Best practice on automatic document generation with process asset library for Small and Midsize Enterprises (SMEs)

So Young Moon, Byungkook Jeon* and R. Young Chul Kim

SE Lab., Dept. of Software and Communications Eng.), Hongik University, Sejong Campus, 30016, Korea
*Dept. of Software, Gangneung-Wonju National University, Wonju City, Gangwon Prov. 26403 Korea

Abstract: Unlike large corporations, our small and midsize enterprises (SMEs) have a lot of trouble in consistently accumulating their technologies and expertise. This causes them to make a poor at software competitiveness against large software corporations. Although they need to improve the quality of software with software development process, they can’t afford to invest more human resources and time. Therefore, is there any option available for them to enhance their software quality for themselves? In this paper, we propose a method of enabling the continuous maintenance of projects based on software process through developing the visualization toolchain system. We make a solution how to automatically generating documents from in-house process assets of project-related byproducts such as experience, knowledge and documentation. For automatically generating documents, we use XML and XSLT on the toolchain system. In addition, we suggest to build some document assets using the proposed method to spend less human resources.

Keywords: Process Asset Library (PAL), Automatic Document Generation, Small and Midsize Enterprises (SMEs)

1 Introduction

Since 2011, the software industry is surpassing the hardware industry. In addition, there is sustaining the continued expansive evolution of the software industry through the convergence of the software and other industries. However, in Korea, the software business infrastructure serves as a factor that prevents many software companies from accumulating their technologies and expertise. As of 2011, there are a total of 6,785 software companies in Korea. 82.6% of them are small and midsize enterprises (SMEs) which aggregate sales amounted to no more than KRW 5 billion [1]. Korean venture companies or SMEs have difficulty in software competitiveness due to poor environment. Nevertheless, they need to invest more human resources and time for improving the qualities of software and software process. In other words, they are focusing on managing their processes to improve the quality of their final products. Our government also pays increasing attention to the quality of software and its process constructed by domestic companies. In order for them to manage their software quality, they need to share hard-earned knowledge and experience involving how to resolve problems in the implementation and documentation of projects [2]. Most organizations have trouble in making output documents with constructing frameworks because of their tight development schedules [3]. Moreover, when developers are asked to prepare various documents including requirement, design, and code documents, they slightly modify the existing documents or submit the
modified ones which result in discrepancies between final software codes and their documentation [4]. Such discrepancies cause many difficulties in software maintenance.

This paper has concerned with the continuity of project maintenance and management on software process. We focus on constructing Process Asset Library (PAL) defined by Capability Maturity Model Integration (CMMI) constructed. Yet, in this paper, we attempt to build in-house process assets using open source and Extensible Stylesheet Language Transformations (XSLT) which is Extensible Markup Language (XML), for venture companies or SMEs. Our proposed method enables them to standardize business processes and documents. This paper is composed of the following sections. Section 2 describes related work. Section 3 describes core activities and PAL required to enhance the performance of software development of projects on Software Process. Section 4 shows the examples applied to Requirement Specification and Project Plan. Section 5 describes the conclusion and future studies.

2. Related Work

Software Process (SP) in the NIPA is included with core activities [5] required to strengthen their capability to perform software development projects for SMEs in Korea. The core activities are considered from a project and an organizational perspectives, which effectively improves capacity levels of processes. Figure 1 shows 5 evaluation areas, 17 evaluation items, and 70 detailed evaluation items. The 5 evaluation areas are project management, development, support, organization management, and process improvement area. For each area, the 17 evaluation items are described as follows. The project management area consists of project planning, project control, and partner management. The development area is composed of customer requirement management, analysis, design, implementation and testing. The support area is made up of quality assurance, configuration management, measurement and analysis. Lastly, process improvement area consists of quantitative process management, problem solving, and process improvement management. The 70 detailed evaluation items consist of activities listed as subcategories of the 17 evaluation items. Figure 1 shows the detailed core activities of software process for the project.
Figure 1. Software Process focused Project Core Activity

The PAL [6, 7, 8] is a well-organized, well-indexed and searchable repository of process assets that is easily accessible by anyone who needs process guidance information such as examples, data, templates, or other process support materials. The purpose of PAL is as follows.

