7e Conférence Internationale de MOdélisation et SIMulation - MOSIM'08 - du 31 mars au 2 avril 2008 – Paris- France
« Modélisation, Optimisation et Simulation des Systèmes : Communication, Coopération et Coordination.».

ROHS COMPLIANCE BASED ON PLM

Chuanhong Zhou

Visiting Professor, Université de Technologie de Troyes, Laboratory of Mechanical Systems and Concurrent Engineering, BP2060, F.10010 Troyes Cedex, France; formerly, Associate Professor, CIMS & Robot Center of Shanghai University, Shanghai, China, 200072r

Benoît Eynard

Professor, Université de Technologie de Compiègne, Department of Mechanical Systems Engineering, BP 60319, F.60203 Compiègne Cedex, France

Lionel Roucoules

Associate Professor, Université de Technologie de Troyes, Laboratory of Mechanical Systems and Concurrent Engineering, BP2060, F.10010 Troyes Cedex, France

Nadège Troussier

Assistant Professor, Université de Technologie de Compiègne, Department of Mechanical Systems Engineering, BP 60319, F.60203 Compiègne Cedex, France

ABSTRACT: *At present, more companies are going to adapt themselves to RoHS (Restriction on Use of Hazardous Substances) compliance and enforcement by IT technology. This paper has presented a solution for RoHS Compliance Declaration and Integration between PLM and RoHS Compliance, which includes two IT frameworks: one is an integration framework that utilizes RoHS Compliance Declaration based on smart document and XML database, which uses SOAP technology to have the flexibility to complicated network environments and multiple operating systems, makes use of XML technology and XML database to acquire reuse and robustness; another is a general framework that integrates the BOM XML files from heterogeneous PLM systems to RoHS compliance system, which uses Smart client technology to have the flexibility to complicated distributed environments, makes use of XML technology and utilizes design patterns to acquire reuse and robustness.*

KEYWORDS: *RoHS Compliance, RoHS Compliance Declaration, Smart Client, XML, XML database, SOAP*

1. INTRODUCTION

Raising awareness and understanding of harmful effects of specific substances in these waste streams, it's necessary to control and reduce the amounts of waste generated by used and discarded products. So laws have been made to limit the hazardous substances contained in products, in which RoHS is very famous in electrical and electronic equipment (OFFICIAL JOURNAL OF THE EUROPEAN UNION. 2003). RoHS is intended to limit the use of substances such as lead and cadmium and is one of several rules that are forcing manufacturers to take more environmental responsibility. Similar codes in China and South Korea are also created.

For manufacturers, detailed data on the material composition of each component will become increasingly important. Additional changes from new legislation and from RoHS directive in other regions will force manufacturers to closely monitor the substances contained in each part. Nevertheless, many manufacturers are still far from complying with RoHS, with collecting and managing the data efficiently still remaining as the biggest hurdles. On the basis of investigation by famous consulting corporations, it is an important fact that collecting and managing component and materials data are proving to be the stumbling blocks for large manufacturers (Eric Karofsky. 2006).

An overall RoHS compliance Process (Nikhil Joshi et al. 2006) is summarized in Figure 1. RoHS compliance Process has centered on materials databases. Firstly, suppliers submit material information, which must be validated before storing in materials databases. Secondly, extracting the product BOM information from PLM system, it's necessary to build a new BOM-Compliance BOM through joining materials information from material databases. Finally, it is general to optimize the RoHS compliance process by Life Cycle Assessment (LCA) and other correlation technology. In this paper, we focus on dealing with a solution for RoHS Compliance Declaration and Integration between PLM and RoHS Compliance (Figure 1 dashed rectangle).

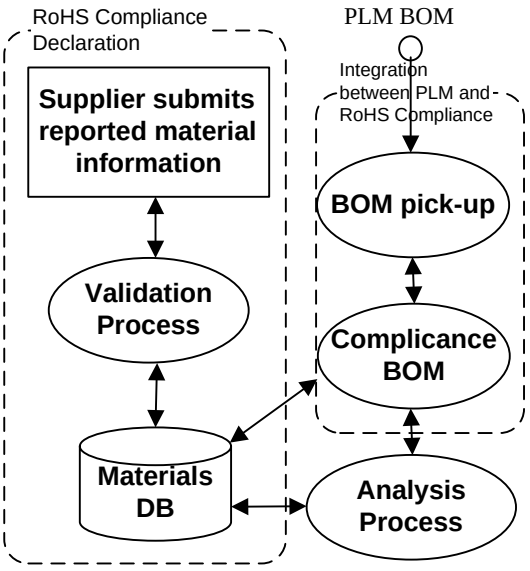

Figure 1 RoHS compliance Process

Recently, there are two main problems in RoHS Compliance:

- There is difficulty in finding and verifying data for thousands of parts from suppliers across the world.
- Reporting to various states, countries, and legislative bodies requires multiple languages and formats.
- There are so many PLM or else similar systems in manufacturers and their Original Equipment Manufacturer (OEM) that it is difficult for them to collect what they want for the analysis of RoHS compliance, because their BOM files have different formats, even if these BOM files belong to XML.

