Journal of International Council on Electrical Engineering Vol. 3, No. 1, pp.79~84, 2013
http://dx.doi.org/10.5370/JICEE.2013.3.1.079

Construction for Ontology of Domain Terms and Tasks (ODT^2)

Junya Miyamoto*, Ryo Kawabata* and Kiyoshi Itoh†

Abstract – This papers proposes Diagram for Ontology of Domain Terms and Tasks (referred to as DODT^2) in order to visualize Ontology. DODT^2 is divided into two diagrams. The first diagram is Diagram for Ontology of Domain Tasks (referred to as DODTasks). DODTasks visualize Ontology of Domain Tasks represented in Case Grammar. A component of this diagram describes a term which corresponds to each case of Case Grammar. In addition, to respond to any of various situations, three notations for DODTasks are proposed: Conditional Branch, Parallel Execution, and Random Order. The second diagram is Diagram for Ontology of Domain Terms (referred to as DODTerms). DODTerms represents the super-/sub-relation of Domain Terms. Moreover, to assist Construction of Ontology, the author implemented DODT^2 Editor. It can edit DODT^2. DODT^2 Editor involves functions to help construction of Ontology.

Keywords: Case grammar, Construction for ontology, Diagram notation, Domain terms and tasks, System analysis

1. Introduction

To describe a specification by system analysis, the use of Ontology has two benefits. First, it is easy to understand the meaning of terms and their semantic relationships. Second, it brings an efficiency to construct Ontology by reusing Ontology of similar Domain that has already been specified in [1]. Ontology is, however, difficult to understand, because it is conceptual.

The purpose of this research is to make Ontology understandable intuitively and to facilitate the construction of Ontology in terms of visualization of Ontology. In this research, Diagram for Ontology of Domain Terms and Tasks (DODT^2) is proposed. Moreover, to assist Construction of Ontology, DODT^2 Editor is implemented.

2. Ontology

In this research, Ontology is a conceptual structure of terms. For a specific term, we define as Ontology what kind of category it exists and what kind of semantic relationships it has. In this research, Ontology is composed for the Domain Terms and Tasks.

2.1 Ontology of domain tasks (ODTasks)

For Domain Tasks, there are two terms that are a subject or an object and these terms are paired with a specific verb. Task Ontology defines such relationships. Even if the same verb is used, a pair of subject and object terms differs according to Domain. This is called Ontology of Domain Tasks (referred to as ODTasks). ODTasks is described by using Case Grammar. Case Grammar is a description method that represents relationships between verbs and nouns. In this research, describing ODTasks uses eight deep cases: Agent, Action, Object, Location, Time, Source, Goal, and Instrument. Additionally, it uses Condition. If Condition is true, then a Task is performed. Table 1 is three sentences described on the basis of Case Grammar.

They are “I write a letter with a pen.”, “I choose a book from bookshelf in the library in the daytime.”, and “He gives allowance to me.”.

The description of ODTasks consists of Case Frames about specific Domain Tasks. Case Frames are frames with semantic role on verb in Case Grammar. A description example of ODTasks is shown in Table 2. Table 2 is ODTasks of “The sale of beer”. First, Customer takes beer to checkout. Next, clerk identifies customer’s age. If customer’s age is larger than Minimum Purchase Age (MPA), he or she can purchase beer.

2.2 Ontology of domain terms (ODTerms)

There are terms that compose a specific Domain and represent concepts: Task, Person, Object, Action, Location, and Time. It is called Domain Terms. There are synonyms for term in Domain Terms, and Domain Terms has a super-

† Corresponding Author: Dept. of Information and Communication Sciences, Sophia University, Japan (itokiyo@sophia.ac.jp)
* Dept. of Information and Communication Sciences, Sophia University, Japan
Received: December 24, 2012; Accepted: December 26, 2012

Table 1. Sentence described on the basis of case grammar

	Agent	Action	Object	Location	Time	Source	Goal	Instrument	Condition
i	I	write	letter					Pen	
ii	I	choose	book	library	daytime	book shelf			
iii	he	give	allowance				I (me)		

Table 2. ODTasks of "The sale of beer"

	Agent	Action	Object	Location	Time	Source	Goal	Instrument	Condition
The sale of beer	customer	take	beer	shop	shop hour		checkout		
	clerk	identify	age	shop	shop hour		customer		
	clerk	sell	beer	shop	shop hour		customer	checkout	Customer's age is larger than MPA

/sub-relation. We can define Domain Term's category and super-/sub-relation as Ontology of Domain Terms (referred to as ODTerms).

