

A Construction Model of Ancient Architecture Protection Domain Ontology Based on Software Engineering and CLT

Zhenbo Bi

Architecture School, Xi'an University of Architecture and Technology, Xi'an, China
School of Mathematics, Physics & Information Science, Zhejiang Ocean University, Zhoushan, China
E-mail: bzb136@sina.com

Huiqin Wang and Ying Lu

Information and Control Engineering School, Xi'an University of Architecture and Technology, Xi'an, China
E-mail: {hqwang, luying}@xauat.edu.cn

Abstract—In allusion to the lack of formal mathematical methods and engineering characteristics on the construction of AAPDO (ancient architecture protection domain ontology, AAPDO), a construction model on AAPDO based on software engineering and CLT (Concept Lattice Theory, CLT) is proposed, and the application steps and methods of the model are also elaborated on. The model uses a guiding ideology of engineering, a formal way of expression and the standardized work steps to complete the construction of AAPDO. Finally, taking Chinese Traditional Roof as an example, an instance on the construction of the field ontology of Chinese Traditional Roof based on software engineering and CLT is given.

Index Terms—Concept lattice, Software engineering, Ontology construction, Ontology for ancient architectures protection

I. INTRODUCTION

Ontology construction in ancient architecture protection is the premise of all the knowledge engineering activities based on its ontology. Improving the formalization and standardization level of ontology construction in ancient architecture protection is the foundation for enhancing interoperability, knowledge expression ability and the ontology inferential capability on the basis of a semantic ancient architecture software intersystem. Ontology construction in ancient architecture protection has attracted increasing attention in the ancient architecture and other related research fields. For example, Nadezhda Govedarova, et al.[1] put forward architecture knowledge management based on ontology in Bulgaria cultural heritage BULCHINO network directory project, for searching and browsing; Liu Qifeng, et al.[2] proposed a design method for ontology repository based on Ontology Definition Metamodel(ODM), to build framework ontology by recognizing the key concepts and their incidence relation

in ancient architecture protection; Bai Weijing, et al.[3] designed and implemented an ancient architecture repository using semantic web technology, thus established an automatic animation system; Song Yu, et al.[4] established an ontology model integrating architecture components, culture and structure, thus explored a retrieval system based on the ancient architecture component ontology. These studies achieve their objectives through the establishment of ancient architecture ontology. However, one common feature is that their ontology construction in ancient architecture basically relies on personal knowledge or experience of researchers, developers and experts, which easily causes the ontology concept deletion or repetition, even the ambiguity and confusion among the concepts due to its strong subjectivity. Particularly, when faced with large-scale data organization, it is time-consuming, laboursome and inefficient. Therefore, ontology construction in ancient architecture by formalization mathematical method is of great significance.

CLT is a branch of applied mathematics, which is a kind of knowledge description and data analysis tools built on the hierarchy of concept based on the mathematics. With the help of CLT, the concepts composed of the extent, the intent and the hierarchical relationships among these concepts can be discovered, constructed and demonstrated. It is a formal and standardized domain ontology construction method. CLT and domain ontology are the two kinds of methods on formal knowledge representation, and the literature [5] pointed out the links and differences between them. There are a lot of published literatures about CLT used in the construction of domain ontology [6-10]. These documents have laid the foundation for the produce of the theory of domain ontology building based on CLT. But when they used CLT to build domain ontology, they mostly focused on some aspects of domain ontology construction or a specific stage, for example, access to formal context, concept lattice constructing algorithm and the improvement of concept lattice. There are no

Corresponding author: Zhenbo Bi

engineered features in domain ontology building process based on CLT. In fact, the ultimate development trend of the construction of AAPDO must be an engineering way to solve the problem on its building.

In allusion to the lack of formal mathematical methods and engineering characteristics on the construction of AAPDO, we propose a construction model on AAPDO based on software engineering and CLT. The model adopts a guiding ideology of engineering [11-13], a formal way of expression and standardized working steps to complete the construction of AAPDO, which improves the engineering degree and degree of formalization of the construction of AAPDO. This will provide a theoretical basis and new ideas of application for the work and some related research.

II. CLT RELEVANT DEFINITIONS

Definition 1. A binary relation R on set M is called an order relation (or shortly an order), if it satisfies the following for all elements $x, y, z \in M$: 1) xRx ; 2) xRy and $x \neq y \Rightarrow \text{not } yRx$; 3) xRy and $yRz \Rightarrow xRz$.

Definition 2. A formal context $K = (G, M, I)$ consists of two sets G and M and a relation I between G and M . The elements of G are called the objects and the elements of M are called the attributes of the context. In order to express that an object g is in a relation I with an attribute m , we write $gIm \in I$ and read it as "the object g has the attribute m ".

Definition 3. For a set $A \subseteq G$ of objects, we define $A' = \{m \in M \mid \forall g \in A, gIm\}$ (the set of attributes common to the objects in A). Correspondingly, for a set B of attributes, we define $B' = \{g \in G \mid \forall m \in B, gIm\}$ (the set of objects which have all attributes in B).

Definition 4. A formal concept of the context (G, M, I) is a pair (A, B) with $A \subseteq G, B \subseteq M, A' = B, B' = A$. We call A the extent and B the intent of the concept (A, B) .

