

University of Toronto Department of Computer Science

Lecture 3: Requirements Elicitation I

Last Week:
Approaches to RE
Processes, methods & techniques
Domains
Systems Theory

This Week:
Elicitation (I)
Traditional approaches
Interviews & Questionnaires
Scenarios, Goals and Use-Cases

Next Week:
Elicitation (II)
Cognitive approaches
Contextual approaches
Ethnography

© 2000-2003, Steve Easterbrook 2

University of Toronto Department of Computer Science

Requirements Elicitation

→ Starting point

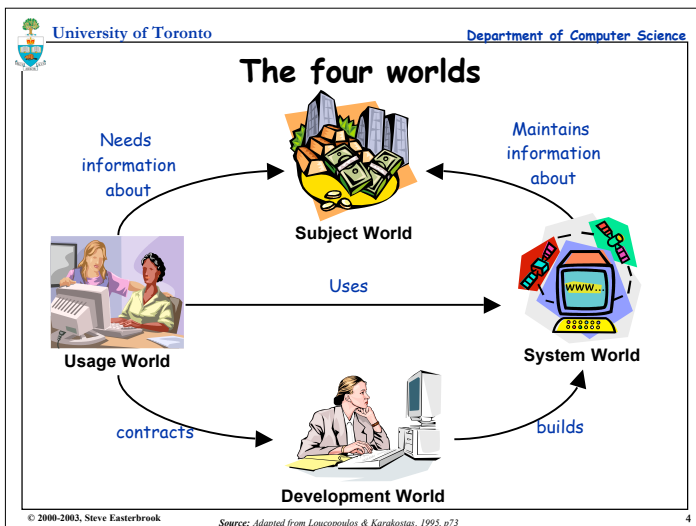
- ☞ Some notion that there is a "problem" that needs solving
 - e.g. dissatisfaction with the current state of affairs
 - e.g. a new business opportunity
 - e.g. a potential saving of cost, time, resource usage, etc.

→ Collect enough information to:

- ☞ identify the "problem"/"opportunity"
 - Which problem needs to be solved? (identify problem **Boundaries**)
 - Where is the problem? (understand the **Context/Problem Domain**)
 - Whose problem is it? (identify **Stakeholders**)
 - Why does it need solving? (identify the stakeholders' **Goals**)
 - How might a software system help? (collect some **Scenarios**)
 - When does it need solving? (identify **Development Constraints**)
 - What might prevent us solving it? (identify **Feasibility and Risk**)
- ☞ become an expert in the problem domain
 - Learn how to find your way round a new problem area quickly
 - Use your (initial) ignorance as an excuse to ask questions
 - Recognise the domain expertise of the people you talk to

W6H
The journalist's technique:
What?
Where?
Who?
Why?
When?
How?
(Which?)

© 2000-2003, Steve Easterbrook 3



University of Toronto Department of Computer Science

Finding out about the four worlds

→ Subject World

- ☞ the subject matter of the information system:
 - e.g., customers, accounts, transactions for a bank information system

→ Usage World

- ☞ the environment within which the planned system will operate
 - e.g., people, such as managers, clerks, customers; also business processes such as handling a withdrawal, a deposit of foreign currency,...

→ System World

- ☞ what the system does within its operational environment, what information it contains and what functions it performs;
 - e.g., system records all transactions in a database, reports on transactions for a particular account, gives account balance,...

→ Development World

- ☞ the development process, team, schedule, required qualities (security, performance,...) etc.
 - e.g., system to be delivered in 12 months, fully tested to MCDC standard, etc.

© 2000-2003, Steve Easterbrook 5

University of Toronto Department of Computer Science

Stakeholders

→ **Stakeholder analysis:**

- ↳ Identify all the people who must be consulted during information acquisition
- ↳ Look for stakeholders associated with each of the four worlds

→ **Example stakeholders**

- ↳ **Users**
 - concerned with the features and functionality of the new system
- ↳ **Designers**
 - want to build a perfect system, or reuse existing code
- ↳ **Systems analysts**
 - want to "get the requirements right"
- ↳ **Training and user support staff**
 - want to make sure the new system is usable and manageable
- ↳ **Business analysts**
 - want to make sure "we are doing better than the competition"
- ↳ **Technical authors**
 - will prepare user manuals and other documentation for the new system
- ↳ **The project manager**
 - wants to complete the project on time, within budget, with all objectives met.
- ↳ **"the customer"**
 - whoever it is that pays for the new system!

