CPT level explanation. At semantic level all the INSs under the same name of CPT share a semantic which is the focus of the implicit intention expressed at corresponding CPT level.)

There is no valid "=" at semantic level because semantics are all unique [3].

G. Towards Modeling Knowledge

Model vs. expression: The modeling of knowledge firstly takes place in the mind of the modeler.

Model::= a reflection of organized things in the mind.

Organized::= related by human being. (Confirmed by Yes/No)

EID-SCE [3]+ CWA based deduction:

Semantic::= related/relationship(REL).

Hypotheses: there is no overlap of the usage of notations. The amount of basic things can be identified by the amount of basic semantics/RELs. The present of the model in the world outside the mind demands the expression of the model to realize the transfer of the information of the model to others' minds.

Expression::= complete presentation of a model which is independent of a mind. (Transfer from subjective/implicit Yes/No → objective/explicit True/False)

Expression::= semantics/RELs + notations
::= concepts(CPTs)

Concept::= originate from reflection of observation by mind or conceptualization.

Computation::= <*CLA, ORD*>*::=* {<$REL|_{source}$, $notation|_{source}$>→<$REL|_{result}$, $notation|_{result}$>}

H. OBJ (objective)vs. SUBJ(subjective)

At semantic level instead of at conceptual level, if OBJ is put as first existence, a consistent OBJ system can be derived. Then the rest which has been considered other than the system in the mind is SUBJ. If the SUBJ is put as first existence, OBJ system has to be identified from the existing SUBJ.

V. Methodology Extended from EID-SCE

Systemically retrospect on the semantics vs. concepts of *EID-SCE(3), the general deduction mode: OWA→CWA along the expansion flow of (CWA→CWA)**

The general mode of *EID-SCE [3]:* Transformation (<DES(description), IMP(implementation)> $|_{<Yes/No,\ True/False>}$ → <DES, IMP>$|_{<True/False>}$)$|_{(Yes/No \to True/False)}$

Criteria of the transformation:

ORD: SUBJ(subjectiv) →OBJ(objective)

Ontology::= <RELs>

A sketch of main ideas of EID-SCE from problem solving perspectives can be found in Fig. 3, Fig. 4, Fig. 5 and Fig. 6.

The common misunderstanding is that a solution is something completely new in the sense of existence. This is actually a subjective assumption which is implicit and not challenged usually.

Unknown::= Firstly from the OWA, a known is not all/complete. Secondly from CWA, the existence of unknown can be identified. Since arbitrary assumptions are applied, it is a subjective decision. When a unknown is confirmed as existence, it starts to have a layer of semantic of known. Unknown need to be distinguished from unexplored.

Problem::= it is a unknown. When its subjectivity is taken

as objectivity, the concept of answers will be created as relative to it.

Answer::= it is actually derived by traversing existing semantics of known and identifying missing links at various phases, e.g., conceptualization, which are referred as unknown in problem descriptions. The present of an answer should be complete which contains no gaps and missing links.

Knowledge ::=<description, implementation>
::=<CLA,ORD>

So knowledge can be used to replace computation as reusable computation.

