Chapter Six

Requirements Management

Introduction

❑Requirements management is the process of managing changes to a system's requirements.

- New requirements emerge and existing requirements change at all stages of the system development process.

➢It is often the case that 50% of a system's requirements will be modified before it is put into service.

- Clearly, this can cause serious problems for system developers.
- To minimize difficulties, requirements management is necessary where changes to the requirements are documented and controlled.

❑The impact of proposed changes to requirements must be assessed and, as requirements changes are accepted, system design and implementation must then be modified.

➢If changes are not controlled, low priority changes may be implemented before high priority changes and expensive modifications to the system which are not really necessary may be approved.

Cont'd…

- Requirements Management is carried out in parallel with other requirements engineering processes and continues after the first version of the requirements document has been delivered.
- The requirements continue to change during system development and these changes must be managed.

- **The principal concerns of requirements management are:**

1. Managing changes to agreed requirements
2. Managing the relationships between requirements
3. Managing the dependencies between the requirements document and other documents produced during software engineering process.

Cont'd…

- **Changes to system requirements may be:**
  - due to errors and misunderstandings in the requirements engineering process/ design or implementation problems.
  - new requirements may emerge as stakeholders develop a better understanding of the system.
  - as a result of changing external circumstances.
  - the strategy or priorities of the business buying the system may change as a result of economic changes or new competitors in its market.
  - new information about the systems environment may become available.

Cont'd…

- Requirements cannot be managed effectively without requirements traceability.
- A requirement is traceable if you can discover:
  - who suggested the requirement,
  - why the requirement exists,
  - what requirements are related to it and
  - how that requirement relates to other information such as systems designs, implementations and user documentation.
- Traceability information is used to find other requirements which might be affected by proposed changes.

Cont'd…

- CASE tools for supporting the requirements management process provide facilities such as:

1. a database system for storing requirements
2. document analysis and generation facilities to help construct a requirements database and to help create requirements documents
3. change management facilities which help to ensure that changes are properly assessed and costed
4. traceability facilities which help requirements engineers find dependencies between system requirements.

Stable and Volatile Requirements

- Requirements change is unavoidable and does not imply poor requirements engineering practice.
- Table 6.1 factors leading to requirements change

Change Factor	Description
Requirements errors, conflicts and Inconsistencies	• As the requirements are analyzed and implemented, errors and inconsistencies emerge and must be corrected. • These problems may be discovered during requirements analysis and validation or later in the development process.

Cont'd…

Change Factor	Description
Evolving customer/end-user knowledge of the system	• As requirements are developed, customers and end-users develop a better understanding of what they really require from a system.
Technical, schedule or cost problems	• Problems may be encountered in implementing a requirement. • It may be too expensive or take too long to implement certain requirements.
Changing customer priorities	• Customer priorities change during system development as a result of a changing business environment, the emergence of new competitors, staff changes, etc.

Cont'd…

Change Factor	Description
Environmental changes	• The **environment in which the system is to be installed** may change so that the system requirements have to change to maintain compatibility
Organizational changes	• The organization which intends to use the system may change its structure and processes resulting in new system requirements

Cont'd…

- Although change is inevitable, it is usually the case that some requirements are more stable than others.

Stable requirements

- are concerned with the essence of a system and its application domain.
- They **change more slowly** than volatile requirements.

Volatile requirements

- are specific to the instantiation of the system in a particular environment and for a particular customer.

Cont'd…

Example:

❑ **Consider a system for managing student records in a university.**

- Such a system will always have to have information about students, the courses they have taken and the assessment of how well they performed in these courses.
  - This are stable features of the system.
- The system may also maintain information about the students attendance at classes, recommended course groupings, standard letters sent to students.
  - Requirements for these are more volatile.
- Courses may be taught remotely over the Internet so that class attendance then means something completely different, course groupings change as a subject evolves and the standard letters also change both with course groupings and with changes to administration.
  - These are, therefore, volatile features of the system.

Cont'd…

- There are at least 4 different types of volatile requirement.

1. Mutable requirements

- These are requirements which change because of changes to the environment in which the system is operating.
- For example, the requirements for a system which computes tax deductions evolve as the tax laws are changed.

2. Emergent requirements

- These are requirements which cannot be completely defined when the system is specified but which emerge as the system is designed and implemented.
- For example, it may not be possible to specify, in advance, the details of how information should be displayed.
  - As stakeholders see examples of possible presentations, they may think of new ways of presenting information that would be useful to them.

