

Requirements Gathering

Dr Kristin Stock

Outline

- Why do we need to gather requirements?
- Types of requirements.
- Determining requirements.
- Requirements analysis strategies.
- Data gathering techniques.
- Case study: SDMS

Introduction

- The systems development process transforms the existing (as is) system into the proposed (to be) system
- Requirements determination
 - The single most critical step of the entire SDLC
 - Changes can be made easily in this stage
 - Most (>50%) system failures are due to problems with requirements
 - The iterative process of OOSAD is effective because:
 - Small batches of requirements can be identified and implemented incrementally
 - The system will evolve over time

Requirements Determination

- Purpose: to convert high level business requirements (from the system request) into detailed requirements that can be used as inputs for creating models
- What is a requirement?
 - A statement of what the system **must do** or a characteristic it must have
 - Will later evolve into a technical description of how the system will be implemented
- Types:
 - Functional: relates to a process or data
 - Non-functional: relates to performance or usability

Functional Requirements

- What should the system do?
- Examples:
 - Allow users to
 - search for something (by certain criteria)
 - buy items
 - trace the location of an object through a process chain
 - communicate by real time voice
 -

Non-functional requirements

- What characteristics should the system have?
 - Operational requirements: what situation must the system operate in.
 - Look and feel requirements (fun, motivating, etc.).
 - Usability (and humanity) requirements.
 - Environmental requirements: location, physical requirements.
 - User requirements: what kind of users will use it, what are their needs?
 - Performance requirements: speed, capacity, reliability.
 - Security requirements.
 - Maintainability and support requirements.
 - Cultural, political, legal requirements.
 - (Data requirements)

Functional vs. Non-functional

- Functional = how much of the problem does the system solve?
- Non-functional = efficiency, maintainability, usability, reliability, testability, etc.

Requirements Definition

- Functional & non-functional requirements listed in outline format
- May be prioritized
- Provides information needed in subsequent workflows
- Defines the scope of the system – useful to avoid scope creep



Sample of Requirements Definition

Nonfunctional Requirements

1. Operational Requirements

- 1.1. The system will operate in Windows environment.
- 1.2. The system should be able to connect to printers wirelessly.
- 1.3. The system should automatically back up at the end of each day.

2. Performance Requirements

- 2.1. The system will store a new appointment in 2 seconds or less.
- 2.2. The system will retrieve the daily appointment schedule in 2 seconds or less.

3. Security Requirements

- 3.1. Only doctors can set their availability.
- 3.2. Only a manager can produce a schedule.

4. Cultural and Political Requirements

- 4.1. No special cultural and political requirements are anticipated.

Functional Requirements

1. Manage Appointments

- 1.1. Patient makes new appointment.
- 1.2. Patient changes appointment.
- 1.3. Patient cancels appointment.

2. Produce Schedule

- 2.1. Office Manager checks daily schedule.
- 2.2. Office Manager prints daily schedule.

3. Record Doctor Availability

- 3.1. Doctor updates schedule

Requirements Management

- An important part of avoiding scope creep.
- Versioning, change control.
- Traceability.
- See Ch 4, BABOK Guide.

Fit Criterion

- ‘Operationalises’ the requirement.
- Makes sure that we can measure whether or not it has succeeded.

So now we know what requirements are – but how do we actually define them?

Determining Requirements

- Who should do it?
 - IT people?
 - Users?



Together...



But how?

Creating a Requirements Definition

- Determine the types of functional and non-functional requirements applicable to the project
- Use requirements-gathering techniques to collect details
- Analysts work with users to verify, change and prioritize each requirement
- Continue this process through analysis workflow, but be careful of scope creep
- Requirements that meet a need but are not within the current scope can be added to a list of future enhancements

Problems in Requirements Determination

- Analyst may not have access to the correct users
- Requirements specifications may be inadequate
- Some requirements may not be known in the beginning
- Verifying and validating requirements can be difficult

We will look at...

