

# Win in the flat world

## Minimizing Rework in a CRM Greenfield Implementation

- Riju Raval

### *Abstract:*

In the world of Information technology, new systems replace existing systems continually. In CRM implementations this is very common, as more and more companies are selecting CRM package solutions such as CRM package to drive their businesses. Quite often, a CRM implementation will replace an existing system being used by the company. This in addition to harnessing the inbuilt processes of the CRM package also requires building the business processes of the existing system by refining/redefining into the CRM package.

Therefore when evaluating or estimating for a CRM Greenfield implementation it is essential to pay greater attention to the following activities:

1. Capture processes of the existing system
2. Map the existing process to the capabilities of a CRM package solution

These activities need to be planned in addition to the normal SDLC activities of Requirement, Design, Development, Testing and Deployment.

Experience suggests that if adequate weightage is given to these activities, then a lot of reworking in terms of expenditure, manpower and time can be saved.

If you are planning for a CRM implementation, then this paper will help you understand why these activities are vital to the success of your program. In addition, the recommended approach for conducting these activities, are also listed.

For more information, Contact [askus@infosys.com](mailto:askus@infosys.com)

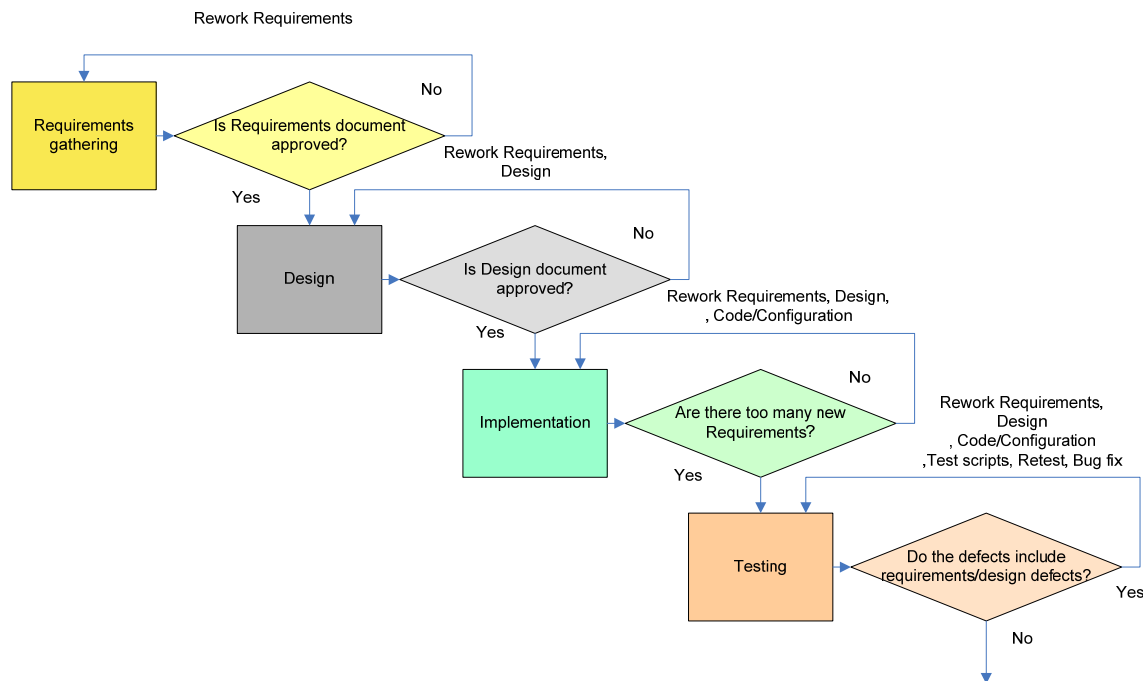


## Introduction

While embarking on a CRM implementation, technology groups often face the following SDLC issues which impede the efficiency of the project –

- Inadequate knowledge of business entities
- Gaps in requirement definition and articulation
- Delay and issues in obtaining approval on requirement, design documents from client technical team
- Application walkthroughs with business users throwing up too many new requirements
- Defects identified during testing phases include requirement/design defects.
- Defects during integration testing exceeding Service level agreement(SLA) defined for integration phase
- Overburdened resources
- Missing delivery deadlines

All these issues eventually lead to rework, thus impacting the overall health of the CRM program. The below figure depicts these issues and their impact in the project lifecycle.