ODTerms is describing hyponymy of Domain Terms on each category. Table 3 is an example of described ODTerms. The first column describes the term name. The second column describes the super-/sub-relation that are "Super" or "Sub". The third column describes the Object term name. It means that Term is super-/sub-class of Object term. For example, in second row of Table 3, term name is "Staff", relation is "Sub", and object term name is Human. Therefore, "Staff" is sub-class of "Human". However, in the first column, third row is not filled, because the term which becomes sub-class of "Human" does not exist. Thus, in description of ODTerms, hyponymy of Domain Terms on each category is described.

Table 3. ODTerms of person of restaurant

Term name	*Relation*	*Object term name*
Human	Sub	
Staff	Sub	Human
Customer	Sub	Human
Waiter	Sub	Staff
Cook	Sub	Staff

2.3 Related works on Ontology

Ontology is systematization of the concepts [2], [3]. Ontology is introduced into AI field for including the semantic information for treating knowledge. Ontology is paid attention to in the areas of the Web retrieval [4] and development of the information system, and is used in various fields over the knowledge system.

Our method proposes a method for Ontology construction with the aim of reuse of requirements analysis in a Domain or between Domains.

The Task Ontology is proposed in (Mizoguchi *et al.* 92) [5] and (Mizoguchi, Vanwelkenhuysen and Ikeda 95) [6]. These papers assume one subject which performs the task in the target of the Task Ontology. The Task Ontology includes the objects and the verbs which represent the action on the objects. The subject of the action is not described explicitly.

Our method takes the subject of the action into account. In systems such as stock control, sales management, etc., the tasks are performed by various subjects such as equipment and personnel in different positions. Information and materials are exchanged between the subjects. The meaning of a task is different depending on a subject. The two tasks represented with the same verb and the same object but different subjects have different meaning. Thus, it is important to clarify the subject of tasks. Our Ontology describes the subject of the task explicitly. The subject and object switch with the progress of the task. These are described by commitment network. The meaning of the task is also different depending on the time and place. These are described based on Case Grammar.

There are other researches of construct Ontology by using the editor, such as (Kozaki, et al. 2002) [7] where one defines structures, attributes and restrictions of concepts as Ontology in terms of class concept and semantic links on the basis of AI.

Our method has an aim of constructing Domain Terms and Tasks for system analysis. For this aim, our method places an important role on the "subject" as well as "verb" and "object" in the Task. To describing Ontology of Tasks, Case Grammar [8] is adopted.

3. Diagram for Ontology of Domain Terms and Tasks (DODT^2)

In this research, to visualize Ontology of Domain Terms and Tasks (referred to as ODT^2), the author proposed Diagram for ODT^2 (referred to as DODT^2) in order to visualize Ontology. DODT^2 is divided into Diagram for ODTasks (referred to as DODTasks) and Diagram for ODTerms (referred to as DODTerms).

3.1 Diagram for Ontology of domain tasks (DODTasks)

DODTasks represent ODTasks described in Case Grammar as shown in Fig. 1. Fig. 1 is fundamental component of DODTasks. It is the rectangle divided into three, and the term which corresponds to three cases that are Subject, Object, and Action is described on each part of the rectangle. Several arrows are extended to up and down, left and right from the rectangle. The solid horizontal arrow represents task flow. It is connected to right side of the rectangle represent the next task from left side of the rectangle represent the previous task. The dashed arrow extended to up and down is connected to the term which corresponds to three cases that are From, To, and With. Direction and connecting point of these arrows differ depending on type of case. In this way, component of the diagram describes a term which corresponds to each case of Case Grammar. Describing such diagrams and connecting them, we can construct DODTasks.

