Chapter 11
Agility in Competency Building—A Pedagogical Approach with a Case Illustration from the Indian IT Industry

Jose Kumar Reddypogu, Ravi Shankar Pillutla and M.G.P.L. Narayana

Abstract People competencies must be constantly upgraded or revised in organizations, since market conditions are constantly change. The importance of having systems and methods to make all aspects of the organization agile is critical. In knowledge industries, people competency plays a major role in the overall business. Systems and methods used to manage people competencies is a must therefore, enabling a quick change in competencies. This scenario is characterized by rapidly changing competency needs, a short time for building them, short scope of requirements, etc. Current methods focus on knowledge content whereas the change imposes changes in context, constraint peculiarities which do not largely impact the knowledge content. The impact is more visible in the application of knowledge, hence such training methods poorly enable competence to deal with change. Consequently, there is a need for low latency and high agility in people competency development pedagogies.

Keywords Agility · Competency building · Organization agile

11.1 Introduction

Change is constant and every organization and the individuals working in it face changes. Software organizations that deliver software to customers, deal with changes either in the form of change in requirements or technology change by

J.K. Reddypogu (✉) · R.S. Pillutla · M.G.P.L. Narayana
Business Systems & Cybernetics Centre, Tata Consultancy Services Ltd,
Hyderabad 500034, Telangana, India
e-mail: jose.reddypogu@tcs.com

R.S. Pillutla
e-mail: p.shankar@tcs.com

M.G.P.L. Narayana
e-mail: mgpl.narayana@tcs.com

Sushil et al. (eds.), *Flexible Work Organizations*, Flexible Systems Management,
DOI 10.1007/978-81-322-2834-9_11

way of versions. 55–60 % of software operations in the Indian context deal with managing change in software. Therefore, change is an important and constant factor that must be factored into every step of the working processes. For example, a sudden rush in the market for a retail product may in a short time create a major increase in the load on the system and hence call for a change in the way the product is supported. Or, this change may call for an upgrade in the product and hence, a new development exercise. Such situations are not uncommon. Similarly, there may be a high attrition of people with the right skills and hence, there is a sudden need to build competencies relevant to the problem and with good ability to deliver. Massive changes due to major market changes and requests for technology or product changes are a cause of worry for the management. Getting people to build new competencies is time taking and it may take longer still before these skills can be implemented for projects and for addressing customer needs. Change may impact any aspect of the work that is undertaken. With the advent of the internet and the world becoming more connected with transactions happening instantly, the pressure of dealing effectively with change has become imperative and methods to address them, particularly in relation to people competencies has become more difficult and more important. The impact of this change not being met is difficult to estimate or determine (Coverity5 Brochure 2009). However, that there is a need to be ready for constantly re-adjusting staff competencies.

In a typical software organization, project induction presents one of the most frequent challenges for change in the competencies of people. New staff and experienced employees must change from one work role to another, and this is a routine phenomenon. Typical scenarios are: new staff is provided Java training and due to business reasons are asked to change to .Net or Business Intelligence work. A person working in Banking on Java applications has to take on a developer role in a .Net project for retail application domains, in such scenarios there will be a need to move from one set of competencies to another. This is the most common scenario in the software industry. Inappropriate staff competencies affects the following: Cost of quality or reworks, auditing, lost time in delivering products, inappropriate solution to requirements, directly impacting the bottom line of the cost, quality and delivery.

The scenarios mentioned above will be used as the backdrop to discuss the pedagogical approach for building agility in staff competency building.

11.2 Current Approaches to Staff Competency Development

The current approach in the industry is to understand competence as knowledge of the technology and domain. Training is an instrument or mechanism to impart these and the task of knowing how to apply this knowledge of technology and domain to the work or requirements and deliver the outcome, is that of the trainees or developers. Competence, as a concept that builds the ‘ability to do’ in

the trainee is not recognized. In fact, By reference to the vocabulary document (normative ref) ISO 9000, you will see that competency is defined as the 'demonstrated ability to apply skills and knowledge' (Ravi and Narayana 2012; Curtis et al. 2009). This is one of the important reasons why pedagogies have not been popular in building competency. Knowledge building is easy and makes it easy and repeatable for trainers. Classroom training, training via popular massively open online courses (MOOCs) are accessible via internet, with reduced involvement of faculty, often results in poor correlation with actual project performance. This results in a high cost for quality since the effectiveness of training is questionable and subjective. The reasons could be varied. Competency requires the context and the process steps that must be executed for delivering the results or outcomes to be known in the specific case. This makes imparting competence difficult and specific to the desired outcomes and to the context.

