Process Asset Library in Software Process Support Technology: A Review of the Literature

Leonardo Bermón, Carlos III University, Avenida de la Universidad 30, 28911 Leganés, Madrid, Spain; E-mail: lbermon@ie.inf.uc3m.es
Antonio Amescua, Carlos III University, Avenida de la Universidad 30, 28911 Leganés, Madrid, Spain; E-mail: amescua@inf.uc3m.es
Javier García Guzmán, Carlos III University, Avenida de la Universidad 30, 28911 Leganés, Madrid, Spain; E-mail: jgarciag@inf.uc3m.es
Paloma Martinez, Carlos III University, Avenida de la Universidad 30, 28911 Leganés, Madrid, Spain; E-mail: pmf@inf.uc3m.es

ABSTRACT
Process assets library (PAL) is a repository of software process-related documents, necessary to implement technologically infrastructures in support of organizational process. This paper offers a review of the literature related to PAL and exposes some questions for solving in Software Process Engineering.

Keywords: Software Process Improvement, CMMI, Process Support Technology, Software Process Assets.

1. INTRODUCTION
The objective of this paper is to present the current situation on software process asset libraries, which are repositories that store information about software process of an organization, allowing that accumulated experience could be used in future software projects.

PAL concept is included nowadays in the CMMI model as a way to achieve the software process definition. This repository is composed generally by a great quantity of no structured information that makes difficult search and reuse of software process assets.

Planning, design and building of a PAL is a task that still is in an initial research phase; therefore, this paper presents a summary of the main studies, advances and questions found in the research of this kind of libraries.

The paper is organized as followed. Second section presents a body of knowledge of the main areas that delimit the concept of PAL. Third section reviews works related to PAL. Fourth section shows a summary of dissertations related. Fifth section discusses issues for solving. Concluding remarks are given in the last section.

2. BODY OF KNOWLEDGE
2.1. Main Areas of Knowledge
Due to the great quantity of meanings that exist in software process engineering domain, a conceptual definition is necessary to avoid serious confusions on having use the PAL term in the area of Software Engineering. This conceptual definition is showed in Figure 1.

- *Software Engineering*: The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software.
- *Software Process*: A set of partially ordered process steps, with sets of related artifacts, human and computerized resources, organizational structures and constraints, intended to produce and maintain the requested software deliverables.
- *Software Process Management:* This area is concerned with the management of the technical aspects of the software development process. It includes knowledge about software process elements: activities, methods, and practice that people use to develop software.
- *Software Process Engineering:* It deals with methodologies, tools, and techniques for the design and implementation of software processes. It includes knowledge about representing the important characteristics of a process as a coherent, integrated set of well-defined software engineering and management processes for organizations, teams, and individuals.
- *Software Process Improvement (SPI):* It is a systematic procedure to improve the performance of an existing process system by changing the current processes or adding new processes for correction or avoidance of the problems identified in the old system by process assessment.

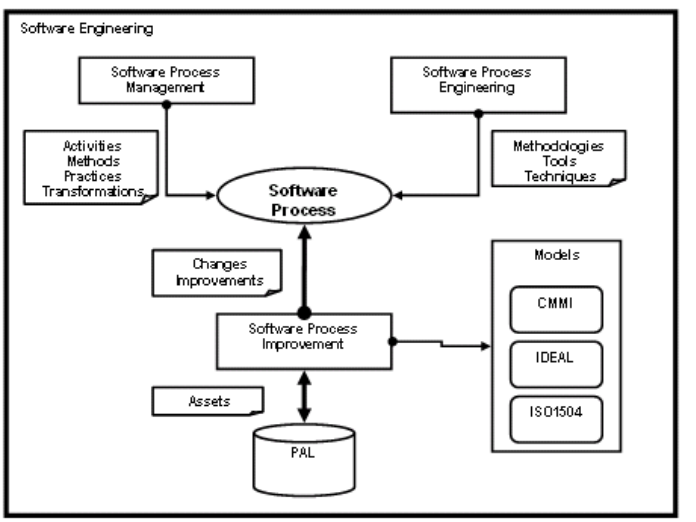
2.2. SPI Models
To achieve a successful SPI, the industry has developed models that assess the reached maturity and to identify improvement strategies.

One of the most known proposals is Capability Maturity Model Integration (CMMI) developed for Software Engineering Institute (SEI). CMMI consists of best practices that address the development and maintenance of products and services covering the product life cycle from conception through delivery.