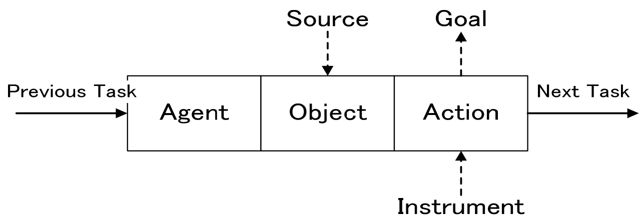

Fig. 1. Fundamental component of DODTasks.

3.2 Specific notations of DODTasks

In describing DODTasks, to respond to any of various situations, the author proposes three specific notations for DODTasks: Conditional Branch, Parallel Execution, and Random Order.

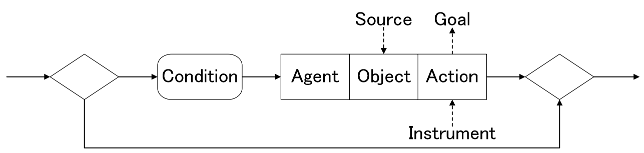

Fig. 2. Conditional Branch.

Conditional Branch is shown as Fig. 2. When the performed tasks differ according to the condition, Conditional Branch is used. Branching and converging are performed from a diamond shape. The number of branching could be more than one. If condition for running the task exists, a rectangle which has rounded corners describes the condition and is placed in front of the task.

Fig. 3 is an example of Parallel Execution. When a number of tasks are executed in parallel, Parallel Execution is used. Dividing and merging are performed from a synchronization bar. Synchronization bar fall into two types. One is Folk which perform dividing, and the other is Join which perform merging. Each task divided by Folk is executed in parallel way. When all tasks have executed, then they are merged by Join and task flow proceed to the next of the Join. Notice that if it has not been executed all tasks, it cannot proceed to the next.

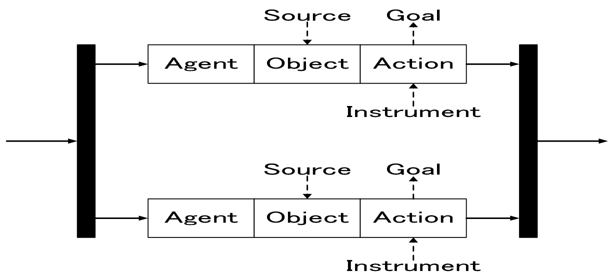

Fig. 3. Parallel execution.

Fig. 4 shows notation of Random Order. Although several tasks exist, execution sequence is not decided. At times like that Random Order is used. Several tasks are put in brackets. Moreover, they don't connect a solid arrow. In this way, several tasks in brackets indicate that they can perform in no particular order.

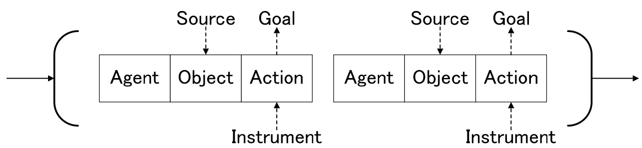

Fig. 4. Random Order.

3.3 Diagram for Ontology of domain terms (DODTerms)

DODTerms is represented the super-/sub-relation of each category in tree structure as shown in Fig. 5. Fig. 5 is DODTerms which represent ODTerms of Table 3. There are rectangles that Term name is described in. If combination which has super-/sub-relation exists in them, the pair is connected with the solid line. It is connected bottom of the rectangle represent super-class term and top of the rectangle represent sub-class term. This way we describe DODTerms by constructing tree structure of terms that belong in each category.

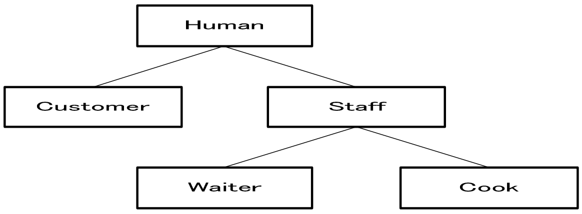

Fig. 5. Example of DODTerms.

