IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received May 9, 2019, accepted June 1, 2019, date of publication June 12, 2019, date of current version June 26, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2922310

A Mapping-Based Sharing Approach for Heterogeneous XML Data Supporting User Personalization

CHAO LI[1,2], QINGTIAN ZENG[1], HUA DUAN[1], AND NENGFU XIE[3]
[1]Shandong Province Key Laboratory of Wisdom Mine Information Technology, Shandong University of Science and Technology, Qingdao 266590, China
[2]Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai 201804, China
[3]Agricultural Information Institute, Chinese Academy of Agricultural Sciences, Beijing 100081, China

Corresponding authors: Chao Li (1008lichao@163.com), Qingtian Zeng (qtzeng@163.com), and Hua Duan (huaduan59@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61702306, Grant 61472229, Grant 61602278, and Grant 31671588, in part by the Ministry of Education in China Foundation for Humanities and Social Sciences Grant 16YJCZH041 and Grant 17YJCZH262, in part by the Shandong Provincial Natural Science Foundation, China, under Grant ZR2017BF015, Grant ZR2018BF013, Grant 2016ZDJS02A11, and Grant ZR2017MF027, in part by the Taishan Scholar Climbing Program of Shandong Province, in part by the SDUST Research Fund under Grant 2015TDJH102, in part by the Innovative Research Foundation of Qingdao (under Grant 18-2-2-41-jch), in part by the Open Project of the Key Laboratory of Embedded System and Service Computing, Tongji University, under Grant ESSCKF 2016-06, and in part by the Scientific Research Foundation of the Shandong University of Science and Technology for Recruited Talents under Grant 2016RCJJ011.

ABSTRACT With the development of computer science and technology, there is an urgent demand on integration of heterogeneous data focus on personalized information service. In this paper, we explore two aspects on the usage of the heterogeneous data stored in the form of XML, and propose a mapping-based sharing approach for heterogeneous data supporting user personalization. First, a method is proposed to automatically standardize the heterogeneous XML data, which are provided by different users. The standardization process includes two steps: 1) An XML document is first parsed to a DOM tree, and then the DOM tree is transformed to a reduction tree. By introducing synonym table, the mapping relation can be obtained after the process that the reduction tree is transformed to a standard tree. 2) With that mapping relation, the XSLT file can be generated, with which the standard XML document containing standard data can be easily generated. Second, a method is proposed to support user personalization on data display. The mapping relation mentioned above is first stored in a well-defined form, which can be used to transform the standard XML document into a personalized XML document so as to display users' personalized data. The methodology proposed in this paper has been applied to an application of a sharing platform for the multi-source agricultural spatial data. The user-friendly interface and experience of the platform show that the methodology proposed is feasible.

INDEX TERMS Data sharing, user personalization, XML, XML schema, mapping relation.

I. INTRODUCTION

With the development of information technology, many companies have gradually realized the business and information management on computer, thus accumulating lots of data within their companies. Because of historical and realistic reasons, different companies, even different departments within a company, have different database systems. As a result, there is a huge and complex heterogeneous data environment between companies and departments. Currently, enterprise business processes get increasingly complex given that they are often executed by geographically dispersed partners or different organizations. To complete a cross-organizational business, one of the key points is to integrate or share the heterogeneous data stored in each organization, Moreover, personalized requirements are increasingly becoming one of the main requirements of system development and integration.Therefore, there is an urgent demand on the integration and sharing of heterogeneous data, to satisfy the personalized information service.

The primary task for the integration of heterogeneous data is to design a common model for systems. The past

The associate editor coordinating the review of this manuscript and approving it for publication was Rashid Mehmood.

heterogeneous data integration systems, i.e. multi-database systems(for example, CIMS and FDBS), usually treat data pattern of relations and objects as global schema, but they can not satisfy the system applications of network age. Generally speaking, the global schema of integration of heterogeneous data have to describe all kinds of data formats, whether they are structured or semi-structured, and whether they can support query language or simple text query [1]. And the global schema of integration of heterogeneous data should also be easy to release and exchange data in many kinds of formats [2]. Those are exactly the unique advantages of XML (Extensible Markup Language). With the development of technology and application of XML, it is regarded as a standard for data exchange and presentation. So researching on integration and sharing of heterogeneous data with XML is the hot spot of the field.

At the same time, personalized information service technologies, which can realize feature intelligence, user-centricity and active information push, gain increasing emphasis from researchers [3]. Presently, a number of intelligent information service methods and models have been proposed in academic community. They can be divided into two categories [4]: one is wide range information service systems based on machine learning; the other is the user individualization-oriented intelligent information service systems.

Traditional integration of heterogeneous data approaches pay little attention on personalized information service, which can no more meet the information requirements of users with different backgrounds or different purposes or at different time. However, more and more personalized information service research focus on the models and the algorithms. Little attention has been paid to the sharing of heterogeneous data. In this paper, we explores the issue of data integration and sharing based on XML. With the consideration of personalized information service, this paper proposes a mapping-based sharing approach for heterogeneous data supporting user personalization. This approach can be used to realize automatic data matching based on synonyms table. Thus the XSLT file can be generated, with which we realized automatic switching of files, which make the system more flexible on the integration of heterogeneous data. We insert this methodology into our a sharing platform for multi-source agricultural spatial data based on Web Service, which we introduced in the [5], we find the users can share their data as their wishes, without considering the compatibility of data formats. Meantime, by storing the mapping relations between standard data structure and users' personal data structure, users can see personalized information. From the users' standpoint, the method seems to be their private systems, which provide a good user experience.

The organization of the paper is as follows: Section II gives the related work and Section III outlines metadata standardization and data heterogeneity analysis. Section IV presents a framework and key algorithms for user's data standardization. The paper describes data display based on mapping supporting user personalization in SectionV. SectionVI shows the application case study of the approach. SectionVII concludes the paper.

II. RELATED WORK

In this section, we present an overview about the related work on sharing heterogeneous data and personalized information service.

There are several middleware technologies available for the XML-based integration of heterogeneous data sources [6]–[8]. On the perspective of XML-based middleware integrating heterogeneous data source or heterogeneous information, for example, the OPAL launched an Information Integration System, named YAT system, which aimed to realize the transformation from relation model and SGML to ODMG using Mediator/Wrapper middleware framework [9]. This research was further funded by OPAL and AQUARELLE program, and continued to focus on the XML middleware-based information integration system, in which the researchers studied on wrapping and querying heterogeneous data through XML integration view, and brought out the corresponding algebraic system [10]. XPERANTO [11], a middleware system, supports publishing object-relational data in the form of XML and query-based XML relational data view. Moreover, it was targeted at DTD schema. But it did not support the primary key constraint or the foreign key constraint description or the conversion of multi-table relation constraints. In [12], a middle-layer structure was proposed to integrate and query XML data sources. To deal with the coordinated sharing and interaction across multiple autonomous database management systems on data grids, a service-oriented system for distributed data querying and integration on grids was proposed in [13]. From the work presented in [9]–[11], we can see that mode conversion is the core of XML-based heterogeneous database integration middleware. On the aspect of the conversion of relational model and XML mode, there are some converting tools and related algorithms already. Some products implemented the transformation from relational model to XML document, such as ODBC2XML, Oracle's XSQL, SilkRoute etc. At present, most of the tools provide services for single data products instead of heterogeneous databases, much less the integration towards XML. Although ODBC2XML and SilkRoute supported a variety of database systems, they still did not support heterogeneous database integration, and the outputs of these tools are mostly not XML Schema but XML DTD. A storage strategy was proposed according to DTD mapping relational mode, which was a structure mapping method [14]. Based on the node model mapping method, a relational storage mode was introduced for XML data, called XRel [15]. XParent proposed in [16], was a relational storage model for XML data based on node model mapping method. Recently, to handle the semantic heterogeneity during the data integration [17], a schema and ontology-aided intelligent information integration was proposed in [18]. The integration of time and space is the key to the research on Temporal Geographic Information System. The [19] proposed an integration method of spatio-temporal data at the spatial database level combining with requirements of cadastral management. XML schema integration aims to create a mediated schema as a unified representation of existing heterogeneous sources sharing a common application domain. The [20] proposed a novel method, named XINTOR, for automating the integration of heterogeneous schemas, which can completely capture all of the concepts and relations in the sources without duplication, provided that the concepts do not overlap.

At the same time, personalized information service has been received more and more attention recently [21], [22]. Web Watcher [23] was a very well-known navigation device, which used an agent called information retrieval assistant, to facilitate the Web navigation for users browsing online. Alexa system [21] collected information usage and preference of users as a raw data for qualitative assessment of the site. Based on the raw data collected, Alexa system determined the relevant links to be provided in recommendation services. Letizia system [21], was used during users' browsing time to recommend links that may interest users. That links may well be relevant with the web content users have accessed. ProFusion Personal Assistant [24], [25] was an information filtering tool, which utilized clear user relevance feedback to detect their areas of interest. On the basis of feedback, the system filtered the results returned from the meta search engine ProFusion, which provides content serving users. In [26], a hierarchical intelligent information filtering model was proposed in response to the changeability of user interests and the uncertainty arisen from the dynamic changes of documents. The system can implement content-based and user-customized model of information filtering, and through optimized relevance feedback method and the limited interaction with the user, it can realize the adaptive learning of user interests. In the personalized information system proposed in [27], understanding behavior patterns can help the system predict users' next moves. The behavior patterns can be identified by machine-learning algorithms.

