Digital Policy Office

AN INTRODUCTION TO
RAPID APPLICATION DEVELOPMENT
(RAD)

[G47a]

Version: 1.8

Jul 2024

TABLE OF CONTENTS

1.   PURPOSE

The objectives of this document are:

- to introduce the RAD concepts;
- to give an overview of the RAD task structure;
- to describe the major roles in RAD methodology; and
- to introduce the essential RAD techniques.

2. SCOPE

Rapid Application Development (RAD) refers to a development life cycle designed to give faster development and higher quality systems than the traditional one. This document gives an introduction of the Rapid Application Development methodology. Some of the main components of RAD are described and explained. End users and potential tenderers are suggested to read this guide to understand the RAD concepts. For details about RAD and related techniques/methodologies, please refer to other manuals listed in Section 3.

3.   REFERENCES

3.1   STANDARDS

- RAD Procedures Guide [G47]

- RAD Documentation Guide [G48]

3.2   OTHER REFERENCES

Nil

4.   DEFINITIONS AND CONVENTIONS

4.1   DEFINITIONS

Nil

4.2   CONVENTIONS

Nil

5.   INTRODUCTION

5.1   WHAT IS RAD?

RAD refers to a development life cycle designed to give much faster development and higher quality systems than the traditional life cycle. It is designed to take advantage of powerful development software like CASE tools, prototyping tools and code generators. The key objectives of RAD are High Speed, High Quality and Low Cost. RAD is a people-centred and incremental development approach. Active user involvement, as well as collaboration and co-operation between all stakeholders are imperative. Testing is integrated throughout the development life cycle so that the system is tested and reviewed by both developers and users incrementally. A typical RAD life cycle is composed of the following Stages, which will be described in detail in Section 7.

```
 ┌─────────────────────────────────────────┐
 │  ┌──────────────┐    ┌──────────────┐    │    ┌──────────────┐    ┌──────────────┐
 │  │ Requirements │    │    User      │    │    │    Rapid     │    │  Transition  │
 │  │  Planning    │──▶ │   Design     │──▶ │──▶ │ Construction │──▶ │    Stage     │
 │  │   Stage      │    │   Stage      │    │    │   Stage      │    │              │
 │  └──────────────┘    └──────────────┘    │    └──────────────┘    └──────────────┘
 └─────────────────────────────────────────┘
```
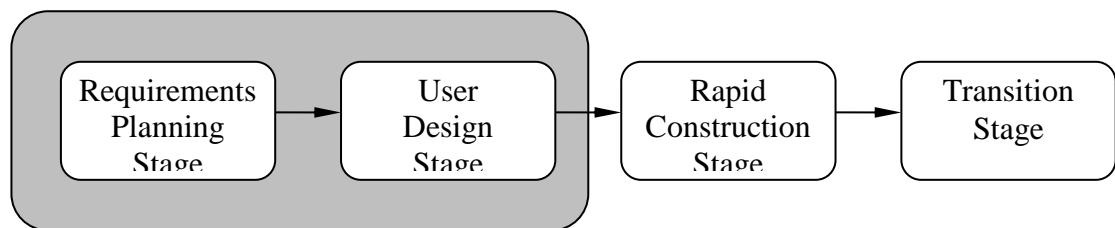
If the requirements are obvious, the first two Stages may be combined.

5.2   RAD IN SDLC

Mapping between System Development Life Cycle (SDLC) and RAD stages is depicted as follows.