Definition 5. If (A, B) and (C, D) are concepts of a context. (A, B) is called a super concept of (C, D) (which is equivalent to that (C, D) is a subconcept of (A, B)), provided that $B \subseteq D$ (which is equivalent to $C \subseteq A$), we write $(C, D) \leq (A, B)$, namely, $(C, D) \leq (A, B) \Rightarrow B \subseteq D (\Leftrightarrow C \subseteq A)$. The relation \leq is called the hierarchical order (or simply order) of the concepts. The set of all concepts of K ordered in this way is denoted by $(\mathbf{B}(K), \leq)$, $(\mathbf{B}(K), \leq)$ is called the concept lattice of the context K .

Definition 6. An order relation \leq is decided by a pair of elements with covering relations (for example, $a < b$)

in a finite set (S, \leq) , if each element in set S is shown with a small circle in the same plane, when b cover a , a and b be connected with a line. Thus obtained graph is called Hasse diagram of order set (S, \leq) . Hasse diagram can be used to express vividly the relationship among elements in order set.

III. CONSTRUCTION MODEL OF AAPDO BASED ON SOFTWARE ENGINEERING AND CLT

As the construction of AAPDO is a systematic project, common domain ontology construction methods, such as "skeleton method", "Assessment method", "Bernaras", "Methontology method" and the "Sensus method", which have not formed a unified standard of domain ontology construction are application-specific ontology construction methods. Therefore, according to the characteristics of ancient architecture protection domain knowledge and the current difficulties, the construction of domain ontology should adopt the development method of increment model [13], as shown in Fig.1. Each linear sequence in the Fig.1 represents a delta; the result would have an ontology which can be used. Among them, the first increment is a core ontology which meets the basic needs (Performance and features); the second increment is an expansion of the previous incremental, as time goes on, a completed ontology will be eventually formed which meets all application requirements in the field. The results of user feedback or comments are the cause of the next increment plan. Incremental plan illustrates the need for core ontology changes, also illustrates the need to increase the features and performance.

In Fig.1, the process flow of each increment can adopt waterfall development paradigm or prototype development paradigm, here is given priority to with waterfall model, the process is divided into six stages: learning from the idea of the waterfall model of software engineering, the domain ontology description model of ancient architecture protection shown in Fig.1 has an important practical significance. Here its build process is divided into six stages: I. Fields and related fields definition; II. Feasibility study; III. requirement analysis; IV. Domain ontology design; V. Domain ontology implementation; VI. Application and maintenance of domain ontology. If there are problems involving a preceding stage with a vague description at each stage, you can return to perfect, as shown in fig.2. In Fig.2, the phase division of software development, the key problems and the standard of the end of the software lifecycle methodology have been the model for a reference. Considering some characteristics of ontology engineering, some improvements have been made to the waterfall model, such as the testing of ontology is omitted and tested on the application. In Fig.2, CLT's role is mainly reflected in the design stage of AAPDO. The detailed corresponding model of CLT combined with software engineering is shown in Fig.3.

A. Fields and Related Fields Definition

The priority of AAPDO engineering is to fully understand the domain knowledge. To understand the domain knowledge, first, we must have a definition on domain knowledge to determine what knowledge is to be

studied in the field of knowledge, at the same time, the cross of one domain knowledge and other fields of knowledge must also be defined. Finally, specifications of fields' definition need to be written.

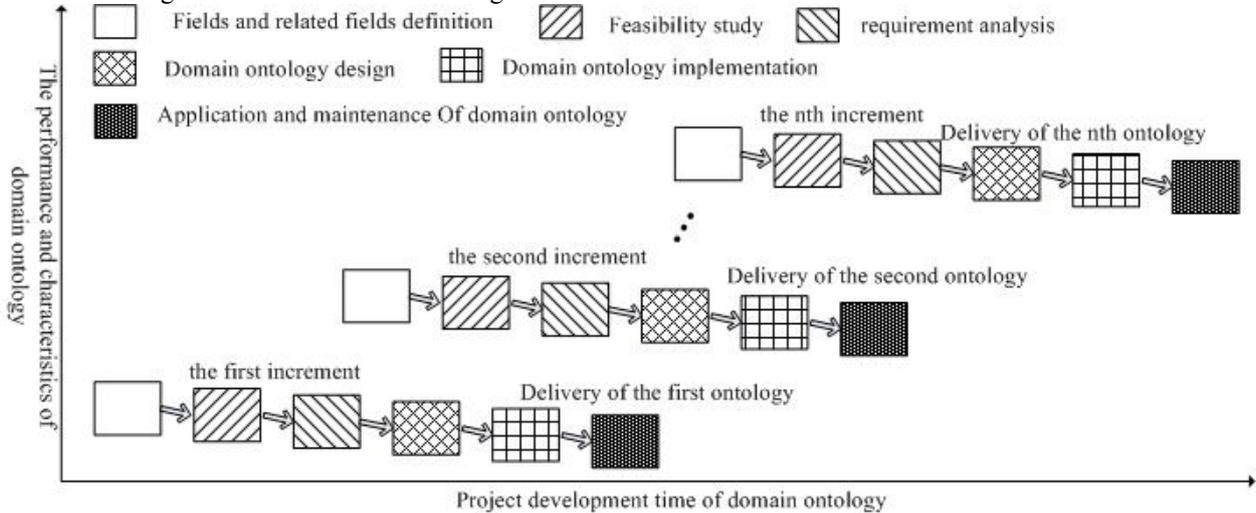


Figure 1. Incremental model of ancient architecture protection ontology

B. Feasibility Study

After the scope of range of fields and related fields definition is defined, we need to analyze whether the existing resources can live up to modeling the defined domain knowledge and whether implementing the entire project is feasible, and we also need to estimate the future workload and difficulty of the domain ontology development. These require knowledge engineers, domain experts and the related personnel to decide through their comprehensive evaluation. If possible, the work can be turned to the next step; otherwise, it is necessary to discover and find the factors that may lead to failure of the project and to determine whether to proceed with the project. At the end of the stage, a feasibility study report needs to be written.

