Chapter 87
The Design of XML-Based Software Information System Schema and Development of the Standard for Information Processing

Chang-Su Kim, Min-Kyo In, Kang-Chan Lee, Seung-Yun Lee and Hoe-Kyung Jung

Abstract Recently, various advance information in industry sectors address the importance of software's which emphasized as a key capacity element of IT convergence in a diverse environments. Therefore, the necessity to enhance the availability of excellent software resources by making the database is also highlighted. Thus, this study aims to design the methods to establish a database for software resources effectively so that XML-based software information system schema can be designed and implemented, and the standard for software information system management can be developed.

Keywords Software information · XML · XML schema · XSLT · Data mapping

87.1 Introduction

As the society becomes more information-oriented, the number of computers distributed increases, and the importance of computer-based processing of documents is highlighted. As computer-related technology advances, electronic

C.-S. Kim · H.-K. Jung (✉)
PaiChai University, Doma2-Dong, SeoGu, DaeJeon, Korea
e-mail: hkjung@pcu.ac.kr

C.-S. Kim
e-mail: ddoja@pcu.ac.kr

M.-K. In · K.-C. Lee · S.-Y. Lee
Electronics and Telecommunications Research Institute, Daejeon, Korea
e-mail: mkin@etri.re.kr

K.-C. Lee
e-mail: chan@etri.re.kr

S.-Y. Lee
e-mail: syl@etri.re.kr

processing of documents becomes essential, and thus electronic document processing systems such as word processor and electronic publication system are becoming more common [1, 2].

A variety of convergence environments, software is highlighted as a key capacity element of IT convergence. Therefore, the necessity to enhance the availability of excellent software resources by making the database is also highlighted.

In addition, the efficiency of information services is of importance to manage or establish a quantity of software resources recently. Existing software materials, however, involve a lot of text and image based data, and thus it is difficult in reality to provide accurate data efficiently and to reuse existing data [3–5].

To effectively manage text-centered data handled through software resources, it is necessary to convert them into hypermedia documents, called XML document, which can support more various types of characteristics and structural information [1, 2, 6].

This study aims to design and demonstrate an effective method to make a database for software resources in order to design and present an XML-based software information system schema, and develop a standard for software information system management.

The future study is expected to suggest plans to use this in a national scale by maximizing the values of software assets, and developing standardized databases and operation instructions.

87.2 Related Researches

87.2.1 The Concept of XML Schema

XML schema is also known as DTD, a standard to define the structure of XML. XML schema is used to describe the XML elements and attributes. XML schema basically consists of statements of attributes and element types, which include content models on the XML elements and attributes within an XML document. XML schema plays a similar role to that of DTD, but it provides more advanced functions than those of DTD.

This study designs the software information system schema in accord with XML Schema 1.1 of W3C [7].

87.2.2 XSLT

XSLT is a document converting language to convert or reconstruct an XML document into another type of document [8]. The process that XSLT is converting an XML document into the desired type of output can be divided into two sections as below [8]:

- Structural conversion: The structure of the input XML document involves conversion of data into a structure that reflects the desired output format.
- Formatting: The structure of the input XML document becomes an output format such as HTML or PDF.

XSLT may specify the format of style documents or include the definitions of mapping rules among applied patterns. Recently, XSLT is used to generate certain XML documents from another type of XML documents of different structure. This system has been designed and implemented in reflection of such characteristics of XSLT.

87.2.3 The Concept of Data Mapping

Mapping connects data elements with the same or similar meaning in two different structures. In addition, it may add another meaning if necessary to define the relation between the two structures [1, 8].

Figure 87.1 shows the structure of two documents through data mapping. The information of the two documents' structures includes the same contents but represent different structures at the same time. The BirthDate structure includes a child elements such as Year, Month, and Day in relation of n:1. Here, the mapping is applied to the single element, which is BirthDate. The address elements are separated in the relation of 1:n and then go through the mapping. In this example, Zipcode in the relation of 1:1 goes through the mapping with Zipcode expressed in elements.

The system designed in this study has the ability to automatically generate XML documents in accord with the software information system XML schema, applies XSLT, and is used to convert XML documents that are valid for the schema.

