



## INFRASTRUCTURE REQUIREMENTS FOR EXPERIENCE BASED FACTORY MODEL IN SOFTWARE DEVELOPMENT PROCESS IN A COLLABORATIVE ENVIRONMENT

Mastura Hanafiah<sup>1\*</sup>, Rusli Abdullah<sup>1</sup>, Masrah Azrifah Azmi Murad<sup>1</sup>, Jamilah Din<sup>1</sup>, Norfaradilla Wahid<sup>2</sup>

<sup>1</sup>Faculty of Computer Science and Information Technology, University Putra Malaysia

<sup>2</sup>Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn, Johor, Malaysia

\*Corresponding author email: [mastura.hanafiah@daimler.com](mailto:mastura.hanafiah@daimler.com)

This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

### ARTICLE DETAILS

#### Article history:

Received 07 August 2017

Accepted 11 October 2017

Available online 3 November 2017

#### Keywords:

Experienced Based Factory Model,  
Software Development Process (EBF-SD)

### ABSTRACT

This paper discuss the infrastructure requirement of the model named Experienced Based Factory Model for Software Development Process (EBF-SD) in order to ensure the implementation of the model will be able support collaborative environment effectively and efficiently.

## 1. INTRODUCTION

Software development today has accumulated data, information and knowledge during its development lifecycle. Software developers go through the different kind of activities or events to achieve their goals which eventually turns out as valuable experiences. The form of the experiences would be either tacit or explicit. Explicit knowledge can be documented in a formal and systematic language, while tacit knowledge is unstructured, highly personal and hard to articulate. It is based on intuition, judgment and experience one has, derived from actions, procedures, ideals, values, emotions [1]. Additionally, due to the growing numbers of software demands, many organizations has adopted distributed development approach in order to find the right skills at the right price by implementing global software development or offshore development where the teams are distributed in geographically distributed locations. In these kinds of development, managing knowledge and experiences is even more challenging. Be it locally or offshore, organizations need to have a knowledge management system (KMS) managed efficiently in a collaborative environment so that the knowledge and experiences gained are not lost and could be reused and enhanced in the future. This research has proposed the adoption of experience based factory model that was initially created by a group of scientist in the domain of software development process [2].

Experience Factory (EF) is a framework that enables organizational learning by recognizing the need to distinguish the support organization and the project organization [2]. The support organization allows the project organization to observe itself by collecting data, building the model, drawing conclusion, packaging as reusable experiences for future reuse and feeding the experiences back to the project organization. EF makes use of reusing products, processes and other forms of knowledge from system lifecycle in order to improve system development at a lower cost. This paper discuss the infrastructure requirement of the model in order to ensure the implementation of the model will be able support collaborative environment effectively and efficiently.

## 2. THE WORK

### 2.1. Proposed Model

The model, namely Experienced Based Factory Model for Software Development Process (EBFSD), is composed of several components that are evaluated during the literature review analysis (Figure 1).

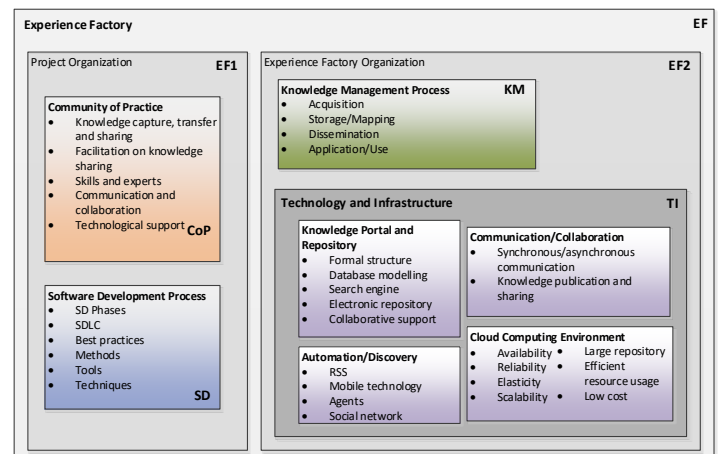


Figure 1: The proposed model EBF-SD

Adapted from the original EF model, the proposed model is composed of two main organizations: Project Organization (PrO) and EF organization (EFO). The model can also be referred in the previous work by a group researcher [3]. PrO consists of the community of practice (CoP), the people that involve in the software development, and the software development process (SD) that are carried out by the CoP. They are responsible to supply the project data to the EFO. EFO consists of the components to support the collection of the data, packaging them, and supplying back to PrO as a useful product. As such, EFO is equipped with the knowledge management (KM) process and appropriate technology and infrastructure (TI). In order to have an efficient and effective infrastructure for collaborative environment, the model is empowered with cloud computing technologies.

### 2.2. Infrastructure Requirement

Cloud Computing is a combination of several computing concepts and technologies such as Service Oriented Architecture (SOA), Web 2.0, virtualization and other technologies that is dependent on the Internet to provide online business applications to serve customers with their computing needs with software and data stored on the servers [4]. For the prototype implementation of EBFSD model, we focus on the infrastructure requirement in the following areas: availability, storage requirement, KM automation, security, and network and performance.

Availability. High availability of the software refers to ensuring that the software is running (available) most of the time. Especially for distributed development environments which normally involves different timezones, it

is recommended that the software has minimum downtime and able to be accessed from anywhere, anytime. Service downtime may affect user experiences, and in the case for KMS, downtimes may affect user willingness to continue using the system as the knowledge sharing platform. High availability in cloud is achieved in several ways, most common are monitoring, replication, and failure detection [5]. In our prototype, we use Google Cloud Platform (GCP) to host our systems mainly by leveraging their PaaS and IaaS [6]. GCP offers second generation instance configuration to overcome any outages or failover, by allowing continuity of the data to be available to clients.

