Lesson Learned in a Self-Sustainable Framework

1 Mensely Cheah Siow Fhang, 2 Wong Wai Tong, 3 Rajashekara Swamy
1 Staff Engineer, ICT Quality, System Engineering, MIMOS Berhad, Kuala Lumpur, Malaysia
2 Senior Staff Engineer, ICT Quality, System Engineering, MIMOS Berhad, Kuala Lumpur, Malaysia
3 Principal Engineer, System Engineering, Software R&D and Infrastructure, MIMOS Berhad, Kuala Lumpur, Malaysia
1 sf.cheah@mimos.my, 2 waitong.wong@mimos.my, 3 rajashekara.swamy@mimos.my

ABSTRACT

The world is being transitioning from the information age to the knowledge age. Successful organizations typically have a higher capability in learning from their past experiences and they are efficient in using their intellectual assets resulting in higher usage of technology and improved processes to support constant changes within the work units and organization. These are essential factors for organizations to become world leader in their core business areas. Today, one of the key challenge of any lesson learned system is the enormous effort required to establish a knowledge base, which ensures data integrity and having the ability to generate useful information from various data sources. When the volume of information data for the project increases over time the complexity of various projects parameter associated for lesson learned management turns into an extravagant process. The paper will elaborate an implementation of a self-sustainable lesson learned framework, which can be independently maintained with minimal effort. This approach outlines a robust methodology to capture the lessons and information that enable the organization to learn efficiently streamlining the knowledge transformation from project level to organizational level and vice versa.

Keywords: Lesson learned, self-sustainable framework, knowledge base, process improvement

1. INTRODUCTION

Today, virtually all companies can be considered to be knowledge organizations where knowledge is their primary resource and source of differentiation [1]. A good lesson learned that is factual and technically correct is reliable in drawing a specific design, process, or decision that reduces or eliminates the potential for failures and mishaps, or reinforces a positive result [2] in an organization.

Similar to any other assets management, e.g. inventory management, document management, lesson learned needs to be effectively managed, with no process and system in place, one will require huge effort in the collection, sharing and application of the lesson learned, thus, many organizational is in search of a Lessons Learned Management Systems (LLMS) in order to provide access to results and outcome which were processed and validated, regarding various activities within their projects which could be reused in their other similar projects or training of new project team members [3].

Knowing the importance of lessons learned and Lessons Learned Management Systems, the authors believes that a self-sustainable framework combining both aspects of genuine lessons learned and effectual Lessons Learned Management Systems are crucial in ensuring lessons learned process could survive in long run.

It’s the paper objective to present a framework for implementing such initiatives, that’s behaving as i) An effortless system, ii) A real-time system, iii) A self-service system, iv) A motivating system and v) A one and all system.

2. LESSON LEARNED IN THE FRAMEWORK

When a lesson learned framework is planned, it’s important to comprehend questions like what, where, when, why, how and who pertaining to lesson learned execution before a proposal is put up. In this paper, Figure 1 below explains about the lesson learned in the framework, with further details available in Figure 2 till Figure 6.

![Figure 1:Overview of lesson learned](image)

2.1 Lesson Learned – What, Where?

A good lesson learned shall focus on studying both the strength and opportunity, which is having focus on the favorable outcome that had occurred and the mistakes that were made. Typically, strength is used to recognize the successful approaches undertaken by the project team (example: processes, decisions) so those can be repeated or improvised in future project / release, whilst opportunity is used to avoid the mistakes by preventing its occurrence in the future.
One must be aware that effective transfer of knowledge from what is learned is not solely to other project teams, but also to the organization as a whole. Many lesson learned systems today are heavily focused on project level data, giving not enough attention to the lesson learned at organization level, and hence, it’s the intention of this paper to explain and elaborate on the 2 distinct and interconnecting levels – i) Learning from/to at Project Level, ii) Learning from/to at Organization Level.

Figure 2 below depicts the types of lesson learned collected and the level of learning that lesson learned reside.

2.2 Lesson Learned – When?
Lesson learned identification, analysis, review and sharing must occur throughout the project life cycle, from project kick off until project End of Life (EOF). Lesson learned could be continual collected at project kickoff, project gates review, project activity, project status meeting, project risks review and project defects (ranging from internal testing, external acceptance and production escaped defects), project audit / assessment and postmortem / retrospective.

Data from each project phase provide insights to the project behavior at different chapters in the project lifecycle.