4. DODT² Editor

4.1 Editing DODT²

To assist Construction of Ontology by editing DODT², the author implemented DODT² Editor. Fig. 6 is DODT² editing screen. To click-hold on the component of DODT² from the window in the left-hand side of the screen and drag and drop it anywhere on diagram editing area, DODT² is described.

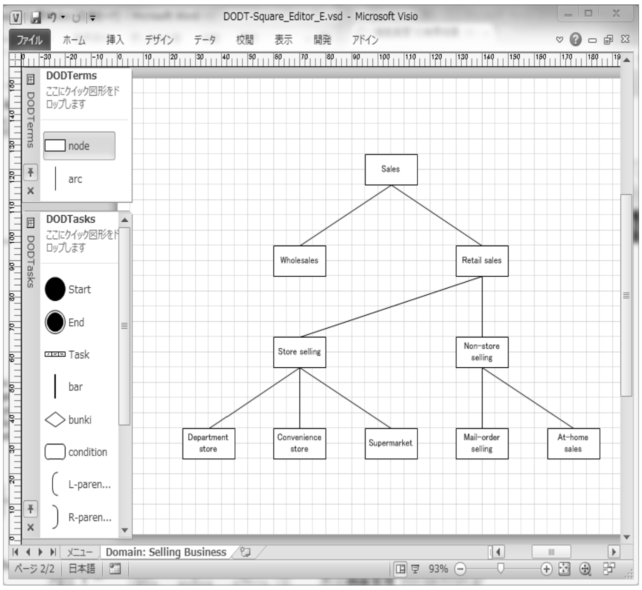

Fig. 6. DODT² editing screen.

Editing contents of DODT² such as Case Frames about Tasks and Term name is supported by using specific edit screen as shown in Fig. 7~9. Fig. 7 and 8 can edit content DODTerms. If the type of Term is Task, edit screen such as Fig. 8 is displayed, otherwise edit screen such as Fig. 7 is displayed. From edit display of ODTerms (only Fig. 8), DODTasks of specific Task can be laid out. Edit screen such as Fig. 9 can edit content of DODTasks and lay out DODTasks of sub tasks.

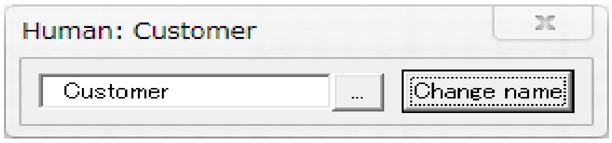

Fig. 7. Editing ODTerms (expect Task).

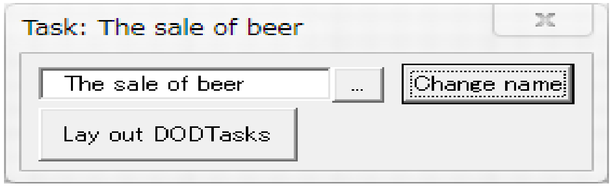

Fig. 8. Editing ODTerms (Task).

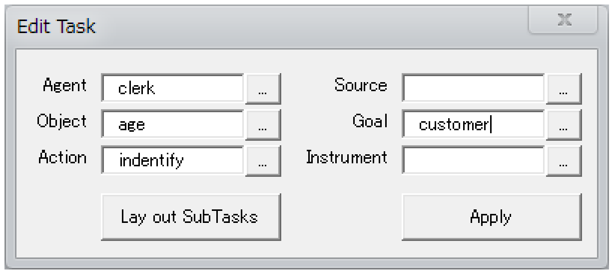

Fig. 9. Editing ODTasks.

4.2 Support functions for construction of Ontology

DODT² editor involves functions to help construction of Ontology. One function is File input-output. Another function is reference ODTerms which already constructed. It can replace the entry of edit screen of DODT² with the term which is selected from ODTerms in the screen as shown in Fig. 10.