Cont'd…

3. Consequential requirements

- These are requirements which are based on assumptions about how the system will be used.
- When the system is put to use, some of these assumptions will be wrong.
- Users will adapt to the system and find new ways to use its functionality.
- This will result in demands from users for system changes and modifications.

4. Compatibility requirements

- These are requirements which depend on other equipment or processes.
- As this equipment changes, these requirements also evolve.
- For example, an instrument system in a power station control room may have to be modified when a new type of information display is added.

Requirements Identification and Storage

- An essential pre-requisite for requirements management is that every requirement must have some kind of unique identification.
- Consequently, effective requirements management is impossible.

➢ The commonest approach to requirements identification is based on numbering the requirements according to the **chapter** and **section** of the requirements document where the requirement is included.

  ➢ Therefore,

    - the 6th requirement in the 2nd section in Chapter 4 would be number 4.2.6;
    - the 8th requirement in the 3rd section of Chapter 2 would be assigned the number 2.3.8; and so on.

Cont'd…

- There are two problems with this style of requirements identification.

1. It isn't possible to assign a unique number until the requirements document has been finished.
   - The chapter and section organization must be stable.
   - When a requirement is elicited, it is unclear where it will appear in the requirements document.
     - It can't be assigned a number and therefore can't be referenced by other requirements.
2. Assigning an identifier based on chapter and section numbers implicitly classifies the requirement.
   - It suggests that the requirement is closely related to other requirements with similar identifiers.
   - Document readers may be misled into thinking that there are no other important relationships between that requirement and other requirements elsewhere in the document.

Cont'd…

- There are alternative approaches to requirements identification which address these problems.
- Some of these are shown in Table 6.2 on the next slide.
- If a requirement has a unique identification number as well as a paragraph number, references to the requirement may use its unique identifier.
- This allows for easy rearrangement of the requirements document but can sometimes confuse requirements readers who mix-up the requirements identifier and the paragraph number.

Cont'd…

Table 6.2 Techniques for requirements identification

Identification method	Description
Dynamic renumbering	• Some word processing systems allow for automatic renumbering of paragraphs and the inclusion of cross- references. • You can therefore assign a number to a requirement at any time. • As you re-organize your document and add new requirements, the system keeps track of the cross-reference. • It automatically renumbers your requirement depending on its chapter, section and position within the section. • All references to the requirement are also renumbered.

Cont'd…

Identification method	Description
Database record identification	• When a requirement is identified it is immediately entered in a requirements database and a database record identifier is assigned. • This database identifier is used in all subsequent references to the requirement.
Symbolic identification	• Requirements can be identified by giving them a symbolic name which is associated with the requirement itself. • For example, EFF-I, EFF-2, EFF-3 may be used for requirements which relate to the efficiency of the system. • The problem with this is that it is sometimes difficult to classify requirements and assign a meaningful mnemonic to them.

Cont'd…

- The requirements must be maintained in a database with each requirement represented as one or more database entities.
  - ✓The facilities of the database can be used to link related requirements and it is usually possible to formulate fairly complex database queries to identify requirements groupings.
  - ✓The database may provide some version control facilities or, at least, provision for these facilities to be implemented.
  - ✓Databases usually include facilities for browsing and report generation.
  - ✓Related requirements are linked and simple scripts which scan the database, extract parts of the record for each requirement and generate skeletons of the requirements document may be developed.

Cont'd…

- Relational databases were designed for storing and managing large numbers of records which have the same structure and minimal links between them.

➢ A requirements database, however, may have relatively few records (hundreds rather than hundreds of thousands) each of which includes many links such as links to documents, text files and other requirements.

- Maintaining these links is possible with a relational database but it is inefficient.
- It requires operations on several different tables.
- For very large numbers of requirements, this type of database may be too slow.

Cont'd…

- **Object-oriented databases** have been developed relatively recently and are structurally more suited to requirements management.

- They are better than relational databases:
  - ✓ when there are many different types of entity to be managed and
  - ✓ where there are direct links between different entities in the database.

- They allow different types of information to be maintained in different objects and managing links between objects is fairly straightforward.

- The figure on the next slide shows a set of object classes which could be defined in an object- oriented requirements database.

