ISSN 2278-7763

Fusion Process Model Implementation Case Studies

Rupinder Kaur, Dr. Jyotsna Sengupta

Punjabi University, Patiala,
Punjab, India.
Email: rupadeo@gmail.com

ABSTRACT

In this paper we have discussed, three case studies. The first one is applied at Web Shrub Solutions, a software development organization, second is applied at web based job portal (stepintojob.com) for leading Indian firm and the third is web design and development for SCL limited, to observe the results of Fusion Process Model. Fusion Process Model follows component driven approach; it applies 3C Model to generalize the process of solving the problem in each phase, which provides firm control over the software development process.

Keywords : Fusion process Model, 3C Model, Component driven approach

1 INTRODUCTION

Developing and maintaining software systems involves a variety of highly interrelated activities [1]. In order to manage these structured set of activities various models have been developed over the years with varying degree of success. These include Waterfall model, Iterative development, Prototyping, Spiral model and RAD etc. Each product can pass through different states, depending on the specific circumstances of each project and hence, there are different development models.

The overall success in the development of software is still not achieved because each software development process or model consider only one or few concerns and specify a phase wise abstraction for the development, but no definite approach or model is specified for the phases of software process model. Various reports exhibit the failure of software products [2]. In current software engineering practices, ever changing requirements during the development process for large software development is still not managed by software process models. The solution space analysis concept of software engineering is very effective, but this concept is not completely integrated to software development yet. Alternative management, a technique which is used in mature engineering disciplines is not explicitly followed in software engineering discipline. The software development models till date follow fixed or iterative design and development approach. There is no scope for dynamic testing in software development process.

To make the software development effective and reliable, a new approach is required. Fusion process model is based on component driven development approach, which is different from component base software development. In Fusion process model, each component implements a problem solving model. It includes the explicit processes for technically analyzing the problem, solution space analysis, alternative management, dynamic design specification and development and scope for dynamic testing. In this paper, we present the new software process model which will address all the concerns and consider each phase of software development as software development process and provide an effective model for software development phases.

The remainder of the paper is organized as follows: In Section 2 we have discussed Fusion Process Model, Section 3 we have analyzed three case studies. In Section 4 the conclusion is done.

2 Fusion Process Model

Fusion is component driven software process model, where each phase implements a problem solving model [4].3C Model is problem solving model that enable generalizing the software development process in which a problem specification is transformed to a solution by decomposing the problem into sub-problems that are independently solved and integrated into an overall solution [3].

2.1 3C Model

3C Model consists of multiple cycles; each cycle in 3C-Model corresponds to a trans-formation from one state to another, consisting of a problem specification state and a design state. The problem specification state defines the set of problems that still needs to be solved. The design state represents the tentative design solution that has been lastly defined. Initially, the design state is empty and the problem specification state includes the initial requirements. After each state transformation, a sub-problem is solved. In addition a new sub-problem may be added to the problem specification state. Each transformation process involves an evaluation step whereby it is evaluated whether the design solutions so far (design state) are consistent with the initial requirements and if there are any additional requirements identified during the evaluation. The 3C-Model in development process divide the problem solving approach of each phase in three fundamental parts: Capture, Control, Context and Domain engineering, as shown in Figure 1.

ISSN 2278-7763

2.1.1 Capture

The real problem is captured or a real problem is defined, which arise due to client requirements in this part. It comprises of five concepts: Need, Problem Description, Solution Domain Knowledge, Alternative, Solution Description and Artifact.

- **Need** represents an unsatisfied situation existing in the context (environment). The function Input represents the cause of a need.
- **Problem Description** represents the description of the problem. The function Conceive is the process of understanding what the need is and expressing it in terms of the concept Problem Description.
- **Solution Domain Knowledge** represents the background information that is used to solve the problem. The function Search represents the process of finding the relevant background information that corresponds to the problem.
- **Alternative** represents the possible alternative solutions. The function Generate serves for the generation of different alternatives from the solution domain knowledge. After alternatives have been generated, the problem description can be refined using the function Refine.
- **Solution Description** represents a feasible solution for the given problem. It uses the relevant background information to provide a solution description that conforms to the problem description. The function Detail is used to detail the description of a selected alternative.
- **Artifact** represents the solution for the given need. The function Implement maps the solution description to an artifact. The function Output represents the deli-very and impact of the concept artifact to the context. The function Initiate represents the cause of a new need because of the produced artifact.

2.1.2 Control

The development process in software engineering starts with the need, while the goal is to arrive at an artifact by applying a sequence of actions. Since this may be a complex process, the concepts and functions that are applied are usually controlled. This is represented by the Control part in the model. The controller observes variables from the system, evaluates this against the criteria and constraints, produces the difference, and performs some control actions to meet the criteria.

- **(Mathematical) Model** represents a description of the concept Alternative. The function Analyze represents the process of analyzing the alternative.

- **(Quality) Criteria** represent the relevant criteria that need to be met for the final artifact. The function Evaluate assesses the alternative with respect to (Quality) Criteria and Constraints.

- **Constraints** represent the possible constraints either from the context or as described in the problem statement.

- **Heuristics/Optimization Techniques** represents the information for finding the necessary actions to meet the criteria and constraints. The function Select/Optimize selects the right alternative or optimizes a given alternative to meet the criteria and the constraints.

2.1.3 Context

Both the control and the problem-solving activities take place in a particular context. Context can be expressed as the environment in which software development takes place including a broad set of external constraints that influence the final solution and the approach to the solution. Constraints are the rules, requirements, relations, conventions, and principles that define the context of software engineering, that is, anything, which limits the final solution. Since constraints rule out alternative de- sign solutions directing engineers into taking action on what is doable and feasible.

The context also defines the need, which is illustrated by a directed arrow from the context to the need concept. Apparently, the context may be very wide and include different aspects like the engineer‘s experience and profession, culture, history, and environment.