Traditional integration of heterogeneous data approaches pay little attention on personalized information service [28]. On the other hand, the current personalized information services, in some respects, have achieved very good effects, such as building and updating interest model, and active information recommendation and so on. However, little effort has been spent on the following two questions:

1) The existing personalized information service systems [21]–[29], mostly take users as information consumers only, ignoring the fact that they are at the same time the information providers. Those systems mainly focus on providing information consistently meeting users' requirements. When users want to provide their own data to share with others, how to ensure users to carry out convenient and flexible data access without any limitation of the process to be dominated by users is an issue needed to be solved. It also reflects a demand of users, though not a kind of specific information needs. However, the current personalized information service systems do not consider much about this aspect, which is very important to data integration and sharing.

2) Personalized information service system should first pay attention to meeting the needs of users. For example, the purpose of user interest model is to find information interesting users and provide the information to them. Another problem is in what form the information should be presented to the users. After all, people tend to be interested in the information which is presented in the way that caters to their individual requirements instead of a way that is rigid and uniform. Presently, most of the personalized information service systems mainly concentrate on the first issue and pay little attention on the second one [21]- [29].

In order to solve the two aforementioned problems, this paper proposes a user-directed data sharing method, in which the format of data is defined by users and the presentation of information is in accordance with their personalized preferences.

III. PRIMARY-METADATA STANDARDIZATION AND DATA HETEROGENEITY ANALYSIS

In this section, we introduce the basic concepts and notations used in this paper.

A. FORMAL METADATA STANDARDIZATION

Standard metadata specification is formally defined based on an XML schema (XSD) file. We know that the tools used to parse XML files also can be applied equally to the XSD files; so an XSD file can be handled like a common XML file. DOM model describes the hierarchical structure of an XML document in memory and provides a powerful approach to process nodes, traverse the document structure and perform other programming tasks. To represent a standard meta-data specification in a tree structure, the tree structure must meet certain conditions in order to establish mapping relations with DOM tree generated by the user's XML document.

In Figure.1, an example is presented to show several conditions that the tree structure needs to meet. To begin with, several concepts are defined:

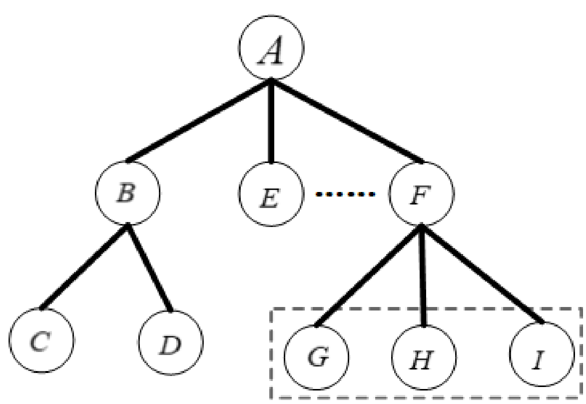


FIGURE 1. Tree structure of metadata standard.

1) **Path**: A path or an addressing path is defined as the orderly collection of all the nodes on the path starting from the root to the specified location of the node. The first element of the collection is the root node, and the last element is the current node. For example, if C is the current node in Figure.1, then $A \rightarrow B \rightarrow C$ is the addressing path of C.

2) **Prefix of a Path**: the path by removing the current node is the prefix of the path. For example, the prefix of the path of C is ($A \rightarrow B$)

3) **Subset of Names**: the collection of all the children of a certain node is called the subset of names. For example, the subset of the names of the node F is G, H, I, which is presented by the dashed box in Figure.1.

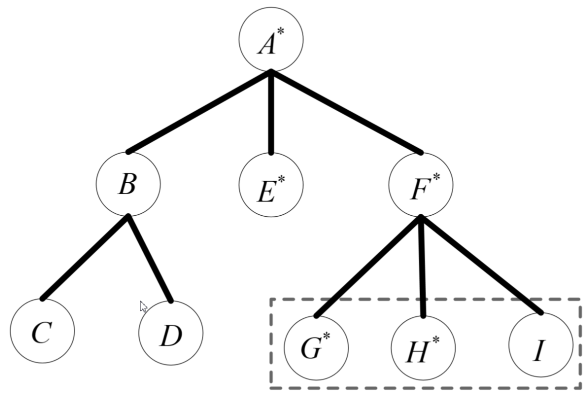


FIGURE 2. An example for DOM of metadata standard.

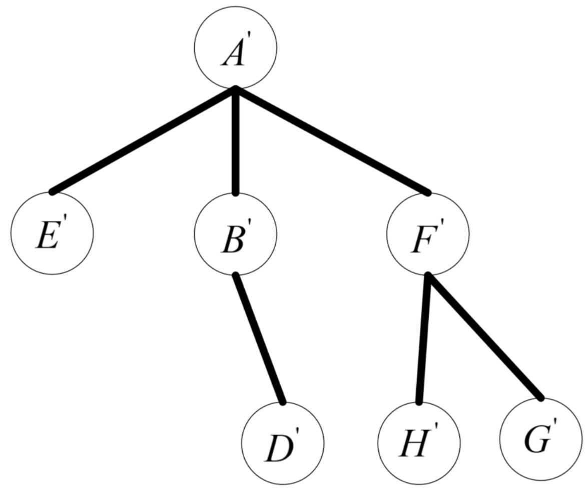


FIGURE 3. An example for allowing heterogeneous DOM.

B. HETEROGENEOUS DATA ANALYSIS PROVIDED BY DIFFERENT USERS

In this paper, the heterogeneous resource sharing model investigated is based on Web services. Service registry center provides the standard definition for describing an object. Besides, this definition is comprehensive, meaning that the number and types of its data items can be extended to include all the items of the objects users describe, if needed,. For example, the registry center may stipulate that the creation date, owner, file size and last update date must be included in the description of a document. And users might describe a document with its creation date, owner, last updated date, or other combinations. But no matter what constitute the description, it should not exceed the standard overall definition. In addition, another condition that user data should satisfy is, the superior and subordinate relations between data items (embodied in the tree as the parent-child relation between nodes) must be consistent with standard description. Although the users can describe items other than standard ones, for the same item, the user can otherwise give it any name. For example, as to ’owner’, it can be substituted by ’manager’ or ’publisher’ to express the same meaning of a word. In addition, as for the items on the same level (in the DOM tree, they are all the children of a node); the sequence can be arbitrary. Users must include compulsive items, while other items are optional.

In the followings, we present some tree structures to illustrate the difference between user-defined description and standard description on structure.

Figure.2 is a standard example of the definition of meta-data specification. In Figure.2, the nodes with $*$ are mandatory nodes, i.e. the descriptive items that users must select when defining objects. Without those nodes, the mapping process cannot guarantee the metadata obtained conform to standard structure. The rest nodes without $*$ labeled indicate that these descriptive items are optional.

Figure.3 shows three kinds of heterogeneous cases, in which the node names with ’′ removed corresponds to the standard name of the data items. Such as A', the corresponding standard data item in Figure.2 is the node A.

1) As to the three child nodes of A, the standard emergence order should be B, E, F, but here the emergence order is E, B, F. Such case of heterogeneity is permitted.

2) There is another heterogeneous case in Figure.3. node B has two children nodes in Figure.2, but there is only one in Figure.3. This heterogeneity is permitted, because the B node and its two child nodes are optional.

3) The last heterogeneity lies in the inconsistency of names, such as A corresponds to the A, and B corresponding to B. B and B may be the same one word. If they are not same words but they express the same meanings, such heterogeneity is also permitted.

These three types of heterogeneity can be allowed in any combination and in a variety of forms.

In addition, for the heterogeneous case that is not allowed, we also give an example, as shown in Figure.4. This DOM structure is not allowed, because the node D is supposed to be a child node of B node, but in the figure it is a sibling of B node, which violates the standard description. Moreover, I node should be a child node of F node, and now it a child of B node. This is also not allowed. Such two heterogeneous

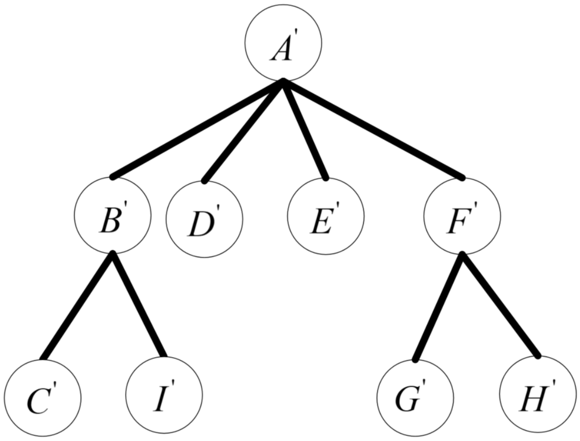


FIGURE 4. An example for prohibited heterogeneous DOM.

cases are not allowed to appear in the users' data provided; otherwise the mapping module may not work correctly.

IV. STANDARDIZATION FOR HETEROGENEOUS XML DATA

In this section, we first proposed a framework which is a standardize the data provided by different users. And then, The solutions to key technical problems are also introduced.

A. FRAMEWORK FOR USER'S DATA STANDARDIZATION

The framework of standardizing user data is shown in Figure.5, which contains XML document parsing, structural reduction, mapping based on synonyms,automatic conversion of XML documents by standard XML document,etc.

1) XML DOCUMENT PARSING

This is a DOM-based parsing approach, which reads XML documents into memory, and organizes them into trees. This component is realized by using the DOM parsing interface in .NET platform. Its inputs are the data in XML format submitted by different users, and its outputs are DOM trees which contain document elements, document declaration, comments, text and other elements, which are called the original trees in this paper. This step is mainly implemented by using the DOM parsing interface in the Xml Document class library.