**Figure 1: Issues resulting in Rework**

Evidently reducing rework is one of the key parameters determining the success of the CRM implementation. Towards minimizing rework, this paper tries to examine some reasons for these issues and suggests that by giving adequate weightage to certain key inputs, rework can be significantly reduced. Suggestions for organization of these activities are also outlined.

## Understanding issues that hamper efficiency

The basic reasons for these issues are –

### **1. Focusing primarily on CRM package capabilities rather than business requirements**

Since CRM solutions/implementations replace existing client systems, the implementation of a CRM solution in such a scenario may involve refining/redefining rebuilding some or most of the current client system processes/capabilities/functionalities. The reason being – the existing system may have been in existence for a while. Therefore over the years it would have multiple business requirements already built in, which the business is now not completely aware of, but requires the CRM package to merge with the package's inbuilt processes (either as-is or with modifications).

During requirements gathering however, the focus is generally on the out of box (OOB) capabilities that the CRM package can deliver and the customizations that the business users would like to have. Often the customizations expressed by the business users are also relative to the out of box processes being discussed. Since business may not be sure of the requirements built into the system currently in use these requirements may get marginalized.

### **2. Lack of sufficient documentation on the existing system**

As discussed above, merging of the existing system processes with the CRM package's out of box processes is integral to a CRM Greenfield implementation. Towards this goal it is necessary to build a better understanding of the existing system which is possible through documentations and discussions.

However, often, the documentation for the existing system may either not be available at all or may not be updated as per the latest functionality available in the system.

## Key activities to resolve these issues

The issues discussed here can be addressed by investing sufficient time and better planning into the following activities –

### **1. Capturing processes of the existing system**

It involves conducting a detailed study of the existing system to obtain a deeper understanding of the business domain, application knowledge, processes (identified for further refinement, for merging with CRM package's OOB processes). This activity is referred to as System appreciation (SA).

The end product of this exercise is a document, known as a System appreciation document which should capture knowledge about the existing system from all possible sources for example, existing documentation, Subject matter experts (SMEs) of the existing system.

### **2. Mapping the existing processes to the capabilities of a CRM package solution.**

The processes built into the existing system will need to be evaluated for replacement, or complete or partial retention depending upon the extent to which the CRM package solution is replacing the existing system.

Depending upon the requirement the CRM package solution can replace the existing system, partially or completely. These scenarios are explained below -

[A] When the CRM package is completely replacing the existing system this activity will involve –

- a. Identifying processes of the existing system that will be replaced completely and the customization required to CRM processes.
- b. Identifying and outlining integration if required with other systems.

[B] If it is not possible to completely replace the existing system with the CRM package solution (for example: Existing system's ecommerce website is integrated with a CRM package application), then this activity will involve –

- a. Identifying processes of the existing system that will be retained as-is
- b. Identifying processes of the existing system that will be replaced completely and the customization required to the CRM package processes.
- c. Identifying processes of the existing system that will be replaced partially and the customization required to the CRM package processes.
- d. Outlining the integration between the existing system processes being retained with CRM processes.
- e. Identifying and outlining integration if required with other systems.

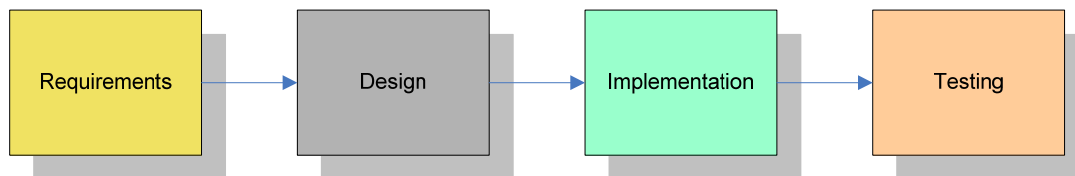
These objectives will be realized through discussions with the technical teams of the existing system, client technical team and CRM technical team. We refer to it as Joint Application Design (JAD).

Since these activities will go into details of implementation, initially in the discovery phase of the project a simplification exercise must be conducted with the business users to identify the high level processes which they require the CRM solution to offer. The output of this exercise will serve to define the structure of the system appreciation and Joint Application Design activities.

