

2.1 The RAD life cycle composes of four stages:

A typical RAD life cycle is composed of the following Stages

2.1.1. Requirements Planning;

2.1.2 User Design;

2.1.3 Rapid Construction;

2.1.4 Transition.

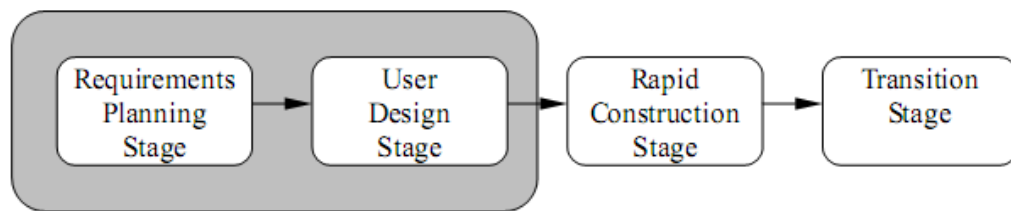


Fig 2.1 RAD life cycle Stages

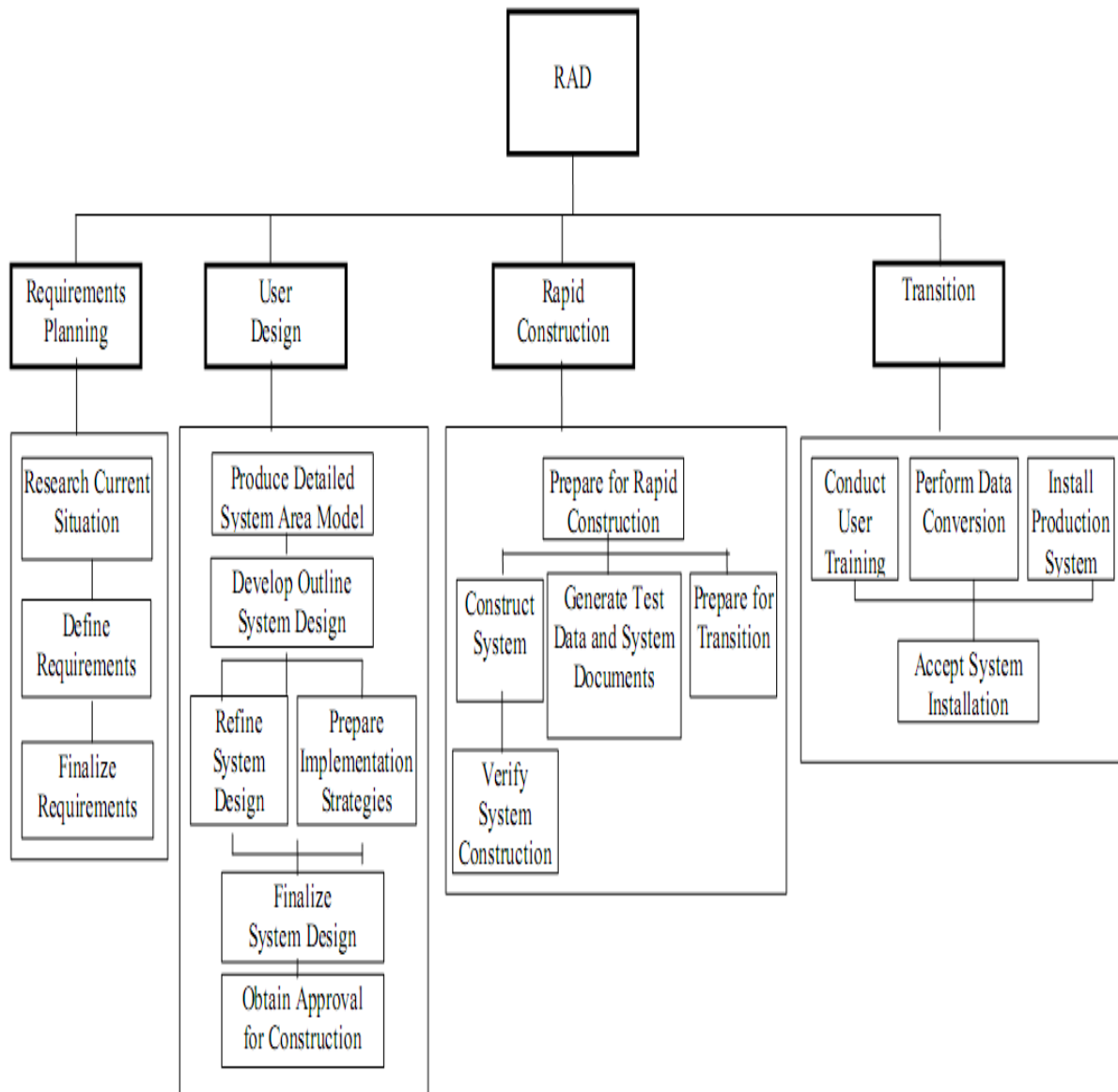


Figure 2.2 Overview of structure

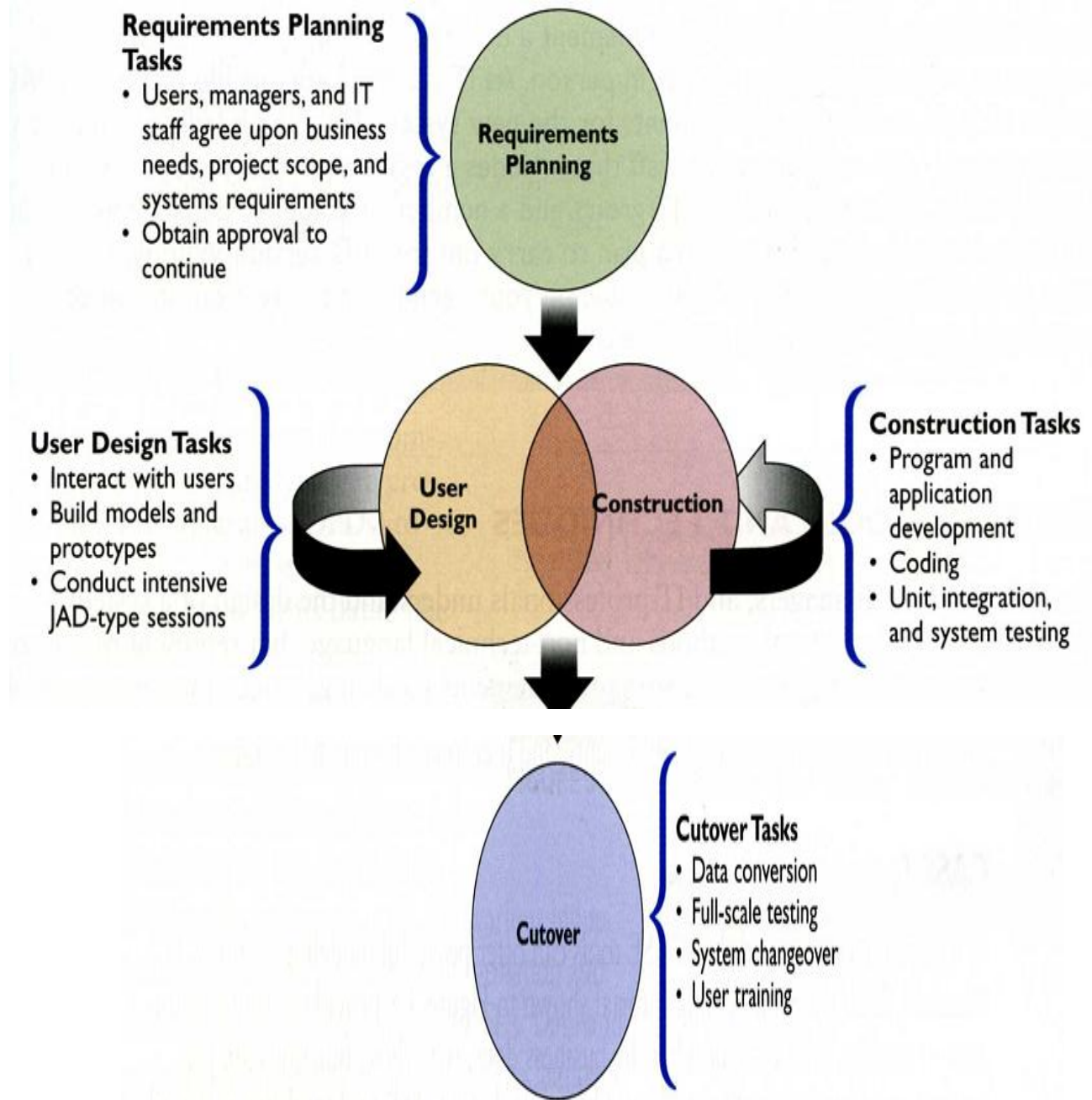


Figure 2.3 the four phases of the RAD model

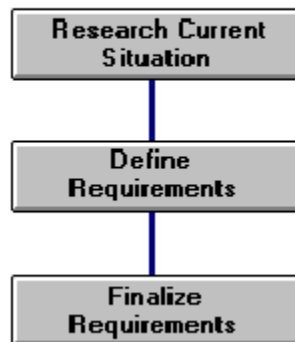
2.1.1 Requirements Planning (RP)

The objectives of the RP stage are (i) to establish a general understanding of the business problems that surround its development and eventual operation; (ii) to become familiar with existing systems and; (iii) to identify the business processes that will be supported by the proposed application.

The RP stage begins with a meeting or series of meetings, between the businesses and IS management. These meetings initiate the development process by establishing a mutual understanding of the objectives of the development project in general and the system in particular.

Business executives, end users and IS professionals, then, take part in Joint Requirements Planning (JRP) workshops which progress through a structured set of steps. During this stage, an outline of the system area and definition of the system scope are developed.