© 2000-2003, Steve Easterbrook 6

University of Toronto Department of Computer Science

Difficulties of Elicitation

→ **Thin spread of domain knowledge**

- ↳ The knowledge might be distributed across many sources
 - It is rarely available in an explicit form (I.e. not written down)
- ↳ There will be conflicts between knowledge from different sources
 - People have conflicting goals
 - People have different understandings of the problem

→ **Tacit knowledge (The "say-do" problem)**

- ↳ People find it hard to describe knowledge they regularly use
 - Descriptions may be inaccurate rationalizations of expert behaviour

→ **Limited Observability**

- ↳ The problem owners might be too busy solving it using the existing system
- ↳ Presence of an observer may change the problem
 - E.g. the Probe Effect and the Hawthorne Effect

→ **Bias**

- ↳ People may **not be free** to tell you what you need to know
 - Political climate & organisational factors matter
- ↳ People may **not want** to tell you what you need to know
 - The outcome will affect them, so they may try to influence you (hidden agendas)

© 2000-2003, Steve Easterbrook 7

University of Toronto Department of Computer Science

Example

→ **The problem area:**

- ↳ Loan approval department in a large bank
- ↳ The analyst is trying to elicit the rules and procedures for approving a loan

→ **Why this might be difficult:**

- ↳ **Implicit knowledge:**
 - There is no document in which the rules for approving loans are written down
- ↳ **Conflicting information:**
 - Different members of the department have different ideas about what the rules are
- ↳ **Say-do problem:**
 - The loan approval process described to you by the loan approval officers is quite different from your observations of what they actually do
- ↳ **Probe effect:**
 - The loan approval process used by the officers while you are observing is different from the one they normally use
- ↳ **Bias:**
 - The loan approval officers fear that your job is to computerize their jobs out of existence, so they are deliberately emphasizing the need for case-by-case discretion (to convince you it has to be done by a human!)

© 2000-2003, Steve Easterbrook 8

University of Toronto Department of Computer Science

Expert Bias

→ **What is bias?**

- ↳ Bias only exists in relation to some reference point
 - can there ever be "no bias"? (reflects reality or truth)
- ↳ We cannot perceive reality directly:
 - It is interpreted through a filter of mental models
 - ...mediated by our senses and neural pathways.
- ↳ All decision making is based partly on personal value systems.

→ **Types of bias:**

- ↳ **Motivational bias**
 - the expert makes accommodations to please the interviewer or some other audience
- ↳ **Cognitive bias**
 - the expert does not follow objective rules or standards

Sources of Bias

- ↳ **Social pressure**
response to verbal and non-verbal cues from an interviewer
- ↳ **Group think**
response to reactions of other experts
- ↳ **Impression management**
response to imagined reactions of managers, clients, etc.
- ↳ **Wishful thinking**
response to hopes or possible gains.
- ↳ **Misinterpretation**
Analyst selectively interprets to support what she currently believes.
- ↳ **Misrepresentation**
expert cannot accurately fit a response into the requested response mode
- ↳ **anchoring**
contradictory data is ignored once an initial solution is available
- ↳ **inconsistency**
assumptions made earlier are forgotten
- ↳ **availability**
some data are easier to recall than others
- ↳ **underestimation of uncertainty**
tendency to underestimate by a factor of 2 or 3.