<table>
<tr><td align="center">SDLC</td><td></td><td align="center">RAD</td></tr>
<tr><td>**Project Request**</td><td></td><td></td></tr>
<tr><td>**Feasibility Study (FS)**</td><td>←</td><td>**Requirements Planning**</td></tr>
<tr><td>**System Analysis & Design (SA&D)**</td><td>←</td><td>**User Design**</td></tr>
<tr><td rowspan="2">**Implementation**</td><td rowspan="2">←</td><td>**Rapid Construction**</td></tr>
<tr><td>**Transition**</td></tr>
<tr><td>**Post Implementation Review**</td><td></td><td></td></tr>
</table>
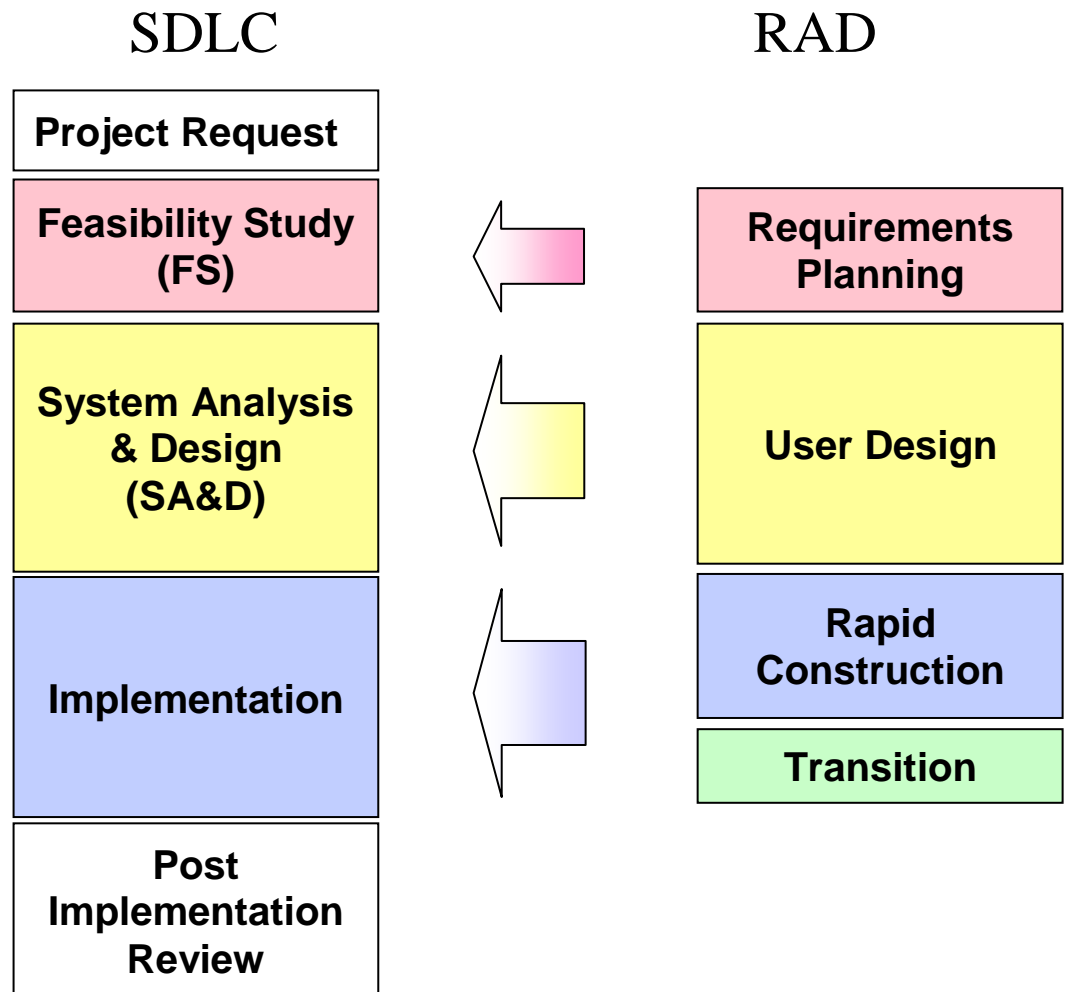
Note: If a combined FS/SA&D is to be conducted, the Requirements Planning (RP) and User Design (UD) stages will be conducted consecutively.
If a SA&D is to be conducted without a previously conducted FS, the activities in the RP stage still have to be done when necessary, before starting UD stage.

5.3   ESSENTIAL INGREDIENTS OF RAD

RAD has four essential ingredients:

- Tools;
- Methodology;
- People; and
- Management.