Information technology has evolved over the last few decades at an extremely rapid pace. In this paper, we present an intelligent, scalable framework for RoHS Compliance Declaration based on the development of synergistic technologies, such as Smart document, XML, SOAP and XML database (S. Dekeyser et al. 2003), which can deal with above problems. At the same time, we present an intelligent, scalable framework for RoHS compliance based on the development of synergistic technologies, such as Smart client, XML, XSLT and design patterns.

The remainder of the paper is organized as follows. First, we review some of the related work. This is followed by a brief background of several key impacts of RoHS Compliance in Section 2. In Section 3, we present a detail record of the integration framework, followed by Smart Document, SOAP, and XML database. In Section 4, we present a detail record of the integration framework, followed by Smart client, XSLT, and design patterns. In Section 5, we describe our prototype implementation and summarize our ongoing evaluation of the framework in the context of a case study. Finally, Concluding remarks along with future research directions are given in Section 6.

2. BACKGROUND

To the best of our knowledge, RoHS Compliance Declaration has not been previously studied in literature. However, some methods to integrate based on XML and the selection of hazardous substance and recyclable content specifications for components have been discussed

From RoHS compliance analysis aspect, Nikhil Joshi and Debasish Dutta have presented a new approach to account for regulatory requirements early in the design phase, with the aim of reducing downstream costs of compliance (Nikhil Joshi et al. 2006).

IPC 1752 (IPC. 2006) is the standard for the exchange of materials declaration data, which developed by a group of OEMs, EMS providers, component manufacturers, circuit board manufacturers, materials suppliers, information technology solution providers, and the National Institute of Standards and Technology. Though IPC 1752 is free and it is in an easy to use standard forms-PDF format, it is lack of total solution for RoHS Compliance Declaration.

XML, due to its structured, platform and language independent, highly extensible and Web-enabled nature, has rapidly become an emerging standard to represent data between diverse applications (Peak R. et al. 2004). By the way, IPC 1752 is based on XML schema to allow electronic data exchange across the web by xml file.

An XML database is a data persistence software system that allows data to be imported, accessed and exported in the XML format. XML databases serve in a complementary role to traditional databases especially as XML becomes prevalent.

3. ROHS COMPLIANCE DECLARATION

3.1. Framework

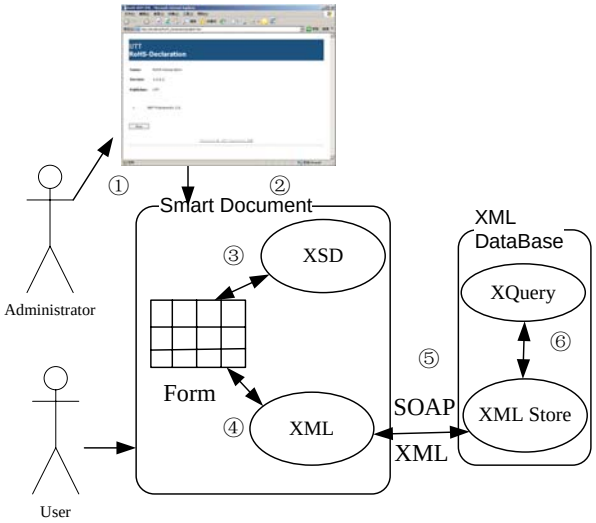

Figure 2 Information framework architecture

In this Section, we present a framework that realizes RoHS Compliance Declaration based on smart document and XML database. The information framework has a complicated structure illustrated in Figure 2 (where numbers indicate the sequence of steps). These steps are explained below:

1. Publish the RoHS Compliance Declaration Smart document. A smart document is smart client solution in Office 2003, which is a Microsoft Word 2003 or Excel 2003 document that has been programmatically customized to provide context sensitive information within the standard product interface. Smart Document is a new technology used in software development, which will be discuss in Section 3.2. First, the IT administrator will publish a software package for the RoHS Compliance Declaration Smart document in his enterprise's website.