2.1.4 Domain Engineering

The phased model use domain analysis to identify do- mains, bounding them and discovering commonalities and variability‘s among the systems. This information is captured in models that are used in the domain implementation phase to create artifacts such as reusable components, a domain-specific programming language, or application generators that can be used to build new systems in the domain. A key idea in systematic software reuse is the domain, a software area that contains systems sharing commonalities.

ISSN 2278-7763

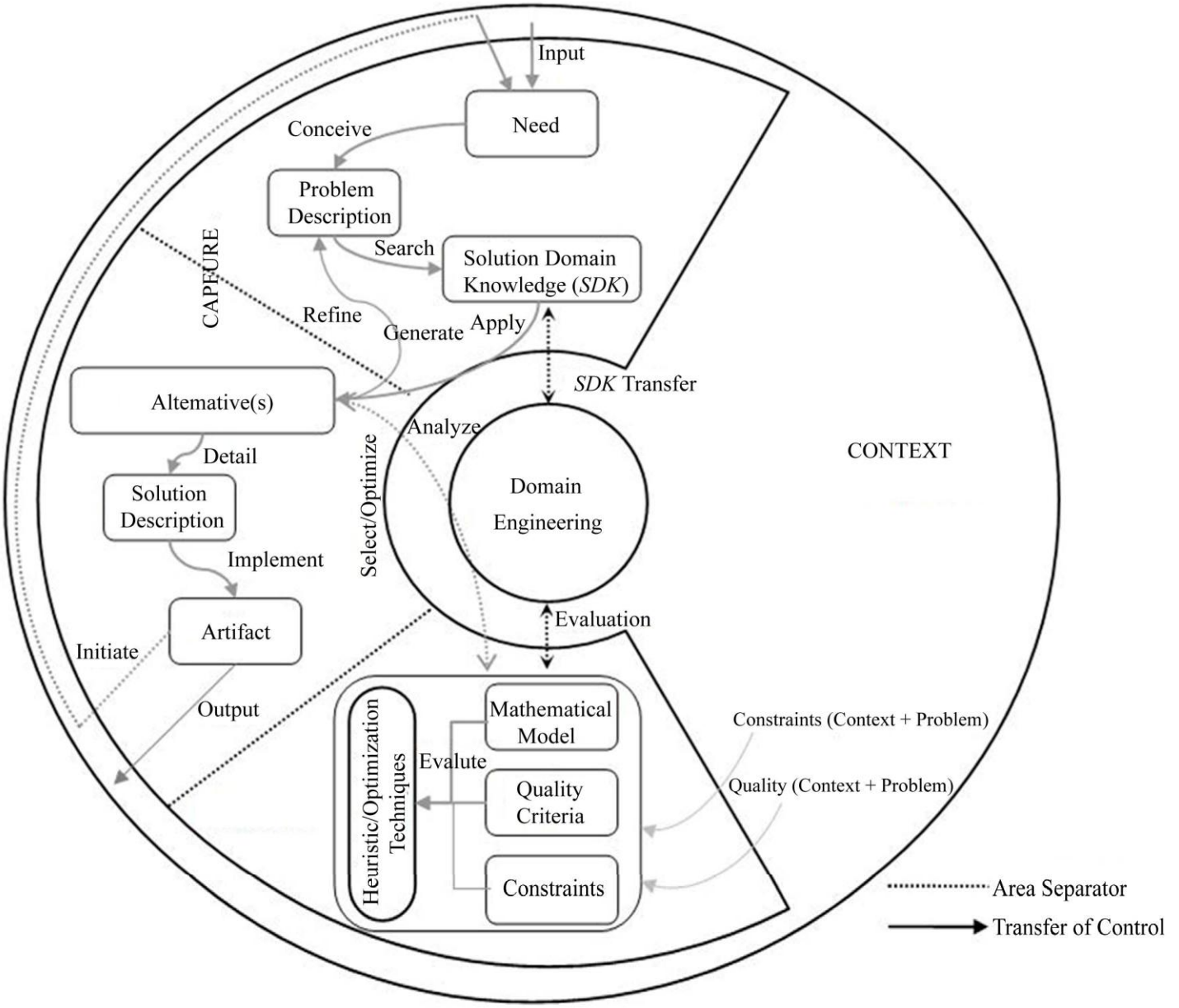

Figure 1. 3C-Model

2.2. Five Fundamental Phases

Fusion is component driven software process model, consist of five fundamental phases and one fusion process controller to control and coordinate the overall development process, as shown in Figure 2. The phases address what is to be built, how it will be built, building it and making it high quality.

2.2.1. Project Preparation

The project preparation phase provides the initial planning and preparation for software development project. Extracting the requirements of a desired software product is the first task in creating it. This process is called requirements elicitation. After requirements elicitation, client requirements are mapped to technical problems in the technical problem analysis process. The problem analysis process consists of the following steps: Generalize the Requirements, Identify the Sub-Problems, Specify the Sub-Problems and Prioritize the Sub-Problems.