2) FROM AN ORIGINAL TREE TO A REDUCTION TREE BY REDUCTION PROCESS

In the reduction process, the node types should be discriminated. Such text nodes, comment nodes, documents, declarations in the original tree are useless in the following procedure, which can be cleaned up. The result of the reduction process is a reduction tree. The implementation of this module also be utilized using the DOM parsing interface in the XmlDocument class library.

3) FROM A REDUCTION TREE TO A STANDARD TREE

This is the most important step, which is completed through mapping an arbitrary node in the reduction tree to a certain node in the standard tree. It outputs the mapping relations and saves them in the mapping table of data dictionary library. Furthermore, the execution of mapping algorithm demands synonym matching. When executing depth-first traversal on reduction tree, the main programming interface is the DOM parsing interface, while the access to the mapping table and the synonym table is related to database operations accomplished with ADO .NET.

4) AUTOMATIC GENERATION OF A XSLT FILE

According to the mapping table, the templates are generated for the conversion of each node in the reduction tree. There are three corresponding template generation methods for the three types of nodes. Section IV will give the template generation algorithms, which are mainly implemented by using the node path information and node names of the reduction tree saved in the mapping table. At the same time, the sequence of peer nodes in the standard tree is also taken into consideration. The output of this step is a XSLT style sheet file, which is equivalent to the conversion rules and makes available to the XSLT processor module.

5) AUTOMATIC CONVERSION OF XML DOCUMENTS

The XSLT processor loads style sheet file containing conversion rules to convert the XML files provided by different users into a standard format. The .NET platform has provided relevant interfaces of processor module. In this paper, the XmlTransform class interfaces are used to realize the conversion.

As is shown in the framework, th mapping, reduction, generation of XSLT template are the three core issues of the normalization process. In Section IV, these three issues will be addressed.

B. KEY STEPS FOR USER's DATA STANDARDIZATION

In this subsection, three key algorithms in the proposed framework for user's data standardization are presented.

1) CONSTRUCTION OF REDUCTION TREE

For an XML document, how does the DOM model represent it as a tree structure in memory? Let us consider the following simple example books. XML from W3School website.

The tree structure of this XML document is shown in Figure.6. We can see from the example, that the root element of the XML document is noted as a root, and other elements in the XML document are the descendants of the root. There are examples like the 'book' element as a child node of 'bookstore', and the 'title' element which is not only the child node of the 'book' element but also the peer node of the 'year' element. It is noteworthy that in the example, the text data and the attribute data are also represented as nodes.

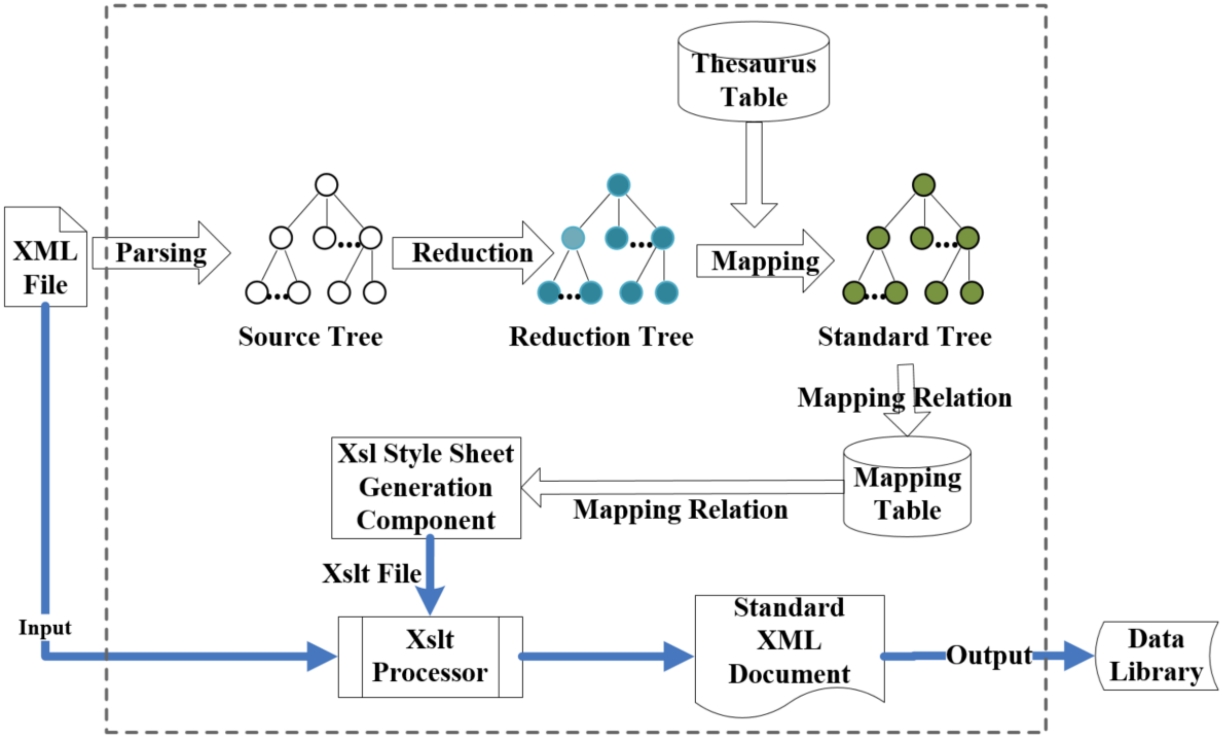


FIGURE 5. The framework of standardizing user data.

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- Copyright w3school.com.cn -->
<!--W3School.com.cn bookstore example -->
<bookstore>
  <book category="children">
    <title lang="en">Harry Potter</title>
    <author>J K.Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>
</bookstore>
```

FIGURE 6. XML for simple example books.

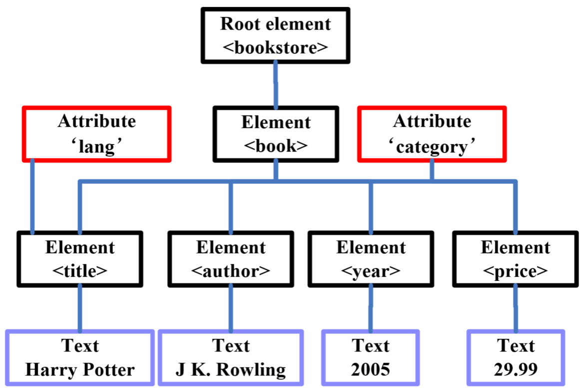


FIGURE 7. The DOM tree of books.xml.

A normal XML document usually has XML declaration, processing orders, XML document type declaration and a number of other processing instructions, comments and document elements. Aside from the document element, others are optional. Considering the convenience of research, we only consider the elements and attributes, and the comments, statements and other nodes are ignored. But it should be noted that these nodes in the DOM structure do exist. So the applications in practice should process these nodes, so as not to affect the generation of reduction tree.

For the text nodes in Figure.7 such as text 'Harry Potter', because it is just the value of data, and has nothing to do with the data structure and does not participate in mapping process, so we ignore the text nodes directly. In addition, attribute nodes, such as the attribute 'lang', represent another aspect of the elements. And it is not nesting there, so it can be seen as a child node of the element 'title'. In this way, the DOM tree structure of Figure.7 evolves into the structure of Figure.8.

As books.xml only contains an element 'book', so when the text nodes are removed, the structure in Figure.8 appears as a reduction tree. However, are all the XML data files of this format? The answer is negative. In most cases, usually there are a number of 'book' elements as the child nodes of the root. Moreover, in practice, the 'book' elements do not always include all the elements such as 'title', 'author', 'year', 'price', 'category'. They contain the mandatory items as well

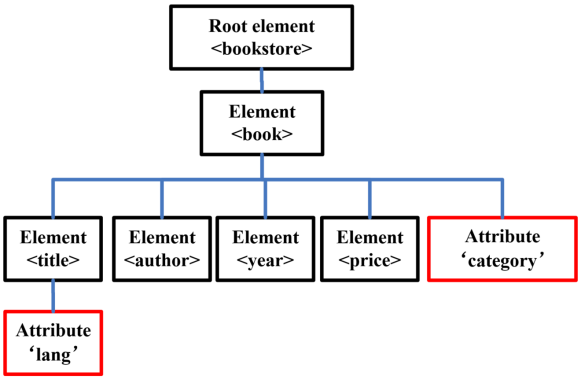


FIGURE 8. The DOM tree without text node.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- Copyright w3school.com.cn -->
<!--W3School.com.cn bookstore example -->
<bookstore>
  <book category="children">
    <title lang="en">Harry Potter</title>
    <author>J K.Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>
  <book category="cooking">
    <title lang="en">Everyday Italian</title>
    <author>Giada de Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
  </book>
</bookstore>
```

FIGURE 9. The expanded books.xml.

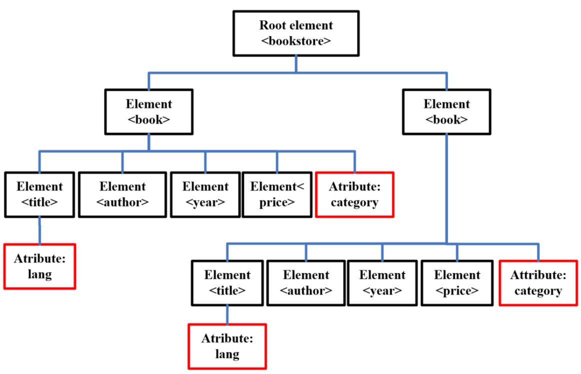


FIGURE 10. The DOM tree with two node named book.

as part of the optional ones. Such as the expanded books. The XML is shown in Figure.9:

What is obtained from the DOM tree generated from the XML document, after text nodes being ignored, should be a tree as shown in Figure.10.