2. As users, compliance engineers install the RoHS Compliance Declaration Smart document and start to run it.

3. With the new XML mapping features in Office 2003, an XML schema (XSD file) can be associated with RoHS Compliance Declaration Form workbook. We utilize XML mapping features to connect RoHS Compliance Declaration Form and XML file, which will be explained in Section 3.2.

4. Though RoHS Compliance Declaration Form workbook is mapped to a XSD file, it is necessary to build the Form contest to a XML file , which will be explained in detail in Section 3.2.

5. Using SOAP protocol, RoHS Compliance Declaration XML will be transfer to XML database, which will be explained in Section 3.3.

6. Finally, XQuery is a query language that is designed to query collections of XML data. XQuery will be applied to pull data from xml databases for the RoHS compliance analysis, which will be explained in detail in Section 3.3.

3.2. Smart Document

3.2.1 Smart Client and Document

Smart Client (David Hill et al. 2004) is a new technology used in software development, generally referring to applications which:

- are delivered over the web
- do not require installation (or provide automated installation and updates)
- automatically update without user action
- have the look and feel of desktop applications

The term "Smart Client" can be designed to combine the benefits of a rich client application (high performance, high productivity) and a thin client application (zero-install, auto-update), although the precise nature of the balance between the two approaches depends on the exact scenario. Smart Client applications bridge the gap between web applications and desktop applications. They provide the benefits of a web application (such as leveraging the internet and offering remote access to data) while still providing the snappy look and feel inherent to desktop applications.

A smart document is smart client solution in Office 2003, which has all advantages of Smart Client. On the other hand, a smart document provides data retrieval and submission in an already familiar and robust environment. Smart document development relies heavily on the use of XML and some programmatic customization. In effect, an XML schema is attached to a document. That same schema is used to develop the custom user interface for the smart document. An XML expansion pack component provided by Microsoft is used to connect the two (Frank Rice. 2005). It is important to note that all of the base functionality associated with Word or Excel is available in a smart document solution. The task pane is used simply guide the end user through the population of specific sections of that template. This facilitation can include interaction with various back-end data sources as a means of providing information lookup and validation.

Smart Document application for RoHS Compliance Declaration can has different versions for different users and different environment by means of changing XML schema and redesigning declaration reports, for example: "A" version for RoHS Compliance Declaration for China RoHS, "B" version for RoHS Compliance Declaration for South Korea RoHS; It is convenient to connect various data sources and services, for example: local file, database and Web service.

Smart Document, in addition to making document and workbook creation easier, are flexible in terms of technology as well. Smart document code can be written in Microsoft Visual Basic 6.0, Microsoft Visual Basic .NET, Microsoft C# .NET, or Microsoft Visual C++. Smart documents can be deployed over a corporate intranet, over the Internet, using XML Web services, or through Web sites based on SharePointTM Team Services from Microsoft. In this paper, we choose Microsoft C# .NET and Office 2003 to develop this Smart Document application for RoHS Compliance Declaration.

3.2.2 XML Mapping

Microsoft Office Excel 2003 improves support for using native XML by adding a XML Schema Definition (XSD) to a workbook. This creates an XML map to make data more meaningful upon importing or exporting data. Thus, import data can be conforming to a custom XML vocabulary or schema, and then write the data back out by using the same XML schema.

This feature is targeted at Office solution developers, but expert users can also put it to good use. This feature enables users to:

- Create a map between an Excel spreadsheet and an XML structure
- Use that map to import subsequent XML data to that same structure
- Use that map to export XML data from the spreadsheet, which conforms to that XML schema
- Persist and reuse that map for subsequent import and export operations