ISSN 2278-7763

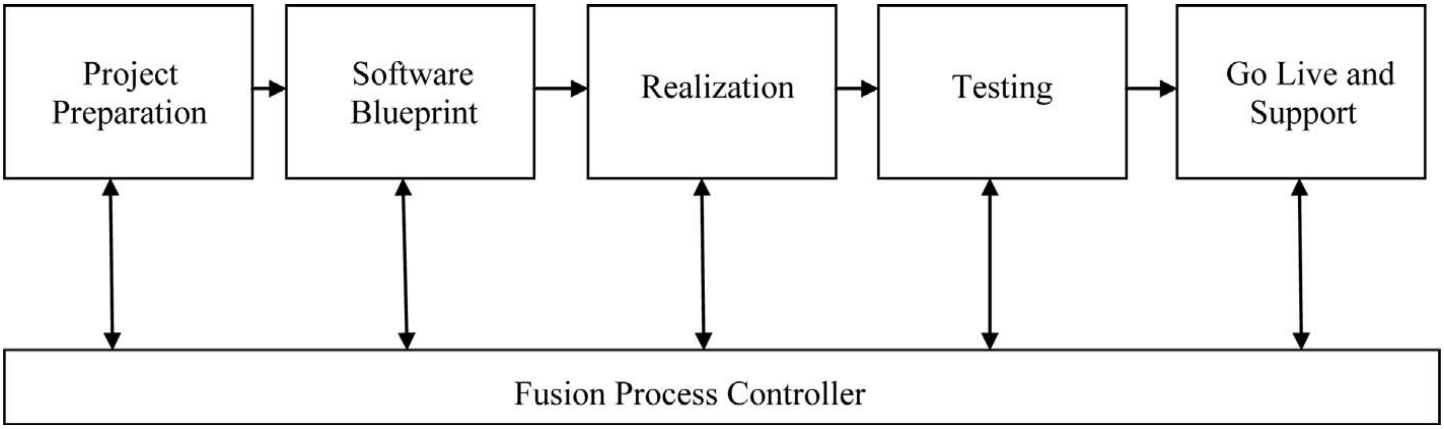

Figure 2. Fusion Process Model.

2.2.2. Software Blueprint

Architecture is established during the design phase. This phase starts with the inputs delivered by the initial phase and maps the requirements into architecture. The architecture defines the components, their interfaces and behaviors. The Solution Domain Analysis process applied in software design phase aims to provide a solution domain model that will be utilized to extract the architecture design solution. It consists of the following activities:

1) Identify and prioritize the solution domains for each sub-problem;
2) Identify and prioritize knowledge sources for each solution domain;
3) Extract solution domain concepts from solution do- main knowledge;
4) Structure the solution domain concepts;
5) Refine the solution domain concepts;
6) Alternative design space analysis.

- Define the alternatives for each concept.
- Describe the constraints.

2.2.3. Realization

The purpose of realization phase is to develop a software system for requirements based on the software design. The team builds the components either from scratch or by composition. Given the architecture document from the design phase and the requirement document from the analysis phase, the team builds exactly what has been requested, though there is still room for innovation and flexibility.

2.2.4. Testing

Quality of software product is very important while developing it. In many software engineering methodologies, the testing phase is a separate phase which is performed by a different team after the implementation is complete. There is merit in this approach, it is hard to see one's own mistakes, and a fresh eye can discover obvious errors much faster than the person who has read and re-read the material many times. Unfortunately, delegating testing to another team leads to a slack attitude regarding the quality of the implementation team. Alternatively, another approach is to delegate testing in the whole organization. If the teams are to be known as craftsmen, then the teams should be responsible for establishing high quality across all phases.

2.2.5. Go Live and Support

The purpose of the Go Live and Support phase is to cut over to live productive operation and to continually support and improve live operations. There are two distinct periods of this phase: Project End and Continuous Improvement.

ISSN 2278-7763

2.2.6. Fusion Process Controller

The controller part is not a phase in the process model, but it is an integral part of fusion process model. The controller part helps to achieve the component driven approach by listing the details of components which are added due to requirement changes or because of new requirements. By implementing Fusion Process Controller the current software development process will not be affected by changes required due to new requirements or modifications. The affected components can be taken care separately till these components matches with the current development process.

3. Case Studies

3.1 Case Study 1

The case study aimed at investigating more practical aspect of fusion process model by means of implementation in commercial software company. Web Shrub Solutions Ltd. is an Indian software company in Noida. It is a technology-enabled product and services company that delivers innovative solutions to meet the demands of today's digital world. The case study involved, developing a web based collaborative solution to allow managing construction activities, migration of data from client's legacy software to new system, the development of integration tools for integrating client's sales and accounting systems with construction management solution.

Client was using legacy software with very limited capabilities for managing their construction activities related to production home building, which was inefficient and lacking in capabilities. They wanted to build a new robust and scalable software system that included all of the capabilities, critical for their business. Moreover, they were dependent on the manual efforts for major set of activities. The development team decides to build complete web based solution to allow managing construction activities. Our approach has following primary elements:

1. Software development process is a structured set of activities, which play a fundamental role in the development of large-scale and complex software system. Managing each of these activities is a necessary and sufficient condition for project success.

2. 3C-Model, which is used to manage the software development model activities.

3. The development time, cost, quality and changing requirement plays an important role for successful software development, and how fusion process model manage this with component driven development approach.

3.1.1 Project objectives

Create the artifacts necessary to establish a successful life-cycle architecture based on component driven development approach for automation of construction process, and provide the scope for future enhancement. These artifacts were:

1. An operational concept definition.
2. A system requirements definition.
3. A system and software architecture definition.
4. Artifacts after completion of each process of 3C-Model.
5. A life-cycle plan and controller to keep track of integration of components.
6. Re-usable components tracking.
7. Evaluation of components after applying heuristic & optimization, mathematical models, quality criteria, constraints and alternatives.

3.1.2 Team structure

Each of the twelve team members was responsible for development of project artifacts. In addition, the team member responsible for architect and design also served as the project lead. The project lead can serve as controller or delegate the responsibility to one of the team members. The team member responsible for the feasibility rationale, served as project manager with the following primary responsibilities:

1. Ensure consistency among the team members' artifacts (and document this in the rationale).

2. Lead the team's development of plans for achieving the project results and ensures that project performance tracks the plans.

3.1.3 Project approach

The team was divided into sub-groups and each group developed the artifacts in each phase implementing 3C-model concurrently. Controller tracked all the integration and reusable components. The team was responsible for evaluation of components after applying heuristic & optimization, mathematical models, quality criteria, constraints and alternatives.

3.1.4 User Negotiations

Each group worked with a user representative for functionality validation and to provide the scope for requirement change, during the validation of each component. This process provides the scope to identify the issues at earlier stages, help to win the customer confidence and scope for requirement changes.

.

ISSN 2278-7763

There were three stakeholders:
1. **Developer:** The architecture and prototype team members represented the developer concerns, such as the use of familiar packages, stability of requirements, availability of support tools and technically challenging approaches.