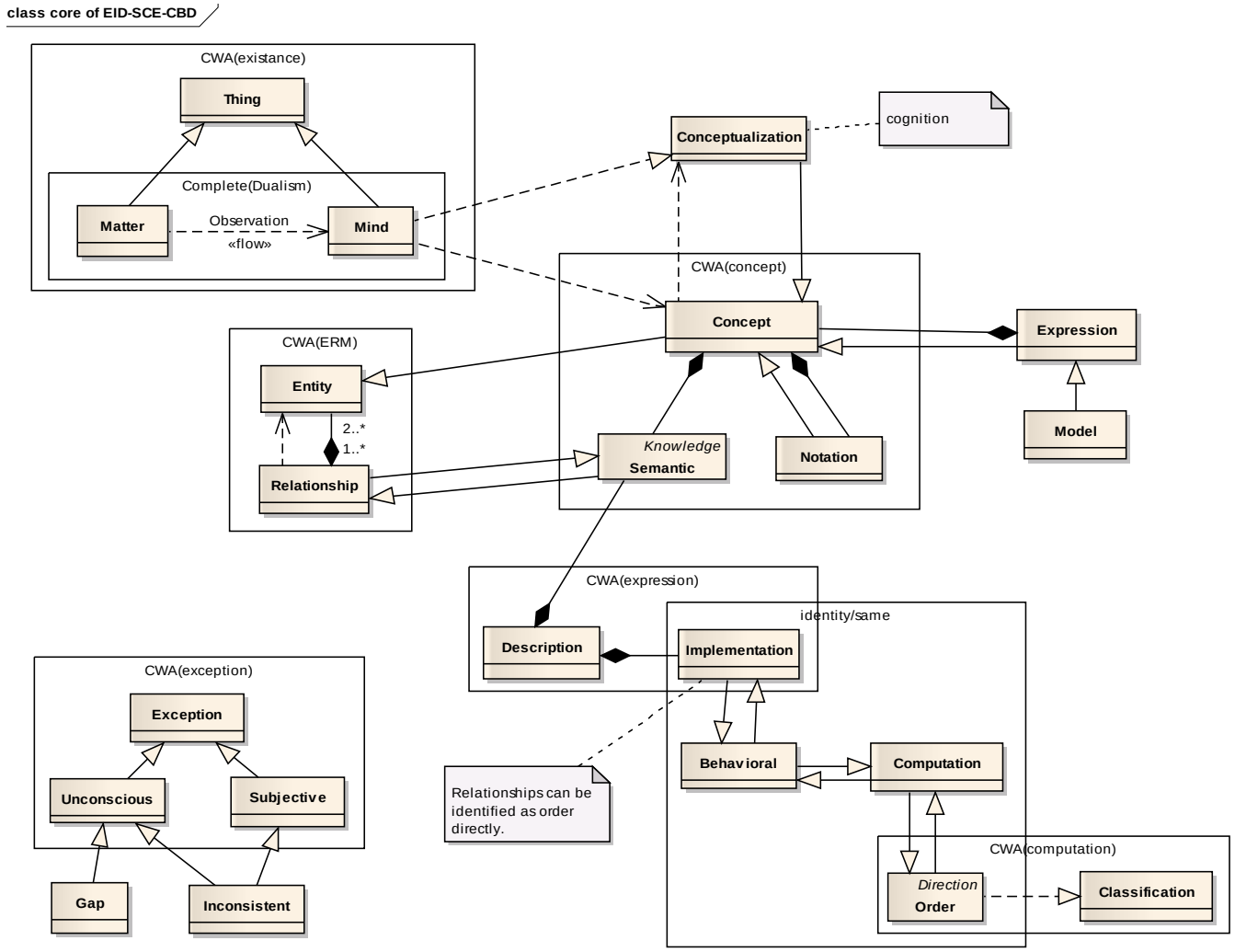

Fig. 3. Structured core classes of EID-SCE[4].

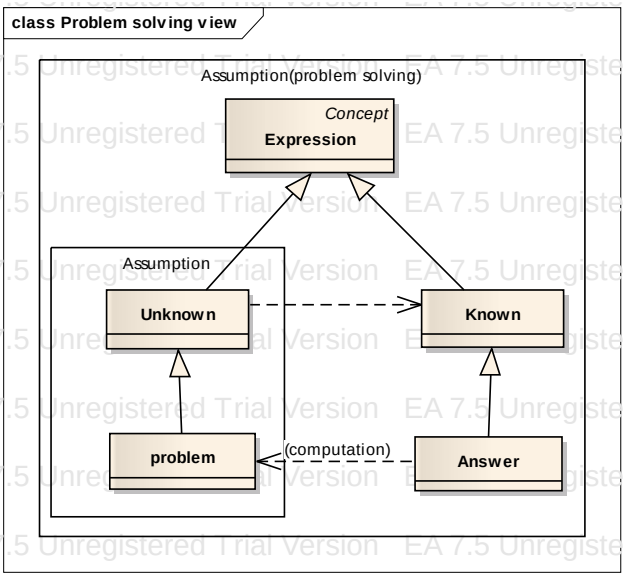

Fig. 4. Class structure of problem solving.