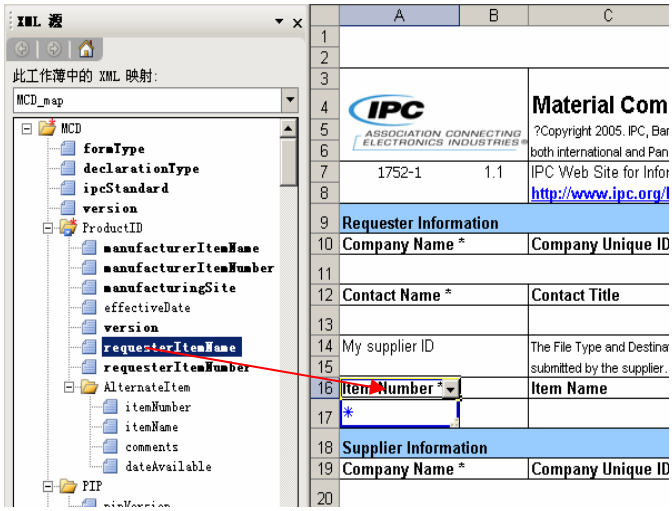

Figure 3 The XML Source task pane and mapping

The process starts by adding an XSD file to an Excel workbook. Once the XSD file is added, Excel creates an XML map in the XML Source task pane (see Figure 3) that users use to map to specific ranges or to a single cell. Excel also uses this map to manage the relationship between those mapped ranges and the elements in the XML schema. A workbook can contain many XML maps where each map is an independent entity. Likewise, users can also have multiple maps refer to the same schema. When XML data is imported or exported, Excel uses the map to relate the contents of a mapped range to elements in the schema. The result of mapping a repeating element to a range of cells is to create a list in Excel (see Figure 3 red arrowhead). Automatically created XML-related lists appear as a single column with a heading at the top (based on the element or attribute name) and a cell for adding data at the bottom of the list. As users add data to the final cell, the list is extended automatically to the next row. On the other hand, XML Schema lets users declare an element in an XML document for RoHS Compliance Declaration to be of a particular type, allowing the parser to validate the content of a document as well as its structure.

The XML Source task pane is a visual tool that enables users to set up spreadsheets quickly that know how to consume and produce XML data in user-specified schemas. The XML Source task pane consists of a tree view that enables users to pick the XML elements they want and drag them into the cell grid in logical groupings and into separate tables (John R et al. 2003).

3.3. XML Database

In Software engineering, an XML database is a data persistence software system that allows data to be imported, accessed and exported in the XML format.

Two major classes of XML database exist (Shalaka Natu et al. 2003):
1) XML-enabled database: In an XML-enabled database, the documents are stored in constituent fragments. Here the XML data is stored in object relational form and one can use an XML SQL Utility (XSU) or SQL functions and packages to generate the whole document from its object relational instances. There are some representative products, for example: SQL server 2000 (Microsoft), Oracle (Oracle), DB2 (IBM) and so on.

2) Native XML database: In a native XML database approach, XML is not fragmented but rather is stored as a whole in the native XML database. This means that documents are stored, indexed, and retrieved in their original format, with all their content, tags, attributes, entity references, and ordering preserved. There are some representative products, for example: eXist (Wolfgang Meier), Timber (University of Michigan), Berkeley DB XML (Oracle) and so on.

In this paper, we choose SQL Server 2005 to store and manage the RoHS Compliance Declaration XML using new XML features in SQL Server 2005: HTTP SOAP and XQuery.

3.3.1 HTTP SOAP

HTTP SOAP is a protocol for exchanging XML-based messages over computer networks, normally using HTTP/HTTPS. SOAP forms the foundation layer of the Web services stack, providing a basic messaging framework that more abstract layer can build on (Nayef Abu-Ghazaleh et al. 2005).

Most people use SOAP because it supports interoperability among many different environments and it supports HTTP, which has led to SOAP becoming an industry standard. This is SOAP's secret weapon and one of the sources of its power. It's almost unheard of to find someone blocking Port 80 with a corporate firewall, so SOAP (as bound to HTTP) should pass through corporate firewalls untouched. The other source of SOAP's power is the fact that the information transported by the HTTP protocol is actually XML. To be more specific, the content-type of the HTTP packet is text/xml.

In order for SQL Server 2005 to be able to listen for SOAP requests, we must set up SQL Server as a Web Service, which means creating and defining HTTP endpoints and their related properties and methods that the endpoint exposes. In this paper, The RoHS Compliance Declaration example creates an endpoint called RoHS_endpoint, with there methods: EditXML, GetXML and InsertXML. These are the methods for which a client can send SOAP requests to the endpoint. Transact-SQL for RoHS_endpoint:

```
DROP ENDPOINT RoHS_endpoint
GO
CREATE ENDPOINT RoHS_endpoint
STATE = STARTED
AS HTTP(
  PATH = '/Compliance/RoHS',
  AUTHENTICATION = (INTEGRATED),
  PORTS = ( CLEAR ),
  CLEAR_PORT =8082
  )
FOR SOAP (
  WEBMETHOD 'EditXML'
        (name='RoHS.dbo.EditXML',
         SCHEMA=STANDARD ),
  WEBMETHOD 'GetXML'
        (name='RoHS.dbo.GetXML',
         SCHEMA=STANDARD ),
 WEBMETHOD 'InsertXML'
        (name='RoHS.dbo.InsertXML',
         SCHEMA=STANDARD ),
  WSDL = DEFAULT,
  DATABASE = 'RoHS',
  NAMESPACE = 'http://zch/Compliance/RoHS/'
  )
GO
```