2. **Customer:** The plan and rationale team members represented the customer concerns, such as the need to develop an IOC in one, limited budgets for support tools and low-risk technical approaches.

3. **User*:*** The operational concept and requirements team members worked with their designated user-community representative to represent user concerns, such as particular multimedia access features, fast response time, friendly user interface, high reliability and flexibility of requirements.

3.1.5 Phases

3.1.5.1 Phase 1 (Project Preparation)

This phase provides the initial planning and preparation for software development project. Extracting the requirements of a desired software product is the first task in creating it. This process is called requirements elicitation. After requirements elicitation, client requirements are mapped to technical problems in the technical problem analysis process.

The problem analysis process mainly consists of the following steps:
1. **Generalize the Requirements:** whereby the requirements are abstracted and generalized.

2. **Identify the Sub-Problems:** whereby technical problems are identified from the generalized requirements.

3. **Specify the Sub-Problems:** whereby the overall technical problem is decomposed into sub-problems.

4. **Prioritize the Sub-Problems:** whereby the identified technical problems are prioritized before they are processed.

Table 1 show the initial component identified and service required.

Later, during the refinement process of 3C-Model, new requirements were added to the project, the client require the mobile based solution. Client required their users to be able to manage their construction activities while on construction sites where they do not have access to their personal computers.

Table1: Component and Service

Required Components (Sub-Problems Identified)	Services Required
• Security role based user login and management. • Recording, viewing and managing construction projects. • Work assignment and management. • Issue and checklist management. • Data import and export. • Reporting for office users, help desk, project managers, and upper management. • Managing vendor payments. • Viewing archived and historical data. • Email notifications. • Integration with sales and accounting systems.	▪ Requirements analysis and solution design. ▪ Java Application development. ▪ Application testing. ▪ Server setup and application deployment. ▪ Database and server administration. ▪ Implementation and training. ▪ Project co-ordination and management.

ISSN 2278-7763

For this mobile application development project, built a custom mobile application to allow users to track and manage their construction activities using their Windows Mobile, Blackberry, and iPhone devices. Mobile Application was integrated with client's existing web application. Application increase efficiency in managing construction schedules, reduced timeline for completing construction. Table 2 exhibit new components which are added due to new requirements and service required.

3.1.5.2 Phase 2 (Software Blueprint)

Architecture is established during the design phase. This phase starts with the inputs delivered by the initial phase and maps the requirements into architecture. The architecture defines the components, their interfaces and behaviors. The Solution Domain Analysis process applied in software design phase aims to provide a solution domain model that will be utilized to extract the architecture design solution.
It consists of the following activities:
5.2.1 Identify and prioritize the solution domains for each sub-problem.
5.2.2 Identify and prioritize knowledge sources for each solution domain.
5.2.3 Extract solution domain concepts from solution domain knowledge.
5.2.4 Structure the solution domain concepts.
5.2.5 Refine the solution domain concepts.
5.2.6 Alternative design space analysis:

- Define the alternatives for each concept.
- Describe the constraints.

Table 3 show technology used in the project and new technology added due to change in client requirement.

The design team got the inputs from the phase 1 during the design process, as per the customer requirement a new framework was required which was integrated with current web based solution. Client required their users to be able to manage their construction activities while on construction sites where they do not have access to their personal computers. After identifying the top-level conceptual architecture, the focus was on each sub-problem and follows the same process. The refinement may be necessary if the architectural concepts have a complex structure themselves and this structure was of importance for the eventual system. The ordering of the refinement process was determined by the ordering of the problems with respect to their previously determined priorities. Architectural concepts that represent problems with higher priorities were handled first and in the similar manner the refinement of the architectural concepts was done. The control phase of 3C-Model evaluates the component during analysis and design process based on the following categories:

- Function
- Performance
- Usability
- Reliability, Availability, Serviceability
- Localization
- Portability
- Maintainability
- Security
- Testability
- Extendibility

Functional requirements define what data the system must produce and what transformations of that data; the system must be able to do. Performance requirements describe how fast it must do the transformations, how many transformations it must do and any limitations on the amount of utilization of the agents used to support the transformation (e.g., amount of machine time, amount of disk space). Usability requirements describe the ergonomics of the system (e.g., ease of correctly interpreting the information on a screen). Reliability defines the degree of accuracy required in the transforms. Such as in billing this would be 100% and in weather forecasting it could be plus or minus 5% for a short term forecast. Availability defines the amount of time the system is actually up during the time periods it is supposed to be up. This is usually defined as a percentage, qualified with standard deviations. Mean time to failure, by type of failure, further defines system availability. Serviceability addresses how quickly the system can be corrected when it is discovered to be unreliable or unavailable. This might be expressed as the mean time to fix. Mean time to fix is usually qualified by the type and severity of the failures.

Localization describes the ability to adapt the application to different languages, character sets, and cultures to support international users. Portability describes the need to be able to quickly adapt the application to run on different technology. Maintainability describes the need for people to be able to quickly and reliably identify where changes must be made to the system. Security describes the system ability to restrict who can do what to what and when. Extendibility describes the system's ability to absorb major modifications to changes in any of the above requirements, while minimizing the impact to the rest of the system. This is usually described in terms of change scenarios accompanied with the probability the change will be needed and the probable time frame in which it will occur.

ISSN 2278-7763

Table 2: New Components and Service Required

New Components Added	Service Required
• Secured user login • Viewing and managing construction projects • Issue and checklist management • Task assignment and management • Integration with web application	• Requirements analysis and solution design • Application development • Application testing • Application deployment • Project co-ordination and management

Table 3: Technology Used and Technology

Added Technology Used	Technology Added (As per Requirement Change)
□ Java / J2EE □ MS SQL Server □ Web Sphere □ Hibernate □ Eclipse □ Reporting Services □ XML Web Services	□ Mobile Platforms: Windows Mobile, Blackberry, iPhone □ Web Technologies: Java/J2EE, SQL Server, Web Services, SOA

Table 4 describes various component attributes, before these components moved to development phase for further processing. Later, these independent components integrated as part of complete solution.