C. Requirement Analysis

According to the problems existing in the ancient architecture protection, we should explicitly put forward the goal to be achieved for the knowledge base in this phase, namely, an AAPDO constructed by the five elements of the conceptual classes, relations, functions, axiom and instances is the goal of the phase, which build a bridge of exchange and communication to achieve transparent access and interoperability between the applications systems and the users and between the different applications based on the ontology. This is the premise of knowledge development based on the ontology. In practice, we should begin from understanding application to solve the core problem, and the core issues of the applications and the core issues of the areas need to be distinguished here, the former involves the concepts of multiple areas, there can be or not a link between each other; the latter involves the core concepts carefully selected of specific areas. Analysis of application requirements should begin from the

decomposition of application problems, and then we can draw lessons from the top-down method of software engineering to do it. Finally, for specific sub-applications, according to the core problems to be solved, we need to use the corresponding techniques [14] to extract the core term set of areas from data sources, such as an existing glossary, documents, databases and domain experts according to the specific types of data sources.

D. Domain Ontology Design

The module's central task is to design a complete concept lattice, including four subtasks:

Subtask 1: According to the Binary relation of "Object – attributes", a context of "object as a line head" and "attribute as the column name" is created through the domain core terms got in the requirements analysis phase, namely, a context is represented by a matrix, a head of each row of the matrix represents an object O_i , a head of each column of the matrix represents an attribute P_j , when $O_i, P_j \in I$, it represents that "the object O_i have the attribute P_j ", then we draw a "*" at the corner of the row i and the column j . After finishing the work, we need to determine whether the context is a standard context, if not, we need to analyze the reasons (such as multi-valued context, non-purification context), and take corresponding measures (such as a multi-valued context into a single-valued context, context purification) to standardize the context.

A multi-valued context (G, M, W, I) is composed of sets G, M, W and the ternary relationship I among the three sets, namely, $I \subseteq G \times M \times W$, moreover, $(g, m, w) \subseteq I$ and $(g, m, v) \subseteq I$, always implies $w = v$, the elements of G are called the objects, the elements of M are called the multi-valued attributes of the context and the elements of W are called the attributes of

the context. $(g, m, w) \subseteq I$ is read as “The value of attribute m of object g is w ”. if W has n elements, (G, M, W, I) is called a n -value context. According to the different types of attribute value of a multi-valued context, it mainly includes numerical multi-valued context, interval multi-value context and language multi-value

context [14]. For different types of multi-valued contexts, we need to take a different approach (such as membership degree transformation method, concepts scaling transformation method and Language scale segmentation transformation method, etc.) for the conversion of single-valued context.