The tasks of the Requirements Planning are shown in the diagram below:



The tasks of this stage are:

(a) Research Current Situation

This task initiates the RP stage for the proposed system by researching the current environment. The information developed in the initial discussions provides a starting point for this investigation. This research is conducted for preparing the requirements definition through JRP workshops. The project team members become familiar with the current situation by investigating current systems, the information available in any existing CASE repositories, and by looking at similar systems for the purpose of looking for any reusable structures.

(b) 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. Management issues that affect subsequent development and transition activities are also identified.

Deliverables of this task are accomplished via a series of JRP workshops. Participants in these workshops include the managers of business areas that will be directly affected by the operation of the proposed system.

(c) Finalize Requirements

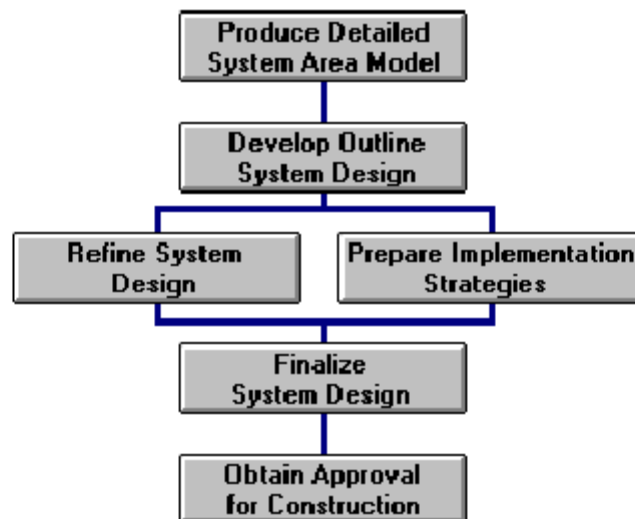
In this task, the scope of the proposed system is formally documented. An estimate of the cost and duration to implement the system is prepared. Sometimes, the cost and duration may already be fixed when the project commences. In this case, the scope must be well defined such that the project is still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

2.1.2 User Design (UD)

The objectives of the UD stage are (i) to analyze in detail business activities associated with the proposed system area; (ii) to analyze in detail the business data associated with the proposed system area; (iii) to develop the system structure in terms of the automated and manual functions that will comprise the system; (iv) to develop proposed screen layouts for the most important automated functions; (v) to select the appropriate construction approach for the system and; (vi) to prepare a work plan defining the steps necessary for transition of the system, the effort required to perform these steps, and a schedule by which these steps can be completed.

The UD stage produces a detailed system area model, an outline system design, and an implementation plan. End users participating in Joint Application Design (JAD) workshops perform the analysis and design activities associated with this stage.

The tasks of the User Design are shown in the diagram below:



The tasks of this stage are:

(a) 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. Once the detailed system area model is completed, it is possible to refine the scope to ensure that the critical functions will be delivered in the required time frame.