3.1.5.3 Phase 3 (Realization)

Development started once client approved the design documents and development of components started based on the priority of component. Initially, teams started working on two independent base components. As the client was extensively involved during design phase, each small level detail was incorporated before approval. Now the team has clear vision for development. These components were immediately approved by client after completion. At this point, the base was ready to build a complete software system on it. The team started work on independent components depending upon the defined priority and other attributes. The client was involved in the development and few minor modification suggested by client, which were immediately applied. Each component was implemented using 3C-Model development process. All the customer suggestion and evaluation results were noted down for predecessor components. The team moves the component immediately to testing phase after completion of development. The testing results of each component were used as guidelines for the development and evaluation of other components. Numerous studies have shown that the earlier you test, the cheaper the costs. The cost difference between finding an error early in the requirements definition versus once it is in production have been found to be as much as 270 to 1500 times as expensive. Prior to system release, the issue is scrap and rework. The later an error is found, the greater the impact on work already completed. It is like finding out after a building is constructed that the plumbing has been omitted. If someone had noticed this omission early in the blue-print stage, the costs would have been minimal. The changes in the independent components have minimal impact on the complete development plan, and serve as guidelines for predecessor components.

ISSN 2278-7763

Given the architecture document from the design phase, and the requirement document from the analysis phase, the team builds exactly what has been requested, though there is still room for innovation and flexibility. If any new requirement was raised by client, independent of any other component, then the new component development process starts from first phase. Once the development process finished, the testing phase started which included individual component testing and regression testing.

3.1.5.4 Phase 4 (Testing)

The complete software system design was based on component driven development approach. Each component directly moved to testing phase after the completion of development phase. There were various similar kinds of components in application; the test case used for one component can be used with little or no modification for other components, which saved a lot of time required to build test cases. The test cases for reusable components were already available with company, which were used to test various scenarios of application. Finally, the integration testing was completed to deliver the complete software solution.

3.1.5.6 Phase 5 (Go Live and Support)

The purpose of the Go Live and Support'phase is to cut over to live productive operation and to continuously support and improve live operations based on project agreements finalized with client. Finally the software deployed on customer landscape within projected cost and time. Due to component driven approach and customer involvement in each component, the software solution had all the functionality as required.

3.1.5.6.1 Project End

During the time when the system is first live, all issues and problems are resolved, transition to the production support team is finalized, knowledge transfer is completed, and the project is signed off.

3.1.5.6.2 Continuous Improvement

Now that the project is over, the production support team monitors the system and resolves live business process issues. Proper change management procedures are established, and ongoing end-user training is conducted. Plans are made to continuously review and improve business processes.

3.1.6. Results

The results of the study show that the model has following key strengths:

3.1.6.1 Flexibility

The model is flexible to handle new & changing customer requirements. It provides a scope for continuous requirement changes until final delivery.

3.1.6.2 Reusability

There will be scope for software reuse (using already existing components) and development of components, which can be used at later stages. The model provides explicit way to increase the reusability.

3.1.6.3 Discipline

Each phase of the development process implements a phased model called 3C-Model, to keep track of the artifacts at each stage and developing each component in much disciplined manner. The control monitor handles the successful integration of various phases and components. It helps to monitor quality through proper verification and validation, using predefined qualitative and quantitative techniques.

3.1.6.4 End User Involvement

The involvement of end user helps to win the trust in software solution. This approach provides full scope for end user involvement, but not like iterative development. The model is open for suggestion and requirements from end user.

3.2 Case Study 2

One of the most important resources in the business industry ever is human resources in presence of new technological advancements in fast running world. Universal truth is human power will always in demand to run new technological machines and devices. Employers will always in search of good employees and employees always seek good opportunities.

To fill the gap between employers and employees there are number of people and organizations are working all over the world. Matching right people at right place to work and satisfaction level at both ends is always a big challenge for the business industry. Customized efforts are always in process to help employers and employees to meet right matches for each other.

In today's competitive environment, getting jobs and searching for candidates assumes greater importance. Earlier, the advertisements of jobs were limited to newspapers. But now the emerging economy demands the situation to change. We have various job portals, online employment exchanges, consultancies, company websites etc. In this situation recruitment should be time saving, cost effective and at the same time should search out the qualified candidates. By using the efficient methods and techniques the recruitment process is made easy.

Table 4 is given below: Various Component Attributes

ISSN 2278-7763

COMPONENT	FUNCTION	DATA STORE	DATA FLOW	DATA ELEMENT	USE CASE	EXTERNAL ENTITY/ ACTOR
PROPERTY						
Identification						
Name	X	X	X	X	X	X
Date	X	X	X	X	X	X
Version Number	X	X	X	X	X	X
Aliasing	X	X	X	X	X	X
Description	X	X	X	X	X	X
Cross-Reference						
Parents	X	X	X	X	X	X
Migration	X	X	X	X	X	X
Source/Sinks			X			
Inputs/Outputs	X	X			X	X
Model	X	X	X	X	X	X
Information Sources	X	X	X	X	X	X
Design Axioms	X	X	X		X	
Component Composition						
Census	X	X	X		X	
Organization		X	X			
Syntax/Structure		X	X			
Trigger					X	
Processes					X	
Process Parameters	X				X	
Algorithim	X					
Processing Characteristics						
Rate of Flow			X			
Volume	X	X				X
Access		X				
Periodicity	X					X
Rate of Processing	X					X
Processing Windows	X					X
Maintenance		X				
Security	X	X	X		X	X
Integrity (R.A.S.)	X	X	X		X	X
Priority	X				X	X
Agent Chartacteristics						
Data Connection Char.			X			
Medium			X			
Size			X			
Encoding Method			X			
Connection Requirement			X			
Type			X			
I.D.			X			
Source/Sink			X			
Flow Requirements			X			
% Capacity			X			
R.A.S.			X			
Environment			X			
Data Storage Char.						
Medium		X				
Size		X				
Encoding		X				
Storage		X				
Type		X				
I.D.		X				
Location		X				
Volume		X				
% Capacity		X				
R.A.S.		X				
Environment		X				
Processor Characteristics						
Type	X					
I.D.	X					
Location	X					
Size	X					
% Utilization	X					
Required Features	X					
R.A.S.	X					
Environment	X					
Costs						
Cost per Unit	X	X	X		X	X
Cost for Availability	X	X	X			X
Value						
Replacement Cost	X	X	X			X
Fair Market Value	X	X			X	
Business Value	X	X	X		X	