- Provides a centralized knowledge base to collect, define and disseminate guidelines for processes related to the organization's work (generally product development, management and improvement).
- Reduces unnecessary duplication of process assets, and regenerates assets in your organization.
- Provides a knowledge sharing mechanism for your organization's process assets and usage
- Supports an effective learning environment for new employees who are expected to use an organization's processes
- Provides the basis for making decisions about developing and coordinating an organization's processes.
- Improves application of content consistency and process guidance throughout the organization.

The PAL also has an advantage to provide guidance to help process developers (project managers, SQA, etc.) in the organization with project cases implemented in the past. It allows process developers to use project estimates, project plans, risk databases, lessons learned, and defect data that are similar to those of previous projects. It can be helpful in performing new projects based on projects that were implemented in the past [9].
3. A Management Method of PAL

To construct the PAL, we have added a process for PAL to the tool-chain system [10, 11, 12, 13] published in the previous study. In this paper, we propose a method of managing all items generated in the development process in Redmine [14], which perform core activities based on Software Process such as requirement, issue, project, and development management. The PAL is a well-organized, well-indexed and searchable process asset repository that has guidance information such as examples, data, templates and other process support materials, which is easily accessible by anyone who needs information in process. Therefore, to construct the PAL, the user queries with MySQL to the user’s stored data in Redmine, and extracts some particular data, which then make data based on XML document formats. That is, the selected data is written in a XML file. When the XML data is entered into the PAL processing program, any documents are automatically created. Figure 2 shows the overall process for PAL. The left side of Figure 2 shows the PAL process with the existing tool-chain. The right side of Figure 2 shows how to automatically create a document as follows: 1) Enters tasks occurring during core activities based on Software Process into Redmine, 2) Converts the extracted data to XML data, 3) Selects a suitable XSLT template stored in the PAL upon the analysis of the XML data, and, 4) then make a document with generating HTML code. For example, the Project Management Plan includes:

- Project concepts including project names, periods, aims, expected effects, etc.
- Project scopes
- Environment of system construction including hardware configuration design and software configuration.
• Project promotion system including project organization, role of organization, and manpower planning.
• Management process plan including work plan, control plan and review of artifact plan.
• Project management methods including project report plans, management of issues and changes, education and technology transfer.

4. Our Automatic Document Generation

This section explains how Requirement Traceability Matrix and Project Management Plan are automatically generated by PAL. Both are generated as follows: 1) Well-defined templates are written in XSLT, 2) XML data is entered into the XSLT, and 3) The data is automatically converted to HTML code through the document templates.