Cont'd…

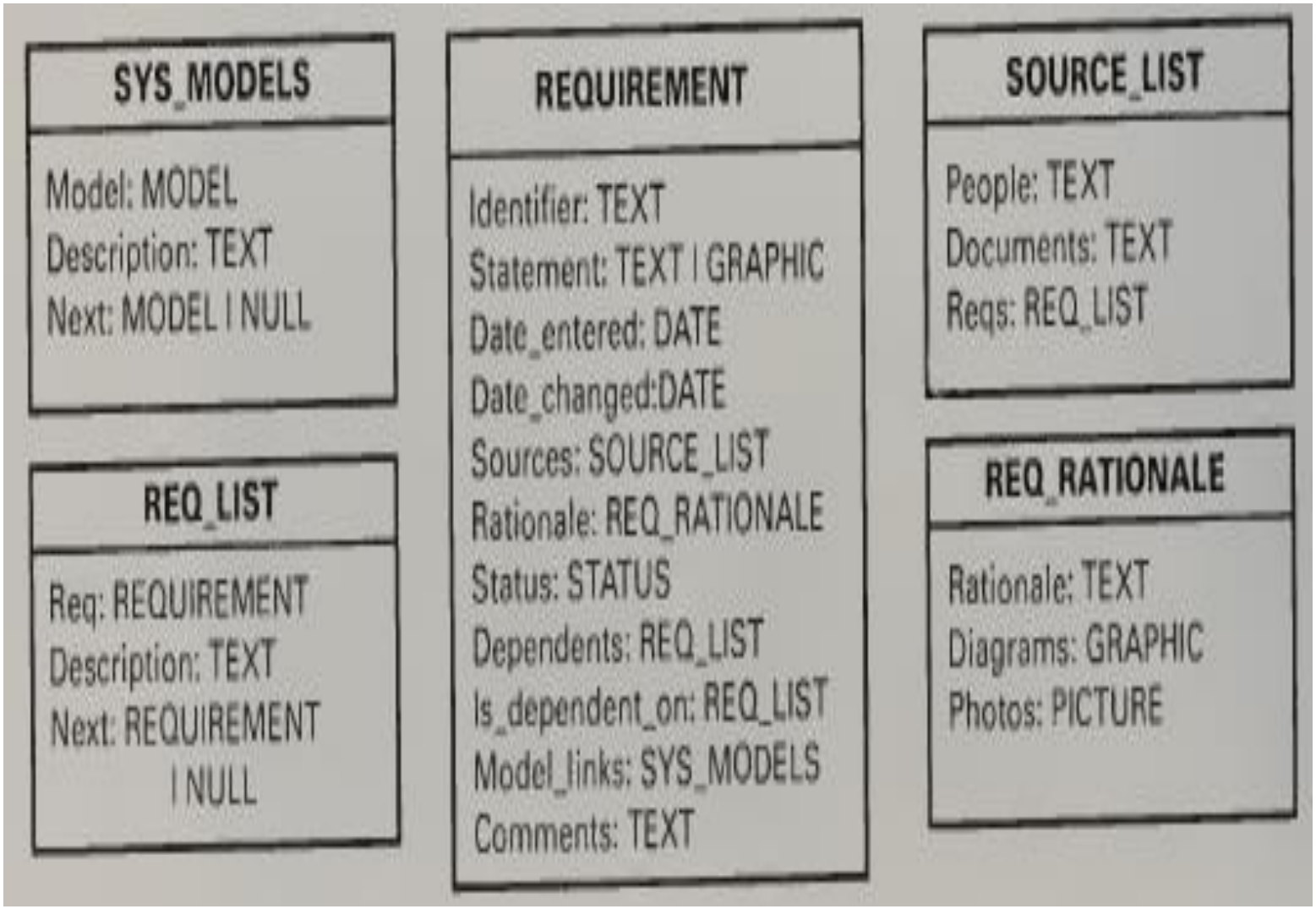


Figure Object classes for a requirements database.

Cont'd…

- The central class is REQUIREMENT which has 11 associated attributes.

1. Identifier

- This is a simple text string which is assigned when a requirement object is created and entered in the database.

2. Statement

- This is a statement of the requirement which may be natural language text or a graphical description of some kind.

3. Date entered

- The date that the requirement was originally entered in the database.

4. Date changed

- The date of the last alteration to the requirement.

Cont'd…

5. Sources

- This is a reference to one or more of the sources of the requirement.
- This helps with analysis when changes to the requirement are proposed.

6. Rationale

- This is a reference to a set of information which provides a rationale explaining why the requirement has been included.
- The associated information may include text, diagrams or photographs.

7. Status

- This is a variable representing the status of the requirement.
- The status may be '**proposed**', '**under review**', '**accepted**', or '**rejected**'.
- Rejected requirements should be maintained in the database as they may be proposed again in future.
- The analysis of the new proposal is simplified if previous information is available.