ISSN 2278-7763

The employers and candidates see things not in terms of what they need, but in terms of best things they want. The client hunted a job portal which is meant for Employers, job seekers, and also which guides the candidates in various areas of job search and related things. Finding and recruiting the best quality candidates seems to get more complicated in the coming days. So the proposed system aimed to provide best class of employment services to job seekers, employers, and recruiters. The system should provide free job search and services. It should be quick, safe and easy to use. The system is meant for the end users -the employer and the candidate. The end users may be registered users or not registered.

The aim of this case study is to describe the development of web based job portal (stepintojob.com) for leading Indian firm; the team developed the complete project following Fusion Process model guidelines.

3.2.1 The Challenge

This project included lots of challenges in its Project Life Cycle, below are few of them:

1. Providing an exclusive search mechanism which fully facilitates a Job Seeker to find a suitable job.

2. More user friendly and advance approach to capture resume and company information facilitating Job seekers and Employer to interact with each other instantly.

3. Giving full control on revenue models (Job Posting, Resume Search & Advertisement) and System should support all currencies.

4. Proving complete control on website theme and admin can choose from a list of available themes.

5. Highly functional CMS system and advanced admin controls to manage website contents.

6. Providing easy process to apply for a suitable Job by new job seekers.

7. Implementing advance advertisement module.

3.2.2 Solution

After discussion with Stepintojobs.com management and development team, it was decided that the job portal development process will follow Fusion Process Model.

We could suggest a better way out for faster recruitment, a job portal which is meant for employers, job seekers and recruiters. For employers looking to recruit the most qualified candidates, it offers modern technology and superior services to simplify your recruitment process. For individuals thinking about new job, new career, or new direction, it helps you explore the possibilities and find the opportunities that are right for you. Whether you are looking for a job or a new member of your staff team, you definitely have the right place. The best way to change your career or get the right people for the jobs is quite easy now.

The system offers Job search by browsing huge list of jobs posted by potential employers; Search employers- know who the top employer in the market is now; it also allows potential employers know about yourself and your valuable skills in a modern and good looking way by the use of free personalized profile, also receive jobs by email regularly. For the employers, advertise job vacancies absolutely free which will allow potential candidates to know about great new opportunities in your company. It allows resume search; browse great number of resumes by simple and defined search.

Thus the recruitment is made faster, easier and the barriers are reduced by advanced searching of the jobs. Employers and candidates are able to explore greater opportunities. The time taken for the recruitment is reduced. The employers can advertise their job vacancies; the candidates can post their resumes, even they give the feedback.

3.2.3 Phase:

3.2.3.1 Project Preparation

To better understand the client requirements development team start discussion with client team.

Based on the client inputs, and based on the information from the solution domain knowledge base initial draft is prepared. As a part of fusion process model development process team was asked to raise questions based on control part of development process and the development team list out key points for the development process:

1. To a job seeker, it provides the feature of Multiple Resume Creation, a Job Crawler which seeks jobs based on the criteria set by the candidate. The candidate can apply online for these selected jobs.

2. The employer can purchase pre-defined packages by admin and create their company profile. Employer can also create company profile in Audio and Video format. The employer can search for resumes as well as post the job requirements online.

3. Employer and job seekers can create Wish-lists for jobs and resumes.

4. We integrated different kind of search facilities to search a Job like Quick search, Advance Search, Search by Distance and Surf for Job.

5. Online recording of Audio and Video resume, Job Seeker can have 3 audio and 3 video resume. We have used audio video recorder and Red5 Flash Server for audio & video streaming.

6. Online chatting bridge integrated between Job Seeker and Employer.

ISSN 2278-7763

7. Fully dynamic revenue models based on the country of operation and selected service and package available. Prices of all revenue models for all countries are defined as base unit only and are finally charged as per base currency defined by admin.
8. Admin can create standard packages for all revenue models as per his needs.

9. Admin can update currency exchange rates based on website's base currency. When user visits the site, prices are displayed as per user locality.

10. Web site supports 5 different themes, Admin can changes website theme to change the looks and feels of the website by just single click.

11. Admin have full control of site from where he can manage everyone's profile and important contents of the website.

12. We implemented One Click Apply concept which allows Job seeker to apply for the Job with minimum information and single step registration process.

13. We have multiple advertisement zones and admin can create different packages for each zone along with its validity.

All these entities were decomposed into final level of decomposition. Along with these entities other supporting requirements were specified in detail like UI design, Reporting mechanism, Internationalization, logging support level. This document was shared with client and updated many times after discussion with client to get understanding of varied industry need, as development team need to make the software system highly generic to support all industry types.

The problem analysis process consists of the following steps:

1.1 Generalize the Requirements: whereby the requirements are abstracted and generalized.
1.2 Identify the Sub-Problems: whereby technical problems are identified from the generalized requirements.
1.3 Specify the Sub-Problems: whereby the overall technical problem is decomposed into sub-problems.
1.4 Prioritize the Sub-Problems: whereby the identified technical problems are prioritized before they are processed.