We can see from Figure.10 that, the two 'book' elements have the same descriptive structure. And such a tree structure is clearly not the regular reduction tree. In fact, for books.xml, merely the use of the DOM structure in Figure.8 can express its meta-data structure. From Figure.10 to Figure.8, some combinations of nodes are needed, i.e. assembling the child nodes of the same parent node, merging the nodes with same names in order to keep nodes unique. The child node must be transferred to the remaining children of this node set. For the collection of its child nodes the same integration process is performed. Ultimately the result or the conditions that reduction tree must meet are the same as the tree structure defined in the standard metadata specification. This process of generating a reduction tree is called the process of reduction. Reduction process is actually a node merging process. The reduction process merges the same nodes in the children of a node. As a result, a regular reduction tree is obtained.

2) MAPPING ALGORITHM FOR USERS METADATA TO THE STANDARD DOM STRUCTURE

Section III has presented a descriptive definition of the simple standard meta-data and analyzed the data heterogeneity provided by different users. If the metadata user provided is of anomalous heterogeneous structure, the mapping relations can not be produced. If the data description is of regular structure, then we can establish mapping relations between nodes in the DOM structures of the customized metadata and the standard metadata.

In order to understand the mapping process clearly, we first give an example shown in Figure.11. In Figure.11, if we remove the broken lines from the figure, two trees are produced. The tree on the left is called a reduction tree, and the tree on the right is called a standard tree. One broken line represents a corresponding relation, for example, A node in the reduction tree corresponds to the A node in the standard tree. Because there is one and only one root node in the tree, the root node in the reduction tree will be reduced directly to the root node in the standard tree. For the nodes in the second layer of the reduction tree, there are three nodes E', B', F', which satisfy brother relations. That node set corresponds to B, E, F in the standard tree. To realize the corresponding relations between E', B', F' and B, E, F is difficult. In response to this kind of corresponding problems, we establish a vocabulary for the standard tree, of which the basic organization unit is the subset of the names. For example, B, E, F is a subset of the names. For each standard name in the subset, we establish a synonym group for it. And all of the synonyms are stored in a synonym table.

Based on the synonym table, the corresponding relation can be determined between the reduction tree and the standard tree. For example, we show how to determine the relations between the nodes in E', B', F' and B, E, F. First of all, assume that the relations is determined between the node root node A' and its corresponding node relation A. As the node E', a child node of A', its path can be found, which is (A', E'), and the prefix of the path is (A'). Because the corresponding relation between the prefix of the path of A' and A has been established, the node names in the prefix of a path in reduction tree will be replaced by the corresponding node names. Then

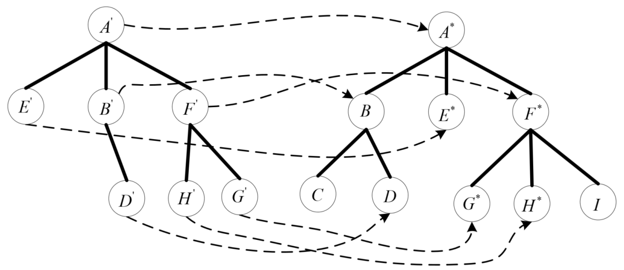


FIGURE 11. Mapping between the nodes of DOM trees.

we can get a prefix of the path in the standard tree, which possesses the only subset of the names (B, E, F) in the standard tree. Thus, as for the node E', the mapping module traverses the synonyms of B, E, and F in their synonym group. The ideal case is that the name E should appear in the synonym table of E', then we can determine the corresponding relation between E' and E. Using the same method, we can discover the correspondence of the other nodes with the standard tree.

Algorithm 1 presents the process for mapping user-defined metadata into the standard DOM structure.

Algorithm 1 Mapping from a metadata DOM structure specified by one user to the standard metadata structure

1: **Input:** Reduction tree based on data reduction
2: **Output:**The mapping relations from the reduction tree to the standard DOM tree
3: Take the root node of the reduction tree as the context node;
4: Obtain the paths of the context nodes and the prefix of each path;
5: If there is no node in the prefix of one path, it means the context node is the root. Establish the mapping relation between the context node and the root of the standard tree directly; otherwise, according to the mapping relations generated, convert the prefix to the prefix of the standard tree. The subset of the names is obtained;
6: Query the synonym table, the mapping relation of the context node in the standard tree corresponding to the context node in the reduction tree is obtained. Save the mapping relations;
7: Set the context node's child node that are far left and unvisited (if it exists) as the context node, go to step 2.
8: If the context node has no children, or all of the child nodes have been visited, set the context node's parent node to be the context node; go to step 5.
9: Repeat the above steps until the context node is the root. If all child nodes of root have been visited, then the algorithm terminates.

Algorithm 1 is a tree-based depth-first traversal algorithm and its output is the node mapping relations. This mapping algorithm is based on the reduction tree, which represents the user's data descriptive structure. In fact, the data provided by different users cannot be represented as a reduction tree, so in Section IV we have described how to use the DOM structure of user's data files to acquire a reduction tree.

```
<xsl:template match="/">
  <A>
    <xsl:apply-templates/>
  </A>
</xsl:template>
```

FIGURE 12. The template of root Node *A*.

3) GENERATION OF XSLT TRANSFORMATION TEMPLATES BASED ON MAPPING RELATIONS

The key step of building an XSLT document is to give the templates of all nodes. According to the mapping relations established by Algorithm 1, we illustrate the process of template construction in this section.

In Figure.1, there are three types of nodes in the standard tree, which are the root A, the leaf nodes C, D, E, G, H, I and the remaining nodes B, F. Template construction of these three types of nodes will be addressed respectively.

a: THE ROOT NODE

For the root node A, its template is:

Two points should be noticed in the template. First, the attribute of the value of 'match' is '/'. Second, because it has child nodes between <A> and </A>, there must exist the 'xsl:apply-templates' element, which does not contain 'select' attribute and which means to apply the templates to all the child nodes of the current node.

b: THE LEAF NODES

For the leaf node, taking D as an example, its template is shown in Figure.13:

```
<xsl:template match="A'/B'/C'">
  <D>
    <xsl:value-of-select="."/>
  </D>
</xsl:template>
```

FIGURE 13. The template of root Node *D*.

The function of element <D> is as same as that of the element <A> in the template of A. In the template of D, two points also need to be explained. First, the value of the 'match' attribute in the template is an xpath expression, which is obtained by the path of the node and the mapping relations. Second, the function of the element 'XSL:value-of' is to retrieve the text value of the objective given by the value of the attribute 'select'. For example, the text data of D node. The value of the attribute *select* is ., indicating the value is itself of the node. The purpose of the template is to identify

```
<xsl:template match="A'/F'">
  <F>
    <xsl:apply-templates/>
  </F>
</xsl:template>
```

FIGURE 14. **The template of root node *F*.**

the *D* node according to *match* attribute and copy its text data to *D* node.

c: THE REMAINING NODES

Except the root and the leaf nodes, the templates of all other remaining nodes are presented as followings. Taking node *F* as an example, its template is:

With regard to the templates of this type of nodes, the meaning of the value of attribute 'xpath' and its generation method of attribute 'match', are basically the same to those of the leaf nodes. In addition, 'xsl:apply-templates' element in the meaning is basically the same to root element.

For each mapping relation, we must judge its node type, and then carry out its template generation. Algorithm 2 presents the generation process of a template for each kind of node.

Algorithm 2 XSLT Transformation Template Generation Algorithm

1: **Input:** A mapping relation record
2: **Output:**A conversion template object
3: Extract the name of the user's data node U_{Title} in the mapping relation. Assume that the corresponding standard node is S_{Title} and node path is 'Path'.
4: Create a new node object 'template', which contains the attribute 'match', and assign the value of 'Path' to 'match';
5: Create a new node named S_{Title} and set its parent node 'template' object;
6: If the type of the node U_{Title} is a leaf, then go to step 5; Otherwise go to step 6;
7: Create a new node object 'value-of' containing an attribute 'select', of which the value is set to '.'. The parent node of the 'Value-of' object is set to S_{Title}. Switch to step 7;
8: Create a new node object 'apply-templates' whose parent node is S_{Title};
9: Template node 'template' has been built.

Before the implementation of Algorithm 2, the DOM object of an XSLT document has been established. Because both the generation of a node object and an attribute object in Algorithm 2 are on the basis of DOM object methods. In addition, the template objects as the output of Algorithm 2 are added to the DOM object. According to the transformation templates, XSLT processor can convert the user's XML file into a standard XML file.

V. MAPPING-BASED DATA DISPLAY SUPPORTING USER PERSONALIZATION

Section IV focuses on how to realize the user-centered aspects during the data import process. From the point of view of facilitating the user-friendly interface, this section discusses how to realize data display supporting user personalization. A framework for data display supporting user personalization is first presented, and then the main steps to realize the framework will be addressed.

A. FRAMEWORK FOR DATA DISPLAY SUPPORTING USER PERSONALIZATION

Figure.15 shows the basic framework of data display supporting user personalization, which mainly includes a request processing engine, a query module, an XSLT processor and an XSLT style sheet generation module.

1) The request processing engine is responsible for receiving the user's query request and starts the query module. At the same time, the request processing engine can also trigger the XSLT style sheet generation module.