### 2.2.1. Storage requirement

The proposed model requires the storage for any kind of knowledge, explicit or tacit. Documents, images, videos or recordings can be the example of knowledge model. It is required that all these assets to be reserved during the lifespan of the applications. Due to the growing number of data, the storage requirement will also increase over time; thus, more investment is required. Cloud storage offers solution for massive data collection. Additionally, some cloud storage solution like Amazon S3 also offers query-in-place functionality that allows powerful analytics directly on the data [7]. S3 provides flexibility in the way they manage data for cost optimization, access control, and compliance. GCP also provides facilities to handle big data with the services like MapReduce, BigTable, BigQuery, etc. Similarly, GCP also provides serverless big data analytics solutions which removes the complexity of building and maintaining data analytics system.

### 2.2.2. Automation of KM processes

KM process involves the activities like knowledge creation, approval, dissemination and recommendation. In order to increase the awareness among the knowledge workers, automation should be incorporated in KMS. For workers to make full use of KM, workflow solution can be viewed as a trigger of KM support activities and a distributor of organizational knowledge for the purpose of providing better judgment and decisions [8]. But, workflow solution approach alone might not be sufficient to aid in the complete automation because workflow solution seems to lack of reactivity, semantics, extensibility, resource management, and heterogeneity; as such, it is suggested that integration with agent technology is necessary [9]. Integration between workflow and agent has been developed previously in KM based applications as in reference [10]. The incorporation between workflow and agent technology ensures the automation objectives are met and that the integration between people and business process applications are more stream-lined.

### 2.2.3. Security

For a system that storing large amount of organizational data, it is crucial that security aspects is handled appropriately. Despite the many benefits of cloud computing, it also has challenges in security due to the possibility of public exposure, especially in the case of public cloud. There is a great deal of uncertainties at many levels e.g. network, host, application and data levels [10]. From the analysis by a researcher, it is found out that security in cloud computing mainly concerns with storage, virtualization and networks [11]. GCP security model is an end-to-end process where by the product and services are built in with security as the core. Some of the platform security features include: secured service APIs and authenticated access, logging, data encryption, secure global network, intrusion detection, security scanning, compliance and certifications. Additionally, GCP provides more capabilities to keep cloud projects secure by regularly applying operating system and application patches, providing user and credential management at project level, maintaining network firewall rule, penetration testing, management of sensitive data, logging and monitoring tools, and verification of security, privacy and compliance controls.

### 2.2.4. Network and performance

Another important aspect of the proposed model is to achieve an acceptable performance especially when serving users with knowledge business activities e.g. creation, storage, search, navigation, dissemination and reuse. With GCP, the network backbone is formed with thousands of miles of fiber optic cable with advanced networking that delivers fast, consistent and scalable performance. In a recent study by Cloud Spectator, it is revealed that GCP ranked the first for price-performance and block storage performance against AWS, Microsoft Azure and IBM SoftLayer [12]. In addition to the persistent disks, GCP also offers users to configure local SSDs for even greater performance and lower latency, by creating the instances depending on the space requirement, performance characteristics and optimizing storage performance.

## 3. CONCLUSION

By empowering the model in cloud computing, it is clearer that the model can support software development process on knowledge sharing activities in a collaborative environment. Though there are many cloud service providers, GCP may stand out especially in terms of performance, storage and availability.

## REFERENCES

- [1] Nonaka, I., Toyama, R., Konno, N. 2000. SECI, Ba and Leadership: a Unified Model of Dynamic Knowledge Creation. *Long Range Planning*, 33 (1), 5-34.
- [2] Basili, V.R., Gianluigi, C. 1995. Improve Software Quality by Reusing Knowledge and Experience. *Sloan Management Review*, 37(1), 55-64.
- [3] Hanafiah, M., Abdullah, R., Murad, M.A.A., Din, J., Nor, M.Z.M. 2017. Experience Based Factory Model For Software Development Process: Item Construct Validation On Questionnaire Design. *Journal of Theoretical and Applied Information Technology*, 95 (1), 177-195.
- [4] Marinos, A., Briscoe, G. 2009. Community Cloud Computing. *IEEE International Conference on Cloud Computing*, 5931, 472-484.
- [5] Endo, P.T., Rodrigues, M., Gonçalves, G.E., Kelner, J., Sadok, D.H., Curescu, C. 2016. High availability in clouds: systematic review and research challenges. *Journal of Cloud Computing*, 5, 16.
- [6] Google Cloud Platform. 2017. <https://cloud.google.com/>
- [7] AWS. 2017. <https://aws.amazon.com/s3/>
- [8] Toledo, C.M., Chiotti, O., Galli, M.R. 2012. Process-aware approach for managing organizational knowledge. *Information Systems*, 62 (16), 203-218.
- [9] Jennings, N. R., Norman, T. J., Faratin, P., O'Brien, P., Odgers, B. 2000. Autonomous agents for business process management. *Applied Artificial Intelligence*, 14 (2), 145- 189.
- [10] Rosado, D.G., Gómez, R., Mellado, D., Medina, E.F. 2012. Security Analysis in the Migration to Cloud Environments. *Future Internet*, 4(2), 469-487.
- [11] Hashizume, K., Rosado, D.G., Medina, E.F., Fernandez, E.B. 2013. An analysis of security issues for cloud computing. *Journal of Internet Services and Applications*, 4(5).
- [12] Cloud Spectator. 2017. <https://lp.google-mkto.com/rs/248-TPC-286/images/Cloud-Spectator-Best-Hyperscale-Cloud-Providers.pdf>