3.2.3.2 Software Blueprint (Design)

Depending upon the inputs from the phase one, various design decisions were taken after considering various alternatives based on the evaluation of control part in phase model. Team performed software design documentation, including Architecture design, Database design doc and Sequence diagram to understand initial flow.
After basic architecture design decisions, based on the component driven approach the project divided into three logical development components from the portal accessibility purpose:
1. Administrator
2. Candidates
3. Employer
All the above users have different accessibility options. The features for the different users are given below:

Administration Panel – The admin panel allows the administrator to:
1. Handle the management of job seekers as well as employers
2. Create and manage different recruitment ad specifications
3. Manage membership pricing plans and payments received
4. Manage the knowledgebase for job seekers
5. Handle the functionality as well as the general settings of the website

Candidates Panel – This panel allows the job seekers to:
1. Register on the website
2. Edit/update their personal information like experience, location, field, functional area, industry, expected salary etc.
3. Upload their CV's onto the website
4. Search/apply for the vacancies in one click

Employer Panel – Under the employer panel, the employers can:
1. Post and manage job vacancies
2. Search for the potential candidates on the basis of defined criteria
3. View the number of candidates who have applied to their ad
4. Download cvs of candidates who have applied

3.2.3.2.1. Identify and Prioritize the Solution Domains

For the overall problem and each sub-problem, search for the solution domains are prepared that provide the solution abstractions to solve the technical problem. The solution domains for the overall problem are more general than the solution domains for the sub-problems. In addition, each sub-problem may be recursively structured into sub-problems requiring more concrete solution domains.

3.2.3.2.2. Identify and Prioritize Knowledge Sources

Each identified solution domain covers a wide range of solution domain knowledge sources. These knowledge sources may not all be suitable and vary in quality. For distinguishing and validating the solution domain knowledge sources we basically consider the quality factors of objectivity and relevancy. The objectivity quality factor refers to the solution domain knowledge sources itself, and defines the general accep-

ISSN 2278-7763

tance of the knowledge source. The relevancy factor refers to the relevancy of the solution domain knowledge for solving the identified technical problem.

3.2.3.2.3. Extract Solution Domain Concepts from Solution Domain Knowledge

Once the solution domains have been identified and prioritized, the knowledge acquisition from the solution domain sources can be initiated. The solution domain knowledge may include a lot of knowledge that is covered by books, research papers, case studies, reference manuals, existing prototypes/systems etc. Due to the large size of the solution domain knowledge, the knowledge acquisition process can be a labor-intensive activity, so a systematic approach for knowledge acquisition is required.

3.2.3.2.4. Structure the Solution Domain Concept

The identified solution domains concepts are structured using parent-child relationship. Here all the attributes and operations associated with the concept are defined.

3.2.3.2.5. Refinement of Solution Domain Concepts

After identifying the top-level conceptual architecture, the focus is on each sub-problem and follows the same process. The refinement may be necessary if the architectural concepts have a complex structure themselves and this structure is of importance for the eventual system.

The ordering of the refinement process is determined by the ordering of the problems with respect to their previously determined priorities. Architectural concepts that represent problems with higher priorities are handled first and in the similar manner the refinement of the architectural concepts is done.

Finally after various reviews, time, cost and development details shared and finalized with client to start the development process.

3.2.3.3. Software Realization - Development

Now the development team has development project divided into various components, the development of components start based on the priority of component. Each component follows different line of development and shared with client before each cycle to get the client feedback (requirement change / new requirement). Each component is monitored using various control techniques defined by development team to keep track of quality, cost and time. Once the development of any component complete, it immediately moves to testing phase.

Given the architecture document from the design phase and the requirement document from the analysis phase, the team builds exactly what has been requested, though there is still room for innovation and flexibility.

3.2.3.3.1. Alternative Design Space Analysis

The alternative space is define as a set of possible design solutions that can be derived from a given conceptual software architecture. The alternative design space analysis aims to depict this space and consists of the two sub-processes: define the alternatives for each concept and describe the constraints. Let us now explain these sub-processes in more detail.

3.2.3.3.2. Define the Alternatives for each Concept

In this approach the various architecture design alternatives are derived from well-established concepts in the solution domain that have been leveraged to the identified technical problems.

3.2.3.3.3. Describe the Constraints

The total set of alternatives per concept may be too large and/or not relevant for solving the identified problems. Therefore, to define the boundaries of the architecture it is necessary to identify the relevant alternatives and omit the irrelevant ones.

If any new requirement is raised by client, independent of any other component, then the new component development process starts from first phase. Once the development process finish, the testing phase start which includes individual component testing and regression tests.

3.2.3.4. Testing

This phase in the process was to test the system to ensure that the developed product is error free. Quality of software product is very important while developing it. Many companies have not learned that quality is important and deliver more claimed functionality but at a lower quality level. It is much easier to explain to a customer why there is a missing feature than to explain to a customer why the product lacks quality. A customer satisfied with the quality of a product will remain loyal and wait for new functionality in the next version. Quality is a distinguishing attribute of a system indicating the degree of excellence.

The testing technique is from the perspective of the system provider. Because it is nearly impossible to duplicate every possible customer's environment and because systems are released with yet-to-be-discovered errors, the customer plays an important, though reluctant, role in testing.

3.2.3.5. Go Live and Support

The purpose of the Go Live and Support phase is to cut over to live productive operation and to continuously support and improve live operations based on project agreements finalized with client.

3.2.3.5.1 Project End

During the time when the system is first live, all issues and problems are resolved, transition to the production support

ISSN 2278-7763

team is finalized, knowledge transfer is completed, and the project is signed off.

3.2.3.5.2 Continuous Improvement

Now that the project is over, the production support team monitors the system and resolves live business process issues. Proper change management procedures are established, and ongoing end-user training is conducted. Plans are made to continuously review and improve business processes.

3.2.3.6 Results

The client got benefit from various ways as the TCO (total cost of ownership) for the client was reduced substantially and quicker ROI (return on investment) was achieved, as the major business processes were automated by software. Customized packages and services designed for employers on the latest technologies gave the client edge and helped the client outsmart its competitors. This allowed client access to highly skilled domain experts with extensive hands-on experience. Real time support ensured minimum uptime and unhindered business services. This helped with client's overall resource optimization enabling client to channelize internal resources towards its core business.