(b) Develop Outline System Design

An outline system design is developed by the JAD workshop participants using the system area model that has been completed and confirmed. Upon completion of the outline design, interactions between procedures (series of functions) and data are identified. That is, data usage per function is defined. The deliverables produced in this task include functions required for the system, reusable design components to be incorporated into the system design, system structures and tentative layouts of critical screens and reports to be supported by the system.

(c) Refine System Design

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

(d) 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 time box development can be adopted. An implementation plan is prepared listing all the 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 summarized into an overall project cost estimate.

(e) Finalize System Design

The outline design of the system and implementation plan needs to be finalized. Usually JAD workshops are conducted. Workshop participants suggest changes to the design based on their experience in reviewing the prototypes and further research into the system requirements. Open design issues are discussed and resolved except those with no impact to the system design such as cultural issues. The implementation plan for the system is presented and discussed, enabling workshop participants to point out any unforeseen problems or invalid assumptions in the plan.

(f) Obtain Approval for Construction

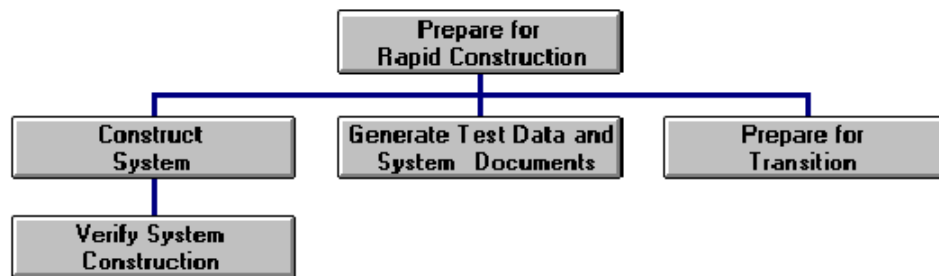
The results of the final JAD workshop are incorporated into the system design and implementation plan, and approval to proceed to the Rapid Construction Stage is sought.

2.1.3 Rapid Construction (RC)

The objectives of the Rapid Construction stage are (i) to complete the detailed design of the proposed system; (ii) to create and test the software that implements the proposed system; (iii) to generate a system that operates at an acceptable level of performance; (iv) to prepare documentation necessary to operate the proposed application; (v) to design, develop, and test the required transition software and; (vi) to perform the steps necessary to prepare for the conversion of the system to production status.

The design of the proposed system, initially described in the UD stage, is completed in the RC stage, and application software to implement that design is developed and tested. Activities to prepare for the transition of the system to production status are also performed.

The tasks of the Stage are shown in the diagram below:



The tasks of the RC stage are:

(a) Prepare for Rapid Construction

The development environment is finalized including workstations and workspace for the system developers, and the CASE software for their use. Additionally, the database is designed based on the preliminary data structure developed in the UD stage.

Finally, the testing strategy for the system is completed and the acquisition of facilities necessary to operate the system after it is constructed is initiated.

(b) Construct System

The detailed definition of the design of each function is completed, based on the requirements of the future end-users. Software to implement the automated functions is developed and tested.

(c) Generate Test Data and System Documents

This task involves developing the necessary test data that will verify the operational capacity of the system. This data will be used during the integration, system and acceptance tests. Documentation explaining how the system is to be operated by users and computer operations personnel is produced.

(d) Prepare for Transition

Activities in this task are to prepare for the subsequent conversion of the application into production status. These activities include:

- o To prepare a detailed work plan for transition activities and a contingency plan to cover failure of the converted application;
- o To develop software needed to convert data from existing formats into formats used by the application;
- o To develop transition procedures and arrangements;
- o To prepare a user training plan and training materials; and
- o To resolve organizational issues related to deployment of the new application.

(e) Verify System Construction

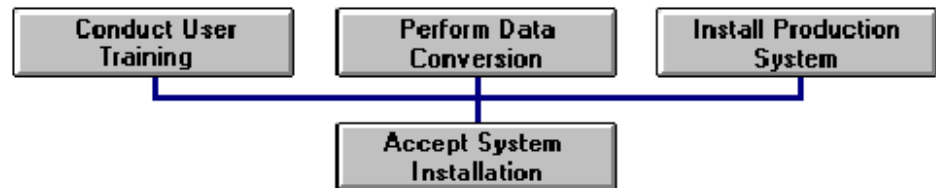
The system must be put through a series of tests to ensure that each component of the system and the complete system performs according to the user requirements.