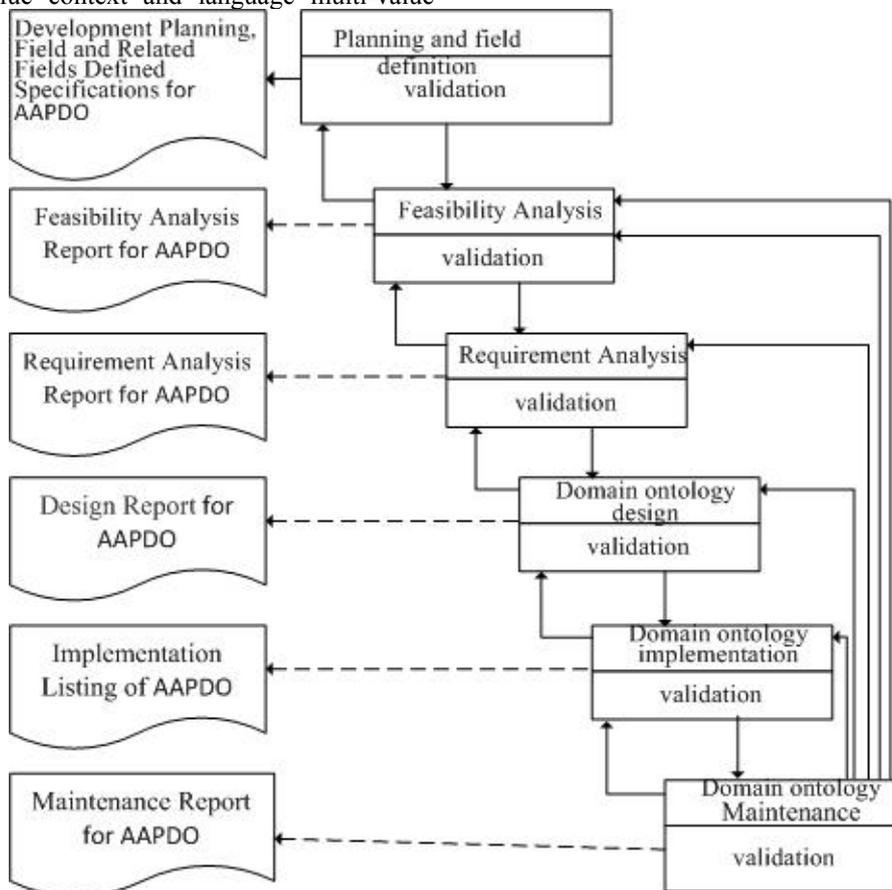


Figure 2. Waterfall model for AAPDO construction

For $k = (G, M, I)$, if $\exists g, h \in G$ and $g' = h'$, there must be $g = h$, then the object g and h are redundant, the lines with G and h can be reduced; Similarly, $\exists m, n \in M$ and $m' = n'$, there must be $m = n$, then the attribute m and n are redundant, the columns with G and h can be reduced. From the perspective of object and attribute redundancy, we combine the same objects and attributes in the context to meet the basic simple purification of the context here. In the same context, if $\exists g, h, t \in G$, $t' \subset g'$, $t' \subset h'$, and $g' \cap h' = t'$, then the object t is redundant, the line with t is reduced [15]. From the perspective of mutual expression of objects and attributes, we remove the extents (the intents) expressed with the intersection of all other objects (attributes) which can be used here.

Through the above basic operation, the standardization of the context can be realized.

Subtask 2: Through concept lattice construction algorithm (algorithm for constructing a batch or

incremental construction algorithm) [15-17], the standard context can be converted into a concept lattice, and through the Hasse diagram, the concept lattice can be visually represented. And then domain experts and knowledge engineers need to determine whether the concept lattice is reasonable on the basis of visualization. For unreasonable concept lattice, we make it a more complete satisfaction concept lattice through adding or removing objects, adding or removing attributes, editing attributes by certain rules. Concept lattice can generate new objects which are not in the table of concept, we can add these objects; The whole process is repeated continuously until the concept lattice is reasonable and perfect.

Subtask 3: The conversion of the edited concept lattice complete mainly includes the naming of top nodes, the labeling of intermediate nodes, the removal of bottom nodes and the conversion of Relationship between nodes to the relationship between concepts in Hasse diagram, the conversion result is a domain ontology prototype.

Subtask 4: According to the actual situation, attribute expansion, instance expansion, axioms expansion and relationships expansion on the basis of the domain

ontology prototype can be done under the participation of experts; finally the expanded domain ontology prototype has been got. Among them, attribute expansion is used to improve the intent of the ontology concept and instance expansion is used to improve the extent of the ontology concept, relationships expansion aims to perfect the relationship of the domain ontology concepts in addition to the classification relationship, axioms expansion is in order to achieve the ontology reasoning.

E. Domain Ontology Implementation

For the expanded domain ontology prototype, it needs to be formalized description through the appropriate ontology description tools and language, namely, an encoding process of ontology is completed, and the

resulting domain ontology is got. The encoding process includes coding of various aspects, such as domain concepts, relationships between concepts, attributes, instances, axioms and inference rules. Next we need to use an ontology reasoning machine to achieve the ontology knowledge reasoning which is based on the concept lattice with a mathematical theory support. Using the concept lattice will effectively help knowledge engineers to complete the logical description of the domain knowledge. Here the ontology reasoning comprises detecting conflicts, optimizing expression and the tacit knowledge acquired by reasoning based on the explicit knowledge.