In Section 3.2, Smart Document Client application for RoHS Compliance Declaration will have the following lines of code (C#) that initiate a new instance of a SQL Server 2005 endpoint (RoHSweb.RoHS_endpoint) as a Web proxy that has a Web method (GetXML) that returns an xml data type row instance value.

```
RoHS_Declaration.RoHSweb.xml outParam=new RoHS
_Declaration.RoHSweb.xml();
 RoHSweb.RoHS_endpoint proxy = new RoHSweb.RoHS
_endpoint();
 proxy.Credentials = System.Net.CredentialCache.Defau
ltCredentials;
object[] results;
results = proxy.GetXML(Convert.ToInt32(textBox1.Text)
, ref outParam);
```

3.3.2 XQuery support

XQuery is a query language (with some programming language features) that is designed to query collections of XML data . It is semantically similar to SQL. XQuery provides the means to extract and manipulate data from XML documents or any data source that can be viewed as XML, such as relational databases or office documents. XQuery uses XPath expression syntax to address specific parts of an XML document. It supplements this with a SQL-like "FLWOR expression" for performing joins. A FLWOR expression is constructed from the five clauses after which it is named: FOR, LET, WHERE, ORDER BY, RETURN.

For SQL Server 2005, Microsoft has added server-side support for XQuery. Coupled with the xml data type, this allows for quick and efficient storage and retrieval of XML data. Server-side support for XQuery means that users get all the added benefits of the XPath language plus additional support for things like better iteration, sorting of results, and the ability to shape the results of users' queried XML (typically called construction). The XQuery data model is what drives the XQuery language, which means, just like the xml data type, users can have typed or untyped results as well as XML fragments (S. Pal et al. 2005).

In this paper, we want to share RoHS Compliance Declaration information in SQL Server 2005 with else IT systems. So we use XQuery to realize this function. This realization is easy. For example: if we want to get the value of Company Name of Requester Information from IPC 1752-1 XML, the entire XQuery expression looks like the following:

```
select @IPC1752.value('(//Requester /@ name)[1]',' nva
rchar (10)');
```

4. INTEGRATION BETWEEN PLM AND ROHS COMPLIANCE

4.1. Integration Framework

In this Section, we present a framework that integrates the BOM XML files from heterogeneous PLM systems to RoHS compliance system. Generally, there are one or more PLM systems in companies, which control all

information through three design stages, namely: conceptual design, embodiment design, and detail design (Pahl, G et al. 1996). We assume that each supplier to report materials content for their supplied parts, and a materials database has already been built. The information integration framework has a complicated structure illustrated in Figure 4 (where numbers indicate the sequence of steps). These steps are explained below:

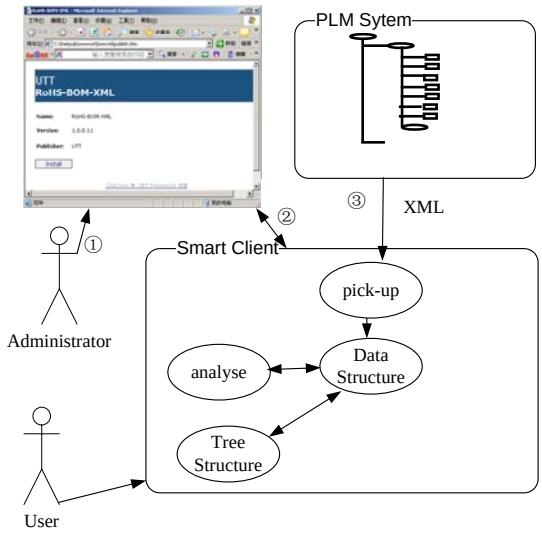

Figure 4 Information integration framework architecture

1. Publish the RoHS compliance Smart Client. Smart Client is a new technology used in software development, which will be discuss in Section 3.2.1

2. As users, compliance engineers install the RoHS compliance Smart Client and start to run it.

3. The function "pick-up" is drawing BOM xml file from PLM system. As there are heterogeneous PLM systems, there are various BOM xml file formats. We utilize XPath and design pattern technology to settle this problem, which will be explained in Section 4.2.