Cont'd…

8. Dependents

- This is a list of references to requirements which are dependent on this requirement (i.e. if this requirement is changed, they may also have to be changed)

9. Is_dependent_on

- This is a list of references to requirements on which this requirement depends.
- The Is_dependent_on relationship is therefore the inverse of Dependents.

10. Model links

- This is a link to one or more system models which add detail to the requirement

11. Comments

- This is any other information which may be useful.
- In practice, it is almost impossible to define a schema which covers everything, and having a general description field is often very useful.

Change Management

- Change management is concerned with, the procedures, processes and standards which are used to manage changes to system requirements.
- Change management ensures that similar information is collected for each proposed change and that overall judgements are made about the costs and benefits of proposed changes.
- Without formal change management, it is impossible to ensure that proposed changes to the requirements support the fundamental business goals.

➢To ensure a consistent approach to change management, organizations may define a set of change management policies.

➢This cover:

1. The change request process and the information required to process each change request.

Cont'd…

2. The process used to analyze the impact and costs of change and the associated traceability information.
3. The membership of the body which formally considers change requests.
   - It is important to have some 'independent' group who considers change requests as they can make an objective decision about the contribution of the change to the overall goals of the system and the cost-effectiveness of the change.
   - In military projects, this is called the 'Change Request Board' or 'Change Control Board' but in other organizations a less formal group may be used.
4. The software support (if any) for the change control process.

Change Management Processes

- The process of requirements change management consists of a set of activities for documenting, reporting, analyzing, costing and implementing changes to management processes such as change management to delivered software.

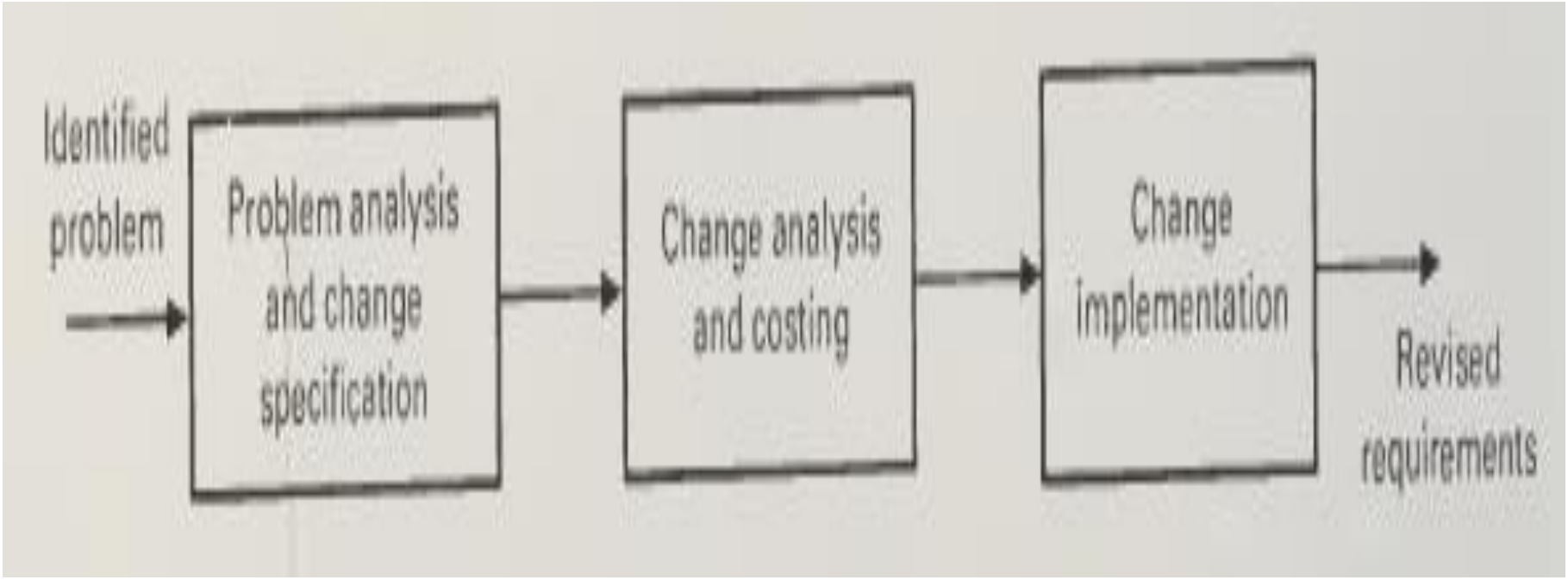


Figure Stages in the change management process

Cont'd…

- The change management process can be thought of as **a three-stage process** as shown in the above Figure.