2.1.4 Transition (TR)

The objectives of the TR stage are (i) to install the system in production operation with minimal disruption of normal business activity; (ii) to maximize the effectiveness of the system in supporting the intended business activities and; (iii) to identify potential future enhancement.

The system developed in the RC stage becomes operational in the TR stage. At this time, the developers prepare existing data for the new system and train users to operate the new application. They also provide support to resolve any problems that arise immediately after the application becomes operational.

The tasks of the Stage are shown in the diagram below:



The tasks of the TR stage are:

(a) 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. The training sessions are based on materials developed during the RC stage and follow the training plan that was developed at that time. The training may include a train-the-trainers program if a large number of users have to be trained or permanent training is required.

(b) 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.

(c) Install Production System

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

(d) Accept System Installation

The new system installation can be regarded as successful and will be accepted when it operates for a specified period within defined tolerances for performance, error rate and usability. This acceptance is based upon agreements among users, production/operations staff, support staff, and the information management organization about schedules, procedures, pricing policies, warranty agreements, software, and systems documentation.

2.2 Core Elements of Rapid Application Development

RAD has many core elements that make it a unique methodology including

2.2.1 Prototyping

2.2.2 Iterative development

2.2.3 Time boxing

2.2.4 Team members

2.2.5 Management approach

2.2.6 RAD tools.

2.2.1 Prototyping

A key aspect of RAD is the construction of a prototype for the purpose of jumpstarting design and flushing out user requirements. The objective is to build a feature light version of the finished product in as short an amount of time as possible, preferably days. The initial prototype serves as a proof of concept for the client, but more importantly serves as a talking point and tool for refining requirements.

Developing prototypes quickly is accomplished with Computer Aided Software Engineering CASE tools that focus on capturing requirements, converting them to a data model, converting the data model to a database, and generating code all in one tool. CASE tools were popular in the late 80's and early 90's, but as technology has changed (and COBOL has become obsolete) few tools take full advantage of the full potential of CASE tool technology. The Rational company is the most well known although its prototype generating potential is limited.

At Automated Architecture our Blue Ink product focuses on generating enterprise level web applications that serve as a prototype due to the speed with which they can be created (in minutes).

2.2.2 Iterative Development

Iterative development means creating increasingly functional versions of a system in short development cycles. Each version is reviewed with the client to produce requirements that feed the next version. The process is repeated until all functionality has been developed. The ideal length of iterations is between one day (which is closer to Agile Methodologies) and three weeks.

Each development cycle provides the user an opportunity to provide feedback, refine requirements, and view progress (in focus group session meetings). It is ultimately the iterative development that solves the problems inherent in the inflexible methodologies created in the 1970's.

2.2.3 Time Boxing

Time boxing is the process of putting off features to future application versions in order to complete the current version in as short amount of time as possible. Strict time boxing is an important aspect of RAD, because without it scope creep can threaten to lengthen development iterations, thus limiting client feedback, minimizing the benefits of iterative development, and potentially reverting the process back to a waterfall methodology approach.

2.2.3 Team Members

The RAD methodology recommends the use of small teams that consist of experienced, versatile, and motivated members that are able to perform multiple roles. As the client plays a vital role in the development process, dedicated client resources must be available during the initial Joint Application Development (JAD) sessions as well as Focus Group Sessions conducted at the end of development cycles. Development teams (also known as SWAT or Skilled Workers with Advanced Tools) should ideally have experience in Rapid Application Development and should have experience with the Computer Aided Software Engineering tools.

2.2.4 Management Approach

Active and involved management is vital to mitigate the risks of lengthened development cycles, client misunderstandings, and missed deadlines. Above all management must be stalwart and consistent in their desire to use the Rapid Application Development methodology. In addition to enforcing a strict timeline, management must focus on team member selection, team motivation, and on clearing bureaucratic or political obstacles.

2.2.5 RAD Tools

One of the primary objectives of the Rapid Application Development methodology developed by James Martin in the late 1980's was to take advantage of the latest technology available to speed development. Clearly the technology of the 1980's is obsolete today, but RAD's focus of the latest tools is as important today as it was when the methodology was initially created. This article has a section dedicated to tools following the process section.