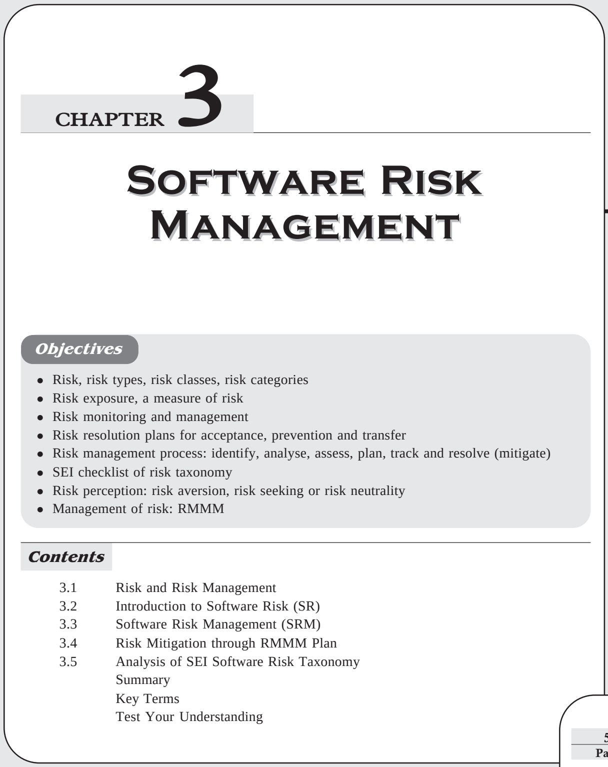
WALKTHROUGH



CHAPTER 3

SOFTWARE RISK MANAGEMENT

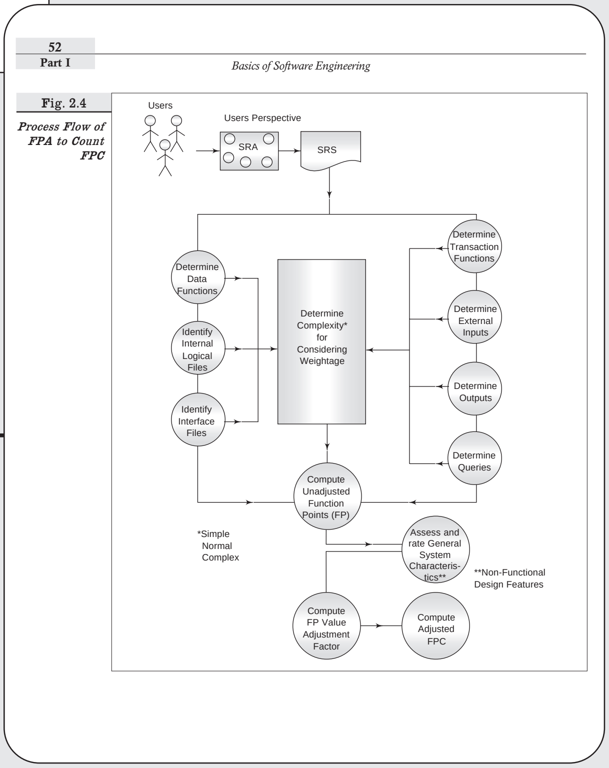
Objectives

- Risk, risk types, risk classes, risk categories
- Risk exposure, a measure of risk
- Risk monitoring and management
- Risk resolution plans for acceptance, prevention and transfer
- Risk management process: identify, analyse, assess, plan, track and resolve (mitigate)
- SEI checklist of risk taxonomy
- Risk perception: risk aversion, risk seeking or risk neutrality
- Management of risk: RMMM

Contents

Chapter Opener

Objectives provide a quick look into the concepts that will be discussed in the chapter
Contents provides a list of all the sections in the chapter.