1. Some requirements problem is identified.
   - The requirements are analyzed using problem information and requirements changes are proposed.
2. The proposed changes are analyzed to see how many requirements (and, if necessary, system components) are affected by the change and roughly how much it would cost, in both time and money, to make the change.
3. The change is implemented.
   - A set of amendments to the requirements document or a new document version is produced.

Cont'd…

- The specific processes for problem analysis and change implementation are dependent on:
  - the type of change,
  - the requirements affected and
  - the type of requirements document.
- However, change analysis and costing is a more general process, as shown in the next figure.

Cont'd…

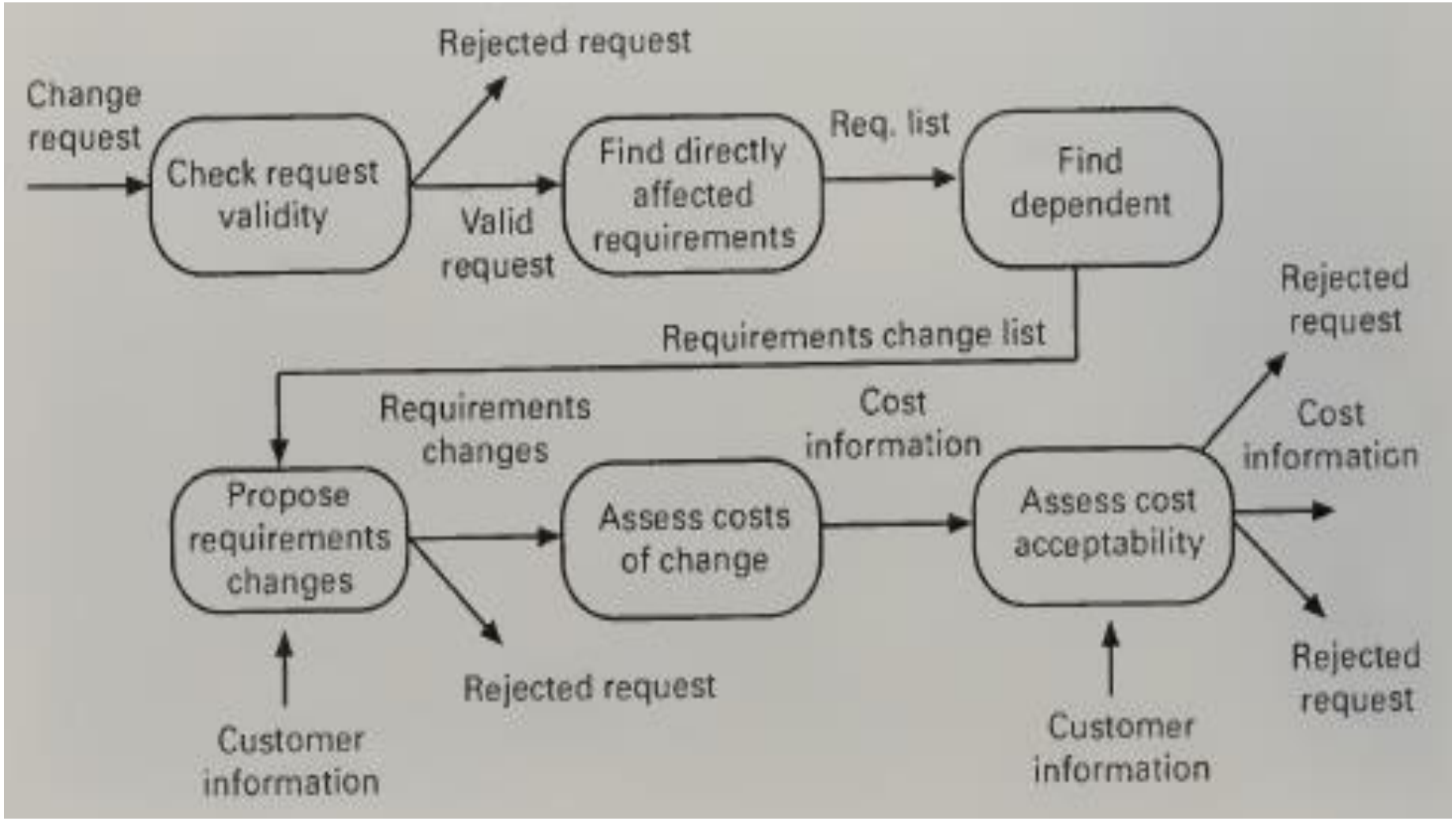


Figure 6.5 The change analysis and costing process.