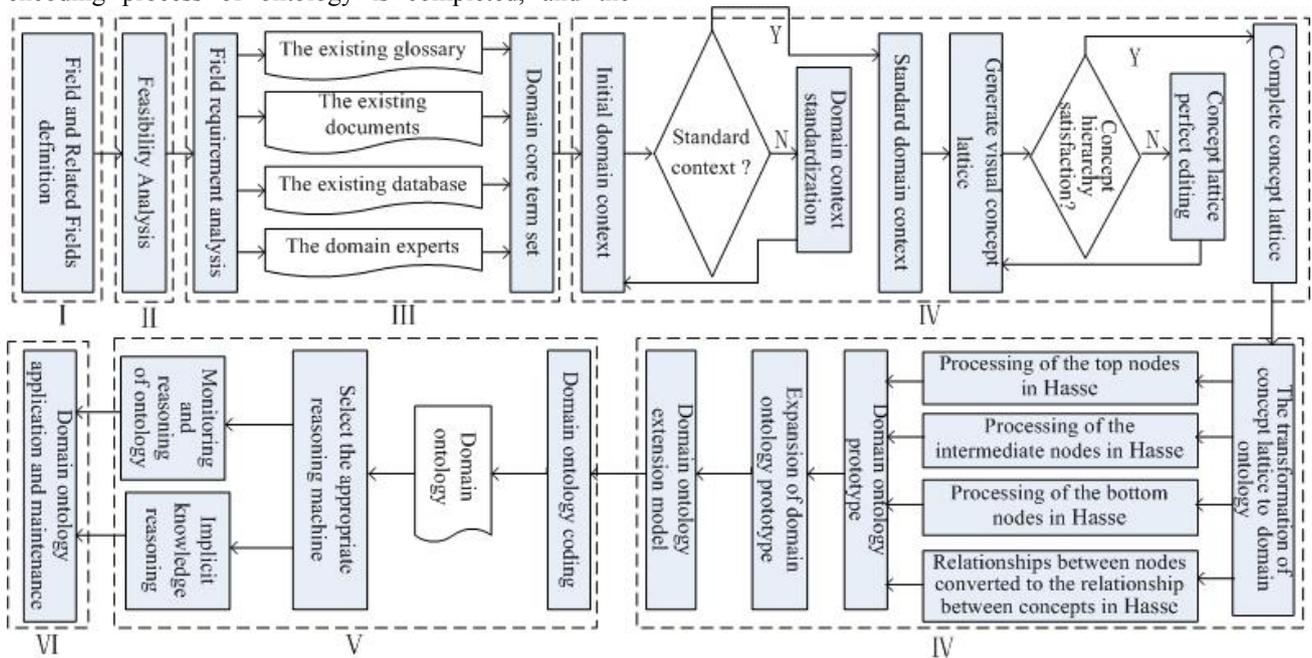


Figure 3. AAPDO construction model based on CLT and software engineering

F. Domain Ontology Application and Maintenance

For the implemented domain ontology, it can be put into practical application. However, domain knowledge is constantly evolving, with new knowledge creation and added in, so the implemented domain ontology cannot be fixed and unalterable, according to the development and changes of domain knowledge, we often need to improve maintenance for the ontology, in this way can we ensure universal applicability of the domain ontology.

IV. EXAMPLE: THE CONSTRUCTION OF TRADITIONAL BUILDING ROOF PROTECTION ONTOLOGY

The purpose of the example aims to verify the practical effect of the proposed model. Therefore, the choice of architecture field does not be too complicated to be able to clarify the correctness, the availability and ease of use of the theory.

A. Step 1: Fields and Related Fields Definition

The field of protection of ancient architecture is a large and complex cross-cutting field which is composed of multiple sub-domain knowledge. To build the entire

AAPDO, we need to divide it into many sub-domain ontologies to build in turn. First, we build a top ontology, and then we built sub-domain ontology. Using the method of the top-down, applications decomposition and divide and conquer contributes to simplify the construction of the ontology. For example, Xi'an is an ancient city with a long history and culture, since thousands of years, many excellent ancient buildings are preserved, such as ancient city wall, ancient gate tower, Dayan Pagoda, palace buildings and Bell Tower, etc. Due to the long time, these ancient buildings urgently needed to be protected through the use of modern advanced technology. Therefore, here we have identified Xi'an 'ancient architecture protection areas and the corresponding sub-fields according to the ancient architecture types. First, we have built the top ontology Xi'an ancient architecture protection ontology, and then we have also built sub-domain ontology and their sub domain ontologies.

B. Step 2: Feasibility Study

The resources of the construction of Xi'an ancient architecture protection ontology consist of economic and

technical resources. Economic aspects, the project has gained national and provincial research funding support, and the related research funds also provide support for the project research; technical aspects, the project team has a large number of ancient architecture protection experts and technical staff who are associated with the research, most of them have rich and practical experience, in addition, there are also a large number of knowledge management engineers, they also have a wealth of knowledge management and application development experience. Therefore, the project implementation of the ontology is feasible, and after the discussion of knowledge engineers, domain experts and relevant personnel, the future workload and technical difficulty of the research projects is also predictable and controllable.

C. Step 3: Requirement Analysis

By analyzing ancient architectural history of china, ancient architecture academic literature, ancient architecture domain knowledge and ancient architecture protection technical literature, we have understood the characteristics of knowledge in the field of ancient architecture protection and extracted key concepts such as ancient architectural complexes, single ancient building, components, damaging state and protection technology

[1]. In protection engineering, we need to use the corresponding protection technology according to the specific damaged status of components. Wooden architecture in Single ancient building is the main; with its roof style divided into flush gable roof, overhanging gable roof, gable and hip roof, hip roof, pyramidal roof, etc. Protection repair documentation for structural description is described through specific technology. In allusion to the damaging of specific components, the corresponding detailed protection technical description is given. To illustrate the problem, here only "Chinese Traditional Roof" as an example to tell, and the "Chinese traditional roofs" is seen as a special component (a combination of many atomic components), which avoid the disadvantages of the context that is too large to be easy to express, at the same time we also necessarily simplify the attributes of the related objects, and added some attributes from the perspective of protection. In this paper, the vocabulary entry "Chinese traditional roofs"[18] in interactive encyclopedia is taken as domain unstructured data, and on the basis of which the ontology of Chinese traditional roof will be built. "Chinese Traditional roofs" is shown in Fig. 4.