If any one of these ingredients is inadequate, development will not be high speed.
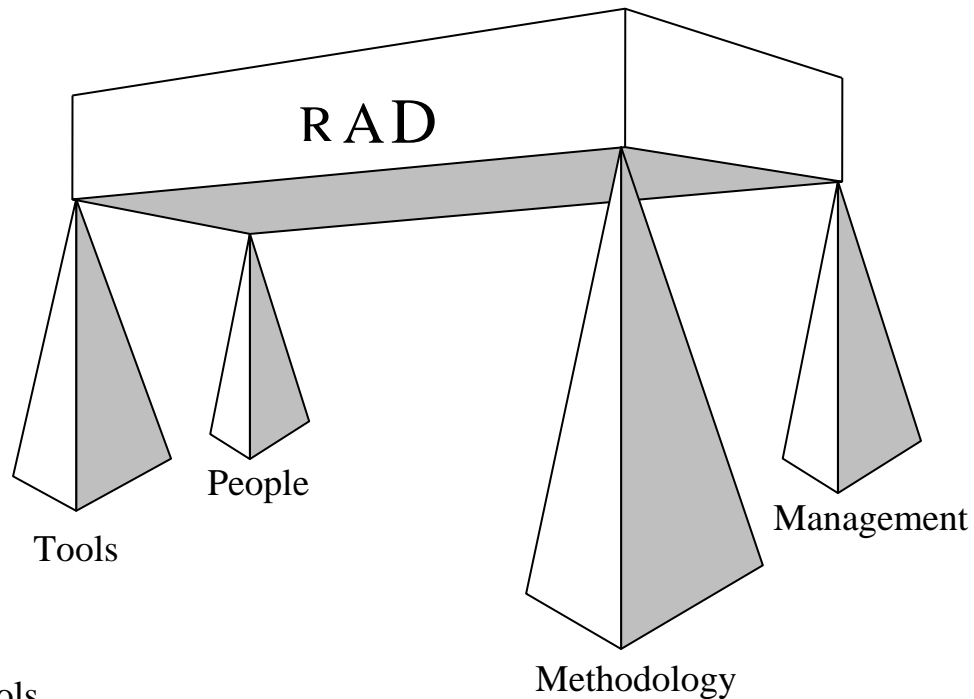
5.3.1   Tools

The use of powerful CASE tools will provide automation support for systems development through features such as code generation and automatic consistency checking. CASE tools that generate prototypes can be used to support an iterative development approach such as RAD, allowing end users to see the application evolves as it is being built.

5.3.2   Methodology

The most effective family of techniques must be formalised and used to deliver the systems. A complete list of tasks is provided to ensure that no essential activity is overlooked, while techniques are fully documented to ensure that a task is performed in the proper way.

5.3.3   People

Bright people must be well-trained in both the methodology and the tools. Small teams that work consistently well together should be grouped together on assignments.

5.3.4   Management

The project must be managed for speed through the use of techniques such as facilitated Joint Requirements Planning (JRP) and Joint Application Design (JAD) workshops to extract users' requirements quickly.   Timebox Management is used in Rapid Construction to iteratively deliver the system to the users.

6.   BENEFITS OF RAD

The following benefits can be realised in using RAD: -

- High quality system will be delivered because of methodology, tools and user involvement;

- Business benefits can be realised earlier;

- Capacity will be utilised to meet a specific and urgent business need;

- Standards and consistency can be enforced through the use of CASE tools.

In the long run, we will also achieve that: -

- Time required to get system developed will be reduced;

- Productivity of developers will be increased.

7.   RAD STRUCTURAL MODEL OVERVIEW

7.1   STRUCTURE COMPONENT

The structure of RAD is divided into Stages. Each Stage consists of a number of tasks. For each Task, the Structural Model defines clearly the objectives, inputs, outputs, techniques to be used, roles involved and a list of sub-tasks to be carried out within the Task.

```
                          ┌─────────┐
                          │  Stage  │
                          └─────────┘
         ┌───────────────────┼──────────────────────┐
     ┌──────┐            ┌──────┐                ┌──────┐
     │ Task │            │ Task │  . . . . . . . │ Task │
     └──────┘            └──────┘                └──────┘
     ┌──────┐ ┌───────┐  ┌──────┐ ┌───────┐     ┌───────┐ ┌───────┐
     │Subtask│..│Subtask│  │Subtask│..│Subtask│  │Subtask│..│Subtask│
     └──────┘ └───────┘  └──────┘ └───────┘     └───────┘ └───────┘
```
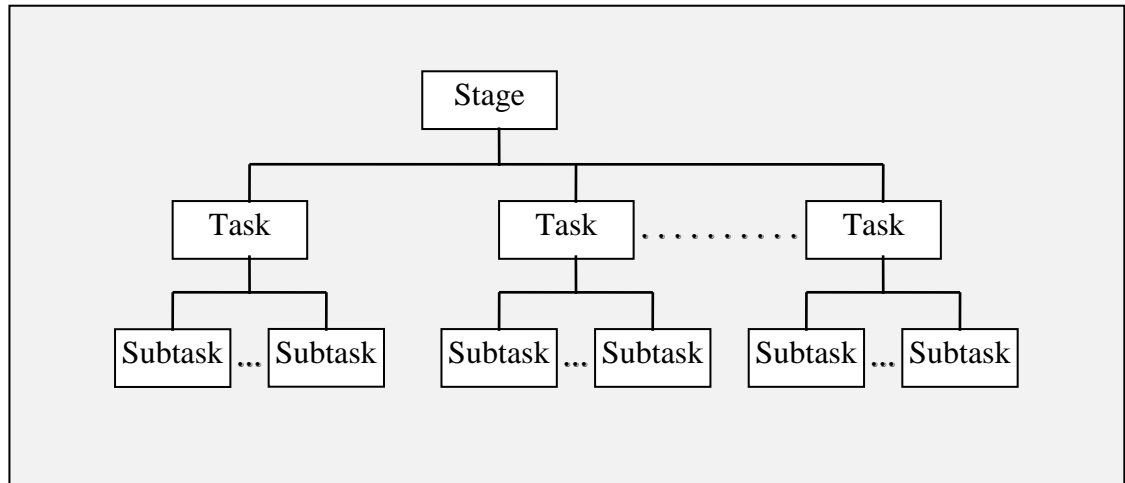
7.2   OVERVIEW OF STRUCTURE