4.2. Extract information from BOM xml file

4.3.1 Understand BOM XML file

Considering BOM XML file is complicated generally, it is important to find key factors to help understand. Understanding BOM tree process is shown in Figure 5, it gives an example for the BOM XML file from Smarteam which belongs to Dassault Systemes.

Firstly, looking for the relationship between parent components and child components are important to understand the BOM XML file. As shown in Figure 5(1), parent component "CARPRD-0045" has several child components, which include child component "CARPRD-0046". Secondly, there is a very interesting XML Element "parentLink" (2), which includes two XML Element "parentPartMaster" and "partMaster" with XML Attribute "href". Finally, Using XML Attribute "href" corresponding "id", it is easy to find parent component "CARPRD-0045" and child component "CARPRD-0046" as shown in Figure 5 (3, 4). Other information linking to components can be acquired from XML Element "partMaster".

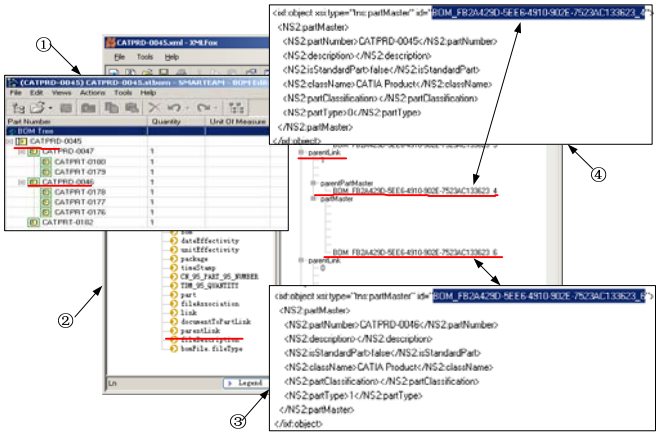

Figure 5 Understanding the BOM XML file from Smarteam

In the same way, the BOM XML file from Teamcenter Engineering (UGS PLM Software) has an important XML Element "Occurrence", which includes two XML Element "id" and "parentRef". It is easy to know that "Occurrence" is similar with the above-mentioned "parentLink".

4.3.2 Template Method design patterns

In fact, there are a lot of formats for BOM XML from various PLM vendors, and there are various RoHS directives from different countries. So, we have to change program code to fit in with the needs of RoHS compliance. How to do few programming and get high reliability? It is a good idea to apply design patterns to software architecture design.

In regard of defining design patterns, some useful definitions of design patterns have emerged as the literature in this field has expanded (Michael Kay. 2005).

Design patterns are recurring solutions to design problems you see over and over. (The Smalltalk Companion)

Design patterns constitute a set of rules describing how to accomplish certain tasks in the realm of software development.

Design patterns focus more on reuse of recurring architectural design themes, while frameworks focus on detailed design and implementation.

A pattern addresses a recurring design problem that arises in specific design situations and presents a solution to it.

Patterns identify and specify abstractions that are above the level of single classes and instances or of components.

Owing to the fact that the function "pick-up", "analyse" and "Tree Structure" will be changed with need of different application demand, but their processing sequence is fixed, we choose Template Method design pattern to overcome this problem.

Template Method is defining the skeleton of an algorithm in an operation, deferring some steps to subclasses in order to let subclasses redefine certain steps of an algorithm without changing the algorithm's structure (Data & Object Factory. 2000).