Fig. 87.1 Data mapping
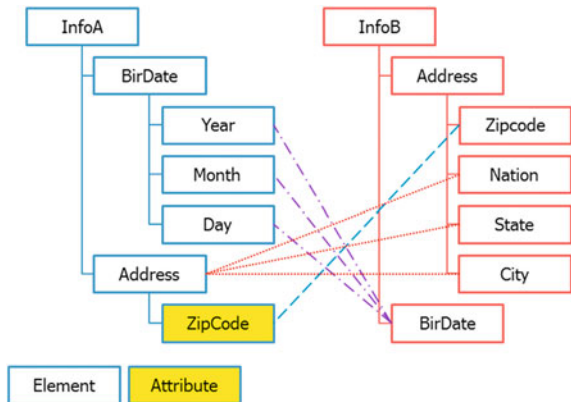
87.3 Design

In this section we illustrate and technically discussed the software information system and schema designs for software products. Figure 87.2 shows the highest class elements that reflect the general structure of software technical information. The software technical information consists of two basic structures.

Software technical information (SWIT) consists of the following two structures: first, the project_Information element that includes the general contents of software technology development; second, technology element that includes contents on software technology.

87.3.1 Definition of the Project_Information Element Structure

Figure 87.3 shows the structure of project_information element. It is consists of three child structures. Firstly, the project_basic, this element defines the development of software technology. Secondly, the project_management, this element defines the information on software quality management, image management, and education and development tools. Lastly, the marketing_plan, this element provides the general contents of the project plan.

87.3.2 Definition of Technology Element Structure

Technology element consists of tech_basic element, market element, software element, development element, test element, application element, tech_transfer element, and outcomes element. The cardinality of each is 1 and of sequence structure.

The tech_basic element includes the technical information including the name of software technology, necessity, usage, applications, and steps of technology. The market element includes market conditions and related technologies including similar technologies, home and abroad, current condition of the market, competitors, and so forth.

The software element includes development goals, achievements, targeted users, applicable criteria, restrictions, and so forth. The development element

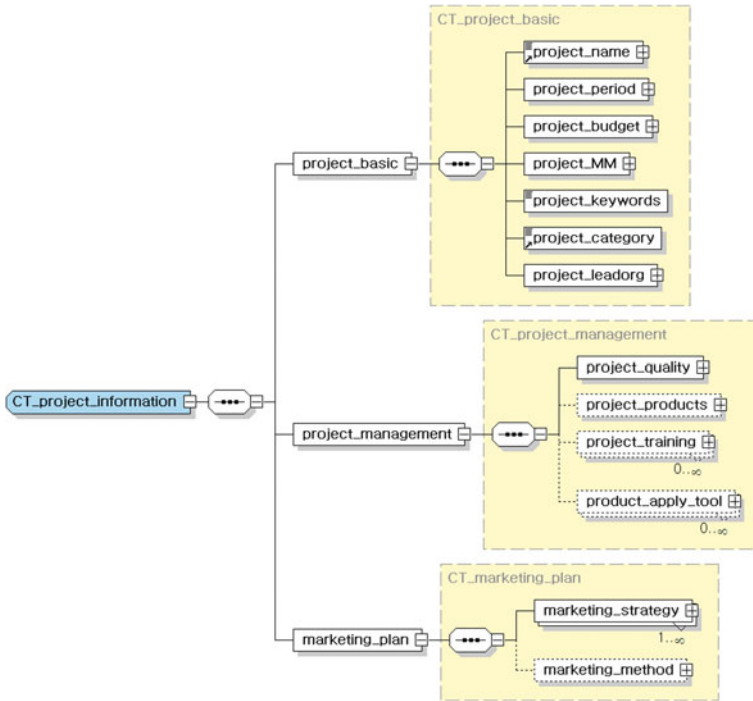**Fig. 87.2** Schema structure of software technical information
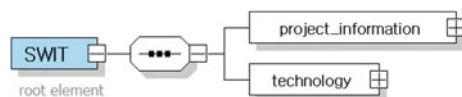
Fig. 87.3 Structure of project_information element

includes technology concept, function, major characteristics, languages, environments, tools, and so forth.

The test element includes test environment, test tool, and test result while the application element includes user interface, working environment, applications of the technology, and so forth. The tech_transfer element includes the manner of technology transfer, scope, costs, and so forth while the outcomes element includes the list of results in each step of development. Figure 87.4 shows the structure of technology element.