The following diagram shows clearly the Stages and Tasks in RAD: -

```
                                    ┌───────────────┐
                                    │      RAD      │
                                    └───────┬───────┘
        ┌───────────────┬───────────────────┼───────────────────────────────┐
┌───────────────┐ ┌───────────────┐ ┌───────────────┐              ┌───────────────┐
│  Requirements │ │     User      │ │     Rapid     │              │   Transition  │
│   Planning    │ │    Design     │ │  Construction │              │               │
└───────────────┘ └───────────────┘ └───────────────┘              └───────────────┘
```

Requirements Planning	User Design	Rapid Construction	Transition
Research Current Situation	Produce Detailed System Area Model	Prepare for Rapid Construction	Conduct User Training — Perform Data Conversion — Install Production System
Define Requirements	Develop Outline System Design	Construct System — Generate Test Data and System Documents — Prepare for Transition	Accept System Installation
Finalise Requirements	Refine System Design — Prepare Implementation Strategies	Verify System Construction	
	Finalise System Design		
	Obtain Approval for Construction		

7.3   RAD STAGES

7.3.1   Requirements Planning Stage

Objectives

- To establish a general understanding of the business problems of existing circumstances;

- To be familiar with existing systems; and

- To identify the business functions that will be supported by the proposed application.

Overview

The Requirements Planning Stage consists of a review of the areas immediately associated with the proposed system. This review produces a broad definition of the system requirements in terms of the functions that the system will support. An outline of the system area and a definition of the system scope are developed. Project board (consisting of Business executives, end users and IS professionals) takes part in workshops which progress through a structured set of steps; with the results recorded in the CASE tool.

The deliverables from the Requirements Planning Stage include an outline system area model of the area under study, a definition of the system's scope, and a cost justification for the new system.

The tasks of the Requirements Planning are shown in the diagram below:
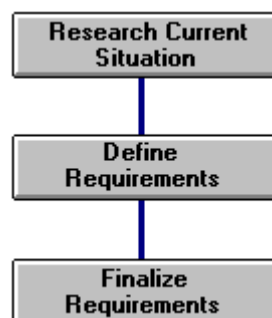
7.3.1.1 Research Current Situation

A research of the current environment is conducted in preparation for the requirements definition of the proposed system. An initial system area model is created in the CASE repository.

7.3.1.2 Define Requirements

The outline system area model and scope of the proposed system are developed in this task. The functionality of the system is expressed in terms of the business processes and the data that the system will support. Potential benefits, costs, and risks associated with the proposed system are identified. Management issues, such as constraints and limitations, that affect subsequent development and transition activities, are documented.

7.3.1.3 Finalise Requirements

The scope of the proposed system is formally documented. An estimate of the resources and the duration to implement the system is prepared. Sometimes, the cost and duration may already be fixed when the project commences. In this case, the scope must be well defined such that the project is still viable with the cost and duration provided. Approval to proceed with the implementation is then obtained.

7.3.2   User Design Stage

Objectives

- To produce a detailed system area model;

- To develop an outline system design; and

- To prepare an implementation plan.

Overview

The User Design Stage consists of a detailed analysis of the business activities and data related to the proposed system. Key users, meeting in workshops, decompose business functions and define data associated with the system. They complete the analysis by defining the interactions between processes and data. The results of the workshops are recorded in the CASE tool. Following the analysis, the design of the system is outlined. System procedures are designed, and preliminary layouts of screens and reports are developed. Prototypes of critical procedures are built and reviewed. An appropriate construction approach for the system is selected. A plan for implementing the system is prepared.