11.3 Competency and Knowledge

In the context of agility and low effective latency, the difference between competency and knowledge is as follows: (Curtis et al. 2009; Ravi and Narayana 2012). Competence is the ability to do or ability to deliver required work, while knowledge is 'The information and understanding that someone must have to perform a task successfully'. Knowledge provides the basis for performing a skill (Ravi and Narayana 2012). Hence, people competence is an effective way of addressing agility than doing it by imparting knowledge content.

Agility is about quickly adapting to change. The creators of the concept of ''agility'' at the Iacocca Institute of Lehigh University (USA) defined it as: 'A manufacturing system with capabilities (hard and soft technologies, human resources, educated management, information) to meet the rapidly changing needs of the marketplace (speed, flexibility, customers, competitors, suppliers, infrastructure, responsiveness)' (Vaughan 2011). Despite the differences, all definitions of 'agility' emphasize speed and flexibility as the primary attributes of an agile organization. However, both these and in general all the definitions do not explicitly mention that the quality of the work delivered has to address the needs of the changed scenario. This reflects an assumption that the quality of work or the ability to meet the needs is met by the people managed by the organization. It is therefore necessary that all who research agility must ensure that the methods they propose to build agility, not only cover the process of agile organizational behaviour, but also includes the way to effectively ensure that quality deliverables can be built, using the proposed method.

The People CMM refers to workforce competencies as a cluster of knowledge, skills and process abilities that an individual should develop to perform a particular type of work in the organization. Consequently, building competencies that are imposed by change in the work or product specifications is the essence of agility. As long as the context does not change significantly, the need for agility is limited

to improvements and addressing innovations. In such cases, traditional methods of competence building will continue to be relevant. However, in today's work environment change is rapid and time to adapt to changes is small. Hence, quick competency building is extremely critical for software and knowledge based industries. It is necessary to understand the root of competence building and how process steps in delivering work outcomes map to the changed features of the requirement.

11.4 The Traditional Induction Program in the Indian IT Sector

The process of inducting new people into projects in software delivery groups in IT companies, uses a combination of classroom training, working with mentors and hands on working with identified content. Table 11.1 depicts a training plan that is used in the induction program (Ravi 2008). Here, the expectation is that if the associates have the knowledge and are exposed to the nuances of the project context, they will be able to apply and deliver results effectively.

Table 11.1 contains the schedule for developers who will be inducted into the project. The time allocations are illustrative in terms of the division of tasks that should be incorporated into the program.

Other methods to train people to meet changing requirements of the IT sector include problem based learning, project based learning, and role playing (Fink 2003). The key aspect of all of these approaches is that the recall, reuse and internalization of the learning are not addressed. In the performance of any activity, be it software or non-software, the key principle of ensuring that every step is executed to build deliverables must correlate to one or another quality in the deliverable. This is the principle of the process step—product quality that has been emphasized by Deming (Ravi and Narayana 2012).

11.4.1 The Impact of Traditional Induction Training

In a test conducted by the authors, in line with the schedule in Table 11.1, trainees attended the initial training programs and joined projects based on the general initial training provided by the organization. The project absorbed these trainees

Table 11.1 A typical induction training program in Indian IT sector

Project induction training—traditional way	
Description	Training (weeks)
Classroom training	2
On the job training (old cases)	8
Working on live projects with mentors	12
Total	22

Table 11.2 The performance of trainees after undertaking traditional induction training

Performance of trainees in pilots			
Team	Pilot type	No. in pilot	% who executed correctly
P1	Paper test	42	52
P2	Paper test	59	53
P3	Live environment	11	55
P4	On system	56	27
P5	On system	33	29
P6	On system	33	20

post training in order to assess the effectiveness of the training. The trainees were organized into teams and they were given tasks to undertake from providing advice to developing and applying programs.