87.4 Implementation

In this section, we briefly illustrate and discuss the design and implementation method on how to automatically generate XML documents by using the designed software technology information schema. Through this, the standard for software information system management is presented as showed in Fig. 87.5.

Spring manages the web server resources while Jersey converts Java Object into XML or XML to Object. TOMCAT is a web server for HTTP communication. Ajax delivers data to a web server and generates XML files by means of XML
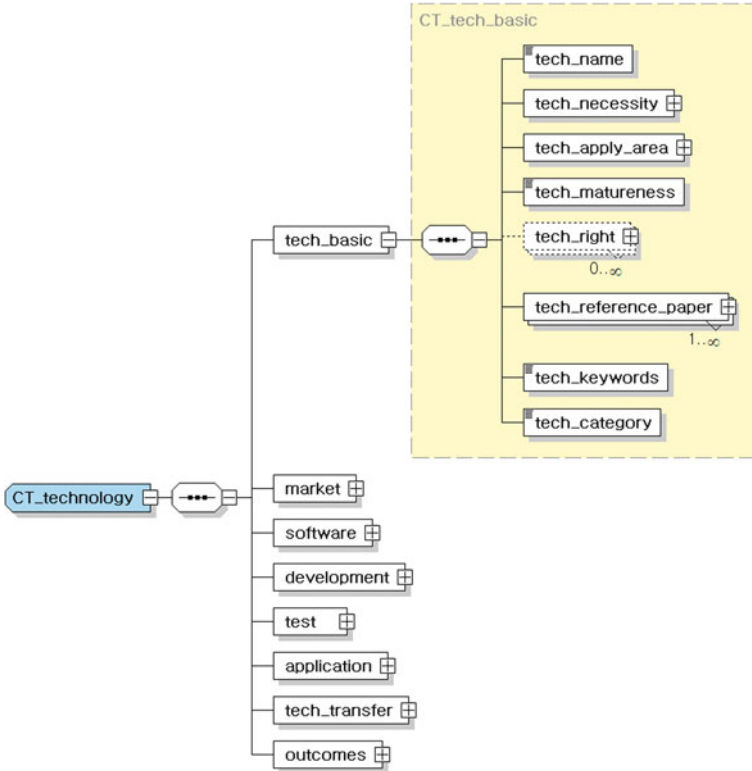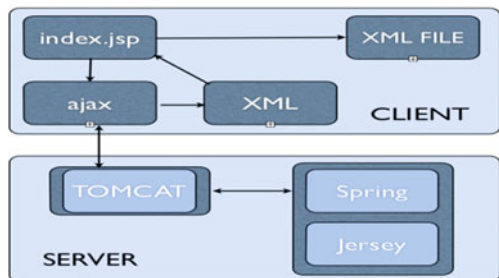
Fig. 87.4 Structure of technology element

structure. Figure 87.6 shows the input form to automatically generate XML documents based on the information of the software technology information schema.

Figure 87.7 shows the work flow of the XML document generation. The original document is prepared according to the schema, the input XML and XSL documents are processed, and then the converted documents are shown through the browser.

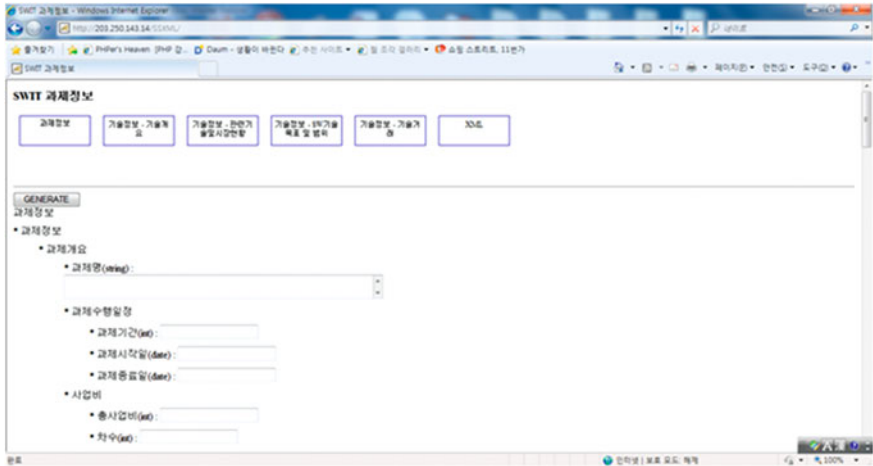**Fig. 87.5** Structure of the client–server