Cont'd…

- There are six basic activities in the change analysis process:

1. The change request is checked to see if it is valid.
   - Sometimes, customers misunderstand the requirements and suggest unnecessary changes.
2. The requirements which are directly affected by the change are discovered.
3. Traceability information is used to find dependent requirements which may also be affected by the change.
4. The actual changes which must be made to the requirements are proposed.
   - There may be consultation with customers at this stage to ensure that they are happy with these changes.

Cont'd…

5. The costs of making the changes are estimated.
   - This estimate should include both the effort required to make the change and the amount of calendar time needed.
   - The availability of resources to implement the change must also be considered.
6. Negotiations with customers are held to check if the costs of the proposed changes are acceptable to them.
   - At this stage, it may be necessary to go back to step 4 to propose alternative changes if the customer feels that the change proposal is too expensive.
   - Alternatively, the customer may modify the change request so that the whole process has to be repeated

Cont'd…

- The change request may be rejected at three stages in this process.

I. If the change request is invalid:
   - this normally arises if a customer has misunderstood something about the requirements and proposed a change which isn't necessary.

II. If the change request results in consequential changes which are unacceptable to the user:
   - for example, a change request to decrease the time required to process a transaction may mean that fewer concurrent transactions can be handled.

III. If the cost of implementing the change is too high or takes too long.

Tool support for change management

- Change management involves handling large amounts of information and passing it between individuals in an organization.
- It is often necessary to keep track of which changes have been proposed, which have been implemented, which are still under consideration, etc.
- Support for requirements change management may be provided by:
  - specialized requirements management tools or
  - by CASE tools designed to support software configuration management.

Cont'd…

- The capabilities which these tools may provide are:

1. electronic change request forms which are filled in by different participants in the process
2. a database to store and manage these forms
3. a change model which may be instantiated so that people responsible for one stage of the process know who is responsible for the next process activity
4. electronic transfer of forms between people with different responsibilities and electronic mail notification when activities have been completed
5. in some cases, direct links to a requirements database.
   - In some cases, this may support the maintenance of multiple versions of a requirement with change information indicating why new requirements versions have been derived.
   - Only the most sophisticated tools provide this functionality.

Cont'd…

- The general problem with these tools is that they all have their own implicit model of the change process.
- Organizations which adopt these tools must conform to that model.
- Special-purpose tools are also fairly expensive and there may be difficulties when integrating these with other CASE tools used in an organization.
- For these reasons, specialized change support tools are mostly used in large organizations such as aerospace companies who are involved in very large projects.
- General purpose tools such as word processors, spreadsheets, and electronic mail systems may be used to implement a more limited change management system.

Traceability

- A critical part of the requirements change management process is the assessment of the impact of a change on the rest of the system.
- To carry out impact assessment, information about requirements dependencies, requirements rationale and the implementation of requirements should be maintained to supplement the information in the requirements document.
  - This is usually called traceability information.
- Change impact assessments depend on this traceability information to find out which requirements are affected by a proposed change.

Cont'd…

- Davis (1993) has classified traceability information into four types.

1. Backward-from traceability
   - links **requirements to their sources** in other documents or people.
2. Forward-from traceability
   - links **requirements to the design and implementation** components.
3. Backward-to traceability
   - links design and implementation components back to requirements.
4. Forward-to traceability
   - links other documents (which may have preceded the requirements document) to relevant requirements.

Cont'd…

- In practice, it is impossibly expensive to collect and manage all types of traceability information.
- Project managers should define traceability policies setting out what essential traceability information must be maintained.

➢ Davis's classifications are a useful way to understand the concept of traceability as the traceability information can be visualized as arrows going forwards and backwards from different documents.

➢ This is illustrated in the following figure which shows how a statement of requirements can include traceability links to and from a design specification and a business plan.