Figure 3 below explains on the types of lesson learned collected and its relationship with the level of learning that lesson learned reside. For instances, lesson learned on requirement may be collected in requirement gathering and analysis phase, lesson learned on design may be revealed in coding and testing phase, lesson learned on validation may be discovered in various internal and external testing phase, while lesson learned on project management, team coordination and project practices may be collected across the project life cycle continuously.

2.3 Lesson Learned – Why?
The motivation for why lesson learned are required, drives the sustainability and usefulness of the Lesson Learned Management System. Understanding how lesson learned is used helps to design a lesson learned framework more robust, catering for expectation at various levels and by stakeholder needs.

Usage of lesson learned is recognized at 2 levels – i) Organization Level (for Senior Management, Process Improvement Team) and ii) Project Level (for Project Management Team, Project Support Team and Project Team).

Figure 4 below elaborate on the expectations from the different stakeholders:

1. Senior management is looking at:
   - What are the lessons learned?
   - How many opportunities remain opened?
   - How many opportunities are critical?
   - How lesson learned can be effectively distribute, search and retrieve org wide
   - How many lesson learned has been propagated to new project / release?
   - How much cost saving with these learning (ROI)?

2. Process Improvement Team is looking at:
   - Background of the gap / improvement needed
   - To have discussion board
   - To have approval workflow
   - Action items assigned
   - Metrics: Aging, severity, priority, due

3. Project Management Team is looking at:
   - To access / acquire knowledge base as inputs to project plan (i.e.: risks management, project task in schedule)
   - How many risks that we have prevented / detected / corrected?

4. Project Support Team and Project Team is looking at:
   - To access / acquire related special cause / common cause @ org wide database
   - To capture and store lesson learned
• To effectively organize, manage and maintain knowledge base on lesson learn

Figure 4: Lesson learned – Why?

2.4 Lesson Learned – How?

It is also vital that project knowledge and experience data / information are preserved. How well lessons are learned depends highly on how well the knowledge can be acquired, how effective the attributes are captured, validated / tested and what are the feedback mechanism established for accessibility to projects and organization stakeholders.

Figure 5 given below elaborates the PDCA (Plan Do Check Act) cycle as one of the fundamental concepts in the lesson learned framework [4]. PDCA to be performed iteratively to ensure the continuity of the Lesson learned practice in the Project and Organizational.

Figure 5: How? Plan–Do–Check–Act

2.5 Lesson Learned – Who?

While lesson learned framework focusing on the concept, methodology, process and implementation structure, the next important elements to make up the framework of lesson learned management is the People and Tool.

Figure 6 below outlines the involvement of People and Tool in the Lesson learned Management overall.

People:
- Identify
- Capture and store
- Analyze and discuss
- Presentation and sharing
- Understand and continuous learning

Tool:
- Organized and managed
- Maintained and distribute
- Search and retrieve
- Group and prioritize
- Allow discussion and comment

Figure 6: Lesson learned – Who?

3. SELF-SUSTAINABLE LESSON LEARNED FRAMEWORK

Lesson learned is a key factor for continuous improvement, as it can bring all the revitalization on future projects and help them to succeed. Thus, it’s essential to build a continuous revolutionized system, that’s functional and self-sustainable to manage the lesson learned.

In establishing a self-sustainable lesson learned framework, considering automation and maintainability in data, information processing and interaction is the first step to encourage the continuous feedback control in long run.

Five main pillars of notions are used building the self-sustainability into the lesson learned framework, which consisting of i) An effortless system, ii) A real-time system, iii) A motivating system, iv) A self-service system and v) A one and all system.

3.1 An Effortless System

Most industry captures and model lesson learned in an independent repository serving purely as the knowledge base and data store, acting as lesson learned database for the organization overall. This approach is good to start off but when the organization grows and begins to establish multiple information systems such as Defects Management System, Risks Management System, Change Management System, etc, significant effort is needed for information compilation, study and reporting. Time taken to analyze the information could be more if the system themselves are not integrated. A standalone lesson learned system from project management system also exhibits that lesson learned is outside the project lifecycle routine.

In any lesson learned management system, it’s important to document and share findings. The best way to do this is by streamlining or embedding the lessons learned data with various project repositories with good meta-data to help with identification. This will make it easier in the process for discovering, collecting, analyzing and disseminating the lessons learned. Lesson learned data is then analyzed thoroughly taking into account the project structures, constraints and implementation plans.