Other models related to SPI are:

- IDEAL: Describes a model defining the main phases and stages to introduce SPI in an organization and to establish supporting infrastructure.

Figure 1. Areas of knowledge



- ISO 15504: It is an international standard, which specifies a reference model with two dimensions: a process dimension and a capability dimension.

2.3. Process Asset Library: PAL

The documentation of software processes to obtain a CMMI certification contains information related to organization's process assets (knowledge about how the organization realizes its processes). The documentation must be available by a repository named PAL. The use of a PAL is a required practice for achieving CMMI Level 3 and achieving a defined process.

PAL belongs to software process support technologies allowing the integration of production and management technologies and supporting for storage and retrieval of organization's process information.

PAL is an organized, well-indexed, searchable repository of process assets that is easily accessible by anyone who needs process guidance information like examples, data, templates, or other process support materials [12]. Some main purposes of a PAL are:

- Provide a central knowledge base for acquiring, defining, and disseminating guidance about processes related to the organization's tasks.
- Provide mechanisms for sharing knowledge about the organization's process assets and how they are used.

Another useful applications of PAL are to assemble sub-processes in constructive fashion y/o derive modified processes by replacing and/or modifying constituent sub-processes.

3. RELATED WORKS

PAL concept is related to many efforts found in the literature about software process improvement: Software Process Reuse, Process Libraries, PSEE, Experience Library, STARS Program, Zahran's Proposal and other proposals (Figure 2). These works incorporate some technical infrastructures and solutions in organizational process.

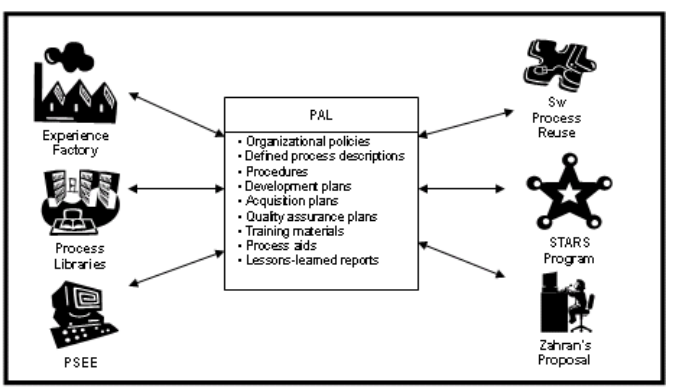
3.1. Software Process Reuse

CMMI indicates that the organization is expected to have one or more approved software life cycles. Individual projects are expected to follow established guidelines and criteria for: selecting the most appropriate of the approved life cycles, tailoring and elaborating the selected life cycle and the organization's standard software process, to fit the needs and particulars of that project [19].

Some studies about software process reuse are:

- Gertrude [29] is a model to re-use software processes, based on object-oriented methods.
- In [25] the characteristics of process components are indicated, allowing projects the freedom to define their own life cycles.
- GPM [17] is a web-based software process management and monitoring tool, providing a mechanism for a software process model to be defined, modified, tracked and measured by a GUI.

Figure 2. Related works



- Case-based reasoning approach has been used on: APSEE [27] a system to identify the similarity between the problem with previous situations stored in the case base and CABS [11] a system that uses a formal notation allowing the user to sketch new processes or adapt template processes, in a matching process which identifies and suggest the reuse of similar processes and process components stored in a library.
- Reuse Architectures: Process Reuse Architecture [10] that uses facets, reuse guidelines and process patterns taxonomy, and Open Architecture for Software Process Asset Reuse [5] through the identification of architectural elements and the specification of element interfaces.
- In [16] a set of process notations and methods for defining processes and tailoring reusable processes is described.
- GUIDE [14] is a rule-based system, for which specific projects are tailored through decision trees used at an arbitrary level of project detail.

3.2. Process Libraries

Process libraries supports reuse capabilities for processes, including the creation, update, deletion, measurement, and management of process assets [33].

A set of works related to building of process libraries is outlined:

- SPLib [24] organizes a process collection constituting a multilevel knowledge base.
- Process Evolution Dynamics Framework [26] is based on an experience-based categorization of process evolution-related activities.
- QUEST [1] is a collection of tools when it highlights a repository, which is a knowledge-based software process model.
- PROGEN [20] is a knowledge-based system for process model tailoring and reuse.