© 2000-2003, Steve Easterbrook 9

University of Toronto Department of Computer Science

Elicitation Techniques

→ **Traditional techniques**

- ↳ Introspection
- ↳ Reading existing documents
- ↳ Analyzing hard data
- ↳ Interviews
 - Open-ended
 - Structured
- ↳ Surveys / Questionnaires
- ↳ Meetings

→ **Collaborative techniques**

- ↳ Focus Groups
 - Brainstorming
 - JAD/RAD workshops
- ↳ Prototyping
- ↳ Participatory Design

→ **Model-based techniques**

- ↳ Goal-based
- ↳ Scenario-Based
- ↳ Use Cases

← This week

→ Next week

→ **Contextual (social) approaches**

- ↳ Ethnographic techniques
 - Participant Observation
 - Ethnomethodology
- ↳ Discourse Analysis
 - Conversation Analysis
 - Speech Act Analysis
- ↳ Sociotechnical Methods
 - Soft Systems Analysis

→ **Cognitive techniques**

- ↳ Task analysis
- ↳ Protocol analysis
- ↳ Knowledge Acquisition Techniques
 - Card Sorting
 - Laddering
 - Repertory Grids
 - Proximity Scaling Techniques

© 2000-2003, Steve Easterbrook 10

University of Toronto Department of Computer Science

Background Reading

→ **Sources of information:**

- ↳ company reports, organization charts, policy manuals, job descriptions, reports, documentation of existing systems, etc.

→ **Advantages:**

- ↳ Helps the analyst to get an understanding of the organization before meeting the people who work there.
- ↳ Helps to prepare for other types of fact finding, e.g. by being aware of the business objectives of the organization.
- ↳ may tell you the detailed requirements for the current system.

→ **Disadvantages:**

- ↳ written documents often do not match up to reality.
- ↳ Can be long-winded with much irrelevant detail

→ **Appropriate for**

- ↳ projects where analyst is not familiar with the organization being investigated.

© 2000-2003, Steve Easterbrook 11

University of Toronto Department of Computer Science

"Hard Data" Collection

→ **Identify Collections of Hard Data**

- ↳ Facts and figures, financial information,...
- ↳ Reports used for decision making,...
- ↳ Survey results, marketing data,...

→ **Sampling**

- ↳ Sampling used to select representative set from a population
 - Purposive Sampling - choose the parts you think are relevant without worrying about statistical issues
 - Simple Random Sampling - choose every kth element
 - Stratified Random Sampling - identify strata and sample each
 - Clustered Random Sampling - choose a representative subpopulation and sample it
- ↳ Sample Size is important
 - balance between cost of data collection/analysis and required significance
- ↳ Process:
 - Decide what data should be collected - e.g. banking transactions
 - Determine the population to be sampled - e.g. all transactions at 5 local branches over one week
 - Choose type of sample - e.g. simple random sampling
 - Choose sample size - e.g. every 10th transaction

© 2000-2003, Steve Easterbrook 12

University of Toronto

Example of hard data

→ **Questions:**

- ↳ What does this data tell you?
- ↳ What would you do with this data?

Agate
Campaign Summary

Date: 23rd February 1999

Client: Yellow Partridge
Park Road Workshops
Park Road
Jewellery Quarter
Birmingham B2 3DT
U.K.

Campaign: Spring Collection 1999

Billing Currency: GB £

Item	Cur	Amount	Rate	Billing amount
Advert preparation: photography, artwork, layout etc.	GB £	15,000.00	1	15,000.00
Placement French Vogue	FFR.	47 000.00	11.35	4,140.97
Placement UK Vogue	GB £	5,000.00	1	5,000.00
Placement US Vogue	US \$	15,000.00	2.47	6,072.87
Total				30,213.84

This is not a VAT Invoice. A detailed VAT Invoice will be provided separately.