Our self-sustainable lesson learned framework suggests that the lesson learned system itself be built
within the Application Lifecycle Management (ALM) system, where the lesson learned data is tracked and managed alongside with the defect data, change management data, task data and any project related data. With this, lesson learned is seen as an integral part of project lifecycle, rather than an isolated activity to be performed at various milestones.

With lessons learned data made part of the project ALM system, all project data and attributes are easily accessible. This makes it easier to analyze recurring problems, to update the data and to maintain the relevancy and accuracy of the data. Also, data categorization, data searching is straightforward and effortless.

### 3.2 A Real-Time System

One of the challenges in establishing a lesson learned system is that information recorded is not useful to improve the project. This is partly attributed because the database will, over time, include old and irrelevant information creating the perception that the whole database is not useful. The problem is compounded with the project teams having to identify lessons at the end of a project and record them in a project closure report for the next project.

Many projects conduct a "lessons learned" review during and at the end of the project, causing huge amount of knowledge and experience that is not generally captured, digest and learned on time. Echo on the historical moment when memories have faded frequently resulted incomplete, outdated and invalid data injected into lesson learned database. Lack of participation from project team is another challenge as project members may be too busy getting to the next task.

Our self-sustainable lesson learned framework suggests that lesson learned system be used to incrementally capture the lessons learned along with the project journey and turning that hindsight into best practices to achieve far greater long-term project success.

### 3.3 A Self-Service System

Lesson learned review is a great venue for team members to discuss how they can capitalize on what they did well, and to agree on what needs to be changed. It’s crucial to build and maintain a high level of confidence in the lesson learned database, the data should always be current and accurate. Having both the organizational and project level lesson learned within the same system, data review, data update and data cleansing at both levels are easy.

To achieve this objective, the self-sustainable lesson learned framework suggests the lesson learned system must enable lesson learned from project level to be propagated at the organization level mechanically so that proper weightage could take place with sufficient data to help in decision making and improvement prioritization.

Lesson learned at organization level shall focus on common practices, which once changed, will be affecting all projects and improving the organization performance all, while lesson learned at project level is focusing on experienced learning at project specific, focusing the improvement to be carried out in the next release or new project of the same lab.

Example of lesson learned at organization level:

- a. To communicate to all parties if there is any change in the generic platform
- b. Need technical support to support continuous integration thru CI server (centralized build server)
- c. To align the applicability of Test Environment Checklist (TEC) and Test Strategy Checklist (TSC)

Example of lesson learned at project level:

- a. When adopting platform in solution, application shall only consume their service, not their UI. This will make the dependencies loosely coupled.
- b. To reduce the waiting time for server, need to have dedicated server for demo
- c. To invite required platform engineers to participate in upcoming Solution requirement review sessions to understand business flow and to learn exactly what users need.

Communication could be made to inform all project teams whenever the organization lesson learned database is updated with new information and, more importantly, raise awareness whenever the data has resulted in a change to the organization’s processes. This close loop approach ensures that the lesson learn system remains relevant and accessible to both stakeholders of the project and the organization.

### 3.4 A Motivating System

The earlier section suggested that organizational and project levels lesson learned must be placed in a cohesive unit and made accessible, but the problem arises when employees are not encouraged to use this database.

Our self-sustainable lesson learned framework suggests the use of lesson learned database be encouraged, and this could be done through the implementation of point reward system at the project level. The system shall allow authorized access to the pool of knowledge and permit comments and feedback, at the same time, invite suggestions for process improvement based on the lessons learned data. Reward point is given to the preferred submission of data that contribute towards organization or cross project learning. Additional points are given for lesson learned that have been implemented at various levels. With this approach, users of the system will be encouraged to contribute and share their experience knowing that their inputs will be used and is beneficial to others.

### 3.5 A One and all System

Lesson learned is not a department, not a project milestone but a routine activity in project lifecycle. It’s not the ownership of the project leader, project manager but collaborative effort from everyone in the project team.
Ownership and empowerment is given to the whole team rather than certain individual.

Also, once we have captured the lessons learned, we have to make sure that the information is easily referenced by other project teams. Project teams is encouraged in searching for problems that exhibit similar patterns and instigate appropriate process changes or improvement proposal as prevention or corrective measures.