Cont'd…

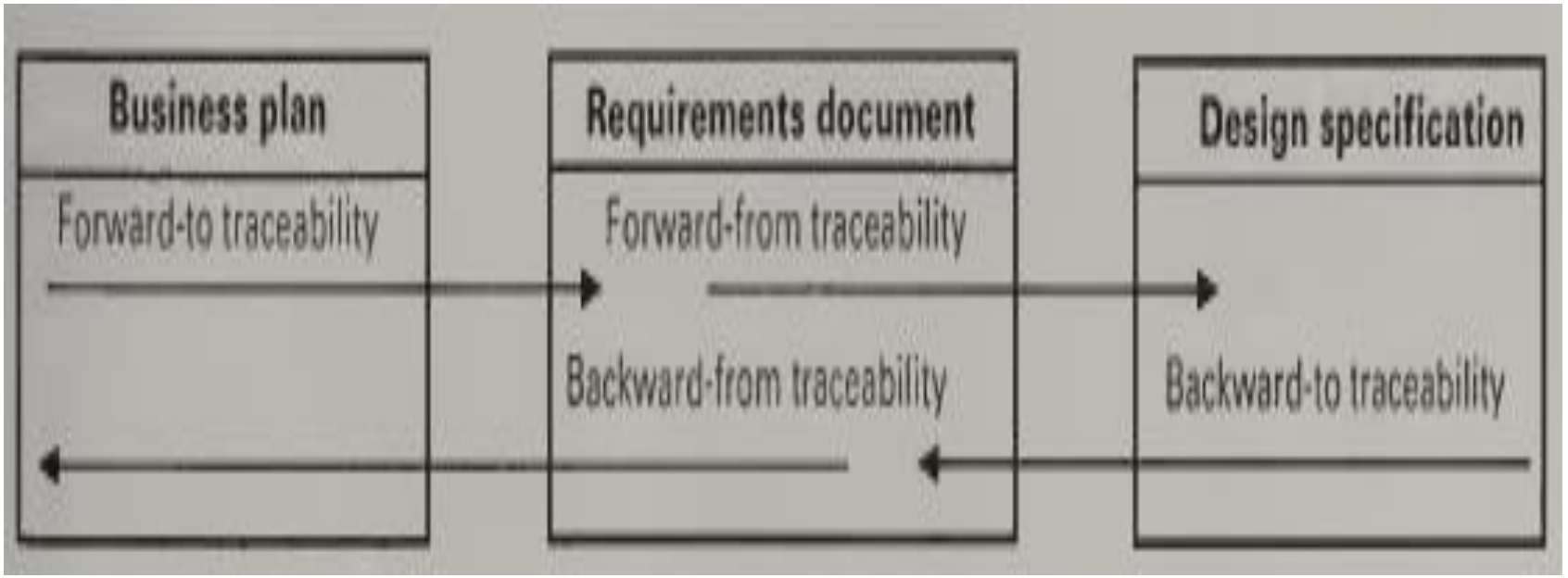


Figure: Backwards and forwards traceability.

- Table 6.1 shows how these different types of traceability can be instantiated more concretely by links between specific information in the requirements document and other system documents.
- In practice, the traceability information which is most commonly maintained during requirements management is requirements-requirements traceability and requirements-design traceability.

Cont'd…

Table 6.1 Types of traceability

Traceability Type	Description
Requirements-sources traceability	• Links the requirement and the people or documents which specified the requirement.
Requirements-rationale traceability	• Links the requirement with a description of why that requirement has been specified. • This can be a distillation of information from several sources.
Requirements-requirements traceability	• Links requirements with other requirements which are, in some way, dependent on them. • This should be a two- way link (dependants and is-dependent on).

Cont'd…

Traceability Type	Description
Requirements-architecture traceability	• Links requirements with the sub-systems where these requirements are implemented. • This is particularly important where sub-systems are being developed by different sub-contractors.
Requirements-design traceability	• Links requirements with specific hardware or software components in the system which are used to implement the requirement.
Requirements-interface traceability	• Links requirements with the interfaces of external systems which are used in the provision of the requirements.

Traceability Tables

- Traceability tables show the relationships between requirements or between requirements and design components.
- Requirements are listed along the horizontal and vertical axes and re1ationships between requirements are marked in the table cells.
- A requirements database is not necessary, although recording traceability information in a database can make it much easier to navigate between dependent requirements.