2) The query module is responsible to find the data from the database. When the query module feedbacks the query results, the results can be processed to conform to the standard XML format and be input into the XSLT processor.

3) The XSLT style sheet generation module is responsible to convert the standard data structure to the XSLT style sheet table of the self-defined data structure for different users. The XSLT style sheet generation module will transfer the generated style sheet files to the XSLT processor.

4) The XSLT processor loads the XSLT style sheet files and converts standard XML data structure to the self-defined XML data structure.

5) The XSLT processor sends the personalized XML data to the request processing engine, and then the engine can display data in different ways to different users.

In the framework presented in Figure.15, the construction of the mapping relations and mapping-based data personalized display are two main steps, which are addressed by the following two sections.

B. CONSTRUCTION OF THE MAPPING RELATIONS

In Section IV, Algorithm 1 generates a number of mapping relations, and these mapping relations must first be used to generate the conversion style sheet XSLT file for converting a reduction tree to the standard tree. We should take into account of the path problem on the storage of these mapping relations, that is, as for the mapping relations among nodes in a reduction tree and in a standard tree, we need to store their paths in the trees, because only through the path can a node be uniquely determined. After generation of the conversion style sheet XSLT file, all the mapping relations are preserved to establish the correlations with the corresponding users. The mapping relations will be used for personalized data display.

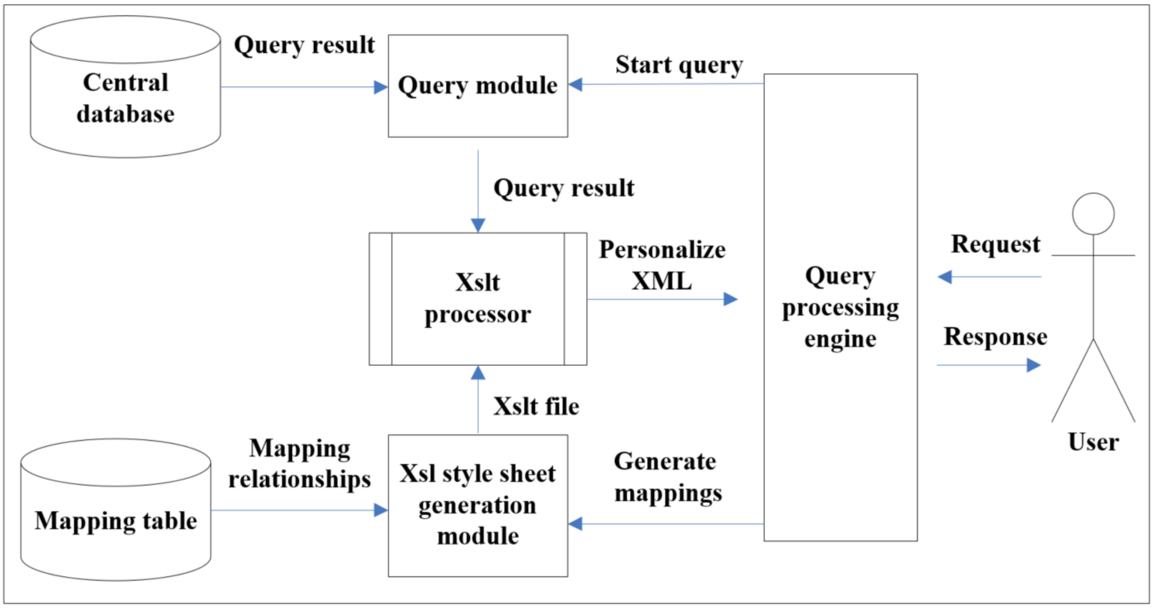


FIGURE 15. Framework for data display supporting user personalization.

MappingId	INT	4	☑
User_Word	VARCHAR	50	☐
Standard_Word	VARCHAR	50	☐
User_Identity	VARCHAR	50	☐
Serial_Number	INT	4	☐
Path_Info	VARCHAR	100	☐
Description	VARCHAR	100	☐

FIGURE 16. Mapping table structure.

The mapping table storing the mapping relations can be regarded as part content of a user's profile model, and the user profile model indicates how to correspond the standard structure of data to his familiar data structure. Each of the mapping relation in the mapping table must be associated with a user, therefore, there must exist a user information table. In addition, in order to reproduce user's data structure, in user's reduction tree, the node sequence of child nodes with the same parent node must be recorded. The user identity, path information, and the sequence of appearance of sibling nodes are three indispensable ingredients to ensure the reproduction of reduction tree. Figure.16 shows the structure of the mapping table. A record in the table is the information about the mapping from a node in the DOM tree of a user's metadata to a DOM tree of the standard metadata.

Algorithm 1 generates a number of mapping relations, and this process does not require user involvement. In this case, the acquisition of the mapping relations is completely automated from the data sharing process. That is, if a user provides data to be shared, the mapping relations are not necessary to be provided again by the users.

```
<Name>rainfall distribution</Name>
<Publisher>Jinan Agriculture Bureau</Publisher>
<AliasName>rainfall distribulture chart</AliasName>
<Version/>
<VersionData>2018-3-20</VersionData>
<Synopsis>rainfall</Synopsis>
<Progress/>
<RepresentationType></RepresentationType>
<SpatialResolution></SpatialResolution>
<Key>rainfall</Key>
```

FIGURE 17. A standard XML fragment.

```
<Name>rainfall distribution</Name>
<Owner>Jinan Agriculture Bureau</Owner>
<AliasName>Rainfall distribulture chart</AliasName>
<Version/>
<VersionData>2018-3-20</VersionData>
<Synopsis>Rainfall</Synopsis>
<Progress/>
<RepresentationType></RepresentationType>
<SpatialResolution></SpatialResolution>
<Key>Rainfall</Key>
```

FIGURE 18. One user's XML fragment.

C. MAPPING-BASED PERSONALIZED DATA DISPLAY

The data provided by the data center to each user is a standard data files, such as the following XML fragment in Figure.17:

The following is the data format that one user provides the data. When this user queries data from the center, he/she wants display the data displayed in the way providing data to the center.

We can see there are no differences between the data contents in these two display fragments, but the description of the data structure is different. For example, the item 'Publisher'

```
<xsl:template match="Document/MetaDataofDS/Publisher">
  <Owner>
    <xsl:value-of-select="."/>
  </Owner>
</xsl:template>
```

FIGURE 19. **The mapping template for 'Publisher' and 'Owner'.**

in the data center should be displayed as 'Owner'. According to XSLT related knowledge we know that, to realize this process, the XSLT processor needs to process the XML file according to a XSLT style sheet file. It is not difficult to generate the XSLT style sheet. However, the key problem is to determine the correlation between the nodes. That is, the XSLT processor should know that the 'Publisher' corresponds to the 'Owner'. Because the mapping table has stored the mapping relations for each user, the XSLT style sheet can be generated based on the mapping relations. Algorithm 2 has presented the generation of the XSLT style sheet based on the mapping relations. According to Algorithm 2, the mapping template for 'Publisher' and 'Owner' is as follows:

VI. APPLICATION CASE STUDY

We apply the methodology proposed to an application, a sharing platform for multi-source agricultural spatial data based on Web Service, which we introduced in the [5]. The user-friendly interface and experience of the platform show that the methodology proposed is more feasible than our previous work [5] .

A. BACKGROUND AND REQUIREMENT

In order to achieve the sharing of distributed agricultural spatial data, we have independently designed and developed a sharing platform for multi-source and heterogeneous agricultural spatial data based on Web Services, which can provide unified services for data sharing and operation class. Using the Web Services technology, we have created a reasonable spatial data sharing mechanism enabling the existing agricultural spatial data distributed in different departments to be integrated and assembled effectively. The sharing mechanism can also realize the establishment of interoperability platform, in which the dynamic access, query, spatial analysis, and decision-making of all kinds of multi-source and heterogeneous agricultural spatial data are realized. Figure.20 shows the architecture of the sharing platform.

In order to solve the problem of the inconsistency among data schemas provided by different user nodes, it is necessary to provide the function of data transformation, i.e., the conversion of the schema of shared data to the same intermediate standardized schema. In the sharing platform, the GML schema is adopted. Besides, a register center is designed as a part of the platform to collect data. As shown in Figure.20, the part circumscribed in dotted line is the register center module, which plays a pivotal role in the operation of whole platform.

B. DEVELOPMENT OF THE REGISTER CENTER

The services provided by the register center consist of the registration and query of data. Figure.21 shows the major interfaces and the principle of this module. From Figure.21, we can see that at least two interface classes are needed: interface class for publishing data and interface class for querying data. The interface class for publishing data performs the function of inputting the metadata of the spatial data files to share from sub-nodes into the central database, while the query interface class enables registered users to find the metadata of spatial data that they need in the central database by address.

To manage the metadata in the central database in the register center more effectively, a standard data structure specification is required. In practice, it is possible that each node has its own data structure specification defined independently; meanwhile, data processing systems based on the independent data structure specification may already exist. In this case, the data provided when sub-node users publish metadata in register center is probably based on self-defined structure. Thus, before inputting data into the central database, it is necessary for the register center to standardize the data to satisfy the unified specification. When a user queries metadata in the register center, if the register center can convert the data to the user-customized schema according to different users, the incompatibility between the schemas for different users' systems and the standardized schema would be avoidable, and moreover, it embodies the idea of personalized information service, and provides users with better services.