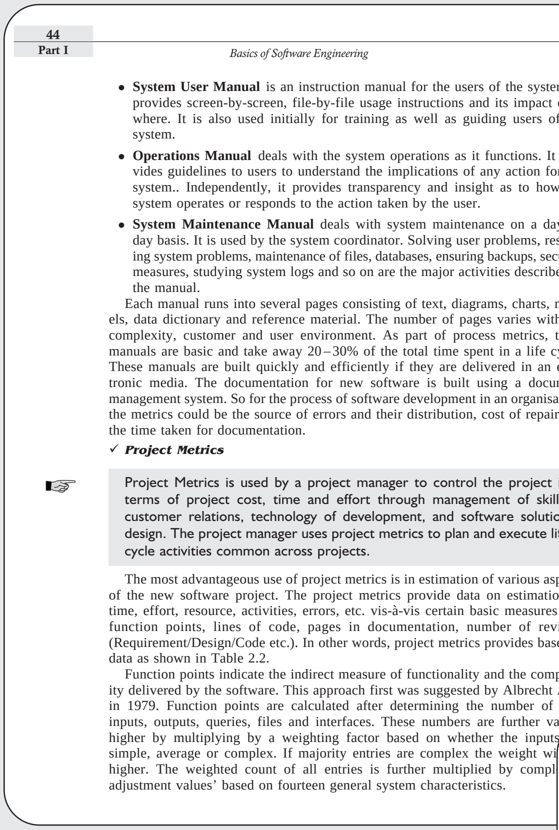
52 Part I *Basics of Software Engineering*

Fig. 2.4 ***Process Flow of FPA to Count FPC***

Users
Users Perspective
SRA
SRS
Determine Data Functions
Identify Internal Logical Files
Identify Interface Files
Determine Complexity* for Considering Weightage
Determine Transaction Functions
Determine External Inputs
Determine Outputs
Determine Queries
Compute Unadjusted Function Points (FP)
*Simple Normal Complex
Assess and rate General System Characteris-tics**
**Non-Functional Design Features
Compute FP Value Adjustment Factor
Compute Adjusted FPC

Figures are used exhaustively to illustrate the concepts and methods described

Walkthrough



- **System User Manual** is an instruction manual for the users of the system. It provides screen-by-screen, file-by-file usage instructions and its impact elsewhere. It is also used initially for training as well as guiding users of the system.
- **Operations Manual** deals with the system operations as it functions. It provides guidelines to users to understand the implications of any action for the system.. Independently, it provides transparency and insight as to how the system operates or responds to the action taken by the user.
- **System Maintenance Manual** deals with system maintenance on a day-to-day basis. It is used by the system coordinator. Solving user problems, resolving system problems, maintenance of files, databases, ensuring backups, security measures, studying system logs and so on are the major activities described in the manual.

Each manual runs into several pages consisting of text, diagrams, charts, models, data dictionary and reference material. The number of pages varies with the complexity, customer and user environment. As part of process metrics, these manuals are basic and take away 20 – 30% of the total time spent in a life cycle. These manuals are built quickly and efficiently if they are delivered in an electronic media. The documentation for new software is built using a document management system. So for the process of software development in an organisation, the metrics could be the source of errors and their distribution, cost of repair and the time taken for documentation.