### Placement of key activities

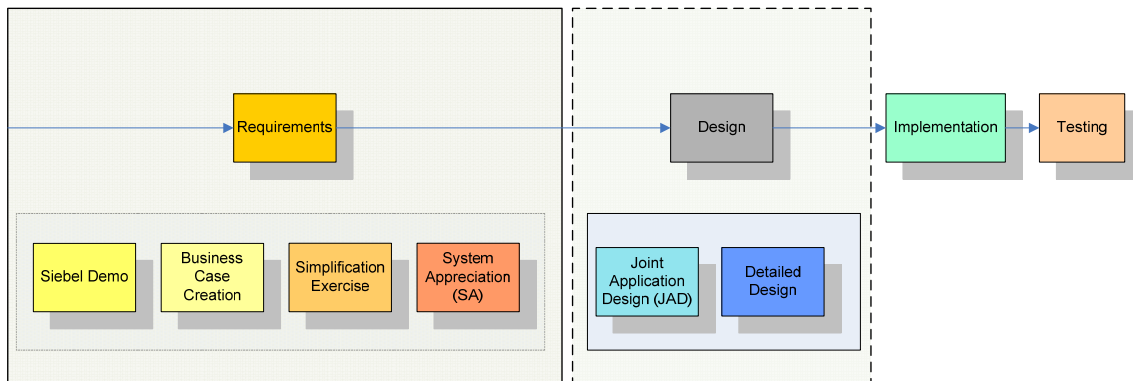
The order, in which these activities may be conducted to ensure effectiveness, is best understood in context to the basic Software life cycle development (SDLC) Model.

The SDLC Model consists of the phases of requirements gathering, design, development and testing.



**Figure 3: Software development life cycle**

The SDLC Model is modified to illustrate the sequence of these activities –



**Figure 4: Software development life cycle depicting process simplification exercise, system appreciation and joint application design**

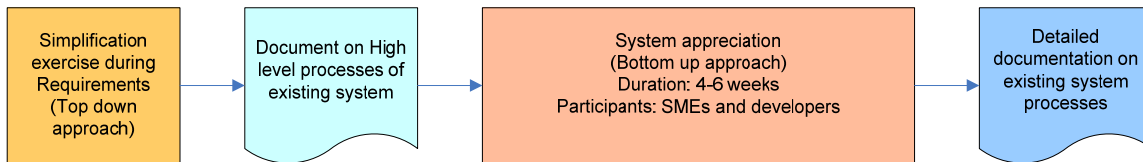
## Adequate weightage

Adequate weightage can be defined in terms of the time invested and the planning for these activities. This is detailed as given below –

### System appreciation

<i>Estimated time for completion</i>	This may be defined based upon the size of the existing system and the processes that need to be documented using the standard methodologies. On an average, this step is usually planned for 4 to 6 weeks ( <i>for a project spanning 60 weeks</i> ), depending upon the complexity of the processes being covered.
<i>Participants</i>	Conducting the system appreciation exercise will require a sound knowledge of the existing system functionalities and the to-be system inbuilt functionalities. Hence SMEs from the existing system team will be required and SMEs from the CRM team will need to drive this exercise. Developers from the existing system will also participate in this activity as a detailed documentation of the existing system processes needs to be prepared. In an Offshoring model, SMEs from both systems, as-is and to-be need to be available both at Onsite (client location) and Offshore (vendor location). The developers may be based at offshore.
<i>Prerequisites</i>	The simplification exercise as mentioned earlier in this paper, needs to be conducted during requirements gathering phase of the project, as this will define at a high level the existing processes that are to be replaced or retained or modified. This high level definition or top-down approach will help in organizing the structure of the SA exercise which can otherwise be overwhelming and unwieldy.
<i>Expected outcome</i>	Documentation on the existing system's functional features, its architecture and the design also known as System appreciation document.