C. DATA STANDARDIZATION

In Figure.22, agricultural spatial data from two distinct districts for sharing are presented respectively. On the left, it shows the spatial metadata of Jinan agriculture bureau to be submitted to the register center. On the right, it shows the data from Qingdao agriculture bureau. We can see that the XML data submitted from these two different nodes, in the aspect of data structure, are different with each other, which means that they are defined separately and independently. In the standard specification of the register center, the fields of 'name', 'publisher', 'alias', 'abstract', 'key' and 'category' are obligatory; these fields are at the same node level, and the sequence of them may be random and unspecified; other fields are dispensable.

On condition that the six obligatory fields and any number of other optional fields are provided, two aspects of the heterogeneity between the standardized and user-defined data structure can be permitted:

(1) The tags of the fields can be named by users separately. For example, in the left of Figure.22, the metadata of the node is 'rainfall distribution', of which the tag name is defined as 'Topic' here, while the standardized name of the corresponding fields should be 'Name'. In the right of Figure. 22, the name of the same field is defined according to the standard specification as 'Name'. In this way, the method

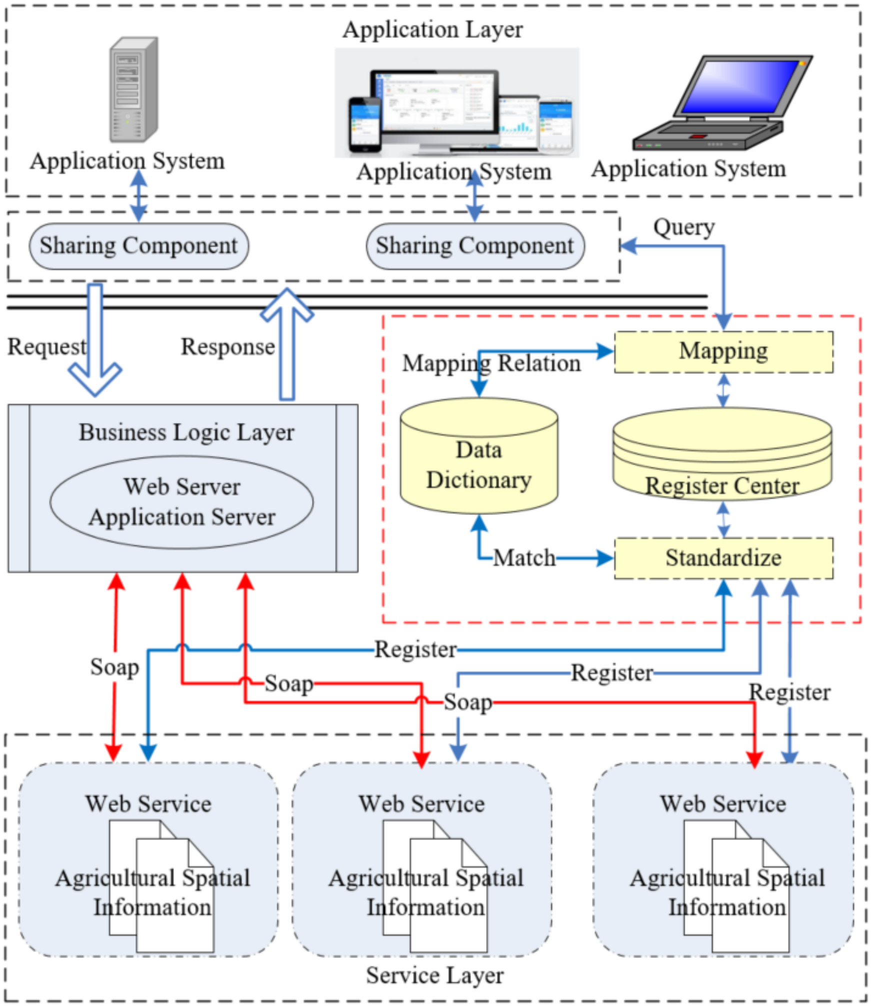


FIGURE 20. Architecture of a sharing platform for agricultural spatial data.

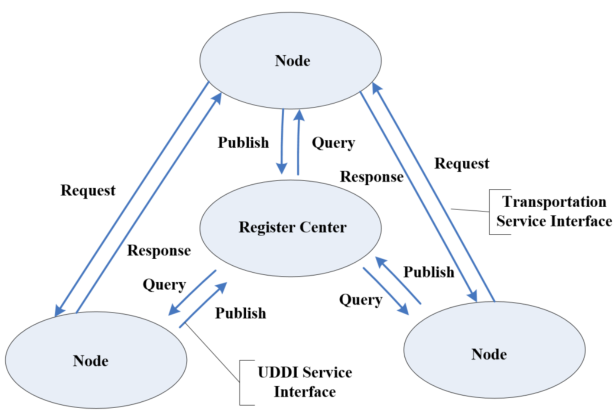


FIGURE 21. Function and principle of registration center.

of naming the field in the left of Figure. 22 is an example of the heterogeneity of naming a particular field.

(2) The sequence of fields can be specified by users. For example, the order of the metadata items in the left of Figure. 22 is:

<Topic> <VersionDate> <Alias> <Abstract> <Provider> <Version> <Progress> <RepresentationType> <SpatialResolution> <Key> <SubjectType> <DataVolume> <Category>.

However, in the right of Figure.22, the sequence of the metadata items is:

<Name> <Summary> <VersionDate> <AliasName> <Owner> <Version> <Progress> <RepresentationType> <SpatialResolution> <Keyword> <SubjectType> <DataVolume> <Classification>.

From this aspect, the disaccord among different sequences of data fields is an example of the heterogeneity of different orders.

The differences on the description of data submitted by these two different nodes are considerable. This kind of problem also happens when other users upload their data. Through the standardization process presented in Section III, we can unify the data submitted by the two users into a standard structure and store the data into the database. During the standardization process, it does not require users providing sharing data to change their own data structures. Figure.23 shows a

```xml
<?xml version="1.0" encoding="utf-8"?>
<Document>
  <MetaDataofNode>
    <Name>Agricultural Pest</Name>
    <Summary>Pest</Summary>
    <VersionData>2018-3-20</VersionData>
    <Alias>Pest Distribulture Chart</Alias>
    <Owner>Qingdao Agriculture Bureau</Owner>
    <Version/>
    <Progress/>
    <RepresentationType>
    <SpatialResolution>
    <Key>pest</Key>
    <SubjectType/>
    <DataVolume/>
    <Classification/>
  </MetaDataofNode>
  <MetaDataofNode>
    <Name>Crop Distribution</Name>
    <Summary>Wheat,Maize</Summary>
    <VersionData>2018-3-20</VersionData>
    <Alias>Crop Distribution Chart</Alias>
    <Owner>Qingdao Agriculture Bureau</Owner>
    <Version/>
    <Progress/>
    <RepresentationType>
    <SpatialResolution>
    <Key>Wheat</Key>
    <SubjectType/>
    <DataVolume/>
    <Classification/>
  </MetaDataofNode>
</Document>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<Document>
<MetaDataofNode>
<Topic>Rainfall Distribution</Topic>
<VersionData>2018-3-20</VersionData>
<Alias>Rainfall Distribution Chart</Alias>
<Abstract>rainfall</Abstract>
<Provider>Jinan Agribulture Bureau</Provider>
<Version/>
<Progress/>
<RepresentationType>
<SpatialResolution>
<Key>rainfall</Key>
<SubjectType/>
<DataVolume/>
<Category/>
</MetaDataofNode>
<MetaDataofNode>
<Topic>Crop Distribution</Topic>
<VersionData>2018-3-20</VersionData>
<Alias>Crop Distribution</Alias>
<Abstract>cotton,soybean</Abstract>
<Provider>Jinan Agributlture Bureau</Provider>
<Version/>
<Progress/>
<RepresentationType>
<SpatialResolution>
<Key>cotton,soybean</Key>
<SubjectType/>
<DataVolume/>
<Category/>
</MetaDataofNode>
</Document>
```


FIGURE 22. **Function and principle of registration center.**

unified data structure which is stored in the database after data standardization. In Figure.23, two sub-trees are expanded. The first sub-tree represents data from Jinan Agriculture Bureau. Compared to the left of Figure.22, Figure.23 shows not only different tag names ('Topic' to 'Name', 'Alias' to 'AliasName'), but also a different sequence of tags as:

<Name> <Publisher> <AliasName> <Version> <VersionDate> <Synopsis> <Progress> <RepresentationType> <SpatialResolution> <Key> <SubjectType> <DataVolume> <Group>.

The naming method and sequence for tags in Figure.23 comply with the standard specification defined in the register center. It is easy to conclude that, the second sub-tree in Figure.23 which comes from the right of Figure .22 is consistent with the first sub-tree structurally. The standardization process regulates them to conform to the standardized schema described in the register center.

This demonstrates the feasibility of the standardization process. The unification of data description structures is conducive not only to storage but also to management. After being standardized, the data will comply with the standard specification defined in the register center precisely. After standardization, the other modules in the register center will no longer need to deal with heterogeneity, as long as their interfaces are designed following the standard data description specification. Thus, the loose coupling of core modules with peripheral modules for interaction in the register center and the code robustness of the data center, are improved.

The standardization process not only facilitates the design of the register center, but it also reduces the workloads of clients, improves the friendliness to data providers and achieves full compatibility of the platform with data from any individual user. In this way, the heterogeneity between user data and the data in the form of standardized schema can be ignored by every data provider, whose custom and experience are thus fully respected.