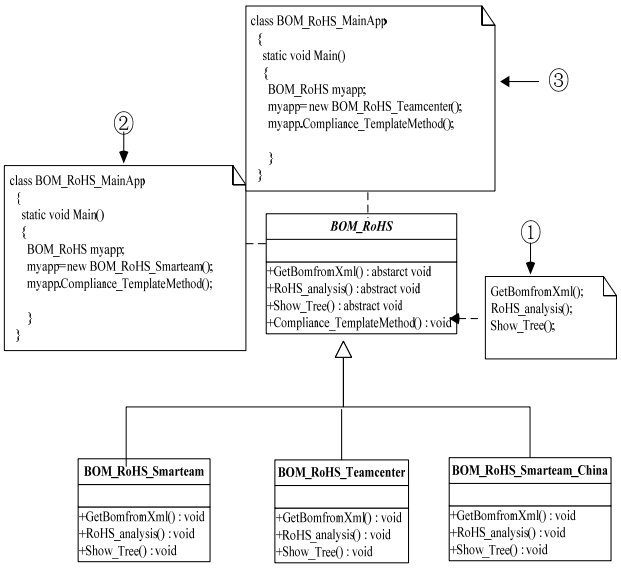

Figure 6 Application of Template Method design pattern

Abstract Class "BOM_RoHS" is created that provides the basic steps of RoHS compliance as shown in Figure 6. These steps (1) are implemented using abstract methods, which are "GetBOMfromXml()", "RoHS_analysis()" and "Show_Tree()" corresponding function "pick-up", "analyse" and "Tree Structure". Later on, subclasses will change the abstract methods to implement real actions corresponding concrete application demand, for instance, subclass "BOM_RoHS_Smarteam" makes these three functions to integrate Smarteam system, and client application realizes as shown (2). Thus RoHS compliance algorithm is saved in one place "Compliance_TemplateMethod()", but the concrete steps may be changed by the subclasses in order to adapt the program to the new condition demand. For example, if we want to integrate Teamcenter system, a new subclass "BOM_RoHS_Teamcenter" will be created at the same time "GetBOMfromXml()", "RoHS_analysis()" and "Show_Tree()" will be updated to be available to the BOM XML file from Teamcenter system, finally, client application realization (3) will just change one line "myapp= new BOM_RoHS_Teamcenter()". On a similar plan, we can create a new subclass "BOM_RoHS_Smarteam_China" to meet the RoHS China compliance demands.

5. PROTOTYPE IMPLEMENTATION

A prototype of the solution for RoHS Compliance Declaration and Integration between PLM and RoHS Compliance has been implemented. The prototype runs on a PC under MS Windows and consists of:
1. A tool for RoHS Compliance Declaration, which is based on Smart Document (see Figure 7).
2. A tool for Integration between PLM and RoHS Compliance, which is based on Smart Client (see Figure 8).

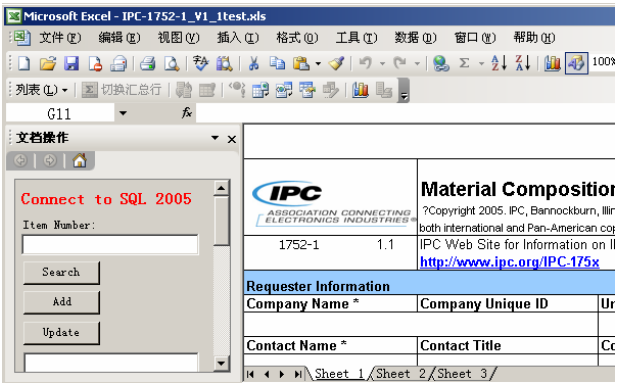

Figure 7 Form for RoHS Compliance Declaration

There are three function-"Search", "Add" and "Update" in this RoHS compliance Smart Document demo as shown in Figure 7. According to "Item Number", "Search" function will get XML content from SQL Server 2005 and automatically fill in IPC 1752 Form in Excel 2003 sheet1. On the analogy of this, "Add" and "Update" function will add a new XML record and update an old XML record. An example XML shows in Figure5.

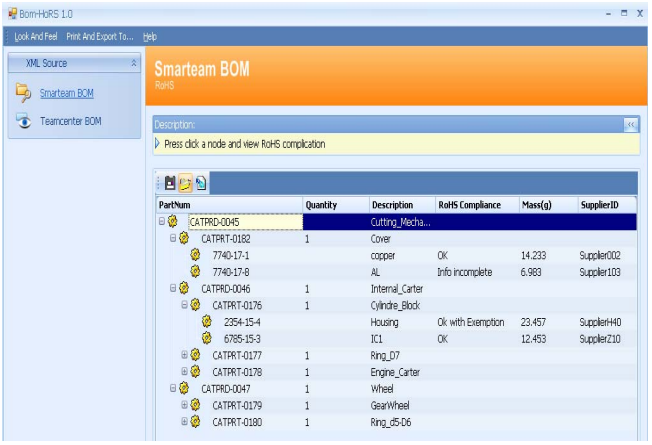

Figure 8 Form for Integration between PLM and RoHS Compliance

As shown in Figure 8, through connecting with materials information and BOM XML files from Smarteam system

to build a Compliance BOM, the RoHS compliance Smart Client show the result.

As shown in Figure 1, the two basic functions of RoHS Compliance have been realized RoHS Compliance Declaration (Figure 7) and Integration between PLM and RoHS Compliance (Figure 8). Although these function prototype are simple, it's a good beginning to research deeply in RoHS compliance.