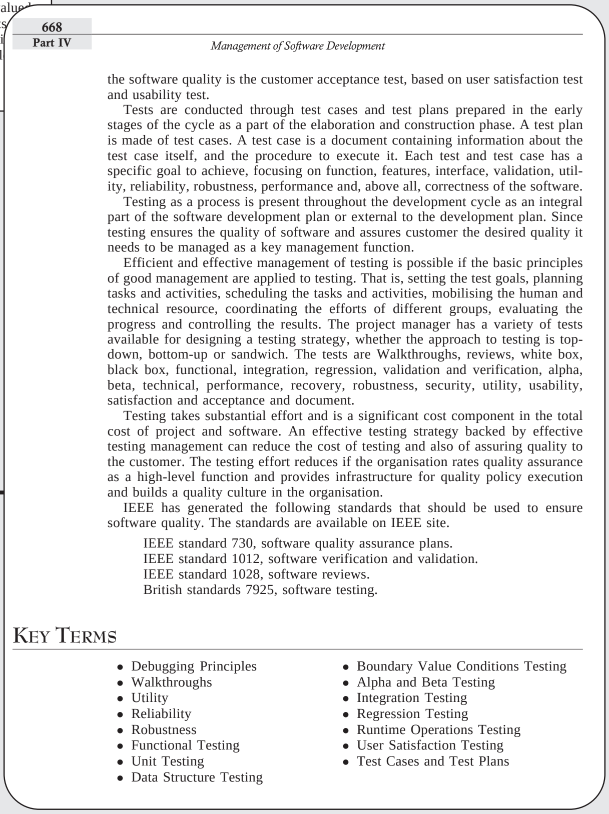
✓ ***Project Metrics***

☞ Project Metrics is used by a project manager to control the project in terms of project cost, time and effort through management of skills, customer relations, technology of development, and software solution design. The project manager uses project metrics to plan and execute life cycle activities common across projects.

The most advantageous use of project metrics is in estimation of various aspects of the new software project. The project metrics provide data on estimation of time, effort, resource, activities, errors, etc. vis-à-vis certain basic measures like function points, lines of code, pages in documentation, number of reviews (Requirement/Design/Code etc.). In other words, project metrics provides baseline data as shown in Table 2.2.

Function points indicate the indirect measure of functionality and the complexity delivered by the software. This approach first was suggested by Albrecht A. J. in 1979. Function points are calculated after determining the number of user inputs, outputs, queries, files and interfaces. These numbers are further valued higher by multiplying by a weighting factor based on whether the inputs simple, average or complex. If majority entries are complex the weight wi higher. The weighted count of all entries is further multiplied by compl adjustment values' based on fourteen general system characteristics.

☞ icon highlights important points—definitions, concepts, formulae—which need special attention of the readers.



the software quality is the customer acceptance test, based on user satisfaction test and usability test.

Tests are conducted through test cases and test plans prepared in the early stages of the cycle as a part of the elaboration and construction phase. A test plan is made of test cases. A test case is a document containing information about the test case itself, and the procedure to execute it. Each test and test case has a specific goal to achieve, focusing on function, features, interface, validation, utility, reliability, robustness, performance and, above all, correctness of the software.

Testing as a process is present throughout the development cycle as an integral part of the software development plan or external to the development plan. Since testing ensures the quality of software and assures customer the desired quality it needs to be managed as a key management function.

Efficient and effective management of testing is possible if the basic principles of good management are applied to testing. That is, setting the test goals, planning tasks and activities, scheduling the tasks and activities, mobilising the human and technical resource, coordinating the efforts of different groups, evaluating the progress and controlling the results. The project manager has a variety of tests available for designing a testing strategy, whether the approach to testing is top-down, bottom-up or sandwich. The tests are Walkthroughs, reviews, white box, black box, functional, integration, regression, validation and verification, alpha, beta, technical, performance, recovery, robustness, security, utility, usability, satisfaction and acceptance and document.

Testing takes substantial effort and is a significant cost component in the total cost of project and software. An effective testing strategy backed by effective testing management can reduce the cost of testing and also of assuring quality to the customer. The testing effort reduces if the organisation rates quality assurance as a high-level function and provides infrastructure for quality policy execution and builds a quality culture in the organisation.

IEEE has generated the following standards that should be used to ensure software quality. The standards are available on IEEE site.

IEEE standard 730, software quality assurance plans.
IEEE standard 1012, software verification and validation.
IEEE standard 1028, software reviews.
British standards 7925, software testing.