4.1 XSLT & XML Design for Requirement Traceability Matrix

In Figure. 3, the XSLT code indicates the template in the Requirement Traceability Matrix. The requirement traceability is distinguished by static and dynamic mechanism [15]. Requirement management is the most important in software process. The requirement traceability is very important issue in complex and large projects, which provides the foundation for requirements change, software defect fixes, and quality assurance of software processes [16]. For example, it includes a project_name, project_period, document_name, version, change_history, and functional requirement. In the functional requirement, there are a requirement_ID, requirement_name, detailed_requirement, priority_of_requirements, and description. Figure. 4 shows the XML code for requirement traceability matrix. The Id_tag has RTM-0001, and the Requirement_id tag has REQ-F-001. The Use-case_id tag has UC-001. It is possible to trace all the elements (use case, test case, class, and so on) of a requirement. Using both XSLT code and XML code, we create a requirement traceability matrix document as shown in Figure. 5. REQ-F-002 in Requirement_id item is to trace index like a query management to detect the relationship. If REQ-F-002 is clicked, it will show to link the detailed information of a requirement such as UC-002(Use-case_ID), IF-003(Interface_ID), CLASS-003(Class_ID), CLASS-003-M-01(Method_ID), TS-002(TestScenario_ID), and TS-002-TC-001(TestCase_ID).
Figure 3. XSLT code for Requirement Traceability Matrix
Figure 4. XML code for Requirement Traceability Matrix

```xml
<?xml version="1.0" encoding="utf-8"?>
<xml-stylesheet type="text/xsl" href="req_matrix_eng.xsl"/>

<task>
  <id>RTM-0001</id>
  <requirement_id>REQ-F-001</requirement_id>
  <requirement_name>Set run options</requirement_name>
  <requirement_link>http://203.248.87.143:1550/redmine/issues/9</requirement_link>
  <usecase_id>UC-001</usecase_id>
  <interface_id>IF-001</interface_id>
  <class_id>CLASS-001</class_id>
  <method_id>CLASS-001-M-01</method_id>
  <test_scenario_id>TS-001</test_scenario_id>
  <test_case_id>TC-001</test_case_id>
  <desc>The tool should be able to run by selecting one of the three options below. #13:
  1. ALL: Perform all steps from source analysis to visualization.#13;
  2. OPT: Get from existing DB files without parsing or parsing. #13;
  3. DUMP: Extract the result of dumping the SNDB file to a file</desc>
</task>

<task>
  <id>RTM-0002</id>
  <requirement_id>REQ-F-002</requirement_id>
  <requirement_name>Query management to detect coupling</requirement_name>
  <requirement_link>http://203.248.87.143:1550/redmine/issues/9</requirement_link>
  <usecase_id>UC-002</usecase_id>
  <interface_id>IF-003</interface_id>
  <class_id>CLASS-002</class_id>
  <method_id>CLASS-002-M-01</method_id>
  <test_scenario_id>TS-002</test_scenario_id>
  <desc>This is to facilitate future query expansion.</desc>
</task>

<task>
  <id>RTM-0003</id>
  <requirement_id>REQ-F-003</requirement_id>
  <requirement_name>Coupling level detection query management</requirement_name>
  <requirement_link>http://203.248.87.143:1550/redmine/issues/9</requirement_link>
  <usecase_id>UC-003</usecase_id>
  <interface_id>IF-003</interface_id>
  <class_id>CLASS-003</class_id>
  <method_id>CLASS-003-M-01</method_id>
  <test_scenario_id>TS-003</test_scenario_id>
  <test_case_id>TC-001</test_case_id>
  <desc>This is to facilitate future query expansion.</desc>
</task>

<task>
  <id>RTM-0004</id>
  <requirement_id>REQ-F-004</requirement_id>
  <requirement_name>Independence of the tools in the code</requirement_name>
  <requirement_link>http://203.248.87.143:1550/redmine/issues/9</requirement_link>
  <usecase_id>UC-004</usecase_id>
  <interface_id>IF-004</interface_id>
  <class_id>CLASS-004</class_id>
  <method_id>CLASS-004-M-01</method_id>
  <test_scenario_id>TS-004</test_scenario_id>
  <test_case_id>TC-001</test_case_id>
  <desc>Each open source tool has the potential to be replaced with additional tools in the future.
  Therefore, you must write code to comply with the OCP at the implementation stage.</desc>
</task>

</xml>
4.2 Project Management Plan

In activities in the project management area, we define aims and scopes of a project, establish plans to achieve the aims, review and control project performance according to the plans, and ultimately manage the project to achieve the aims. Therefore, the project management plan should basically include a project outline, project name, project period, aim, expected effect, project scope, project schedule, a project organization chart, and human resources plan. Figure 6 shows parts of the XSLT code to create a document’s template of a Project Management Plan. Figure 7 shows parts of the XML code that can fill data values into the XSLT template. Then we use a browser to automatically generate a documentation.
Figure 6. XSLT for Project Management Plan
Figure 7. XML code for Requirement Traceability Matrix