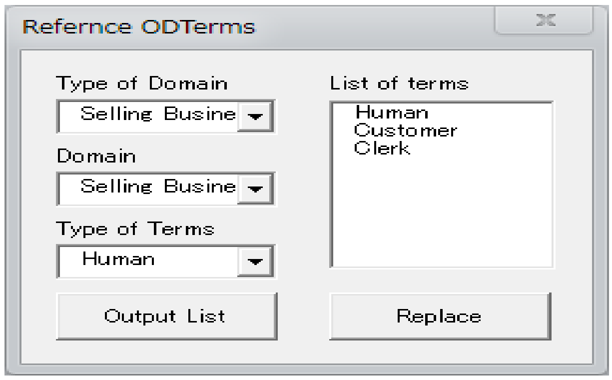

Fig. 10. Reference ODTerms.

In addition, DODT² Editor edits class structure of Domain as shown in Fig. 11. It is like DODTerms and represents Domain relationships between superior and inferior. From Domain edit screen, DODTerms of the Domain can be laid out.

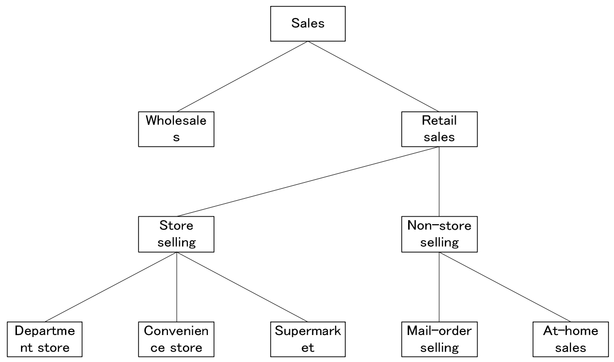

Fig. 11. Class structure of Domain.

5. Reuse of Ontology with DODT^2

When create new DODT^2, in case of DODT^2 of similar Domain is already constructed, the efficiency of Ontology construction is raised by reusing some contents. For example, in Fig. 12 and 13, we construct DODT^2 of electronics retail store by replacing terms of bookstore with terms of electronics retail store.

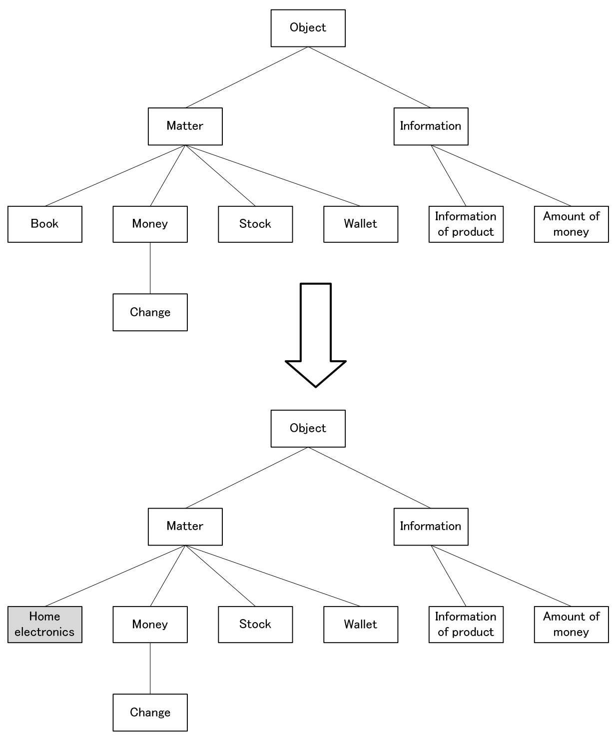

Fig. 12. Reusing of ODTerms.

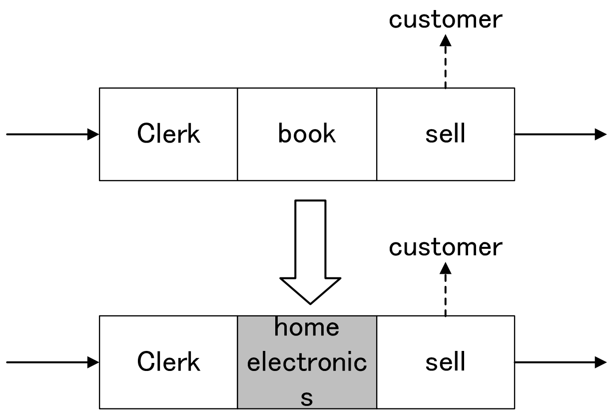

Fig. 13. Reusing of ODTasks.