Various tests were provided to the trainees and the results in Table 11.2 demonstrated that the trainees in general were not able to effectively execute the assigned tasks. In turn this suggests that induction training should be restructured and modified if it is to meet its objectives.

Within the industry the effectiveness of induction training programs and the variation in the uptake of competence is high and the clear alignment of learning achieved to the needs of project teams is poor. In other words, the correlation of the performance in induction training and the actual performance of the trainees is visibly low. Table 11.2 shows this variation in the 'ability to do' among a number of teams after training (Ravi and Alladi 2013).

11.4.2 Understanding the Failure of the Induction Training Programs

In the above process the process of how an individual learns, understands and relates to outcomes expected is not visible. The 'how' in the problem requires attention. The focus in these methods is on the business process of providing content and bringing together named experts in contact with learners thereby providing opportunity and exposing learners to the tacit knowledge of experts. The actual learning, recall, application ability, perspective and reuse are expected to happen from this contact. The validation of the competence that the trainees have acquired is not explicitly carried out. The performance of individual associates in actual projects after the induction schedule is complete and the trainees are inducted into the project is observed. Thus, induction process objectives are subjectively achieved. The variety in the learning curves, the complexity in the deliverables and the contexts, the variation in the level of knowledge and familiarity in the wide group of learners in not factored into the process. As result the achievement of agility in people competencies with currently known methods, is therefore not sustainable.

People competencies must be constantly upgraded or revised in organizations, since markets and other factors change constantly. The importance of having systems and methods to make all aspects of the organization agile is critical. Most traditional methods of adapting people to newer competency needs are based on imparting knowledge related to new scenarios. These approaches have a number of shortcomings and hence result in a high 'effective latency' in people being able to deliver in the changed scenario. 'Effective latency' is the time taken for the person to acquire competence in the changed scenario and to build quality deliverables in the given scenario. This is in contrast to the traditional idea that agility is the time taken to acquire the knowledge needed by the changed scenario. Current methods focus on imparting the knowledge relevant to the changed context. Hence, the time to acquire this knowledge and to effectively apply it to the new context is important. Consequently, there is a need for methods that ensure low effective latency and high agility in people competency development. In this chapter, the required 'ability to deliver' is developed and explained in the new or changed scenarios in software industries, ensuring agility in the ability to perform.

11.5 Research

Localizing Change—To Ensure Agility
It is traditionally understood in mature industries such as manufacturing that change effects the Operating Procedures of the changed product or work. It is also understood that the operating procedure is written following Deming's process step—product quality correlation (Ravi 2008).

These imply that any specification or modification of a software, product or service will cause a change or modification in the Operating procedures. Extending the idea that competence is simply the ability to execute the process steps or operating procedure, competency building can be made agile by bringing about a change in the ability of the person to execute changes in the operating procedure. This is the assumption that the person who needs agility in acquiring competence, will have the competence in dealing with the basic operating procedure.

11.6 Comparability of Software Products and Services

Most applications are similar and with small variations is an ensemble of various smaller reusable functional components. The total set of components that constitute an application will be large, but the unique list is a reasonable fraction of this larger subset. Variations in contexts, specifics and configuration items are there, but for these, the applications are simple to assemble. For instance, a Login functionality in software applications has UI components such as HTML Pages, JSP

tags or Strut tags, JavaScript Validations. Other aspects such as session management, exception handling, data querying and so on are common functions and are required in every web app. Hence, if competence building is dependent on such a breakdown, then the task is simpler. If it is know how the original project was executed and what competencies were included in it, the same basis can be used for changed competencies from one project to another. This concept has been discussed in detail by Ravi and Narayana (2012). They indicate that the suggested way to build competence is to understand and interpret the context in terms of process steps and thereby make competence a sustainable, and deliverable indexed capability. This approach is developed in the following sections.

11.6.1 The Proposed Training Model

Competence is a cluster of knowledge, process ability and skill and these are needed to build competency in ourselves to deliver the required output. The authors have (Ravi and Narayana 2012) detailed the model that is useful in developing assured competencies to build anything that is required. Every person who acquires competence must know the process steps that must be executed to realize the requirements of the application or deliverable. In simple words, process ability is knowing what process steps must be executed to realize requirements. These process steps are the steps that are defined in the operating procedure for developing the product or deliverable.