- Requirements gathering techniques:
 - How do we actually get data?
- Requirements analysis strategies:
 - What kinds of data do we get, and how do we analyse it to identify requirements?
- Part of it is a process of helping people to move through the design process – getting them on board

Requirements Analysis Strategies

- Problem analysis
- Root cause analysis
- Duration analysis
- Activity-based costing
- Informal benchmarking
- Outcome analysis
- Technology analysis
- Activity elimination

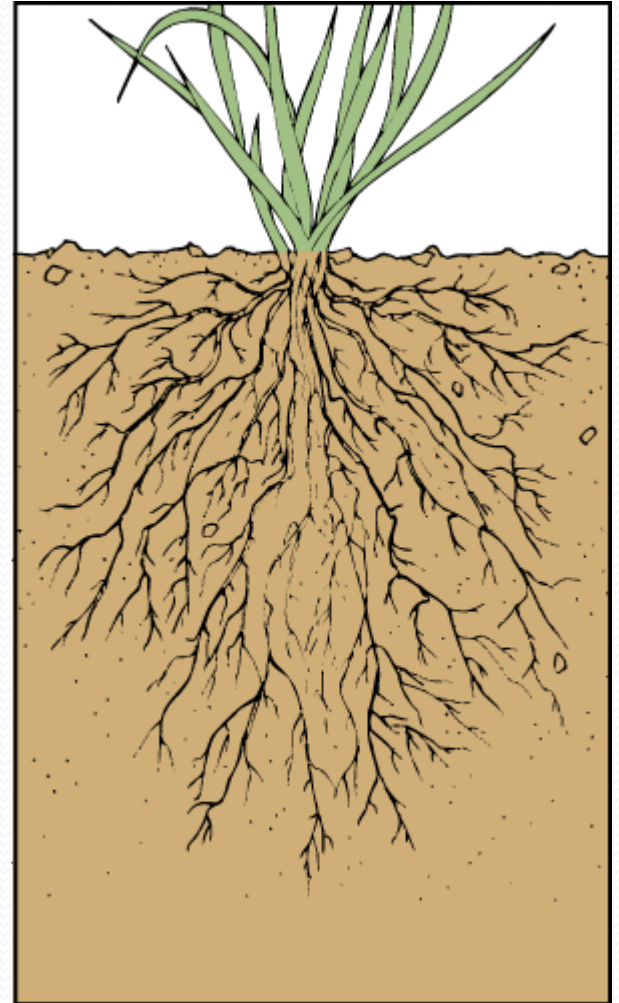
Problem Analysis

- Ask users to identify problems with the current system
- Ask users how they would solve these problems
- Good for improving efficiency or ease-of-use



Root Cause Analysis

- Focus is on the cause of a problem, not its solution
- Create a prioritized list of problems
- Try to determine all the possible root causes for each problem
- Once the causes are known, solutions can be developed



Duration Analysis

- Determine the time required to complete each step in a business process
- Compare this to the total time required for the entire process
- Large differences suggest problems that might be solved by:
 - Integrating some steps together
 - Performing some steps simultaneously (in parallel)



Activity-based costing

- Same as duration analysis but applied to costs.
- Analysts identify costs for each step in the process.
- Direct cost of labour and materials for each input, plus indirect costs.