A lot of time spent on planning and design. But the time spent during first two phases help the development team to fully understand the requirements – problems/sub-problems till final level of decomposition. This decomposition helps in the final delivery of the product functionality and monitors the development process of each component separately at unit level to keep track of cost, quality and time schedule. Much of this design could be changed later or the plan could be trashed in turn of a more efficient aspect of the plan. Also, the design doesn't ever get tested until coding begins. If a flaw is found in the plan, major changes will need to be made. This could result in a waste of productive time. But as the development process followed component driven approach, all the requirement changes or new requirement were easily accommodated in development process. As development team was able to monitor the development at unit level, the problems identified at earlier stages and modified within scheduled time and cost. Finally the testing team was able to work parallel to the development team on delivered component, bugs/issue raised by testing team fixed during development and again delivered for testing phase. This insures the delivery of quality product within given time frame.

3.3 Case Study 3:

SCL Group is a leading company involved in the manufacture of Bulk APIs like Clopidogrel, Sertraline, Ketorolac, Pantoprazole, Rabeprazole, Amiodarone etc., Advanced Pharma intermediates - Bromo Acid, Chloro acid etc. & in the development of therapeutic recombinant proteins and monoclonal antibodies.

Established in early 90's, SCL built its business working alongside large Indian Pharmaceutical Companies like Ranbaxy, Lupin etc. Progressing rapidly from intermediates to API manufacturing, SCl has established several strategic alliances in the domestic and international markets. Focusing on international markets, SCL manufacturing facilities have been built to comply with EU-GMP and USFDA norms. In keeping with the times, SCL has taken initiatives in growth areas like Biotechnology. All the trading activities now being upgraded into Distribution and Logistics business.

3.3.1 Requirements

With world-class production facilities, R&D and quality assurance procedures in place, the only aspect in which SCL Group lagged was a respectable online presence to promote its services in the B2B marketplace. Its existing website was no more than a brochure on the internet, simply listing out the products. Further, they were also facing bottlenecks in communication due to an inefficient email server, which could not even bear the load of unsolicited inquiries.

So they contacted software development team with the goal of redesigning their website to showcase their full range of products, get more leads, increase B2B customer conversion rates and create a strong, memorable brand. After discussing with the SCL management team they were ready to implement Fusion Process Model for overall development process of their website.

3.3.2 The Solution

The analysts and designers following the Fusion Process Model approach identified several problems with the existing website, such as inappropriate sections, unprofessional images and lack of marketing depth that prevented the creation of a strong and memorable brand. These issues needed immediate redressal.

The immediate challenge for developers was to leverage the standing of SCL group as a company into a congruent solution for their website. We started with a powerful core design and layout that incorporated SCL logo and branding strategy. We used the client's preferred colour—consistent with their primary products—and set up corresponding imagery, logo and web elements, taking care into to neglect clarity and usability in the process. Development team also provided a resilient hosting and email solution satisfying all the mailing and communication-intensive needs of the client.

More specialized sections added in the website Products Section, Manufacturing Section, International Trading Division, SCL Biotechnology Division, where the clients can get in depth detail of each section and contact details of responsible person to get immediate reply from concerned department.

Products Section

ISSN 2278-7763

- Bulk APIs
- New Introductions
- Under Development
- Advanced Intermediates
- Cephalosporin and Carbapenem Intermediates
- Other Intermediates

Manufacturing Section

- Location
- Equipment
- Process Equipment
- Utility Equipment
- E.T.P.
- Quality Control
- Chemical Expertise
- Contract Manufacturing

International Trading Division

- APIs
- Personal Care
- Neutraceuticals
- Intermediates
- Domestic Product Offering

SCL Biotechnology Division

Services, Experties and clients.

3.3.3 Results

Sauravchemicals.com has been instrumental in gaining significant business benefits for the client, because the website has given them a definitive brand and professional image commensurate with their status as a large-scale manufacturer in the dyes and pigments industry. SCL group had this to say: —We experienced a doubling of inquiries and near-tripling of conversion rates in the three months following the launch of the revamped website, which I can directly attribute to the better user experience it provides.

4 Conclusion

We have talked about Fusion Process Model for software development process and 3C-Model for each phase of development process model. It has been implementation in three companies to investigate the practical aspect of fusion process model. The vital results in this paper include the Fusion Process Model, 3C Model, analysis of Fusion Process Model and results based on the experience of project manager and team leader using Fusion Process Model. Fusion Process Model ensures the overall quality of software system; reduce the development cost and time by considering the changing requirements of customer, risk assessment, identification, evaluation and composition of relative concerns at each phase of development process.

REFERENCES

[1] H.Dieter Rombach and Matrin Verlage, "Directions in Software Process Research", Advances in Computer, Marvil V. Zelkowitz (editor), Academic Press, Volume 41, pp. 1-3, 1995.

[2] Rupinder Kaur and Jyotsna Sengupta, "Software Process Models and Analysis on Failure of Software Development Projects", International Journal of Scientific and Engineering Research, Volume 2, Issue 2, pp. 1-4,
February, 2011.
Link:
http://www.ijser.org/onlineResearchPaperViewer.aspx?Sowar e_Process_Models_and_Analysis_on_Failure_of_Software_De velopment_Projects.pdf

[3] Rupinder Kaur and Jyotsna Sengupta, "Development and Analysis of 3C- Model for Software Development Lifecycle", IEEE 2nd International Conference on Computer Engineering and Technology (ICCET 2010), Volume 6, pp. 688-691, April 16 -18, Chengdu, China,2010.
Link:
http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=548597 0

[4] Rupinder Kaur and Jyotsna Sengupta, "A New Approach in Software Development - Fusion Process Model", Journal of Software Engineering and Applications, USA, Volume 3, Issue 10, pp. 998-1004, October
2010.
Link:
http://www.scirp.org/journal/PaperInformation.aspx?PaperID =3079&JournalID=45