D. MAPPING-BASED PERSONALIZED DATA PRESENTATION

The examples given in Figure.22 show four items of data provided by Jinan and Qingdao Agriculture Bureau, respectively, that present the standardized schema after being normalized in the register center. Two of the four items are shown in Figure.23, i.e. one named 'crop distribution' provided by Jinan Agriculture Bureau and the other named 'agricultural pest' provided by Qingdao Agriculture Bureau. While the

```xml
<?xml version="1.0" encoding="utf-8"?>
<Document>
  <MetaDataofDS>
    <Name>Crop Distribution</Name>
    <Publisher>Jinan Agriculture Bureau</Publisher>
    <AliasName>Pest Distribulture Chart</AliasName>
    <Version/>
    <VersionData>2018-3-20</VersionData>
    <Synopsis>cotton,soybean</Synopsis>
    <Progress/>
    <RepresentationType>
    <SpatialResolution>
    <Key>pest</Key>
    <SubjectType/>
    <DataVolume/>
    <Group/>
  </MetaDataofDS>
  <MetaDataofDS>
    <Name>Agricultural Pest</Name>
    <Publisher>Qingdao Agriculture Bureau</Publisher>
    <AliasName>Pest Distribulture Chart</AliasName>
    <Version/>
    <VersionData>2018-3-20</VersionData>
    <Synopsis>pest</Synopsis>
    <Progress/>
    <RepresentationType>
    <SpatialResolution>
    <Key>pest</Key>
    <SubjectType/>
    <DataVolume/>
    <Group/>
  </MetaDataofDS>
</Document>
```

FIGURE 23. **Standardized data in the registration center.**

left two items of data which are submitted to the register center in Figure.22 are not expanded in Figure.23, i.e. the data named 'rainfall distribution' from Jinan Agriculture Bureau and the data named 'crop distribution' from Qingdao Agriculture Bureau.

In Figure.23, the data in the register center is organized according to the standard data specification. Then, if a user from Jinan Agriculture Bureau wants to view data in the data center, what kind of result it would be? When the user queries the data in the data center, the platform will provide data on the working principle illustrated in Figure.16. First, the platform receives query command from Jinan Agriculture Bureau, and then finds all data that meet the query. Second, according to the user ID of Jinan Agriculture Bureau and the Dictionary Database, the mapping relations between the user's customized data schema and the standardized data schema are discovered. Based on the mapping relations, the XSLT file is created to transform the data from standard schema to the customized schema of the Jinan Agriculture Bureau. As the data in standardized schema retrieved and the XSLT file generated are input into the XSLT processor, the result data that qualifies the user's customized schema can be obtained.

As illustrated in Figure.24, to save page space, the query user interface is not provided here. By default, all data in the central database can be retrieved and showed when users log in. Furthermore, the formatted display of the retrieved XML file is actualized here for convenience. In Figure.24, the login user is from 'Jinan Agriculture Bureau', and the four data items from Jinan Agriculture Bureau and Qingdao Agriculture Bureau are all displayed in the same format. Here, a GridView control is applied for the formatted display of data result. The tags of XML file are displayed as the content of table head of the GridView control. We can see from Figure.24 that, the sequence of the fields in the table head is:

<Topic> <VersionDate> <Alias> <Abstract> <Provider> <Version> <Progress> <RepresentationType> <SpatialResolution> <Key><SubjectType> <DataVolume> <Category>

The order and the field names here are as same as those in the left of Figure.22, from which we can see, our method for the personalized data presentation based on mapping is able to recover data in the form of user's customized data schemata effectively and precisely.

Moreover, in Figure.24, it shows that in the column of 'Provider', there are two data items for Jinan Agriculture Bureau and the other two for Qingdao Agriculture Bureau. As we know, these two users provide self-structured data, but the data displayed here are organized in accordance with the user's customized data schema. Figure.25 shows the results for the user from Qingdao Agriculture Bureau. The four data items that this user receives are also equal to the data schema defined by Qingdao Agriculture Bureau structurally whether the data comes from the user or not. In Figure.25, the order and the field names are:

<Name> <Summary> <VersionDate> <AliasName> <Owner> <Version> <Progress> <RepresentationType> <SpatialResolution> <Keyword> <SubjectType> <DataVolume> <Classification>

As we can see, the data contents are all the same in Figure.24 and Figure.25. Exactly, they are the two data items from Jinan Agriculture Bureau and the two from Qingdao Agriculture Bureau, which are all stored in the database of register center. The sequence and field names in Figure.23 are:

<Name> <Publisher> <AliasName> <Version> <VersionDate> <Synopsis> <Progress> <RepresentationType> <SpatialResolution> <Key> <SubjectType> <DataVolume> <Group>

For example, the field and the data for the tag 'Name' in the standardized data schema are listed in the first column, which correspond to the first field 'Topic' in Figure.24 and the first field 'Name' in Figure.25. The field and the data for the tag 'Publisher' in the standardized data schema are listed in the second column, which correspond to the fifth field 'Provider' in Figure.24 and the fifth field 'Owner' in Figure.25. The field and the data for the tag 'Synopsis' in the standardized data schema are listed in the sixth column, which correspond to the fourth field 'Abstract' in Figure.24 and the second field 'Summary' in Figure.25. From these examples, we can see that the sequence and the names of fields that users see are different to those in the standard data

Personalized Data Presentation for Agricultural Spatial Information Sharing Platform

Current User: Jinan Agriculture Bureau OK

Generate XML File | Generate XSL File | Transformation Result | View Result

Topic	VersionDate	Alias	Abstract	Provider	Version	Progress	RepresentationType	SpatialResolution	Key
Rainfall Distribution	2018-3-20	Rainfall Distribution Chart	rainfall	Jinan Agricultural Bureau					rainfall
Crop Distribution	2018-3-20	Crop Distribution Chart	cotton, soybean	Jinan Agricultural Bureau					cotton, soybean
Agriculture Pest	2018-3-20	Pest Distribution Chart	pest	Qingdao Agricultural Bureau					pest
Crop Distribution	2018-3-20	Crop Distribution Chart	wheat, maize	Qingdao Agricultural Bureau					wheat

FIGURE 24. **Data presentation for jinan agriculture bureau.**

Personalized Data Presentation for Agricultural Spatial Information Sharing Platform

Current User: Qingdao Agriculture Bureau OK

Generate XML File | Generate XSL File | Transformation Result | View Result

Name	Summary	VersionDate	AliasName	Owner	Version	Progress	RepresentationType	SpatialResolution	Keyword
Rainfall Distribution	rainfall	2018-3-20	Rainfall Distribution Chart	Jinan Agricultural Bureau					rainfall
Crop Distribution	cotton, soybean	2018-3-20	Crop Distribution Chart	Jinan Agricultural Bureau					cotton, soybean
Agriculture Pest	Pest	2018-3-20	Pest Distribution	Qingdao Agricultural Bureau					pest
Crop Distribution	Wheat, Maize	2018-3-20	Crop Distribution Chart	Qingdao Agricultural Bureau					Wheat

FIGURE 25. **Data presentation for Jinan Agriculture Bureau.**

description specification, and in the data views for Jinan and for Qingdao Agriculture Bureau the sequences and names of fields are also different, which fully embodies the personalized data presentation.

VII. CONCLUSION

By the research on XML-based data integration and personalize service, this paper proposes a method of implementing user-driven data sharing. It achieves not only standardized access of users' data based on XSLT, but also the personalized presentation based on mapping between user's metadata and the standard metadata. There are two main contributions in this paper:

1) Standardization of heterogeneous XML data. By introducing synonym table, we can realize automatic matching between the standard DOM tree and that provided by users, thus we can obtain the mapping relations. According to the mapping relations obtained, the XSLT file can be generated so as to generate the standard XML document that contains standard data. With that method, the process from user-defined data structure to standard data structure can be transformed automatically, thus reducing much manual participation of users.

2) Personalized data presentation. Based on the mapping relations from the standardization procedure and XSLT, the system can show different information to different users without their participation, reflecting the user differences between users and improving their experiences.

To achieve the two contributions, this paper has solved the following main problems: defining the standard schema of data center and the conditions, presenting the reduction algorithm of raw DOM tree to obtain the schema defined by users, creating XSLT templates to support the procedure of creating XSLT Style Sheet, establishing the mapping relations between users' metadata and the standard metadata, and providing users with the function of creating mappings manually, etc. The approaches proposed to solve the problems above have been verified in the application of a sharing platform for the multi-source agricultural spatial data. Although prototype application shows our methodology is feasible, there are still a lot of problems:

1) Data standardization process requires a complete synonym table to support. There is no reasonable solution when the automatic matching fails because of the incompleteness of synonym table. So the synonym table should be big enough.

(2) The approach can only provide personalized information service to the users who have already provided data to be shared. To other users without sharing data, it is more difficult to provide personalized information service with the approaches proposed in the paper.

(3) A public metrics need to present. There is no public evaluation index to compare different methods. In our paper, we illustrate the effectiveness and practicability of our

method from the perspective of application of a sharing platform for the multi-source agricultural spatial data. There is no public evaluation method to evaluate the scheme proposed method. Therefore, we will try to propose a public evaluation system on integration and sharing heterogeneous data with the consideration of personalized information service.

REFERENCES

[1] A. Y. Halevy, N. Ashish, D. Bitton, M. Carey, D. Draper, J. Pollock, A. Rosenthal, and V. Sikka, "Enterprise information integration: Successes, challenges and controversies," in *Proc. ACM SIGMOD Int. Conf. Manage. Data (SIGMOD)*, 2005, pp. 778–787. [Online]. Available: http://doi.acm.org/10.1145/1066157.1066246

[2] G. Zacharewicz, S. Diallo, Y. Ducq, C. Agostinho, R. Jardim-Goncalves, H. Bazoun, Z. Wang, and G. Doumeingts, "Model-based approaches for interoperability of next generation enterprise information systems: State of the art and future challenges," *Inf. Syst. e-Bus. Manage.*, vol. 15, no. 2, pp. 229–256, 2016.