The following figure captures the plan for System appreciation –

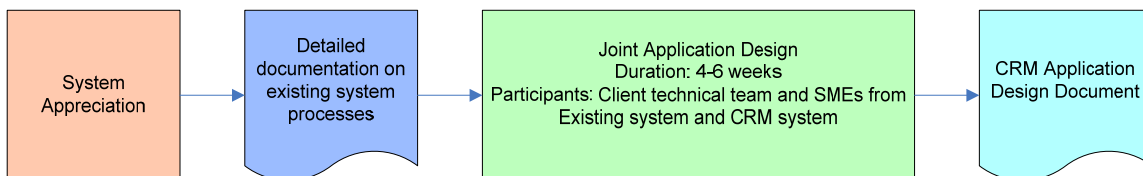


**Figure 5: Defining adequate weightage for system appreciation in terms of time and planning**

### Joint Application Design

<i>Estimated time for completion</i>	This may be defined as approximately same as the time taken to perform system appreciation, since the processes studied in the SA phase will be the inputs for the discussions during Joint application design (JAD).
<i>Participants</i>	<p>The purpose of the Joint Application design is to review the System appreciation document, and define the to-be landscape of the CRM Implementation.</p> <p>Since the discussions on the high-level design will require expertise on both existing and the new systems, this activity needs to involve the client technical team, SMEs from the existing system team and CRM package implementation team.</p>
<i>Prerequisites</i>	System appreciation exercise must commence prior to Joint application design.
<i>Expected outcome</i>	The outcome of the JAD sessions will be a High level design document which will cover the to-be system processes and integration to the current system or other external systems if required.

The following figure captures the plan for Joint application design –



**Figure 6: Defining Adequate Weightage for Joint Application Design in terms of time and planning**

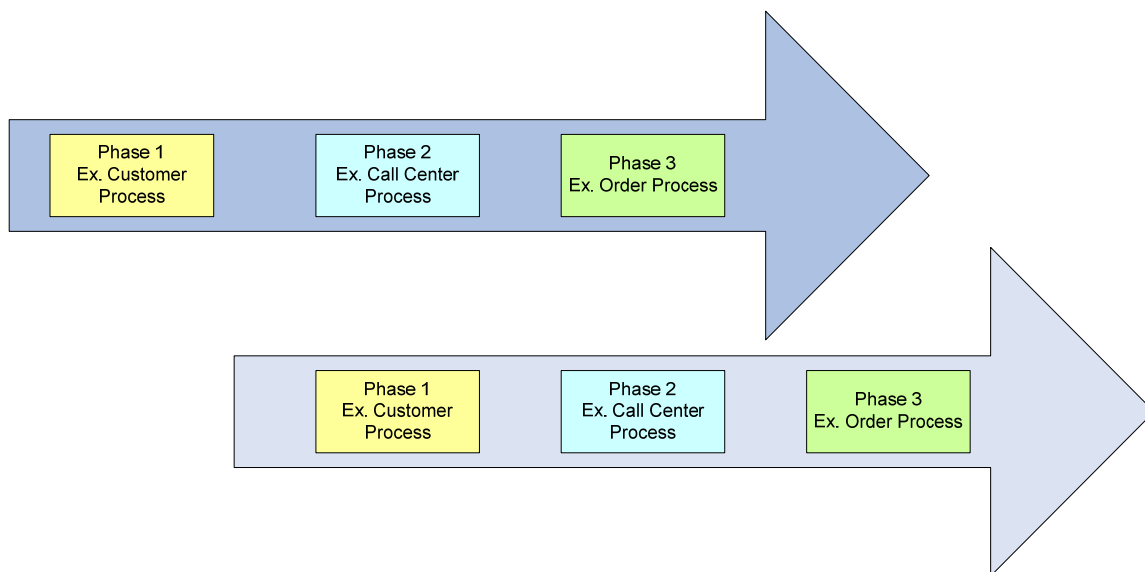
## Other considerations

### **Phased-out Joint application design:**

It is not essential for System appreciation to be completed prior to start of Joint application design. In fact SA and JAD may be conducted in parallel provided the SA and JAD are organized in functional phases. For example, the Output of phase 1 of SA can be the input to phase 1 of JAD.

The phases will be a logical grouping of processes. For example, the High level processes of the CRM Implementation can be grouped into – Customer processes, Transaction (Order) processes for example. Hence System appreciation for Customer processes can be conducted first, which will be the input to the JAD sessions on these processes.

This phased out approach to conducting SA and JAD is depicted in the figure given below:



**Figure 7: Phased out approach to System appreciation and Joint application design**

### **Spill-Over SA and JAD:**

The paper has described the sequence of the System appreciation and Joint application design activities in relation to the Software development lifecycle. Based on our experience we find that SA and JAD may be continued into the subsequent phases of build and testing. Ofcourse these activities will be conducted on a much smaller scale in the remaining phases, mainly on a need basis since majority of the processes would have been covered during Design. The need would arise when the existing system (which is still in production) processes undergo any changes.

Since the time required for this activity will be much less as compared to the activity in design phase, we are using the term 'Spill-Over'.