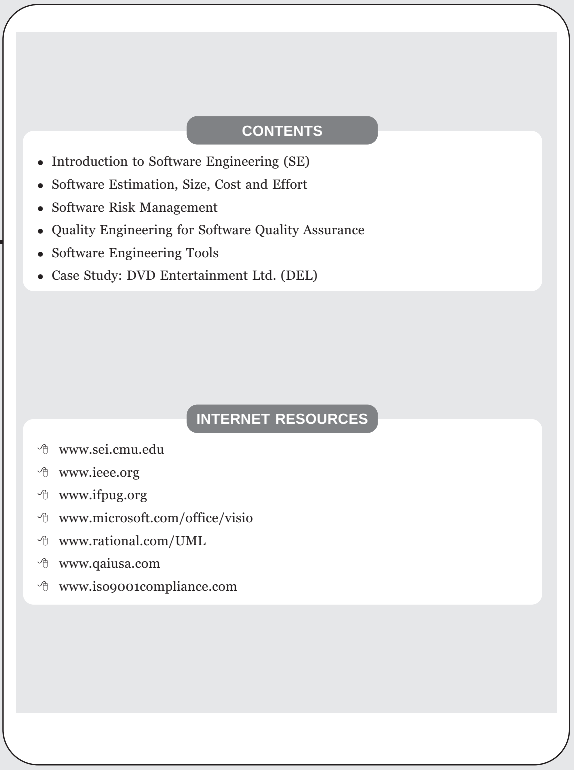
Key Terms

- Debugging Principles
- Walkthroughs
- Utility
- Reliability
- Robustness
- Functional Testing
- Unit Testing
- Data Structure Testing
- Boundary Value Conditions Testing
- Alpha and Beta Testing
- Integration Testing
- Regression Testing
- Runtime Operations Testing
- User Satisfaction Testing
- Test Cases and Test Plans

Key Terms give a list of the important words discussed in the chapter.

WALKTHROUGH

Internet Resources points to where the readers can find additional information on the Web

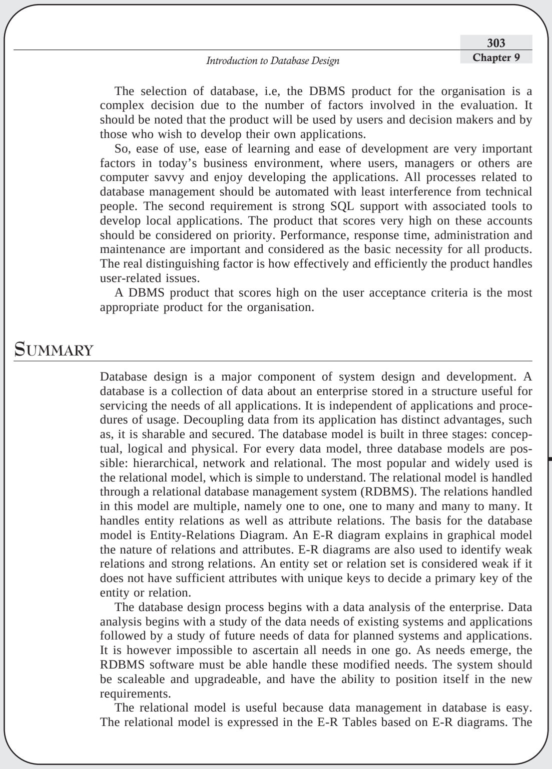


CONTENTS

- Introduction to Software Engineering (SE)
- Software Estimation, Size, Cost and Effort
- Software Risk Management
- Quality Engineering for Software Quality Assurance
- Software Engineering Tools
- Case Study: DVD Entertainment Ltd. (DEL)

INTERNET RESOURCES

- www.sei.cmu.edu
- www.ieee.org
- www.ifpug.org
- www.microsoft.com/office/visio
- www.rational.com/UML
- www.qaiusa.com
- www.iso9001compliance.com



303
Chapter 9
Introduction to Database Design

The selection of database, i.e, the DBMS product for the organisation is a complex decision due to the number of factors involved in the evaluation. It should be noted that the product will be used by users and decision makers and by those who wish to develop their own applications.