3.3. PSEE

Process-centered Software Engineering Environments (PSEE) can be considered a new generation of software development environments with possibility of adapting to every specific software project, providing a better automated support.

A PSEE interacts with a process model developed by Process Modeling Language and a Process Engine can then enact the process model. The process engine has three components [8]:

- An interpreter that executes the process model.
- A user interaction environment (UIE).
- A repository that stores the artifacts produced during the process. This repository can offer some attributes and purposes of a PAL

3.4. Experience Library

Other proposals to capture critical knowledge from software projects are experience-based approaches and organizational learning, named Experience Library or Repository of Experience [4].

These approaches consist in apply past knowledge to current projects while software practitioners engaging in knowledge creation processes. BORE [15] is a tool that supports this experience-based approach, which collects and disseminates project experiences as "cases" representing emerging knowledge of development practices in an organization.

3.5. STARS Program

The mission of DARPA's Software Technology for Adaptable, Reliable Systems (STARS) program is stimulate the productivity from process reuse technologies. Key constituents of the model are: a PAL, process definition, process enactment, process measurement, and process evolution [13]. PAL provides reusable process assets from public repositories over broad-area networks.

3.6. Zahran's Proposal

Zahran [30] provides a framework for establishing an environment for SPI. The framework has two infrastructures:

- Organizational and management: with roles and responsibilities to manage SPI activities.
- Technology and tools: with facilities and tools for automating process activities and supporting the various process improvement roles and responsibilities.

3.7. Others Proposals

Other ideas related to development of software process information repositories are:

- DAGAR [21] is a repeatable and documented process with accompanying language and is based on a domain architecture process.
- Collaborative Parallel Enactment (COPE) Process Architecture [28] has basic capabilities for specifying process elements, their characteristics and interconnections.

4. RELATED DISSERTATIONS

There are several dissertations related to repositories for Software Process Engineering. Some of them are outlined as followed.

- A Software Process Asset Management and Deployment System is developed in [18] that captures the existing processes, as well as the improvements and changes to these processes into a set of assets that can be reused in future process instances.
- A Workflow Reuse Management System is developed in [22] to unify efforts in PSEEs and Workflow.
- In [7] a study is realized on Process Support Systems that generally are slightly flexible when unexpected situations arise during process execution, producing deviations and inconsistencies. A framework to formalize the concepts of deviation and inconsistency is proposed.

Some studies have been developed in Alabama University about Software Process Engineering to support projects related to *NASA's Flight Software Group at the Marshall Space Flight Center*. One project is Standards Advisor Project, a software system designed to improve the quality and productivity of space systems development by making it easier to find and use standards and other information that guide system development. Some works developed are:

- A Web portal is developed in [23] for delivery information related to process and capturing experience.
- The use of metadata is proposed in [2] to storage standards.

5. ISSUES FOR SOLVING

In spite of the quantity of published results, a lot of questions for solving related to SPI and PAL still continue.

Issues as questions with regard to process model languages, which must be tolerant and allow for informal and partial specification that facilitates their adoption for the practitioners.

Also there are similar questions in PSEEs, since many systems have been developed without a wide acceptance and use, because mainly to their lack of flexibility.

Issues relating to software processes expect to be clarified [9]: requirements for process modeling notations and formalisms, representing global process constraints, granularity, integrating product models, process capture, process-centered environment, enforcement and process automation. There is little information about: what kinds of processes to capture, what processes to ignore, and the need to consider different representations of these processes and other materials for the different roles to be played and the different purposes for which it will be used [6].

Some aspects to consider in future researches are [27]: separation of details in process modeling, level of detail, abstraction mechanisms, search criteria for process elements, mechanisms to ensure the correct reuse of retrieved elements.

Atkinson [3] exposes characteristics that a software library must include: nature of the asset, scope of the library, query representation, asset representation, storage structure, navigation schema, and relevant criterion. Also he defines issues of technical design for software libraries that must be considered: assets definition, assets modeling, definition of relationship between assets, and definition of insertion, removal, update and access policies.

All these issues presented about PAL wait to be faced and solved in future works.

6. CONCLUSIONS

To create a PAL that achieves to standardize and reuse a software process, it is necessary: to describe the process; to store it in a suitable format; to identify the desired process in a database and to retrieve it; and to adapt it to organizational needs [29].

The bibliographical review has discovered that the researchers have carried out many efforts in the first aspect (process description) leaving aside other aspects that even have not been solved.