The tasks of the User Design are shown in the diagram below:
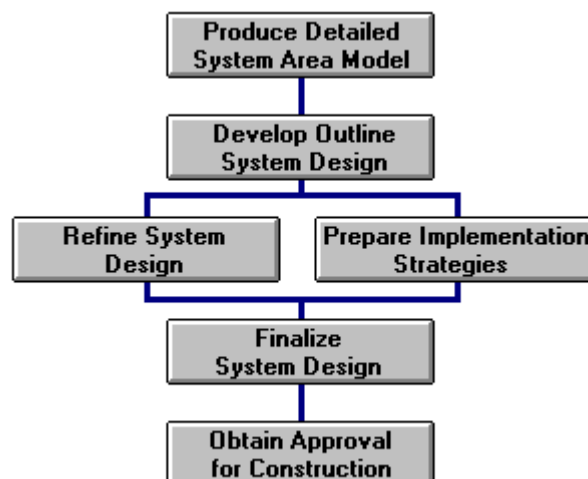
7.3.2.1 Produce Detailed System Area Model

JAD workshops are conducted to complete the analysis of the business activities and data associated with the proposed system, and produce a detailed system area model. The scope of the system to be developed is reviewed and refined to ensure that the critical system functions will be delivered in the required time frame.

During requirements planning, broad definitions of business activities and data are defined. Then, during this stage, the definitions are revisited to explore the details. For instance, the business rules to be applied in each business activity and the attributes for each entity.

7.3.2.2 Develop Outline System Design

The procedures required for the system are identified. Tentative layouts of critical screens and reports are developed. An outline system design is developed. Upon completion of the outline design, interactions between system procedures (series of system functions) and data are identified. That is, data usage per system function is defined.

7.3.2.3 Refine System Design

The completeness of the detailed system area model and outline system design is reviewed and verified. The consistency of the analysis and design is confirmed via interaction analysis and prototyping. By interaction analysis, the interactions between system functions and data are analysed to identify missing/extraneous functions (no data usage) or data (not used by any function). When inconsistencies are resolved, prototypes of screens and dialogues are developed and shown to the users for review and revision. Adjustments to the list of open issues are then made.

7.3.2.4 Prepare Implementation Strategies

A plan for the implementation of the system is developed. The implementation approach is selected after reviewing the design of the system. To speed up delivery, parallel development and timebox development can be adopted. An implementation plan is prepared listing all tasks that must be performed to develop the system and convert it into operational use. An estimate of the effort necessary to complete each task is made and summarised into an overall project cost estimate.

7.3.2.5 Finalise System Design

The outline design of the system and implementation plan need to be finalised. Open design issues are discussed and resolved except those with no impact to the system design such as cultural issues. The transition plan for the system is presented, discussed and confirmed.

7.3.2.6 Obtain Approval for Construction

The results of the final JAD workshop are incorporated into the system design and transition plan. Approval to proceed with the Rapid Construction Stage is sought.

7.3.3   Rapid Construction Stage

Objectives

- To complete a detailed design of the proposed system;

- To develop and test the application software that implements the proposed system;

- To prepare documentation necessary to operate the proposed application;

- To perform the activities to prepare for the conversion of the system to production status.

Overview

A small team of developers which is known as Skilled With Advanced Tools team (SWAT team) works directly with users, finalises the design and builds the system. The software construction process consists of a series of "design-and-build" steps in which the users have the opportunity to fine-tune the requirements and review the resulting software implementation. The transition plan to production is also prepared.

The Rapid Construction Stage deliverables include documentation and instructions necessary to operate the new application, routines and procedures needed to put the system into operation.

The tasks of the Stage are shown in the diagram below:
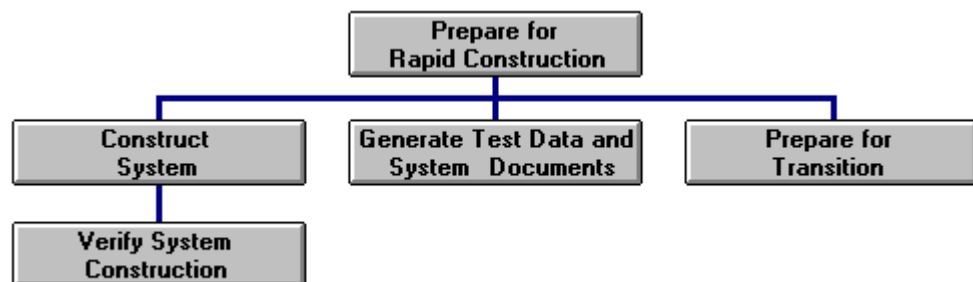
7.3.3.1 Prepare for Rapid Construction

The development environment is finalised. The database is designed based on the preliminary data structure developed in the User Design Stage. Teams of IS professionals and users who will construct the system are established. Standards that will be applied in the final design and construction of the system are reviewed.