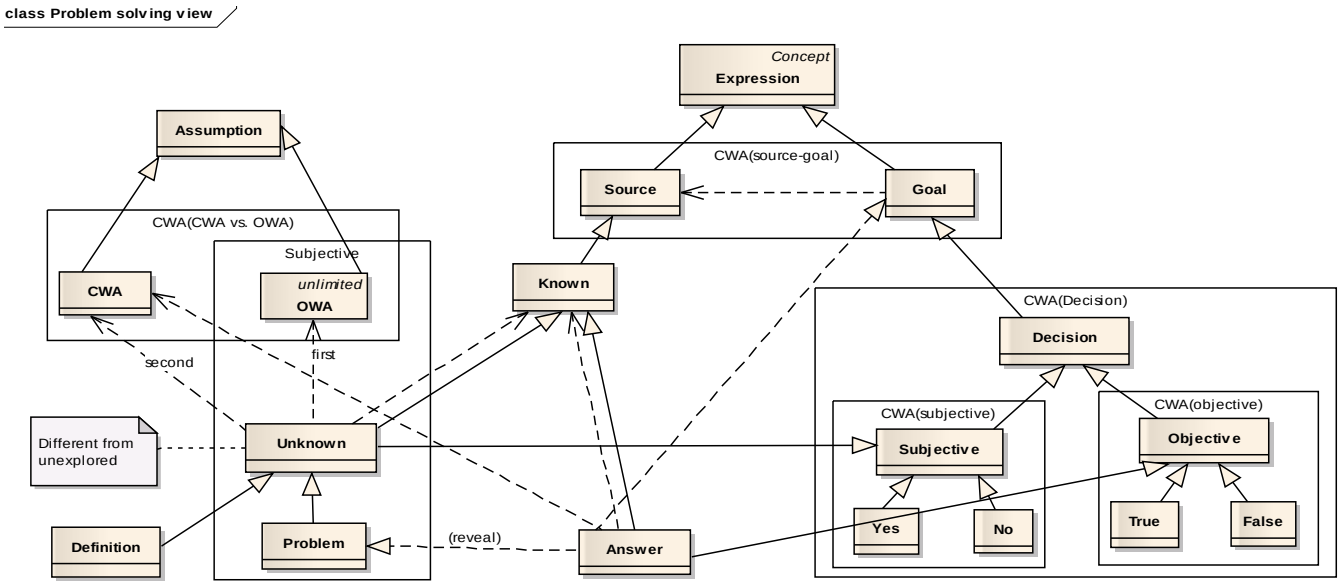

Fig. 5. Problem solving metamodel[12].