So, ease of use, ease of learning and ease of development are very important factors in today's business environment, where users, managers or others are computer savvy and enjoy developing the applications. All processes related to database management should be automated with least interference from technical people. The second requirement is strong SQL support with associated tools to develop local applications. The product that scores very high on these accounts should be considered on priority. Performance, response time, administration and maintenance are important and considered as the basic necessity for all products. The real distinguishing factor is how effectively and efficiently the product handles user-related issues.

A DBMS product that scores high on the user acceptance criteria is the most appropriate product for the organisation.

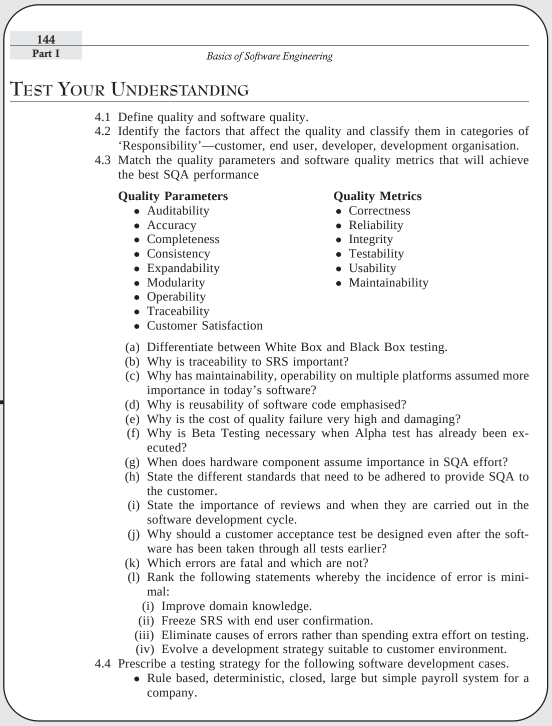
SUMMARY

Database design is a major component of system design and development. A database is a collection of data about an enterprise stored in a structure useful for servicing the needs of all applications. It is independent of applications and procedures of usage. Decoupling data from its application has distinct advantages, such as, it is sharable and secured. The database model is built in three stages: conceptual, logical and physical. For every data model, three database models are possible: hierarchical, network and relational. The most popular and widely used is the relational model, which is simple to understand. The relational model is handled through a relational database management system (RDBMS). The relations handled in this model are multiple, namely one to one, one to many and many to many. It handles entity relations as well as attribute relations. The basis for the database model is Entity-Relations Diagram. An E-R diagram explains in graphical model the nature of relations and attributes. E-R diagrams are also used to identify weak relations and strong relations. An entity set or relation set is considered weak if it does not have sufficient attributes with unique keys to decide a primary key of the entity or relation.

The database design process begins with a data analysis of the enterprise. Data analysis begins with a study of the data needs of existing systems and applications followed by a study of future needs of data for planned systems and applications. It is however impossible to ascertain all needs in one go. As needs emerge, the RDBMS software must be able handle these modified needs. The system should be scaleable and upgradeable, and have the ability to position itself in the new requirements.

The relational model is useful because data management in database is easy. The relational model is expressed in the E-R Tables based on E-R diagrams. The

Summary gives the essence of each chapter in brief



144
Part I
Basics of Software Engineering

TEST YOUR UNDERSTANDING

4.1 Define quality and software quality.
4.2 Identify the factors that affect the quality and classify them in categories of 'Responsibility'—customer, end user, developer, development organisation.
4.3 Match the quality parameters and software quality metrics that will achieve the best SQA performance

Quality Parameters
- Auditability
- Accuracy
- Completeness
- Consistency
- Expandability
- Modularity
- Operability
- Traceability
- Customer Satisfaction