7.3.3.2 Construct System

Software to implement the system is developed and completed by the SWAT (Skilled with Advanced Tools) team under a firm time limit. The construction assistance team works with the SWAT team(s) to develop the software, using an iterative prototyping technique. The resulting code is tested at the system level and across all system components assigned to the SWAT team.

7.3.3.3 Generate Test Data and System Documents

This task involves developing a test plan, test cases and test data, and operational materials and instructions.

7.3.3.4 Prepare for Transition

A detailed work plan for transition activities and a contingency plan to cover failure of the converted application are prepared. Computer software needed to convert existing data, and procedures required to perform the system transition are developed. Any organisational issues related to deployment of the new application are resolved. The transition preparation work performed during the Rapid Construction Stage is an extension of the transition planning work performed in the Requirements Planning and User Design Stages.

7.3.3.5 Verify System Construction

Each component of the system and the complete system activities are verified and confirmed according to the user requirements.

7.3.4   Transition Stage

Objectives

- To install the system in production operation;

- To identify potential future enhancements.

Overview

The Transition Stage involves implementing the new system and managing the change from the old system environment to the new one. This may include implementing bridges between existing and new systems, converting data for the new system, training users to operate the new application and providing support to resolve any problems that arise immediately after the application becomes operational. User acceptance is the end point of the Transition Stage.

The tasks of the Stage are shown in the diagram below:
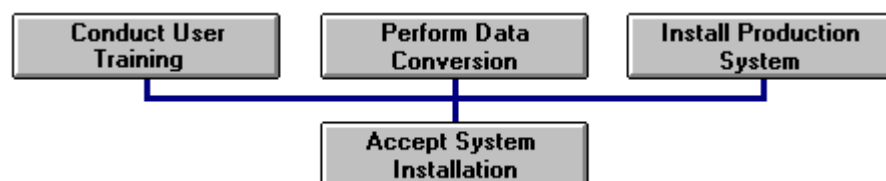
7.3.4.1 Conduct User Training

Training sessions are conducted to instruct future users of the new system on how it operates. This training is completed before the application is placed into production.

7.3.4.2 Perform Data Conversion

Information necessary for the operation of the new system is converted from existing data sources into a format accessible by the new system. The converted data is then loaded into the data structures associated with the system.

7.3.4.3 Install Production System

Steps necessary to start operation of the system in the production environment are performed. Necessary adjustments to the hardware and system software configuration are completed; instructions are given to the operations personnel who will be operating the system; and software libraries are loaded with the production versions of the application software.

7.3.4.4 Accept System Installation

This section presents those tasks to be accomplished in order to achieve specifications identified in the system's design and acceptance which is based upon agreements among users, production/operations staff, support staff, and the information management organisation.

8.   MAJOR RAD ROLES

8.1   FACILITATOR

The Facilitator plays a crucial role in conducting intensive group sessions. This person is responsible for ensuring that the objectives of the session are achieved. The Facilitator organises and conducts the user workshops, and prepares materials for each. The individual must have good communication and leadership skills. This role should be considered as a profession. The skills of a person in this role are critical to the success of a user workshop. He/she should be a mediator and arbitrator during conflict resolution, and also be a fair witness by providing an objective mirror of the team's progress through feedback.

8.2   SCRIBE

The Scribe is an active participant who is responsible for producing the outputs of the workshops, using the CASE tool. The Scribe is usually an IS professional. The Scribe records information throughout the day and at breaks, and in the evening may polish the results, possibly adding to it or raising questions for further discussion. The Scribe may interrupt the workshops frequently to highlight inconsistencies in the discussions.

8.3   SWAT TEAM

A SWAT (Skilled With Advanced Tools) Team is a small group of high-quality developers skilled with powerful tools. They work well together to provide fast and interactive development with the assistance from end users. The team members are responsible for the detailed design and construction of the system. They operate CASE tools to record design information provided by the end users, generate the prototypes of the system procedures, and review these with end-users in the Construction Assistance Team.

8.4   MODEL ADMINISTRATOR

The model administrator coordinates the development and maintenance of architectures and models resulting from multiple projects or from different persons within one project.

8.5   DATABASE ADMINISTRATOR

The database administrator is responsible for the performance, integrity and physical security of the organisation's data.

8.6   PLANNING WORKSHOP TEAM

The participants of the requirements planning workshop assist in the definition of the requirements and the scope of the proposed system. Participants in the workshop are the management-level personnel who are responsible for the business activities that will be affected by the proposed system.