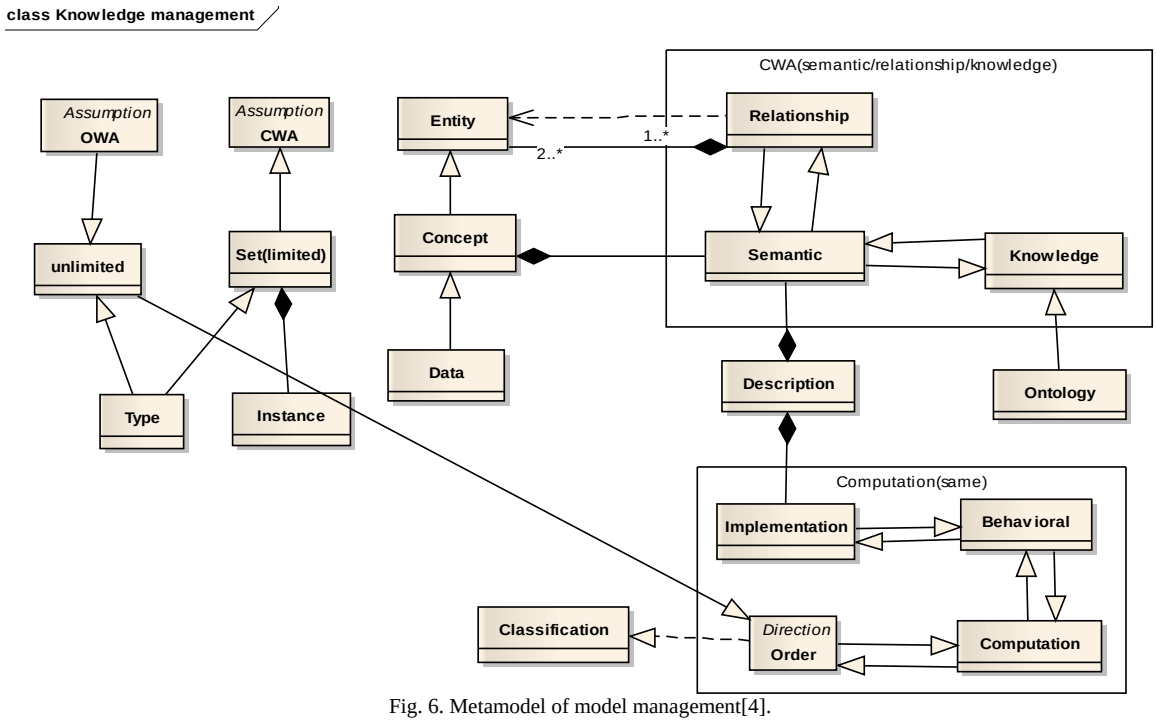

Fig. 6. Metamodel of model management[4].

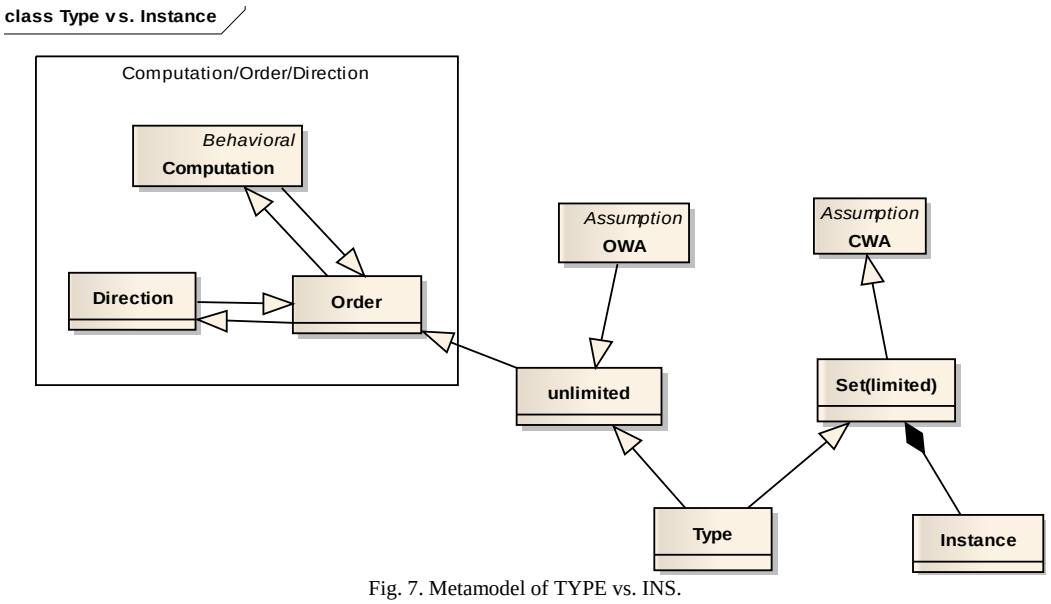

Fig. 7. Metamodel of TYPE vs. INS.