[3] F. Ribeiro, J. Metrôlho, J. Leal, H. Martins, and P. Bastos, "A mobile application to provide personalized information for mobility impaired tourists," in *Trends and Advances in Information Systems and Technologies*, Á. Rocha, H. Adeli, L. P. Reis, and S. Costanzo, Eds. Cham, Switzerland: Springer, 2018, pp. 164–173.

[4] J. I. A. Shengbin and X. Yang, "Temporal semantic understanding for intelligent service systems," *J. Comput. Appl.*, vol. 38, no. 3, p. 620, 2018.

[5] W. Wang, Q. Zeng, X. Hao, X. Tang, N. Xie, and X. Yang, "Development of an agricultural spatial information sharing platform for supporting user personalization," *Inf. Technol. J.*, vol. 9, no. 5, pp. 927–934, 2010.

[6] S. Li, D. H. Zhang, J. T. Zhou, G. H. Ma, and H. Yang, "An XML-based middleware for information integration of enterprise heterogeneous systems," *Mater. Sci. Forum*, vols. 532–533, pp. 516–519, Dec. 2006.

[7] W. Li, L. Yan, F. Zhang, and X. Chen, "A formal approach of construction fuzzy XML data model based on OWL 2 ontologies," *IEEE Access*, vol. 6, pp. 22025–22033, 2018.

[8] X. Chen, L. Yan, W. Li, and Z. Ma, "Reengineering fuzzy spatiotemporal UML data model into fuzzy spatiotemporal XML model," *IEEE Access*, vol. 5, pp. 17975–17987, 2017.

[9] S. Alagic, "Type-checking OQL queries in the ODMG type systems," *ACM Trans. Database Syst.*, vol. 24, no. 3, pp. 319–360, 1999.

[10] Z. Wang, Q. Rui, and L. Qing, "Aircraft support information system integration based on SOA and Web service," *Comput. Eng.*, vol. 44, no. 1, pp. 91–97, 2018.

[11] M. J. Carey, J. Kiernan, J. Shanmugasundaram, E. J. Shekita, and S. N. Subramanian, "XPERANTO: Middleware for publishing object-relational data as XML documents," in *Proc. 26th Int. Conf. Very Large Data Bases (VLDB)*, 2000, pp. 646–648. [Online]. Available: http://dl.acm.org/citation.cfm?id=645926.671862

[12] F. V. Silveira and C. A. Heuser, "A two layered approach for querying integrated XML sources," in *Proc. 11th Int. Database Eng. Appl. Symp. (IDEAS)*, 2007, pp. 3–11. doi: 10.1109/IDEAS.2007.8.

[13] C. Comito, A. Gounaris, R. Sakellariou, and D. Talia, "A service-oriented system for distributed data querying and integration on grids," *Future Gener. Comput. Syst.*, vol. 25, no. 5, pp. 511–524, 2009.

[14] A. A. A. El-Aziz and A. Kannan, "Storing XML rules in relational storage of XML DTD," in *Proc. 2nd Int. Conf. Comput. Sci., Eng. Inf. Technol. (CCSEIT)*, 2012, pp. 408–412. [Online]. Available: http://doi.acm.org/10.1145/2393216.2393285

[15] M. Yoshikawa, T. Amagasa, T. Shimura, and S. Uemura, "XRel: A path-based approach to storage and retrieval of XML documents using relational databases," *ACM Trans. Internet Technol.*, vol. 1, no. 1, pp. 110–141, Aug. 2001. [Online]. Available: http://doi.acm.org/10.1145/383034.383038

[16] H. Jiang, H. Lu, W. Wang, and J. X. Yu, "XParent: An efficient RDBMS-based XML database system," in *Proc. 18th Int. Conf. Data Eng. (ICDE)*, 2002, pp. 335–. [Online]. Available: http://dl.acm.org/citation.cfm?id=876875.879007

[17] A. M. Rinaldi and C. Russo, "A matching framework for multimedia data integration using semantics and ontologies," in *Proc. IEEE 12th Int. Conf. Semantic Comput. (ICSC)*, Jan. 2018, pp. 363–368. doi: 10.1109/ICSC.2018.00074.

[18] J.-L. Seng and I. L. Kong, "A schema and ontology-aided intelligent information integration," *Expert Syst. Appl.*, vol. 36, no. 7, pp. 10538–10550, 2009.

[19] Y. T. Fan, J. Y. Yang, D. H. Zhu, and K. L. Wei, "A time-based integration method of spatio-temporal data at spatial database level," *Math. Comput. Model.*, vol. 51, nos. 11–12, pp. 1286–1292, Jun. 2010. doi: 10.1016/j.mcm.2009.10.032.

[20] H.-Q. Nguyen, D. Taniar, J. W. Rahayu, and K. Nguyen, "Double-layered schema integration of heterogeneous XML sources," *J. Syst. Softw.*, vol. 84, no. 1, pp. 63–76, 2011.

[21] D. Mabrouk, S. Rady, N. Badr, and M. E. Khalifa, "A survey on information retrieval systems' modeling using term dependencies and term weighting," in *Proc. 8th Int. Conf. Intell. Comput. Inf. Syst. (ICICIS)*, Dec. 2017, pp. 321–328.

[22] Z. Zhao, C. Li, X. Zhang, F. Chiclana, and E. H. Viedma, "An incremental method to detect communities in dynamic evolving social networks," *Knowl.-Based Syst.*, vol. 163, pp. 404–415, Jan. 2019.

[23] R. ARMSTRONG, "Webwatcher: A learning apprentice for the world wide Web," in *Proc. AAAI Spring Symp. Inf.Gathering Heterogeneous, Distrib. Environ.*, Stanford, CA, USA, 1995, pp. 6–12. [Online]. Available: https://ci.nii.ac.jp/naid/10010357682/en/

[24] A. H. Keyhanipour, B. Moshiri, M. Kazemian, M. Piroozmand, and C. Lucas, "Aggregation of Web search engines based on users' preferences in WebFusion," *Know.-Based Syst.*, vol. 20, no. 4, pp. 321–328, May 2007. [Online]. Available: http://dx.doi.org/10.1016/j.knosys.2006.08.001

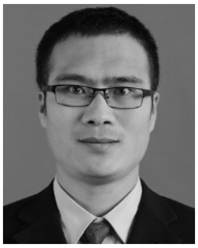
[25] M. M. Rahman, S. Yeasmin, and C. K. Roy, "An IDE-based context-aware meta search engine," in *Proc. 20th Work. Conf. Reverse Eng. (WCRE)*, Oct. 2013, pp. 467–471.

[26] S. A. Macskassy, "New techniques in intelligent information filtering," Ph.D. dissertation, Rutgers, The State Univ. New Jersey, New Brunswick, NJ, USA, 2003.

[27] C. Yang, F. Yan, and S. V. Ukkusuri, "Unraveling traveler mobility patterns and predicting user behavior in the Shenzhen metro system," *Transportmetrica A, Transp. Sci.*, vol. 14, no. 7, pp. 576–597, 2018. doi: 10.1080/23249935.2017.1412370.

[28] S. Kausar, X. Huahu, I. Hussain, Z. Wenhao, and M. Zahid, "Integration of data mining clustering approach in the personalized E-learning system," *IEEE Access*, vol. 6, pp. 72724–72734, 2018.

[29] Z. Zhao, W. Liu, Y. Qian, L. Nie, Y. Yin, and Y. Zhang, "Identifying advisor-advisee relationships from co-author networks via a novel deep model," *Inf. Sci.*, vol. 466, pp. 258–269, Oct. 2018.



CHAO LI received the B.S. and M.S. degrees from the Shandong University of Science and Technology, in 2007 and 2009, respectively, and the Ph.D. degree from the Chinese Academy of Sciences, in 2014. He was a Visiting Scholar with Hong Kong University of Science and Technology from 2014 to 2015. He is currently a Lecturer with the Shandong University of Science and Technology. He is a member of CCF, and a member of Key Laboratory of Embedded System and Service Computing, Tongji University. His research interests include social media, natural language processing, data mining, and network embedding learning.

QINGTIAN ZENG received the B.S. and M.Sc. degrees from the Shandong University of Science and Technology, in 1998 and 2001, respectively, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, in 2005. In 2005 and 2008, he was a Visiting Fellow with the City University of Hong Kong. He is currently a Professor with the Shandong University of Science and Technology. He is a Senior Member of CCF. His main research interests include process mining, Petri net, data mining, Web mining, and machine learning.

HUA DUAN received the B.S. and M.S. degrees from the Shandong University of Science and Technology, in 1998 and 2001, respectively, and the Ph.D. degree from Shanghai Jiao Tong University, in 2006. She is currently with the Shandong University of Science and Technology. Her main research interests include machine learning, privacy protection, and graph theory.

NENGFU XIE received the Ph.D. degree in computer software and theory from the Institute of Computer Technology, Chinese Academy of Sciences, in 2005. In 2005, he visited the City University of Hong Kong. He is currently with Knowledge Engineering Research, Department of the Institute of Agricultural Information, Chinese Academy of Agricultural Sciences. His main research interests include agricultural knowledge management, large-scale agricultural knowledge processing, and the Internet of Things.