8.7   USER DESIGN TEAM

The User Design participants provide detailed information describing the business functions and processes that will be affected by the proposed system. They also assist in the development of the design of the proposed system, and review prototypes and implementation plans. These activities are performed within the user workshops. User Design participants are usually future end users of the proposed system.

8.8   CONSTRUCTION ASSISTANCE TEAM

The Construction Assistance Team ensures that end user needs are met in the completed system. The Construction Assistance Team does this by providing detailed knowledge of business activities and feedback on system usability to the SWAT Team.

8.9   TRANSITION TEAM

Members of the Transition Team perform the tasks necessary to prepare for and carry out conversion of the system into production status.

9.   ROLE/TASK MATRIX

M - Mandatory, O - Optional	Requirements Planning			User Design						Rapid Construction					Transition			
	Research Current Situation	Define Requirements	Finalized Requirements	Produce Detailed System Area Model	Develop Outline System Design	Refine System Design	Prepare Implementation Strategies	Finalize System Design	Obtain Approval for Construction	Prepare for Rapid Construction	Construct System	Generate Test Data and System Documents	Prepare for Transition	Verify System Construction	Conduct User Training	Perform Data Conversion	Install Production System	Accept System Installation
Project Steering Committee																		
Executive	O	O	M			O		O	M					O				M
Senior User			M					M	M									M
Senior Supplier			M					M	M									M
Project Assurance																		
Senior User[1]																		
*Planning Workshop Team	M	M	M															
*User Design Team				M	M	M		M	M									
*Construction Assistance Team							M				M	M		M				
*Transition Team													M		M	M	M	
Senior Supplier[2]		M	M							M								
Executive[3]			M						M									
Project Team																		
Project Manager/Team Manager	M	M	M	O	O	O	M	O	M	O	O	O	O	M	O	O	O	M
*Facilitator	M	M	M	M	M	M	M	M										
Team Member																		
*Facilitator	M	M	M	M	M	M	M	M										
*Scribe		M		M	M	M	M	M										
*Model Administrator	M	M	M	M		M					M							
*Database Administrator			M			M				M	M		M			M	M	
*SWAT Team						M		M		M	M		M	M				

"*" Denotes the RAD roles.

"1" Senior User or his/her delegates on project assurance.
"2" Senior Supplier or his/her delegates on project assurance.
"3" Executive or his/her delegates on project assurance.

10.   MAJOR RAD TECHNIQUES

10.1   USER WORKSHOP

There are two types of user workshop:

- Joint Requirements Planning (JRP) Workshop
- Joint Application Design (JAD) Workshop

JRP workshops are conducted in the Requirements Planning Stage. The basic idea is to gather the developers and users together in the workshop to work out the requirements and agree the system boundary. One of the advantages of using a workshop approach is that it ensures that they work with the same goals and fully participate in their job with minimal interruption. JRP workshop often involves high-level management who understands the overall business requirements and has the authority to decide the scope of the system. The duration of each JRP workshop is usually one day and it does not involve too much technical details.

JAD workshops are conducted during the User Design Stage. The detailed design of the system will be worked out in a number of JAD workshops. The JAD workshop involves users who will own or work with the system. Some of the users involved in the JRP workshop should also be present. The JAD workshop is very similar to the JRP workshop but it usually involves more technical details and the duration of each workshop is usually one day. In addition, JAD workshop uses one powerful technique called Prototyping so that the users can visualise parts of the future system and suggest any refinement necessary.

10.2   FACILITATION

Facilitation techniques are used by a Facilitator to facilitate user workshop sessions, such as JRP and JAD workshops. The facilitator employs various techniques when conducting workshops for two primary reasons: to accelerate the capture of planning, analysis and design information and to ensure that session participants work effectively together as a team.

User workshop facilitation is the process of harnessing the knowledge of the participants while managing participant behaviour to accomplish a set of pre-defined objectives.

10.3   TIMEBOX MANAGEMENT

A timebox is a limited time period within which a well-defined deliverable must be produced with given resources and therefore can be used whenever appropriate. It is generally applied during Rapid Construction where a deadline is set and must not be changed. Within the timebox, continuous iteration development is done.

When a timebox is used, the scope of the system is one of the variables of project management. The quality, however, is never a variable. The functionality of the system

may be revised in order to complete it within the timebox. The system produced by the end of the timebox must be a system that is intended to be implemented.