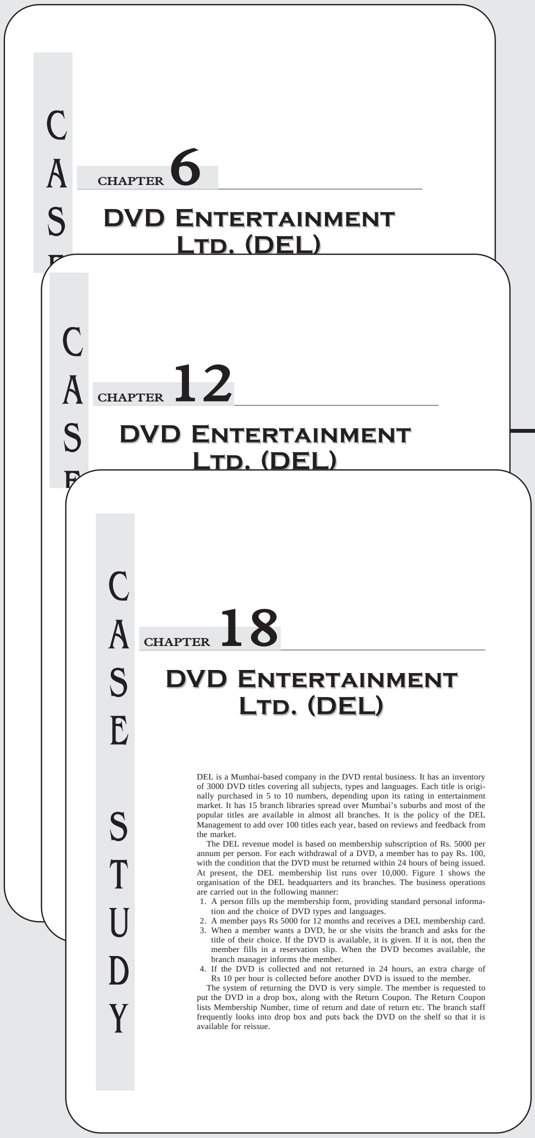
Quality Metrics
- Correctness
- Reliability
- Integrity
- Testability
- Usability
- Maintainability

(a) Differentiate between White Box and Black Box testing.
(b) Why is traceability to SRS important?
(c) Why has maintainability, operability on multiple platforms assumed more importance in today's software?
(d) Why is reusability of software code emphasised?
(e) Why is the cost of quality failure very high and damaging?
(f) Why is Beta Testing necessary when Alpha test has already been executed?
(g) When does hardware component assume importance in SQA effort?
(h) State the different standards that need to be adhered to provide SQA to the customer.
(i) State the importance of reviews and when they are carried out in the software development cycle.
(j) Why should a customer acceptance test be designed even after the software has been taken through all tests earlier?
(k) Which errors are fatal and which are not?
(l) Rank the following statements whereby the incidence of error is minimal:
  (i) Improve domain knowledge.
  (ii) Freeze SRS with end user confirmation.
  (iii) Eliminate causes of errors rather than spending extra effort on testing.
  (iv) Evolve a development strategy suitable to customer environment.

4.4 Prescribe a testing strategy for the following software development cases.
- Rule based, deterministic, closed, large but simple payroll system for a company.

Test Your Understanding will help the readers review their comprehension of the concepts discussed in each chapter

WALKTHROUGH



CAS

CHAPTER 6

DVD ENTERTAINMENT LTD. (DEL)

CAS

CHAPTER 12

DVD ENTERTAINMENT LTD. (DEL)

CASE STUDY

CHAPTER 18

DVD ENTERTAINMENT LTD. (DEL)