4. SELF-SUSTAINABLE LESSON LEARNED IMPLEMENTATION

Lesson learned systems are motivated by the need to preserve an organization’s knowledge and convert individual knowledge into organizational knowledge so that, when experts become unavailable; other employees who encounter conditions that closely match some lesson’s context may benefit from applying it. Therefore, a lesson learned is a validated working experience that, when applied, can positively impact an organization’s processes [5].

Many of the lesson learned systems failed due to its inability to efficiently promote knowledge sharing among projects and to advocate the required changes to the organization decision making for process change / improvements.

The self-sustainable lesson learned framework proposed that lesson learned system itself must enable lesson learned from project level be propagated to organization level mechanically so that proper weightage could take place with sufficient data to help in decision making and improvement prioritization.

The integration of Project Level and Organization Level lesson learned are shown in Figure 7 and Figure 8.

Figure 7: Lesson learned at project level

Figure 8: Lesson learned at organization level

In implementing the lesson learned framework, technologies must efficiently facilitate the collection, storing validation and dissemination of data / information motivating the sharing and reuse at all levels. Hence, lesson learned framework is motivated by the notion of automation and self-sustainable, employing various agents such as workflow, dashboard, search, cloning, linker, vote and watch, planning board, etc.

Figure 9 below shows the various agents contributed in the framework.

Figure 9: Lesson learned framework

5. MATURITY OF LESSON LEARNED MANAGEMENT

Was the lesson truly learned by other projects in the organization? Has the improvement been applied constructively? These are the frequently asked question (FAQ) in any lesson learned initiative. Unfortunately, generally same or similar problems or defects are still echoed in different projects, why?

The authors feels that it’s not the issue of not contributing into lesson learned knowledge database, but it’s the way lessons learned is apprehended and applied ineffectively, more often than not, lesson learned is not captured on time, information is incomplete and knowledge base often outdated for the current trends causing incorrect improvement implementation.

Maturity models seek to establish levels of development of processes, called maturity levels that characterize stages in the implementation of improvement processes in the organization [6].
These raise the questions of whether the Lesson learned Management is effective, and how to assess and report the maturity level for a Lesson learned Management System.

6. FUTURE WORK

For the long term success of the lesson learned framework, technology must be studied and invested aligning with the market trend and industry best practices. The technology would be enablers towards supporting a more socially accepted, useful, and practical system for users.

Fundamentally, the framework extension may employ technology to focus on the system’s intelligence that is capable of promoting data intelligence, data discovery, information interaction and self-promotion. Figure 10 below shows the various concepts suggested covering Semantic / Ontology, met knowledge, smart traceability, etc.

![Figure 10: Lesson learned framework extension](image)

7. CONCLUSION

Promoting organizational innovation, effectiveness, efficiency through organizational learning has been a focus of increasing attention of industry of all sizes. A self-sustainable lesson learned framework needs to be established to foster and embrace the continuous learning system and culture, where an organization aspires to contribute and learn. Lessons learned will eventually become part of the organization’s values and part of its continuous improvement process.

REFERENCES


AUTHOR PROFILES

Mensisely Cheah Siow Fhang is a Staff Engineer in MIMOS Bhd, Kuala Lumpur, Malaysia who has 10 years’ experience in software engineering processes and software development domain. She is certified by QAI as CSQA (Certified Software Quality Analyst), holding professional level of competence in the principles and practices of quality assurance in the IT profession. She is also a Six Sigma Green-belt holder from Motorola University. She has participated significantly and contributed to 2 organizations in CMMI maturity level 5 assessments (Motorola Global Software Group, Malaysia) and MIMOS Berhad, she is also proficient in ISO standards related to software engineering.

Wong Wai Tong is a Senior Staff Engineer in MIMOS Bhd, Kuala Lumpur, Malaysia who has 15 years’ experience in software engineering processes and software development domain. He is a Six Sigma Black-belt holder from Motorola University, CMMI SCAMPI Appraiser and Certified Internal Auditor for ISO 9001:2008 QMS. He is part of the Process Quality unit that was instrumental in driving MIMOS Berhad and Motorola Malaysia Software Center to achieve CMMI Maturity Level 5. He is also proficient ITIL Version 3 Framework, and ISO standards related to software engineering.

Rajashekara Swamy is a Principal Engineer at MIMOS Bhd, Kuala Lumpur Malaysia. He has over 20 years of technical & management experience in the areas of Software & Systems development, Testing, Quality management, Project office, Process improvement. He has expertise in developing data acquisition & supervisory systems for safety critical applications. He is a certified lead auditor for ISO 27000, ISO 9000:2008.