10.4   PARALLEL DEVELOPMENT

Parallel Development is required for the high-speed development of large applications having a high level of complexity or significant size. The system is split into subsystems that can be developed and tested autonomously by multiple Skilled With Advanced Tools (SWAT) teams operating in parallel. Integration testing is performed and adjustments are made where necessary. Although the subsystems are built and tested independently, they must fit together finally. To achieve this, a co-ordinating model is used to help co-ordinate the work of the separate teams.

Parallel Development in the RAD environment provides a means to develop large and complex systems much more rapidly than using traditional techniques so as to reduce cost. Additionally, the subdivision of a large project into sub-projects makes each of these sub-projects much more controllable.

10.5   LOGICAL DATA MODELLING

Logical Data Modelling (or Entity Relationship Modelling) enables you to describe data and its inherent structure. It is represented as a diagram, known as the logical data structure (or entity relationship diagram) which is used to show your understanding of data and in later stages as a basis for Database Design.

10.6   BUSINESS PROCESS MODELLING

The purpose of Business Process Modelling is to understand the current business processes and activities in order to provide a basis for identifying issues, problems and outlining a proposed system. The Business Process Model (BPM) will be eventually used to model the proposed system. A BPM consists of the followings:

- Business Process Diagram
- Business Process Description
- Process Decomposition Diagram

The Process Decomposition Diagram shows a decomposition of the processes executed by the organisation or parts of the organisation within project scope. Decomposition is carried out as an effective method of finding processes quickly. As the business processes are decomposed, a level is reached in the hierarchy where a clear process thread (one level above the Elementary Business Process) can be identified.

The Business Process Diagram is used to show the flow of activity for a business process that forms a process thread. Basically, process diagram shows the flow of activities and the transition from one activity to the next activity. It DOES NOT model the data flow.

10.7   FUNCTION DECOMPOSITION

Function Decomposition is used to facilitate the comprehension and management of function information within the business. Represented in form of a hierarchy, it depicts the decomposition of a function into smaller components with progressively increasing details. However, the hierarchy so developed is independent of organisational structure, existing or planned, and of current procedures.

The benefit of using function decomposition is that it acts as a basis for verifying the understanding of what the proposed system is about from a business standpoint and for planning more detailed analysis.

10.8   INTERACTION ANALYSIS

The Interaction Analysis technique investigates design inconsistencies by analysing data usage per business activity/function. By establishing a matrix with activity/function on the rows and data on the columns, which data are used by which activity/function is defined. Missing or extraneous activity/function (no data usage) and data (not used by any activity/function) are identified.

10.9   PROTOTYPING

Prototyping is a technique for building a quick and rough version of a desired system or parts of that system. The prototype illustrates the system to users and designers. It allows them to see flaws and so look for ways to improve the system. It serves as a communications vehicle for allowing persons who require the system to review the proposed user interaction with the system. For this purpose it is far more effective than reviewing paper specifications.

During User Design stage, prototypes are developed to help users visualise the system design and ensure it matches what the users want. The users exercise the prototypes and help the developers refine the system design. It is important that all prototypes and design data should be developed and kept in a CASE tool which will be directly taken to the Rapid Construction Stage in which further design details such as recovery from failures, handling exceptions, audit and security features, performance considerations, etc. are added for the development of the system.

Before prototyping sessions are actually conducted, plan the number and duration of the sessions that must be conducted with respect to the list of tasks that has been established in the Implementation Plan. Prototyping sessions should be geared towards the rapid construction of core business functions rather than those trivial functions that are largely of aesthetic value. Due attention should be paid to the timeboxes that have already been committed. Shortly after the completion of a specific prototype version, schedule review sessions with end users to collect their feedback and uses their feedback to further enhance the prototype. In planning these review sessions, it is important to invite all the stakeholders of the proposed system to attend the review sessions such the interests of all stakeholders are taken into account.

10.10   CLUSTER ANALYSIS

The Cluster Analysis technique investigates similarities among entities and identifies those that need to be dealt with as a group. These clusters then define the scope for further analysis and development. The main benefit of this technique is that it provides you with sensible collections of entities and processes to be handled together.

10.11   ENTITY STATE ANALYSIS

Entity State Analysis is the study of what can happen during the lives of instances of a given entity. Specifically, it identifies the significant states that an entity can have and the permitted state changes as a result of the processes that cause those changes. It clarifies our understanding of the conditions under which processes can operate on these entities.

10.12   EVENT DEPENDENCY ANALYSIS

Event Dependency Analysis is concerned with understanding the significant events in the business, the business processes they trigger, and the relationship between processes and entities. It relates happenings (events) to the business response (processes) as cause and effect within the business.