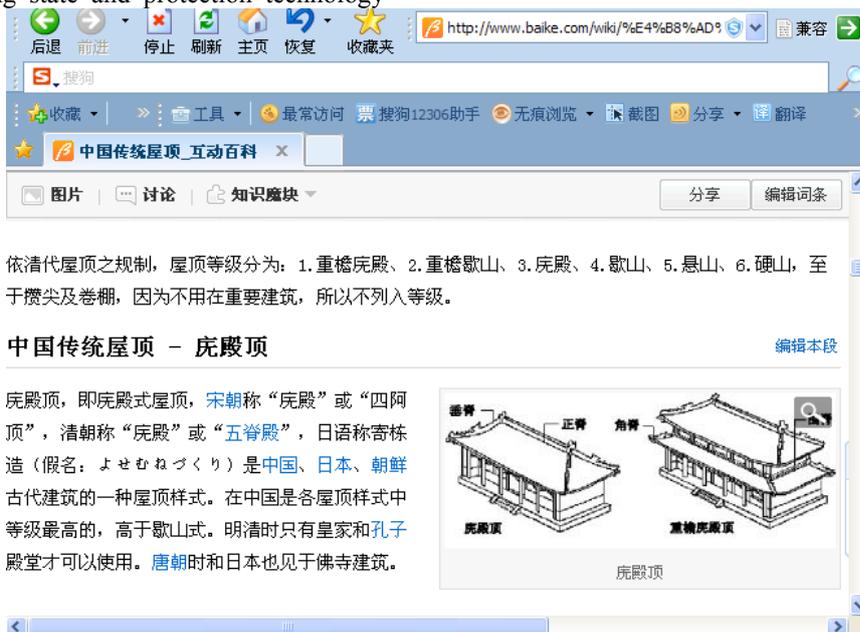


Figure 4. Chinese traditional roof initial data

After the text in the vocabulary entry "Chinese traditional roofs" is preprocessed, we use natural language processing techniques (such as Pao Ding Xie Yang Chinese Word Segmentation) combined with manual analysis techniques to get the domain unstructured data (text), and on the basis of which a core terms set has been extracted. The core terms set includes an attribute set and an object set, the attribute set: { P_1 main ridge, P_2 double eave roof, P_3 slope face, P_4 palace architecture, P_5 civil architecture, P_6 leaking, P_7 damaged tile, P_8 weeding}; the object set: { O_1 hip roof, O_2 gable and hip roof, O_3 pyramidal roof, O_4 overhanging gable roof, O_5 flush gable roof, O_6 round ridge roof}.

D. Step 4: Domain Ontology Design

An initial context based on the above attributes and object sets is shown in Table 1. At this time, the context isn't a regularization context, the attribute P_2 is no corresponding object, so the object O_1 is changed to the object O_{11} single eave hip roof and the object O_{12} double eave hip roof; the attributes P_6, P_7 and P_8 are important public attributes, we put them into the attribute set of the father object "Chinese traditional roofs" of the objects with these attributes. The objects O_4 and O_5 should be combined into one object according to Subtask 1 of Section 2.4, but according to the ancient architectural

knowledge, the objects O_4 and O_5 are actually two different objects, here is the same, because some attributes are simplified, to make the difference, the simplified attribute “P” protruding gable wall” needs to be added in the context shown in table 2. P_3 is a multi-valued attribute, it needs to be changed to Single-valued attributes, so P_3 is changed to the attributes P_{31} 1-slope face, P_{32} 4-slope face, P_{33} 6-slope face and P_{34} 8-slope face, at the same time, in order to make the context more rationalized, the object O_3 is changed to the objects O_{31} round pavilion roof, O_{32} 4-angle pavilion roof, O_{33} 6-angle pavilion roof and O_{34} 8-angle pavilion roof. The final context is shown in Table 2.

TABLE 1.
INITIAL CONTEXT OF CHINESE TRADITIONAL ROOF

G M	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8
O_1	*		*	*		*	*	*
O_2	*		*	*		*	*	*
O_3	*		*			*	*	*
O_4	*		*		*	*	*	*
O_5	*		*		*	*	*	*
O_6			*		*	*	*	*

And when building them, we had communication with domain experts to ensure that the concept lattice can meet the practical requirements, or they are easy to deviate from the field as they really are. Among them, the context of Chinese traditional roof was converted into a concept lattice shown in Fig. 5 by using Concept Explorer.

TABLE 2.
CONTEXT OF CHINESE TRADITIONAL ROOF

G M	P_1	P_2	P'	P_{31}	P_{32}	P_{33}	P_{34}	P_4	P_5	P_6	P_7	P_8
O_{11}	*				*			*		*	*	*
O_{12}	*	*					*	*		*	*	*
O_2	*	*			*			*	*	*	*	*
O_{31}		*		*						*	*	*
O_{32}		*			*					*	*	*
O_{33}		*				*				*	*	*
O_{34}		*					*			*	*	*
O_4	*		*						*	*	*	*
O_5	*								*	*	*	*
O_6			*						*	*	*	*

There are 16 concepts in Fig. 5, in addition to the circular nodes with two colors, and the concept represented by the other nodes are actually hidden, which are domain concepts generated by automatic clustering, which represents objects with a number of attributes. The concept lattice achieves automatic classification of domain concepts or object, the node hierarchy presents hierarchy or hyponymy of concepts or classes.

For each node in Fig. 5, if domain experts think that the concept lattice can't accurately describe the domain knowledge, the concept lattice can be edited in the help of knowledge engineers in accordance with the relevant rules, here to skip this step. After obtaining a complete concept lattice, we need to deal with nodes and relationships between nodes in the concept lattice to get the prototype of the domain ontology. First, with the help of domain experts, the corresponding concepts of the nodes are named; Second, all the attributes (including inherited properties from the ancestor nodes) and instances of the nodes are labeled, namely, the intents and the

extents of the concepts need to be defined; third, the relationships between the nodes are converted into relationships between the corresponding concepts. The following are the three typical nodes:

Node 1: Chinese traditional roof ($\{O_{11}, O_{12}, O_2, O_{31}, O_{32}, O_{33}, O_{34}, O_4, O_5, O_6\}, \{P_6, P_7, P_8\}$), This node contains all instance and the common attributes of all instances in the domain, and represents a chinese traditional roof with P_6, P_7 and P_8 .

Node 10: single eave hip roof ($\{O_{11}\}, \{P_1, P_{32}, P_4, P_6, P_7, P_8\}$).

Node 16: ($\Phi, \{P_1, P_2, P', P_{31}, P_{32}, P_{33}, P_{34}, P_4, P_5, P_6, P_7, P_8\}$), the bottom node represents that no roofs meet all the attributes, the concept needs to be removed.