Models for software process improvement as CMMI indicate the need to build and manage this kind of repositories. Nevertheless CMMI does not specify what technical characteristics are needed to implement a PAL successfully. For which, it is an obligatory requirement that already has begun to obtain results, with the emergence of research proposals that try to address the problem of defining infrastructures for supporting software process assets (standardized and integrated in the future) to create, to organize, to access and to reuse information concerning software process improving.

ACKNOWLEDGMENT

This work on PAL was supported by GPS – Software Process Management Platform: Modeling, Reuse and Measurement (TIN-2004-07083) of the Spanish National Research Program.

REFERENCES

[1] Ahn, Y et al. **Design of knowledge-based integrated software process improvement tools.** IEEE. 1998.
[2] Arla, B. An XML-based Metadata Repository for NASA Standards Advisor. University of Alabama. 2003.
[3] Atkinson, S. A Formal Model for Integrated Retrieval from Software Libraries. In Technology of Object-Oriented Languages and Systems. 1996.
[4] Basili, V. et al. Experience Factory. In Encyclopedia of Software Engineering: Wiley & Sons, 1994.
[5] Boehm, B. et al. An open architecture for software process asset reuse. Proceedings of the 10th International Software Process Workshop. 1996.
[6] Carter, L. Managing the Invisible Aspects of High-Performance Teams. In Crosstalk Journal of Defense Software Engineering. 2001.
[7] Cugola, G. Inconsistencies and Deviations in Process Support Systems. Politecnico di Milano. 1998.
[8] Cugola, G. et al. Software Processes: a Retrospective and a Path to the Future. In Software Process. 1998.
[9] Dowson, M.: Software Process Themes and Issues, International Software Process Symposium. 1990.
[10] Fiorini, J. et al. Process Reuse Architecture. In Computer Science. Proceedings of the 13th International Conference on Advanced Information Systems Engineering. 2001.
[11] Funk, P. et al. Reuse, Validation and Verification of System Development Processes. First International Workshop on the Requirements Engineering Process. 1999.
[12] Garcia, S. Improving Process Improvement with Process Asset Libraries. 2004.
[13] Hart, H. et al. Panel: The STARS Process Approach and Ada Projects. Proceedings of the conference on TRI-Ada. ACM Press. 1992.
[14] Henninger, S. An Environment for Reusing Software Processes. Fifth International Conference on Software Reuse. 1998.
[15] Henninger, S. Tool Support for Experience-Based Methodologies. In Advances in Learning Software Organizations. 2003.
[16] Hollenbach, C. et al. Software Process Reuse in an Industrial Setting. Fourth International Conference on Software Reuse. 1996.
[17] Hutchens, K. et al. Web-based software engineering process management. In Proceedings of the Thirtieth Annual Hawaii International Conference. 1997.
[18] Kaltio, T. Software Process Asset Management and Deployment in a Multi-Site Organization. 2001.
[19] Kellner, M. Connecting reusable software process elements and components. Process Support of Software Product Lines. 1996.
[20] Kerschebrg, L. et al. PROGEN: A Knowledge-based System for Process Model Generation, Tailoring and Reuse. 1996.
[21] Klingler, C. et al. DAGAR: a process for domain architecture definition and asset implementation. Proceedings of the conference on TRI-Ada. 1996.
[22] Kruke, V. Reuse in Workflow Modeling. Norwegian University of Science and Technology. 1996.

[23]Ma, H. Standards Advisor: Activity Modeling and Model-Driven Web Portal for a Software Engineering Process. University of Alabama. 2003.
[24]Mi, P. et al. A Knowledge-Based Software Process Library for Process-Driven Development. In Proc 7th Knowledge-Based Software Engineering Conference. 1992.
[25]Mogilensky, J. et al. Applying Reusability to Software Process Definition. Proceedings of Tri-Ada. 1989.
[26]Nejmeh, B et al. A Framework for Coping with Process Evolution. In the Proceedings of the Software Process Workshop. 2005.
[27]Reis, R. et al. APSEE-Reuse: A Case-Based Reasoning Model for Reuse and Classification of Software Process Assets. Brazil. 2001.
[28]Riddle, W. Coping with Process Specification. Integrated Design and Process Technology. 2003.
[29]Succi, G. et al. Standardizing the reuse of software processes. StandardView. 1997.
[30]Zahran, S. Software Process Improvement. Addison-Wesley. 1998.