VI. Case Studies

A. Case from the process of learning new language

Problems related to unconsciously mistaken concepts as semantics towards forming intended semantic chain can be revealed explicitly in scenarios of translations among languages. For the case of translation from English to Chinese: "A pigeon covers the distance within five minutes", the word "cover" can be matched from English to Chinese words in a dictionary. However that matching could only represent a limited match of several semantics from source to target concepts. A translation actually demands the translation at semantic level. In this case, an alternation for the intended semantic for this sentence must be found to replace the answer indicated in a word by word form.

Some hypotheses can also be draw from this case:

1. Language concepts are bundled to semantics not strictly but by accumulated experiences.
2. There is still much space for semantics to be adopted into related concepts.
3. Since that most semantics in natural languages are affiliated to a collection of relatively independent concepts, efficiency can be brought by the explicitly introduction of a systematization/integration in the form of ontology, etc.

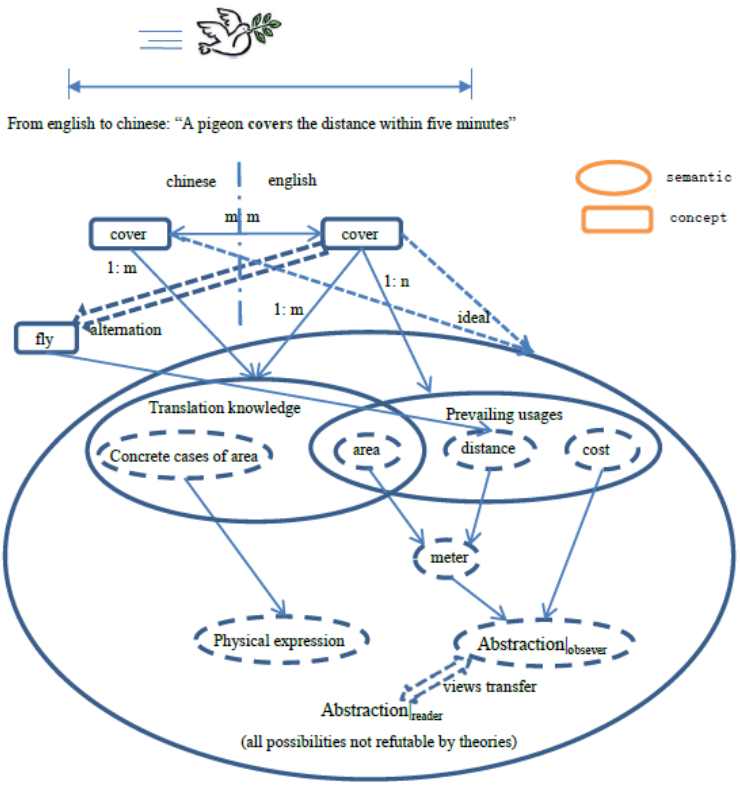

Fig. 8. A case on literal multiple semantics.

B. An alternation of TYPE-INS consistency: the case of "3x+1"[7]

$3x+1::=2^{n->unlimited}$

Is this expression at TYPE level?

$2^{n->unlimited}$ ::= <unlimited extension, elements from N>. It is a TYPE.

About "1":

$1^{n->unlimited}$ ::= <unlimited extension, elements from N>. It is a TYPE.

Also E::=O+1 or O::=E+1 can confirm that it is a TYPE.

x::=O. It is at TYPE level.

3::=1+1+1. It is at TYPE level.

So "3O+1" is at TYPE level.

The problem is whether the result at TYPE level is a fixed point or a scope containing live cycle of fixed points. So it is necessary to study all possible situations which might result from the expression.

The repeated scope/modes of "3O+*n*" are: For the study of the numbers, it is important to study them from a uniform coordinate to avoid the confusion of "E::=O+O" vs. "O::= E/2".