In this way, reusing constructed DODT^2 makes Ontology construction efficiently.

6. Conclusion

To visualize ODT^2 by DODT^2, it can be easier to understand and to construct intuitively. Furthermore, Ontology's reusability and efficiency of construction can be increased by some functions to help construction of Ontology: file input-output, reference existing ODTerms, and so on.

However, there are many challenges in describing Ontology. At present, DODTasks is not described about place and time. Therefore, we have to propose the notation of DODT^2 that Time and Location are described. Moreover, there are synonymous terms. If it is defined in ODTerms, search efficiency of terms may be raised. Solving these challenges will lead to more generalization of Ontology construction.

Acknowledgements

This research was partially supported by Open Research Center Project of MEXT in Japan of 2008-2012.

References

[1] Junya Katoh, Ryo Kawabata and Kiyoshi Itoh, "Construction and Visualization of System Ontology for Enhancing Reusability and Understandability", *SDPS IDPT2010*, Dallas, USA, June 2010.

[2] Mizoguchi, R. and Ikeda, M., "Towards Ontology

Engineering", *The Seventh Pacific Rim International Conference on Artificial Intelligence 13(1)*, pp.1-10, 9-10, 1996

[3] Mizoguchi, R., "A Step Towards Ontological Engineering," < http://www.ei.sanken.osaka-u.ac.jp/english/step-onteng.html > , (accessed March 2012) , 1998

[4] Fensel, D., Hendler, J.A., Lieberman H. and Wahlster W., "Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential". *MIT Press*, 2003.

[5] Mizoguchi, R. Tijerino, Y. and Ikeda, M., "Task Ontology and its Use in a Task Analysis Interview System – Two-level mediating representation in MULTIS", *Proc. of the JKAW'92*, pp.185-198, 1992.

[6] Mizoguchi, R. and Vanwelkenhuysen, J and Ikeda M., "Task Ontology for reuse of problem solving knowledge", *Proc. of KB&KS'95*, pp.46-59, 1995.

[7] Kouji Kozaki, Yoshinobu Kitamura, Mitsuru Ikeda and Riichiro Mizoguchi, "Development and Use of an Environment for Building/Using Ontologies "Hozo"-A Case Study on a Real-scale Plant Ontology-", *Transactions of the Japanese Soc. for Artificial Intelligence*, Vol.17, No.3, pp.196-208, 2002 (in Japanese).

[8] C.J. Fillmore, "The Case for Case", *in Bach and Harms (Ed.): Universals in Linguistic Theory, New York: Holt, Rinehart and Winston*, pp. 1-88, 1968.

Junya Miyamoto received academic degree of B.S in information and communication sciences from Sophia University. He is now a student of graduate course in Sophia University. His research interests are systems and Ontology engineering.

Ryo Kawabata Dr. Kawabata is an Associate Professor of the Department of Information and Communication Sciences at Sophia University, Japan. He has received a B. Eng. and M. Eng. in Mechanical Engineering. He received his Dr. Eng. in Information Systems Engineering from Sophia University, Japan in 2004. His research interests include software engineering, knowledge engineering, Ontology engineering and simulation engineering. His publications include four books, more than twenty journal papers and over forty conference papers.

Kiyoshi Itoh Dr. Kiyoshi Itoh has been working in Sophia University, Tokyo, Japan since 1979. Currently, he is a Professor in the Department of Information and Communication Sciences of Sophia University. He received B. Eng, M. Eng, and Dr. Eng in computer science from Kyoto University in 1974, 1976 and 1980, respectively. His research interests include Domain engineering, software engineering, collaboration and concurrent engineering, knowledge engineering, Ontology engineering and simulation engineering. His publications include sixteen books, over seventy journal papers and over ninety conference papers. One of his main works is "Domain Oriented Systems Development: Perspectives and Practices (ed.), Taylor & Francis, 2002". He is SDPS Fellow. He is a member of IEEE, ACM, IPSJ, IEICE and IEEJ.