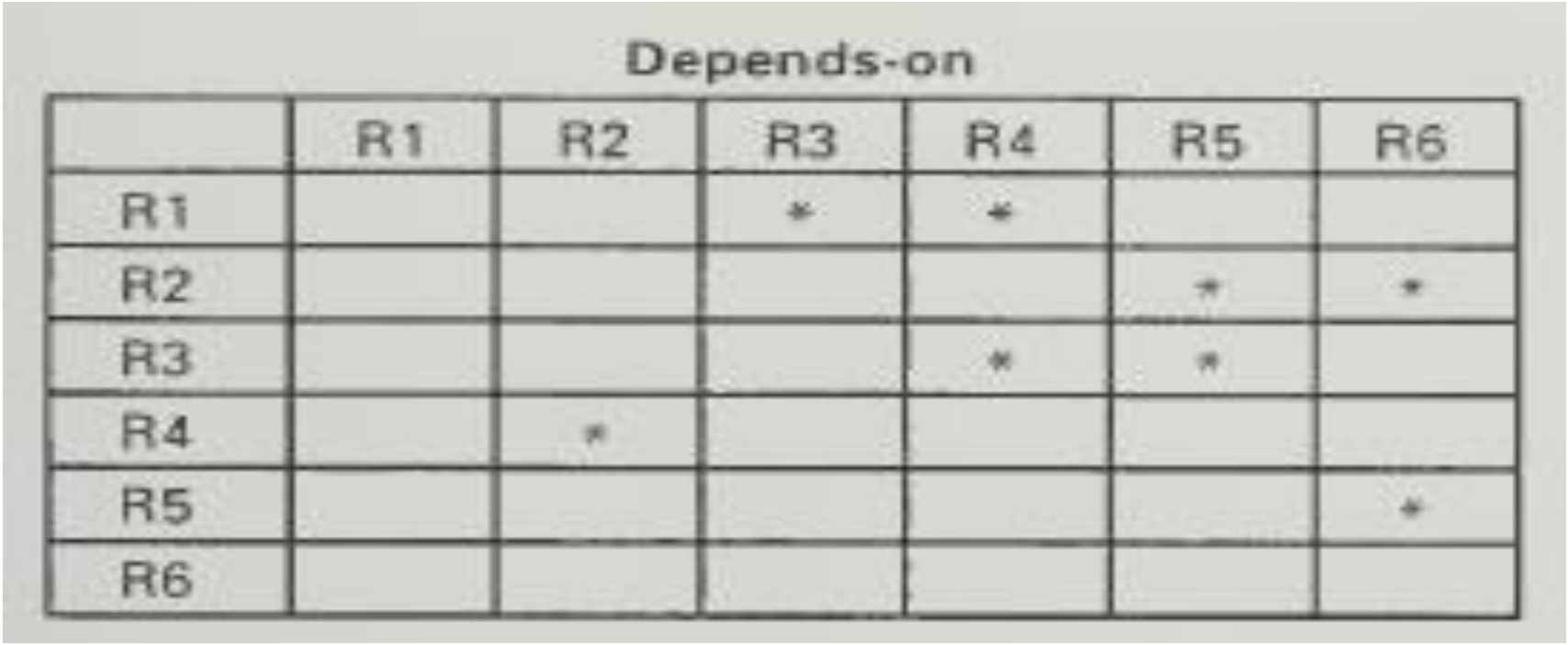
Cont'd…

- Traceability tables for showing requirements dependencies should be defined with requirement numbers used to label the rows and columns of the table.
- In the simplest form of traceability table, you simply put some mark, such as an asterisk, in the table cell where there is some kind of dependency relationship between the requirements in the cell row and column.

➢ That is, if the requirement in row X (say) depends on the requirement in columns P, Q, and R, you should mark table cells (X, P), (X, Q), and (X, R).

- By reading down a column, you see all requirements which depend on a requirement; by reading across a row, you see all requirements which the requirement in that row depends on.

Cont'd…



Depends-on

	R1	R2	R3	R4	R5	R6
R1			*	*		
R2					*	*
R3				*	*	
R4		*				
R5						*
R6						

Figure : A simple traceability table

- Each row in the table shows dependencies, so that R1 is dependent on R3 and R4, R2 is dependent on R5 and R6, etc.
- Therefore, if a change to R4 is proposed, you can see by reading down the R4 column that requirements R1 and R3 are dependent requirements.
- The impact on R1 and R3 of the proposed change to R4 can therefore be assessed.

Cont'd…

- As the number of requirements grows and matrices become unmanageable, a simplified form of traceability table may be used where, along with each requirement description, one or more lists of the identifiers of related requirements are maintained.

➢ Traceability lists are simple lists of relationships which can be implemented as text or as simple tables.

- The table on the next slide shows a traceability list for the dependencies shown in the previous figure.

Table 6.2 A traceability list

Requirement	Depends-on
R1	R3, R4
R2	R5, R6
R3	R4, R5
R4	R2
R5	R6

Cont'd…

- Traceability lists are more compact than traceability tables and do not become as unmanageable with large numbers of requirements.
  - ✓They are therefore less prone to error than traceability tables.

- The disadvantage of these lists compared to traceability tables is that there is no easy way to assess the inverse of a relationship.
- You can easily see that R1 is dependent on R3 and R4 but, given R4, you must look through the whole table to see which requirements depend on it.
- If you wish to maintain this 'backward-to' information, you need to construct another table showing these relationships.

Reading Assignment

- Traceability policies

Thank You!

?