(a) $(3O+1::=E)/2^n \rightarrow O$

(b) $(3O+2::=E) \rightarrow 3O+1$

(c) $(3O+3 ::=3E) \rightarrow E$

(d) $E/2^n \rightarrow O$

By restricting all the variation of TYPEs to {1, 2, 3, E, O}, it can be found that "3O+1" $\rightarrow$ $(E/2^n \rightarrow O)$ will be the final mode for the computation. By considering all variations of the computation mode, there is no live cycle. No live cycle inside means that all states in this cycle will not be eternal. In another word, the computation will not be prohibited to reach any number including 2^n. That is all what can be concluded related to this expression. Here confirmation of no negation will mean reachability or confirmation at TYPE level. TYPE level is a CWA which takes relationship of unlimited from TYPE to INS as an element of the discussion.

The case of "5*x*+1" is an expression at TYPE level. The problem is that the result can construct live cycles which will never form a 2^n since that the phenomena of "5*x*+1" and "5*x*+3" cannot be distinguished or they can be viewed as equal in composing a live cycle of <"5*x*+1"↔"5*x*+3">. Semantically <semantic$|_{result}$> is determined by the problem

description, everything which cannot be excluded by exploring all possibility must be considered as positive/T(true) at TYPE level discussion. The semantic which is implied in a TYPE level expression will be confirmed by a case at INS level. Here the appearance of a live cycle of <"5*x*+1" ↔"5*x*+3"> is not distinguishable as E and can construct a live cycle.

C. The Case of "E(even)::=P(prime)+P(prime)"

P(prime) is a TYPE in the sense of that it is a TYPE identified inside N(natural) by negation/excluding C(composed). C(composed) is a TYPE as that it meets the requirement of being TYPE: <unlimited extension(compose), elements from N>. N is a CWA for P.

"E(even)::= P+P" is an expression at TYPE level. There are more than one case at INS level meeting the T(true) of the expression. So the expression is supported by this TYPE-INS system.

Similar cases include: E::=E+E, E::=O+O, E::=P+O, etc.

D. Description View vs. Implementation View

Implementation *(IMP)/behavioral (BHV) view::= the change is connected by RELs which only represent direction(ORD) to connect entities. A composed form of ORD is classification(CLA).*

EID-*SCE:*

Description*(DES) view::= relative to IMP/BHV view, it refers to the indirect presentation of the RELs which link the target and source.*

Knowledge::= expressions of RELs.

Empirically it refers to indirect presentation in the form of RELs other than direct expression of <CLA, ORD>. Knowledge is expected to be useful for guiding the achieving of the direct RELs of <CLA, ORD>.

Notice: Discrete knowledge expressions which loss links to their conceptualization will result in various gaps/misunderstandings of a cognition process which has been claimed as various questions. The ultimate solution towards a throughout cognitive connection can only be achieved by revealing missing links. Any otherwise claimed solutions usually are just replacing existing gaps with other gaps or stop at some beliefs/hypotheses unconsciously.

Computation$|_{IMP}$::= change of {<$REL|_{source}$, $notation|_{source}$> →<$REL|_{result}$, $notation|_{result}$>}

Computation$|_{DES}$*::= can be expressed as knowledge from expressive view.*

VII. Related Work

A. In Comparison with Mathematical Induction

The idea of mathematical induction shares a seemingly similar idea of ours. However mathematical induction was used without distinguishing at conceptual level or semantic level. So the result cannot be ultimate persuasive when it was applied to explicit conceptual expressions which might implie multiple semantics. Also the conclusion could contain multiple semantics under a seemingly uniform conceptual expression. A key point is that they lack an explicit treatment with "limitless"/"unlimited" which is a key point in our approach.