```
<xml version="1.0" encoding="UTF-8"/>
<xml-stylesheet type="text/xsl" href="project_plan.xsl"></xml-stylesheet>

<name>The name of this project is called "Software Visualization".</name>
<goal>Independent visualization system through intermediate model that can visualize various languages through one tool through metamodel M13</goal>
<effect>Various languages can be visualized as a single tool.</effect>
</figure>
```

Figure 7. XML code for Requirement Traceability Matrix

Figure. 8 shows to generate a 3) document from both the 1) XSLT code to create the document Template and the 2) XML data produced by the query. As a user uses any browser with XML data, the browser shows the documentation of a project management plan at the right side of Figure. 8. Like Figure 8, we can make any templates what the project manager asks any documents. Therefore, we can reduce the labor time for creating some documents.

4.3 Assets of Documentation in software development lifecycle

To make assets on own process, we can reuse any documents on continuous activities of new projects. In this paper, we suggest to build a PAL process to make assets, and also realize six documents for new project. Table 1 shows a list of document assets. We can make PAL assets in which generate Project Management Plan at a project management activity, and also generate Software Structure Analysis, Requirement Traceability Matrix, Requirement Specification, Source Code Specification, and Test Scenario Result at a development activity.
Table 1. Asset Type List

<table>
<thead>
<tr>
<th>Asset Type</th>
<th>Document name</th>
<th>Description</th>
<th>Version</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Management</td>
<td>Project Management Plan</td>
<td>Project management plan is to conduct project successfully.</td>
<td>2.0</td>
<td>2017.11.01</td>
<td>Jang</td>
</tr>
<tr>
<td>Implement</td>
<td>Software Construct Analysis</td>
<td>Including date, department, class diagram, etc…</td>
<td>1.0</td>
<td>2017.11.02</td>
<td>Kang</td>
</tr>
<tr>
<td>Implement</td>
<td>Requirement Traceability Matrix</td>
<td>Side effect report from requirement change, requirement trace</td>
<td>2.0</td>
<td>2017.11.10</td>
<td>Kwon</td>
</tr>
<tr>
<td>Implement</td>
<td>Requirement Specification</td>
<td>Requirement from user</td>
<td>1.0</td>
<td>2017.11.05</td>
<td>Moon</td>
</tr>
<tr>
<td>Implement</td>
<td>Source Code Specification</td>
<td>Name of source code, return type, parameter, etc.</td>
<td>1.0</td>
<td>2017.11.03</td>
<td>Moon</td>
</tr>
<tr>
<td>Implement</td>
<td>Test Scenario Result</td>
<td>Test result</td>
<td>2.0</td>
<td>2017.11.15</td>
<td>Moon</td>
</tr>
</tbody>
</table>

5. Conclusion

In order to enhance the quality of process in venture or SMEs, we suggest how to build process management and process asset library. They can hardly construct an expensive PAL system because of no enough to invest human resources. Therefore, with the proposed PAL system to manage their in-house assets, they can reduce costs, time and human resources to develop their projects. That is, we can automatically generate any documents with PAL assets in the system without their time on preparing documents. Our PAL system leads to traceability of requirements and traceability between documents. We expect to help Korean domestic small and midsize software development companies to make quality of software.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2017R1D1A3B03035421), and supported by Development of Information Communication and Broadcast R&D(2018 Open OS Environment Development and Diffusion) through NIPA(National IT Industry Promotion
Agency)(S1113-18-1001), and supported by a grant(18CTAP-C133299-02) from Technology Advancement Research Program funded by Ministry of Land, Infrastructure and Transport of Korean government.

References

[4]. Chang-Woo Lee, IoT vaccine designed and developed personally”, Hanbit Media,
[5]. NIPA SP Certification Center, “SW Process Training Focused on SP Certification”, NAPA.

Corresponding Author : Byungkook Jeon, R. Young Chul Kim