Fig. 87.6 Input form

Source document Input   Schema& document   XML document created   XSL apply
                             input form mapping

Fig. 87.7 Flow of the XML document generation

Figure 87.8 shows the XML document converted from the automatically gen-erated XML document in application of XSLT based on the software technology information schema.

87.5 Conclusion

Recently, various advance information in industry sectors address the importance of software's which emphasized as a key capacity element of IT convergence in a diverse environments. Therefore, the necessity to enhance the availability of excellent software resources by making the database is also highlighted.

This study presented the standard for the XML-based software product database schema, which is necessary for registration, searching, and sharing of software
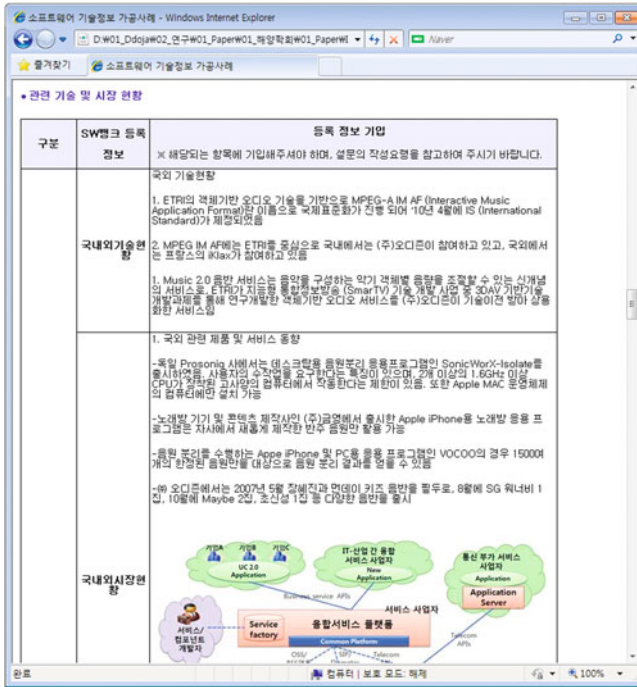
Fig. 87.8 Conversion of XML document

products so as to make a database of the software product assets and share and transact them through a software bank.

This study includes the definition of the standard for the XML-based SW product DB schema in order to provide the guideline for schema definition. Specifically, it defines the common information to register software products, product overviews, basic conditions for use, information of product excellence, multi-language support, sales/distribution/maintenance, product documentations, contact information, types and lengths, and so forth.

Lastly, the method to automatically generate XML documents by means of the designed software technology information schema is designed and implemented.

It is expected that this study presents a method to effectively make a database for software products, maximizes the values of software products, and contributes to developing a standardized database and operation instructions for it.

References

1. Kim CS, Jung HK (2012) XML-based EDI document processing system with binary format mapping rules. J Inf Commun Convergence Eng 10(3):258–263

2. Kim CS, You SK, KIm YW, Kim HJ, Jung HK (2012) Design of presentation language for sensor node data representation. J Korea Inst Inf Commun Eng 16(2):378–383
3. Liu Y, Wang F, Deng H, Fu X, Ji K (2012) A study on modeling of lightweight scientific workflow systems using XML schema. LNCS—Issue 7235
4. Chang Y-H, Huang C-C, Chien P-H (2012) Efficient grouping and ordering processing on XML data. J Chin Inst Eng 35(6):697–709
5. Vilas W, Kiyoshi A, Chutiporn A, Ekawit N (2003) A data model for XML databases. J Intell Inf Syst 20(1):63–80
6. Extensible Markup Language (XML) 1.0 (Fifth Edition). http://www.w3.org/TR/2008/REC-xml-20081126/
7. XML Schema. http://www.w3.org/XML/Schema
8. The Extensible Stylesheet Language Family (XSL). http://www.w3.org/Style/XSL/