11.6.2 Operating Procedures

The operating procedures are the starting point. The operating procedure is really 'Standard Operating Procedures (SOP)' which are a detailed explanation of how (what steps must be used for) a policy/requirement is to be implemented. The details in an SOP standardize the process and provide step-by-step instructions of how to perform the task in a consistent manner.

MA-041.052002: Computer Worm Incident Handling Standard Operating Procedure
Original Issue Date: 02 May, 2002
(*Source* My CERT: Malaysia Computer Emergency Response Team)

MyCERT received a growing number of computer worms incidents reported primarily since August 2001. There are more organizations that are spending endless hours repeating processes that are non-effective in completely eradicating the worm within the network due to uncoordinated effort within the organization.

Due to the nature of email-borne worms and/or internet-borne worms, which replicate via more than one means of propagation, i.e. via email, unauthenticated folder sharing and network scanning. MyCERT provides the procedure for

handling mass worm infection incidents especially within medium to large organization distributed networks. Most organizations in which the network is physically distributed in multilevel facilities, worm problems are difficult to eradicate, if the problem is not dealt with in a coordinated manner.

MyCERT suggests that organizations that are facing these problems to form an operation centre. The Worm Incident Handling Standard Operating Procedure is as depicted in Fig. 11.1.

Standard operating procedures (SOP) upfront are not a common practice in the software or IT industry. This industry follows guidelines when needed, but the operating procedure for the routine work is largely subjectively understood and followed, but rarely documented. The advantage of such an SOP is that it enables quality, reuse and all risks are well handled by defining the way the work is done or programs are done.

Competency, as stated earlier is being able to execute the operating procedure correctly and in time. In the approach to competence building known as 'Vegam', the competencies needed are identified using a product breakdown structure and the operating procedure that must be used to build each of these requirement slices or competencies are also identified. When it is possible to execute the steps correctly and do it in time, one is competent to execute the requirement.

11.7 Leveraging Operating Procedures to Build Agility in Competency Building

When operating procedures are written for developing a product or a service, and when it is known that a different product or service has to be built, another operating procedure has to be written since the qualities, outcomes and productivity needed will be different for both. If the products are similar and both are web applications, then the extent of change will be small. The differences will then be visible in the operating procedure and on studying them, it will be possible to identify the differences and build competence to execute changes. The process will be fairly simple. However, the software industry does not build SOPs and relies on individual knowledge and content based training to handle the new competencies that are needed.

However, in view of the non-availability of the SOPs, the Vegam method of competency building uses the product breakdown structure to identify the changed competencies needed for the new role. Vegam, using this approach, defines sets of reusable steps and ask the trainees to execute the steps. The reusable steps are in the form of competency assets (CAs) and they can be executed to build the needed competencies. They mainly contain process steps, and a measuring framework to check if the person has actually gained the necessary competence.

In case of a change scenario, operating procedures for both the deliverables are compared and the competency needed for both are identified independently.