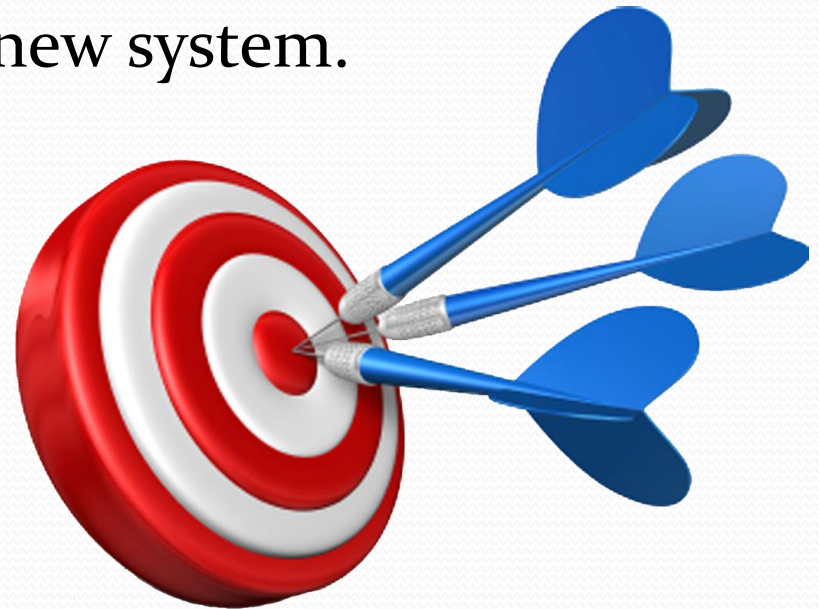
Informal Benchmarking

- Analyzes similar processes in other successful organizations.
- Especially common for customer-facing business processes.
- Brings new ideas into the organisation.
- Can also use this to get ideas for a new system.



Outcome Analysis

- What does the customer want in the end?
- Not always the most obvious thing.
- What is the **ultimate** goal?
- Also helpful for developing a new system.



Technology Analysis

- Make a list of important and interesting new technologies.
- Examine each in terms of what it might offer to the organisation.
- Must only be justified by benefit to the organisation.



Activity Elimination

- Eliminate each activity in a business process.
- People are often reluctant, but must choose something!

~~A~~ ~~B~~ ~~C~~ D ~~E~~

Requirements Analysis Strategies

Method	Finds and analyses
Problem analysis	Problems with the current process
Root cause analysis	The root cause of problems with the current process
Duration analysis	The duration of activities in the current process
Activity-based costing	The cost of activities in the current process
Informal benchmarking	How other organisations perform the process
Outcome analysis	The ultimate outcomes required by the users
Technology analysis	Available technology
Activity elimination	Superfluous activities in the current process.

So now we have some idea
of what data to collect – but
how do we collect it?

Requirements Gathering Techniques

- Process is used to:
 - Uncover all requirements (those uncovered late in the process are more difficult to incorporate)
 - Build support and trust among users
- Which technique(s) to use?
 - Interviews
 - Joint Application Development (JAD)
 - Questionnaires
 - Document analysis
 - Observation
 - Focus Groups
 - Requirements Workshops

Document Analysis

- Provides information about the “as-is” system
- Review technical documents when available
- Review typical user documents:
 - Forms
 - Reports
 - Policy manuals
- Look for user additions to forms
- Look for unused form elements

Interviews

- Most popular technique—if you need to know something, just ask
- Process:
 - Select people to interview & create a schedule
 - Design interview questions (Open-ended, closed-ended, & probing types of questions)
 - Prepare for the interview (Unstructured vs. structured interview organized in a logical order)
 - Conduct the interview (Top-down vs. bottom-up)
 - Follow-up after the interview

Who and How?

- Usually one on one is best to get individual opinions.
- Identify key people – often becomes apparent.
- Best to do some review of documents and other materials before hand.
- May go back to the same individual again later, to ask other or more specific questions.
- Should not ask for information you can get from another source.
- Can be politically sensitive.