B. In Comparison with Other Knowledge Acquisition Modes

Most other existing knowledge acquisition modes including so called deduction, complete or incomplete knowledge acquisition, etc., are at the level of conceptual reasoning/modeling while they are unconsciously taken as at semantic level. So they will suffer from the relativity of conceptual and the vague transfer among multiple semantics. Even the claimed completeness of related knowledge might be incomplete since basic concepts are with multiple semantics as well. In general, they are incomplete semantic knowledge acquisition limited at conceptual level from the view of semantic level. Any confirmations of observations and conceptual expressions are only confirmation at conceptual level which can be mapped to some of multiple semantics which could be bundled to conceptual expressions. A case is the black vs. white swan problem. Actually the definition or semantic of "swan" is an inseparable part of the semantics. If this part is omitted, the reasoning will be a case of the taking an engineering/empirical problem as a scientific one.

C. Other Literature

Giovanni Vico's work [8] on exploring the "science of reasoning" can be seen as an interesting and empirical effort of semantic computing. Our previous work [3], [4], [5] have shared some similar views with Giovanni Vico's ideas such as "the true itself is made". Since that a conceptualization process cannot derive concepts such as Y/N vs. T/F directly from a dualism, they have to be introduced subjectively. Giovanni Vico[9] has proposed empirically/intuitionally in natural language as "we can prove geometry because we created them. If we are said to be able to prove physics, it is because that we once created them." We are also going to reveal in a formal manner one of the semantics inside this expression at TYPE vs. INS discussions from our prerequisite that all actions/behaviors which take place after the finish of the corresponding expressing of expressions.

Some of past philosophers contended that realize recognition of things through creation [9]. We have tried to justify this belief at a conceptualization process of new concepts.

VIII. Summary and Future Work

A. Summary of Our Approach

We propose that semantics computing is an ultimately integration of the abstraction of thought and the concrete of implementation. Semantic computation is based on semantic exploration at ontological/existence level. Every identified computation is actually semantic computation if the cognitive process is reminded explicitly. We found that semantics can be located by organized concepts such as organized concepts, e.g., ontology, metamodel, models, etc. Our hypothesis is that a human think with his/her languages. Then expressions composed by these languages construct the content and define the <BOUNDARY> (boundary) of what we can think as a problem or not a problem, and also as an answer or not an answer. We construct computer to aid the automatic processing of what we construct as a problem or for

implementation purpose only. For finding an answer at TYPE level, the unlimited will invalidate all attempts since that concrete computation by machine is always limited. The essence of computation by machine can do nothing more than <ORD>. If a machine seems to be capable of <CLA>, the <CLA> is a result of transformation of <CLA> → (SUBJ→OBJ) → <ORD>. The completeness of the <BOUNDARY> actually means that answers and problems are always coexisting inside expressions of humans. We see semantic of these problems of these problem expressions because we see part of the semantics of these expressions. Or in other word, we see these problems because that we do not see the complete semantics from the expression which actually matches always to a completeness of integration of partial semantics at TYPE level.

Revelation of this principle of semantics at TYPE level will saves the effort to explore and understand related problem expressions at non TYPE level. For problem solutions exploring, this revelation will point out a road which aims at exploring the answers through identifying the completeness of semantics of problem expressions at TYPE level to guide implementation at non TYPE level. Similar cases include that "P vs. NP" can be matched to "P→ORD→INS vs. NP→CLA→TYPE" at TYPE level.

A summary of principles of our semantic exploration is that:

1. A piece of absolute semantic must be identified as the start or coordinate for the extension of subsequent semantics reasoning. That piece of semantic is not necessarily to be identified as an independent/complete piece of semantic from other considerations. Then this piece of semantic is not necessary to be the only choice from related semantics.
2. The target to be dealt with should be individual semantic instead of an integration of them or CPTs directly.
3. The individual semantics are the confirmed existence of interest.

The key ideas proposed in our methodology include:

Unlimited: the semantic of unlimited is conceptualized as a noun instead of an adjective/verb in our approach in contrast to most existing analysis. Then the semantic of unlimited is considered as part of the semantic content of TYPE.

Existence: := the confirmation/(Y→T) of the result of conceptualization in the form of either INS or TYPE.