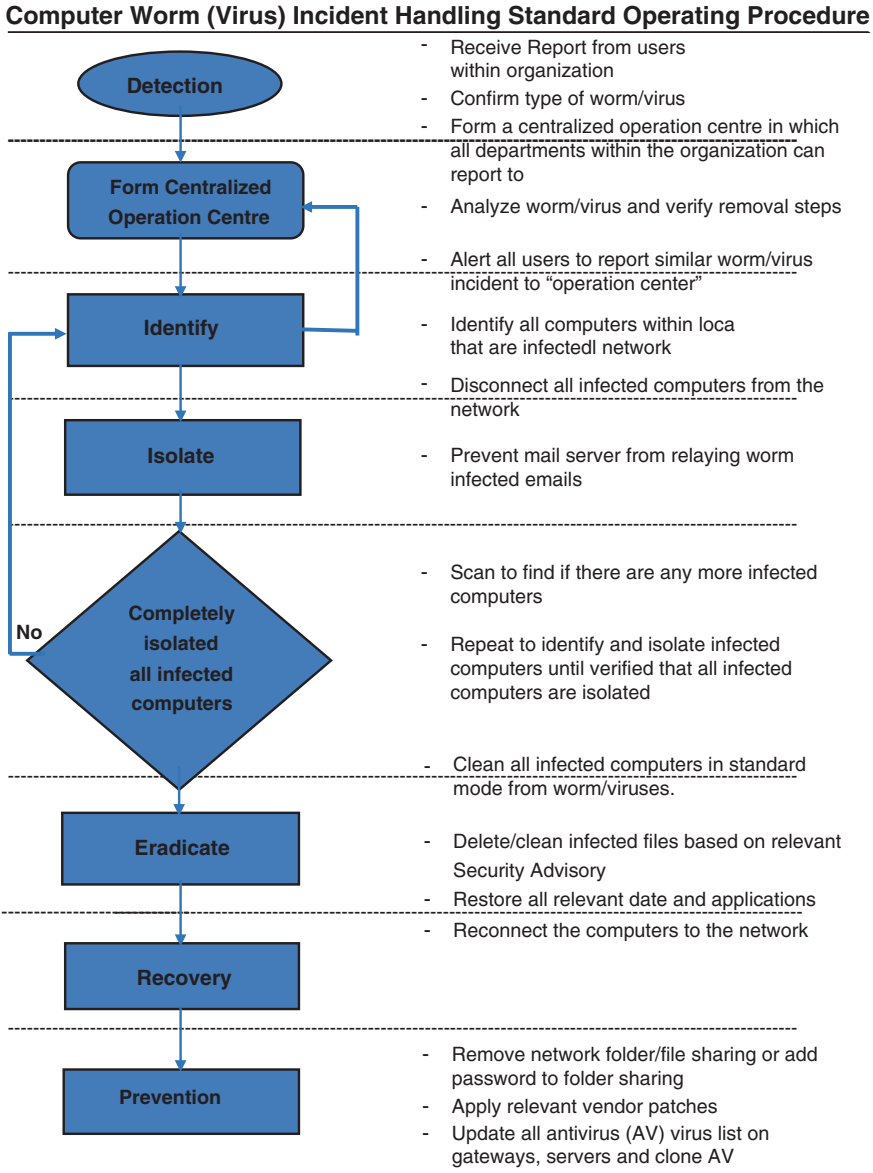

Fig. 11.1 A typical operating procedure. *Source* My CERT: Malaysia computer emergency response team

The differing competencies are the ones which need to be developed. Figure 11.1 depicts how change is localized in the SOPs, and only a few slices of process steps need a change. This way competency development can be done quickly for changed roles.

11.8 Vegam as an Approach to Agile Competency Development

Vegam (Ravi and Narayana 2012) provides the structure for a Product identifying competencies. Figure 11.2 a snapshot of the offering competency map (OCM). A competency menu can be extracted out this OCM and the needed competencies for a role. The changed competencies in the new project will be identified and the needed competencies are then organized to meet the change.

11.9 Pilot Analysis

The authors have used this pilot approach in a number of project scenarios. One such instance is mentioned here for validating the model and its efficacy. The Pilot shows the case of a team of people trained in basic Java using the Vegam approach, moving to a new project. The competency change was done and the benefits realized have been depicted here.

Offering Competency Mapping (OCM) for Basic Web Applications

Offering	Sub-Domain-1	Sub-Domain-2	Re-usable techniques			
			GUI	Action Handler / DD	Middle Tier	Back-End
Development	Web-App Development	Login and Security	Create HTML Page	Session Management	Password encryption / decryption	Query to validate login
			JavaScript validations	Service Integration	Database connection Management	Dynamic menu generated from database
			JSP Tags	Redirection to appropriate view	Exception handling	
			Struts tags in JSP	Exception handling	Connection Leakage	
			JSTL / Core tag libraries	Filters		
			Post method	Role declaration in DD		
				URL Patterns in DD		
		Create / Save new data	Create HTML Page	Capture request parameters and populate DTO as required	Database connection Management	Query / Stored procedure to insert data
			JavaScript validation	Reading property files	Exception handling	Throw exception in case of duplicate records
			JSP Tags	Session Management	Keeping database records in VO.	Dynamic form fields fetching from database
				Service integration	Connection leakage	
				Exception handling		
				Redirection to appropriate view		

Fig. 11.2 The offering competency map or identifying competencies needed by the changed scenario