After completing the conversion of concept lattice, we have gotten the ontology prototype of "Chinese traditional roof", the ontology prototype removed the bottom node is shown in Fig. 5.

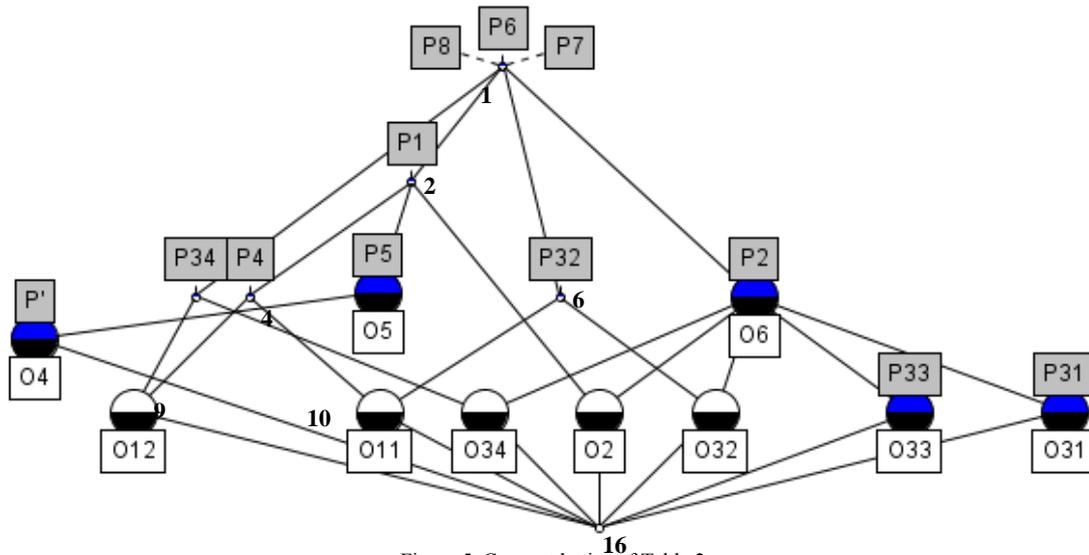


Figure 5. Concept lattice of Table 2

Attributes, instances and axioms in the prototype may appear imperfect, so their expansions need to be done with the help of domain experts and knowledge engineers. For example, for the concept 10 (node 10): single eave hip roof ($\{O_{11}\}, \{P_1, P_{32}, P_4, P_6, P_7, P_8\}$), the axiom $\{\text{single eave hip roof class } (P_4, P_1) \forall (P_{32}, P_1)\}$ is added.

E. Step 5: Implementation of the Domain Ontology

We use protégé and ontology language RDF/OWL to describe the ontology, in fact, protégé can automatically generate the RDF/OWL code of the ontology, the ontology of Chinese traditional roof contains 15 concepts and 11 attributes, which clarify the relationship between the concepts, the attributes of concepts and instances. Parts of concepts and their relationships in Chinese traditional roof ontology represented in RDF are as follow:

```

.....
<rdfs: Class rdf: ID=" O11"
  <rdfs: subClassOf  rdf: resource="# O1" />
</rdfs: Class>
.....
<rdfs: Class rdf: ID=" O1"
  <rdfs: subClassOf  rdf: resource="#Croof" />
</rdfs: Class>
<rdfs: Class rdf: ID=" Croof "
  <rdfs: subClassOf  rdf: resource="# Ccomponent
" />
</rdfs: Class>
.....
<rdfs: Class rdf: ID=" Ccomponent "
  <rdfs:comment> component class </rdfs:
comment>
  </rdfs: Class>
.....
<rdfs: Property rdf: ID=" P8"
  <rdfs: domain rdf: resource="# Croof " />
  <rdfs: range rdf: resource="&rdf; Literal" />
</rdfs: Property >
.....

```

Ontology reasoning is based on RDF file using reasoning machine RacerPro to achieve the ontology reasoning process. Description logic is the basis of ontology reasoning, therefore, it is particularly important how to accurately obtained logical relationships between the domain ontology concepts from domain ontology model. Combining with Fig. 5, we summarized the actual situation of the use of concept lattice to improve description logic. For example, node $2 \subseteq$ node 1, this mean that node 2 is its sub concept of node 1; with $HasPr\ operty(1, P_6)$, we can get $kindOf(1, 2) \wedge HasPr\ operty(1, P_6) \Rightarrow HasPr\ operty(2, P_6)$, among them, $HasPr\ operty(1, P_6)$ represents that node 1 has attribute P_6 , $kindOf(1, 2)$ represents node $2 \subseteq$ node 1; node $4 \cap$ node 6 represents the intersection of all attributes of the two concepts; node $9 \cup$ node 10, represents the union of all attributes of the two concepts; $\neg P_1$ represents the roof without a main ridge; $\exists P_4.P_1$ represents that there are at least a subsequent attribute for node 4; $\forall P_4.P_1$ represents that there are any a subsequent attribute for node 4; single eave hip roof class $\equiv (\forall P_4.P_1) \cap (\forall P_{32}, P_1)$, represents the two are equivalent.