6. CONCLUSIONS AND FUTURE WORK

Over the past several years, manufacturers have faced an increasing array of new environmental mandates from jurisdictions such as the EU, China, Japan, and some U.S. states. The RoHS directive is famous in these environmental mandates. Many companies are going to adapt themselves to RoHS compliance by leveraging IT technology. This paper has presented an IT framework that utilizes RoHS Compliance Declaration and Integration between PLM and RoHS Compliance.

After the discussion and studying above case, the main conclusions are drawn as follows:

- Smart Document and Smart client make this integration framework with both benefits of a rich client application, a thin client application and Microsoft Office;
- Making use of XML technology XML Schema validate the content of a RoHS Compliance Declaration XML document as well as its structure, XML database contributing to store the RoHS Compliance Declaration XML and extract information using XQuery technology for sharing information with else system. Microsoft XML Parser contributing to extract key data from complicated BOM XML file, XSL file contributing to transform the RoHS compliance result to else XML file for sharing information with else system;
- Flexible application of SOAP and XML makes this integration framework obtain the property of reuse and robustness, which is available for multiple operating systems and complicated network environment;
- Flexible application of design patterns and data structure makes this integration framework obtain the property of reuse and robustness.

Future research works may focus on studying in a deepgoing way of RoHS compliance. It will optimize the RoHS compliance process by Life Cycle Assessment (LCA) and other correlation technology.

REFERENCES

ASSOCIATION CONNECTING ELECTRONICS INDUSTRIES (IPC), 2005. IPC 1752 for Materials Declaration. http://members.ipc.org/committee/drafts/2-18_d_MaterialsDeclarationRequest.asp

Data & Object Factory, 2000. "Template Design Pattern," http://www.dofactory.com/Patterns/PatternTemplate.aspx

David Hill, et all., 2004. "Smart Client Architecture and Design Guide," http://msdn2.microsoft.com/en-us/library/ms998468.aspx

Eric Karofsky, December 19, 2006. "RoHS—The Data Collection Problem" http://www.amrresearch.com/Content/View.asp?pmillid=19996.

Frank Rice, Microsoft Corporation. February 2005. Creating XML Mappings in Excel 2003. http://msdn2.microsoft.com/en-us/library/aa203737(office.11).aspx.

Peak R., Lubell J., Srinivasan V., Waterbury S., 2004. STEP, XML, and UML: Complementary Technologies. Journal of Computing and Information Science in Engineering, 4(4), pp. 379-390.

Jim Brown, April, 2006. Environmental Compliance in Electronics: Creating a Successful Strategy. Aberdeen Group, Inc.

John R. Durant, Microsoft Corporation. June 2003. Importing XML Maps, XML Lists, and Dynamic Chart Sources in Excel 2003. http://msdn2.microsoft.com/en-us/library/aa203727(office.11).aspx.

Michael Kay, 2005. "What kind of language is XSLT?" http://www.ibm.com/developerworks/xml/library/x-xslt/?article=xr

Nayef Abu-Ghazaleh, Michael J. Lewis. Differential Deserialization for Optimized SOAP Performance. Proceedings of the 2005 ACM/IEEE conference on Supercomputing SC '05 Publisher: IEEE Computer Society, November 2005.

Nikhil Joshi and Debasish Dutta. 2006. "Towards Regulatory Compliance through PLM." In Proceedings of the IDETC/CIE 2006 ASME 2006 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, Philadelphia, PA, Sep 10 - 13 2006.

OFFICIAL JOURNAL OF THE EUROPEAN UNION, 2003. Directive 2002/95/EC of the European Parliament and of the Council of 27 January 2003 on the restriction of the use of certain hazardous substances in electrical and electronic equipment, February.

Pahl, G., and Beitz, W., 1996. Engineering Design - A systematic approach, 2nd ed. Springer-Verlag, London, UK.

S. Dekeyser, J. Hidders, 2003, A commit scheduler for XML databases, Proceedings of the fifth Asian-Pacific Web Conference, pp. 83–88.

Shalaka Natu and John Mendonca, "Digital Asset Management Using A Native XML Database Implementation", CITC4'03, October 16–18, 2003, Lafayette, Indiana, USA.

S. Pal, M. Fussell, I. Dolobowsky. XML support in Microsoft SQL Server 2005. MSDN Online, http://msdn.microsoft.com/xml/default.aspx?pull=/library/en-us/dnsql90/html/sql2k5xml.asp.