Table 11.3 The list of competencies

Competencies possessed	New competencies needed
Connection leakage	Build SOAP-based Web Service and mapping data objects
Create HTML page	Capture selected products and populate list be sent to service. Map the response from service call, exception handling, redirecting to appropriate view
Database connection management	Map the response from the service call
Dynamic form fields fetching from database	Redirecting to the appropriate view
Exception handling	Create a details page to show complete product details
JavaScript validations	Create an HTML page
JSP Tags	Create a JSP page to see available products
JSTL/Core tag libraries	Database connection management
Keeping database records in VO.	DB configuration (Spring)
Post method	Dynamic form fields fetching
Query/stored procedure to insert data	Exception handling
Redirection to appropriate view	Integrate with DAO Layer
Service integration	JavaScript validations
Session management	JSP Tags
Struts tags in JSP	JSTL/Core tag libraries
Throw exception in case of duplicate records	Keeping database records in VO.
	Post method
	Product details, read properties files (product information)
	Query/stored procedure to insert data
	Redirection to appropriate view
	Service input object validations
	Service integration
	Session management
	Session management, sending the product details as input to service and getting response
	Struts tags in JSP
	Throw exception in case of duplicate records
16 Competencies	26 competencies

Table 11.3 depicts the competencies needed by a Java/J2EE trained trainee who has to work on a project that uses SOAP-based Web services. The competencies he/she possesses are those of simple J2EE and the transition is huge. The associate participated in knowledge training sessions in order to learn advanced SOAP and Web Services. This approach that has been discussed and depicts that out of 26 competencies he already knows 16 and some have been stated repeatedly. One

Table 11.4 Impact of using Vegam over traditional approach

Impact of using Vegam over traditional methods		
Induction time	Vegam-based approach (No. of days)	Traditional approach (No. of days)
Classroom training	15	15
Mentoring support		15
Hands on work		15
Vegam-based competency building (@2 days per competency for 26 – 16 − 3 = 7 competencies)	14	
Net benefit	29	45

competency is stated three times in the competency menus. Hence, the associate has to work on seven competencies only on the Vegam method.

Table 11.3 depicts the list of competencies needed by the shift of trainees.

11.10 Effective Agility

Improved agility arises as a result of the reduced induction time. This is largely due to the competency identification as a section of the operating procedure. In this way, the number of competencies that the person has when he completed Java competency using the Vegam approach were 16 and the new role needed 26. Of these three additional competencies were repeated. This saved time by approximately 30 %. The details have been provided in Table 11.4.

11.11 Conclusion

The understanding of competency and the focus on knowledge are the key factors for inducting people into projects leading to long induction and learning periods. Not understanding the significance of the operating procedure in the development of products and systems has lead to lack of effectiveness and resulted in a long time being needed for effective induction. This research has enabled identifying competencies that are derived from the operating procedure and resulted in training and development process being developed that has been effective in its deployment for over 1200 trainees. The success rate has been encouraging and it is now being planned to extend this to softer competencies and thus make the induction process even more effective.

References

Curtis, B., Hefley, W., & Miller, S. (2009). *The people capability maturity ver 2.2*. Carnegie Mellon, 2001, Software Engineering Institute, SEI (Glossary of Terms).

Fink, D. L. (2003). *Creating significant learning experiences* (p. 4) San Francisco, CA 94103-1741: Jossey-Bass, A Wiley Imprint. www.josseybass.com

Ravi, S. P. (2008, January 17–19). Ensuring associate competence to build quality in deliverables. In *Publication of the Fourth Technical Architects Conference TACTiCS*. Trivandrum, India.

Ravi, S. P., & Narayana, M. G. P. L. (2012, July 30th–August 1st). Developing agile teams for project execution—A cybernetic approach, presented at *12th Global Flexible Systems Management Annual Conference GLOGIFT 2012*. Austria: University of Vienna.

Ravi S. P., & Alladi A. (2013, July 2–5). *Methodology to bridge the gaps between engineering education and the industry requirements*. Zagreb, Croatia: Eurocon.

Reddypogu, J., Ravi, S. P., & Naresh S. K. (2013, August 26–29). *Framework to reduce cycle time for enhancing productivity in associates*. Bali, Indonesia: TALE.

Vaughan, M. (2011). Three ways to increase organizational agility. http://vaughanmerlyn.com/2011/01/11/three-ways-to-increase-it-organizational-agility/