B. In Contrast with Conceptual Modeling

From our analysis, we can foresee that the core of next generation of computation will be semantics computation or shifting from current partially unconscious conceptual computation with partially implicit semantics toward explicit semantics computation.

At a strategic level, semantic computing will mean that:

1. We will be clearer with our intentions after shifting from thinking/expressing with languages concepts at conceptual level to thinking/expressing with semantics.
2. We will be clear with the semantics instead of the concepts of current knowledge expressions. The knowledge can be reorganized at explicit semantics levels instead of at implicit semantics levels embedded in concepts of expressions with multiple semantics possibilities.
3. Based on above, we can understand the semantics instead of the concepts of the problem expressions and provide optimized answers to decompose complexities.

It will be different from current practices of modeling at conceptual level and solving with implicit assumptions if not unconsciousness.

In the future, we are going to extend the application to extensive expressions of areas including artificial intelligence, cognitive sciences, logic research and natural language literature, etc., to reveal implicit vs. explicit semantics of Y/N vs. T/F in a consistent manner.

ACKNOWLEDGMENT

The ideas contained in this paper chapter are the author's own, and do not represent the views of the organization he works for.

REFERENCES

[1] Y. Duan, C.Cruz. "Formalizing Semantic of Natural Language through Conceptualization from Existence". *IJIMT, V2(1)*, pp. 37-42.

[2] Y. Duan. "Attaining and applying consistency from semantic evolved from conceptualization" ,*IC4E2011*, IEEE CS press, pp353-359.

[3] Y. Duan. "Catering quality evaluation design for service/cloud computing through visualized semantics locating", *SSNE 2011*, IEEE CS press. (in press)

[4] Y. Duan. "Semantics computation: towards identifying answers from problem expressions", *SSNE 2011*, IEEE CS press . (in press)

[5] Y. Duan, C. Cruz and C. Nicolle. "Propose Semantic Formalization for 3D Reconstruction of Architectural Objects", IJCIS, 11(1), 2010, pp 1-10.

[6] Y. Duan, "A Dualism Based Semantics Formalization Mechanism for Model Driven Engineering", *IJSSCI* vol. 4, IGI press 2009, page 90-110.

[7] De Nostri temporis studiorum ratione (1709); On the Study Methods of our Times, translated by Elio Gianturco, with, as an addendum, *The Acadamies and the Relation between Philosophy and Eloquence* [*1737*], translated by Donald Phillip Verene. (Ithaca: Cornell University Press, 1990).

[8] Yucong Duan, Christophe Cruz, Christophe Nicolle. "Identifying Objective True/False from Subjective Yes/No Semantic based on OWA and CWA", *ICSCT 2010*, IEEE CS press, pp1-5.

[9] De antiquissima Italorum sapientia ex linguae originibus eruenda librir tres (1710); *On the Most Ancient Wisdom of the Italians Unearthed from the Origins of the Latin Language, including The Disputation with "The Giornale de' Letterati D'Italia"* [*1711*], translated by L.M. Palmer. (Ithaca: Cornell University Press, 1988).

Yucong Duan is a postdoc fellow in DISCO, University of Milano Bicocca. He received a PhD in software engineering from Institute of Software, Chinese Academy of Sciences, China in 2006. He was Post-doc fellow in School of Software, Tsinghua University, China from 2006-2007. He was a Post-doc fellow in the Software Engineering Laboratory at Pohang University of Science and Technology (POSTECH), South Korea, from 2007-2008. He was a lecturer of Biomedical Engineering Institute, Capital University of Medical Sciences, Beijing, China, from 2008 to 2009. He was a Post-doc fellow in Le2i, CNRS, University of Bourgogne, France, from 2009-2010. He was a Post-doc fellow in Distributed system lab, Charles University, Czech, from 2010-2011. He has been a member of IEEE since2005. His research interests include: theoretical and empirical software engineering, model driven software development, knowledge engineering, ontology modeling, Semantic web, quality evaluation, etc.