F. Step 6: Domain Ontology Application and Maintenance

For the ontology has been achieved, we have intended to apply it to Xi'an ancient architecture protection knowledge management systems and ancient architecture protection knowledge retrieval systems based on the ontology. Those systems are developed based on a Web platform, which make full use of the advantages of internet. In the course of the systems 'running, if we find that some parts of the ontology in the bottom of the systems cannot meet the actual needs, or new knowledge appears, we will keep upgrading the ontology to ensure its effectiveness.

V. CONCLUSIONS

The construction of AAPDO is a type of large and complex system engineering. Our work reasonably applied software engineering and CLT to the process of the domain ontology building, which is with engineering characteristics of ontology building and rigorous mathematical theory; the proposed model supports all the phases of the domain ontology of ancient architecture protection. And it has effectively improved the standardization level, engineering level and formal level on the domain ontology description of ancient architecture protection. In consideration of the small scope of cases in the work, the related studies should be taken as the focus in the next step to ensure the effective domain ontology building of ancient architecture protection in a wide range.

ACKNOWLEDGEMENTS

This work is partially funded by the specialized research fund for the doctoral program of higher education of the ministry of education (20126120110008) and Xi'an Construction Science and Technology Project (2011008). Thanks for anonymous reviewers' valuable comments.

REFERENCE

- [1] Govedarova, Nadezhda, Stanimir Stoyanov, Ivan Popchev. An ontology based CBR architecture for knowledge management in BULCHINO catalogue. Proceedings of the 9th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing. ACM, 2008.
- [2] Liu, Qifeng, Bingguo Chang. ON ODM-BASED DESIGN APPROACH FOR ONTOLOGY-BASED KNOWLEDGE BASE OF ANCIENT BUILDINGS PROTECTION TECHNOLOGY. *Jisuanji Yingyong yu Ruanjian* 4 (2010) 71-73.
- [3] BAI, Wei-jing, Song-mao ZHANG, Chun-nian LIU. A knowledge base of traditional Chinese architecture and its efficient implementation demonstrated through. *CAAI Transactions on Intelligent Systems* 6 (2010) 012.
- [4] Song Yu, Qian Liping. Ontology Model of Ancient Chinese Architecture Members and Culture Design and Application. *Journal of Beijing University of Civil Engineering and Architecture* 4(2012)48-52.
- [5] HUANG Mei-Li, LIU Zong-Tian. Research on Domain Ontology Building Methods Based on Formal Analysis. *computer science* 1(2006) 210-212.
- [6] Cimiano P. Staab S, Tane I. Automatic acquisition of taxonomies form text: FCA meets NLP. In: *Proceeding of the International Workshop on Adaptive Text Extraction and Mining*, 2003.
- [7] Haav H M. A Semi-automatic Method to Ontology Design by Using FCA. In: *Snasel V. Belohlavek R, eds. Concept Lattices and their Applications. Proceedings of the 2nd International CLA Workshop, TU of Ostrava* (2004) 13-25.
- [8] Tho, Quan Thanh, et al. Automatic fuzzy ontology generation for semantic web. *Knowledge and Data Engineering, IEEE Transactions on* 18.6 (2006): 842-856.
- [9] Li, Man, Xiao-Yong Du, and Shan Wang. Learning ontology from relational database. *Machine Learning and Cybernetics*, 2005. Proceedings of 2005 International Conference on. Vol. 6. IEEE, 2005.
- [10] Li, Junli, et al. A geographic ontology fusion method based on granular theory. *Geomatics and Information Science of Wuhan University* 38.4 (2013).
- [11] Li M, Zhou J, Liang X. Modeling and Description of Organization-Oriented Architecture[J]. *Journal of Software*, 2014, 9(4): 867-872.
- [12] Jiang B, Zhu M. Ontology-Based Information Extraction of Crop Diseases on Chinese Web Pages[J]. *Journal of computers*, 2013, 8(1).
- [13] Kruchten P. Contextualizing agile software development[J]. *Journal of Software: Evolution and Process*, 2013, 25(4): 351-361.
- [14] ZHANG Yunzhong. Research on Domain ontology Construction Method Based on Formal concept Analysis. *Jilin University* (2009).
- [15] YANG Li. The research on concept lattice and reduction method based on dynamic multi-value context. *Southwest Jiaotong University* (2006).
- [16] Thomas Tilley. formal concept analysis applications to requirements engineering and design. *The University of Queensland* (2003).
- [17] WANG Shaofei. Concept lattice construction algorithm and application based on the ontology[D]. *Dalian Jiaotong University* (2006).
- [18] Interactive encyclopedia. (2013). Chinese traditional roof. Available at: <http://www.baike.com/wiki/%E4%B8%AD%E5%9B%BD%E4%BC%A0%E7%BB%9F%E5%B1%8B%E9%A1%B6>.



Zhenbo Bi is a University Lecturer at School of Mathematics, Physics & Information Science, Zhejiang Ocean University, Zhejiang, P.R. China. He has received B.S. degree in Computer Information Management at Xidian University and M.S. degree in Software Engineering at Xi'an Jiaotong University. Now, He is working for

PH.D degree in Digital Architecture at Xi'an University of Architecture and Technology. His main research interests are information processing technology in digital architecture and digital ocean.

Huiqin Wang is a professor at Information and Control Engineering School, Xi'an University of Architecture and Technology, Shanxi, P.R. China. Her current interests include signal and digital image processing.

Ying Lu is a student at Architecture School, Xi'an University of Architecture and Technology, Shanxi, P.R. China. Now, she is working for PH.D degree in Digital Architecture. Her current interest is information processing technology in digital architecture.