Question Types

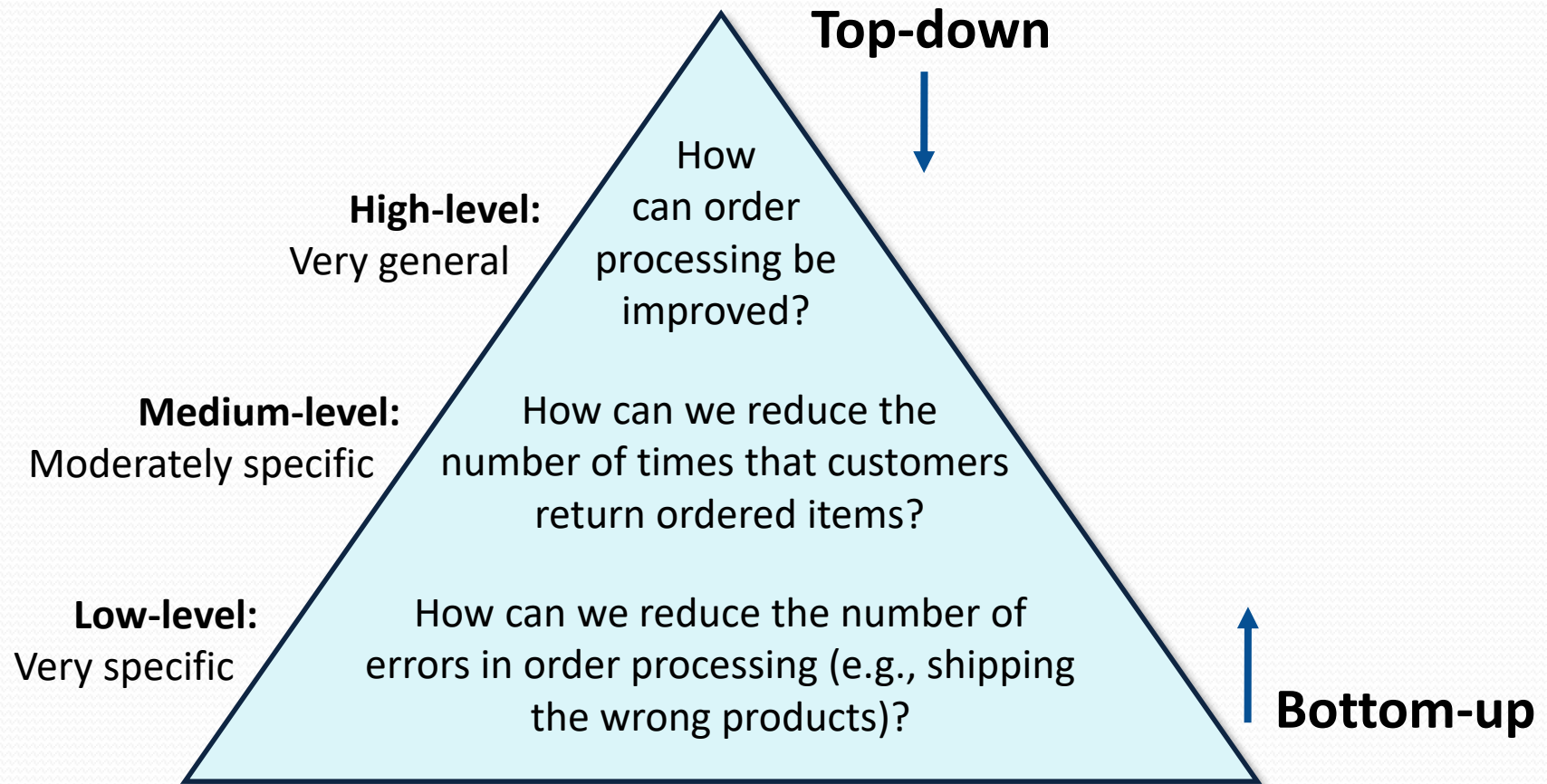
Types of Questions	Examples
Closed-ended questions	<ul style="list-style-type: none">• How many telephone orders are received per day?• How do customers place orders?• What information is missing from the monthly sales report?
Open-ended questions	<ul style="list-style-type: none">• What do you think about the current system?• What are some of the problems you face on a daily basis?• What are some of the improvements you would like to see in a new system?
Probing questions	<ul style="list-style-type: none">• Why?• Can you give me an example?• Can you explain that in a bit more detail?

Design of interviews

- Different levels of detail at different points in the process.
- Ask questions in a logical sequence.
- Start with non-controversial, non-threatening questions.
- Give the interviewee information about what will be covered.



Interviewing Strategies



Conducting the Interview

- Try to build rapport.
- People will often give their opinion, but you need to stay objective, collect facts as well.
- Examples:
 - “This system is rubbish”
 - “I can’t do any of the things I want to”
 - “It always crashes”
 - “I told the programmer to build the system like this...”
- Ask them to show you the problems.