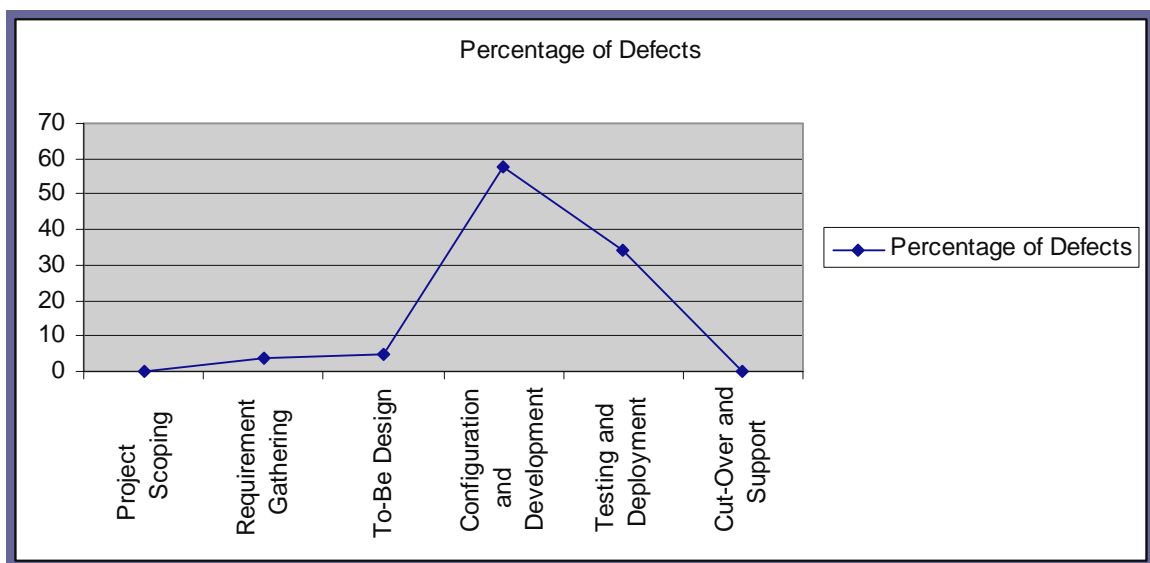
Availability of Subject matter experts from the existing system during the remaining phases of the project therefore will be essential if this 'Spill-Over' activity is being planned.

## Case study

The study presented here is the learning gained from a real life CRM Greenfield Implementation project. The System appreciation and Joint application design phases introduced in this highly complex Siebel implementation which has close integration with the existing web Java based application had the following benefits –

- a. Minimal requirements rework owing to minimum defects injected in Requirements gathering phase. This was achieved by building in the System appreciation phase into the project lifecycle.
- b. Minimal design rework owing to minimum defects injected in Design phase. This was achieved by building in the Joint application design phase into the project lifecycle.

The graph below depicts the percentage of defects injected by each LC stage in the said project –



**Figure 8: Percentage of defects injected per stage in a project where the suggested approach was implemented.**

## Conclusion

Through this paper, I have highlighted the issues many a CRM Greenfield Implementation often face. The analysis presented emphasizes the necessity of giving appropriate attention to the two activities System appreciation and Joint application design in order to minimize rework owing to defects injected during requirements and design phases of a project. The paper also offers suggestions on conducting these activities.

Investing judiciously on these activities, will help CRM implementations in staying aligned to user expectations throughout the project lifecycle, thus minimizing rework and enabling the CRM implementation to stay on track with respect to schedule and budget.

## References:

**System Appreciation Document - A Creation Guide:** By Rupak\_Das, Nitin\_Verma  
(Infosys Knowledge Shop)

## About the Author

**Riju Raval** is a Technical Architect with Infosys' Customer Relationship Management practice.

*She has over 6 years of work experience on various CRM implementations, upgrade and maintenance projects across Transportation and Logistics, Financial and Hi Tech domains. She has led multiple teams both from client side and offshore to design and develop CRM package implementations.*

© 2009 Infosys Technologies Limited.

ALL RIGHTS RESERVED

Copyright in whole and in part of this document "**Minimizing Rework in a CRM Greenfield Implementation**" belongs to Infosys Technologies Limited. This work may not be used, sold, transferred, adapted, abridged, copied or reproduced in whole or in part in any manner or form or in any media without the prior written consent of Infosys Technologies Limited.