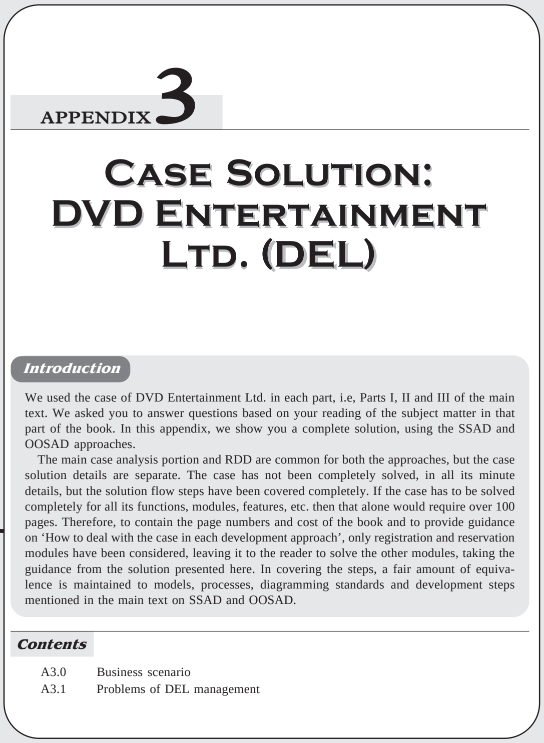
DEL is a Mumbai-based company in the DVD rental business. It has an inventory of 3000 DVD titles covering all subjects, types and languages. Each title is originally purchased in 5 to 10 numbers, depending upon its rating in entertainment market. It has 15 branch libraries spread over Mumbai's suburbs and most of the popular titles are available in almost all branches. It is the policy of the DEL Management to add over 100 titles each year, based on reviews and feedback from the market.

The DEL revenue model is based on membership subscription of Rs. 5000 per annum per person. For each withdrawal of a DVD, a member has to pay Rs. 100, with the condition that the DVD must be returned within 24 hours of being issued. At present, the DEL membership list runs over 10,000. Figure 1 shows the organisation of the DEL headquarters and its branches. The business operations are carried out in the following manner:

1. A person fills up the membership form, providing standard personal information and the choice of DVD types and languages.
2. A member pays Rs 5000 for 12 months and receives a DEL membership card.
3. When a member wants a DVD, he or she visits the branch and asks for the title of their choice. If the DVD is available, it is given. If it is not, then the member fills in a reservation slip. When the DVD becomes available, the branch manager informs the member.
4. If the DVD is collected and not returned in 24 hours, an extra charge of Rs 10 per hour is collected before another DVD is issued to the member.

The system of returning the DVD is very simple. The member is requested to put the DVD in a drop box, along with the Return Coupon. The Return Coupon lists Membership Number, time of return and date of return etc. The branch staff frequently looks into drop box and puts back the DVD on the shelf so that it is available for reissue.

A running **Case Study** throughout the text at the end of every part with questions related to the topics just learnt.



APPENDIX 3

CASE SOLUTION: DVD ENTERTAINMENT LTD. (DEL)

Introduction

We used the case of DVD Entertainment Ltd. in each part, i.e, Parts I, II and III of the main text. We asked you to answer questions based on your reading of the subject matter in that part of the book. In this appendix, we show you a complete solution, using the SSAD and OOSAD approaches.

The main case analysis portion and RDD are common for both the approaches, but the case solution details are separate. The case has not been completely solved, in all its minute details, but the solution flow steps have been covered completely. If the case has to be solved completely for all its functions, modules, features, etc. then that alone would require over 100 pages. Therefore, to contain the page numbers and cost of the book and to provide guidance on 'How to deal with the case in each development approach', only registration and reservation modules have been considered, leaving it to the reader to solve the other modules, taking the guidance from the solution presented here. In covering the steps, a fair amount of equivalence is maintained to models, processes, diagramming standards and development steps mentioned in the main text on SSAD and OOSAD.

Contents

A3.0 Business scenario
A3.1 Problems of DEL management

Case Solution in the appendices gives an integrated solution to the questions raised in the case study using both the approaches, SSAD and OOSAD.