Post-Interview

- Prepare notes and send to the interviewee for verification

Interview Notes Approved by: Linda Estey
Person Interviewed: Linda Estey, Director, Human Resources
Interviewer: Barbara Wixom
Purpose of Interview: <ul style="list-style-type: none">• Understand reports produced for Human Resources by the current system• Determine information requirements for future system
Summary of Interview: <ul style="list-style-type: none">• Sample reports of all current HR reports are attached to this report. The information that is not used and missing information are noted on the reports.• Two biggest problems with the current system are:<ol style="list-style-type: none">1. The data are too old (the HR Department needs information within two days of month end; currently information is provided to them after a three-week delay)2. The data are of poor quality (often reports must be reconciled with departmental HR database)• The most common data errors found in the current system include incorrect job level information and missing salary information.
Open Items: <ul style="list-style-type: none">• Get current employee roster report from Mary Skudrna (extension 4355).• Verify calculations used to determine vacation time with Mary Skudrna.• Schedule interview with Jim Wack (extension 2337) regarding the reasons for data quality problems.
Detailed Notes: See attached transcript.

Joint Application Development (JAD)

- Joint user-analyst meeting hosted by a facilitator
 - 10 to 20 users
 - 1 to 2 scribes as needed to record the session
 - Usually in a specially prepared room
- Meetings can be held electronically and anonymously
 - Reduces problems in group settings
 - Can be held remotely
- Sessions require careful planning to be successful
 - Users may need to bring documents or user manuals
 - Ground rules should be established

Joint Application Development (JAD) (2)

- As with interviews, questions should be planned, participants prepared.
- Vary in length.
- Users bring along documents.
- Potential challenges:
 - Can be hard to get the right people there
 - Group dynamics are a big issue.
 - Require a skilled and trained facilitator.

Focus Groups

- Address a specific product, service or opportunity.
- 6-12 participants.
- Moderator and recorder.
- Can be good work working on an existing model.

Requirements Workshops

- More general than JAD (which involves specific techniques).
- E.g. thematic experts to elicit requirements for a data model.
- All group activities probably best in non-contentious projects.

Questionnaires

- A set of written questions used to obtain information from individuals
- Not flexible, so have to be designed very carefully. Best for very specific questions.
- Common uses:
 - Large numbers of people
 - Need both information and opinions
 - When designing for use outside the organization (customers, vendors, etc.)

Questionnaire Steps

- Select the participants
 - Identify the population
 - Use representative samples for large populations
- Designing the questionnaire
 - Careful question selection
 - Remove ambiguities
- Administering the questionnaire
 - Working to get good response rate
 - Offer an incentive (e.g., a free pen)
- Questionnaire follow-up
 - Send results to participants
 - Send a thank-you

Good Questionnaire Design

- Begin with non-threatening and interesting questions
- Group items into logically coherent sections
- No important items at the very end
- Do not crowd a page with too many items
- Avoid abbreviations
- Avoid biased or suggestive items or terms
- Number questions to avoid confusion
- Pretest to identify confusing questions
- Provide anonymity to respondents

Observation

- Users/managers often don't remember everything they do
- Checks validity of information gathered in other ways
- Behaviors may change when people are watched
 - Workers tend to be very careful when watched
 - Keep a low profile
 - Try not to interrupt or influence workers
- Be careful not to ignore periodic activities
 - Weekly ... Monthly ... Annually

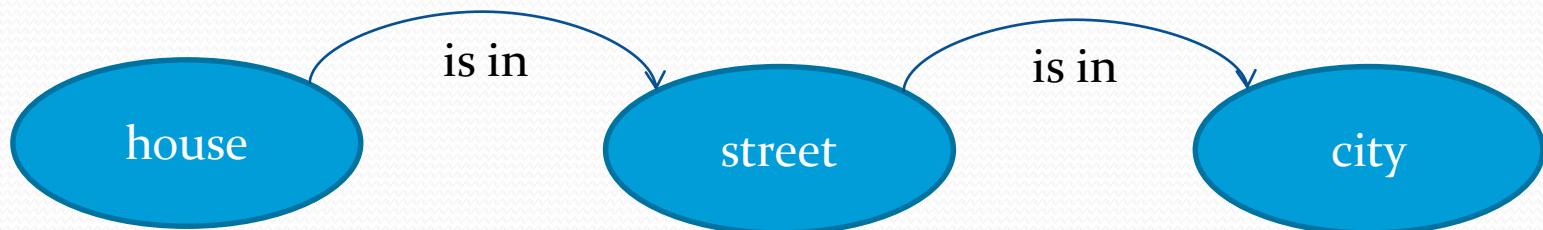
Requirements-Gathering Techniques Compared

- A combination of techniques may be used
- Document analysis & observation require little training; JAD sessions can be very challenging

	Interviews	Joint Application Design	Questionnaires	Document Analysis	Observation
Type of information	As-is, improvements, to-be	As-is, improvements, to-be	As-is, improvements	As-is	As-is
Depth of information	High	High	Medium	Low	Low
Breadth of information	Low	Medium	High	High	Low
Integration of information	Low	High	Low	Low	Low
User involvement	Medium	High	Low	Low	Low
Cost	Medium	Low-Medium	Low	Low	Low to Medium

Concept Maps

- Represent meaningful relationships between concepts
- Focus individuals on a small number of key ideas
- Can be used to model:
 - Requirements,
 - Concepts
 - Processes, etc.
 - Then fleshed out in more detail in later parts of the process.



Story Cards & Task Lists

- Associated with agile development methods
- Very low tech, high touch, easily updatable, and very portable
- Capture requirement using story cards (index cards)
- File card with single requirement
- Each requirement (card) is discussed
 - How much effort is required to implement it
 - A task list is created for each requirement (story)
 - Large requirements can be split into smaller sections
 - The story can be prioritized by risk level and importance

System Proposal Template

1. Table of Contents

2. Executive Summary

A summary of all the essential information in the proposal so a busy executive can read it quickly and decide what parts of the proposal to read in more depth.

3. System Request

The revised system request form (see Chapter 2).

4. Workplan

The original workplan, revised after having completed analysis (see Chapter 2).

5. Feasibility Analysis

A revised feasibility analysis, using the information from analysis (see Chapter 2).

6. Requirements Definition

A list of the functional and nonfunctional business requirements for the system (this chapter).

7. Functional Model

An activity diagram, a set of use case descriptions, and a use-case diagram that illustrate the basic processes or external functionality that the system needs to support (see Chapter 4).

8. Structural Models

A set of CRC cards, class diagram, and object diagrams that describe the structural aspects of the to-be system (see Chapter 5). This may also include structural models of the current as-is system that will be replaced.

9. Behavioral Models

A set of sequence diagrams, communication diagrams, behavioral-state machines, and a CRUDE matrix that describe the internal behavior of the to-be system (see Chapter 6). This may include behavioral models of the as-is system that will be replaced.

10. Appendices

These contain additional material relevant to the proposal, often used to support the recommended system. This might include results of a questionnaire survey or interviews, industry reports and statistics, and so on.

Case Study: Australian Capital Territory SDMS

- Goal: to automate, record, enable land administration
- Two existing systems:
 - CAD system
 - Oracle based textual system
- Requirement: a single integrated system to support all business functions



ACT Planning &
Land Authority

Case Study: Australian Capital Territory SDMS

- Some requirements defined in the project brief.
- Documents of various kinds:
 - data models;
 - registers and spreadsheets of work progress;
 - legislation;
 - web pages.
- Some staff felt threatened.

Summary

- Presented in this lecture:
 - Discussion of functional and non-functional requirements determination
 - Requirements analysis strategies
 - problem analysis, root cause analysis, duration analysis, activity-based costing analysis, informal benchmarking analysis, outcome analysis, technology analysis and activity elimination
 - Requirements gathering techniques
 - Interviews, joint application development, questionnaires, document analysis and observation
 - Alternative requirements documentation techniques
 - concept maps, story cards and task lists
 - The system proposal