210-212 Carstairs Street, Birmingham B1 6TG
Tel:0121 111 2204 Fax:0121 111 2245
Email:agate@agatelltd.co.uk

© 2000-2003, Steve Easterbrook

University of Toronto Department of Computer Science

Interviews

→ **Types:**

- ↳ Structured - agenda of fairly open questions
- ↳ Open-ended - no pre-set agenda

→ **Advantages**

- ↳ Rich collection of information
 - Good for uncovering opinions, feelings, goals, as well as hard facts
- ↳ Can probe in depth, & adapt followup questions to what the person tells you

→ **Disadvantages**

- ↳ Large amount of qualitative data can be hard to analyze
- ↳ Hard to compare different respondents
- ↳ Interviewing is a difficult skill to master

→ **Watch for**

- ↳ Unanswerable questions ("how do you tie your shoelaces?")
- ↳ Tacit knowledge (and post-hoc rationalizations)
- ↳ Removal from context
- ↳ Interviewer's attitude may cause bias (e.g. variable attentiveness)

© 2000-2003, Steve Easterbrook Source: Adapted from Goguen and Linde, 1993, p134. 14

University of Toronto Department of Computer Science

Interviewing Tips

→ **Starting off...**

- ↳ Begin the interview with an innocuous topic to set people at ease
 - e.g. the weather, the score in last night's hockey game
 - e.g. comment on an object on the person's desk: "My... what a beautiful photograph! Did you take that?"

→ **Ask if you can record the interview**

- ↳ but put tape recorder in front of person
- ↳ say that they can turn it off any time.

→ **Ask easy questions first**

- ↳ perhaps personal information
 - e.g. "How long have you worked in your present position?"

→ **Follow up interesting leads**

- ↳ E.g. watch for things people say that indicate that your plan of action may be wrong,
 - e.g., "Could we pursue what you just said a little further?"

→ **Ask open-ended questions last**

- ↳ e.g. "Is there anything else you would like to add?"

© 2000-2003, Steve Easterbrook 15

University of Toronto Department of Computer Science

Questionnaires

→ **Advantages**

- ↳ Can quickly collect info from large numbers of people
- ↳ Can be administered remotely
- ↳ Can collect attitudes, beliefs, characteristics

→ **Disadvantages**

- ↳ Simplistic (presupposed) categories provide very little context
 - No room for users to convey their real needs

→ **Watch for:**

- ↳ Bias in sample selection
- ↳ Bias in self-selecting respondents
- ↳ Small sample size (lack of statistical significance)
- ↳ Open ended questions (very hard to analyze!)
- ↳ Leading questions ("have you stopped beating your wife?")
- ↳ Appropriation ("What is this a picture of?")
- ↳ Ambiguous questions (I.e. not everyone is answering the same question)

Questionnaires MUST be prototyped and tested!

© 2000-2003, Steve Easterbrook Source: Adapted from Goguen and Linde, 1993, p134. 16

University of Toronto Department of Computer Science

Meetings

→ **Used for summarization and feedback**

- ↳ E.g. meet with stakeholders towards the end of each stage:
 - to discuss the results of the information gathering stage
 - to conclude on a set of requirements
 - to agree on a design etc.
- ↳ Use the meeting to confirm what has been learned, talk about findings

→ **Meetings are an important managerial tool**

- ↳ Used to move an information system development project forward.
- ↳ Need to determine objectives for the meeting:
 - E.g. presentation, problem solving, conflict resolution, progress analysis, gathering and merging of facts, training, planning,...
- ↳ Plan the meeting carefully:
 - Schedule the meeting and arrange for facilities
 - Prepare an agenda and distribute it well in advance
 - The meeting itself may be structured or unstructured depending on objective;
 - Keep track of time and agenda during the meeting
 - Follow up with a written summary to be distributed to meeting participants
 - Special rules apply for formal presentations (and how to prepare them), project walkthroughs, brainstorming,...

© 2000-2003, Steve Easterbrook 17

University of Toronto Department of Computer Science

Group Elicitation Techniques

→ **Types:**

- ↳ Focus Groups
- ↳ Brainstorming

→ **Advantages**

- ↳ More natural interaction between people than formal interview
- ↳ Can gauge reaction to stimulus materials (e.g. mock-ups, storyboards, etc)

→ **Disadvantages**

- ↳ May create unnatural groups (uncomfortable for participants)
- ↳ Danger of *Groupthink*
- ↳ May only provide superficial responses to technical questions
- ↳ Requires a highly trained facilitator

→ **Watch for**

- ↳ sample bias
- ↳ dominance and submission

© 2000-2003, Steve Easterbrook 18

University of Toronto Department of Computer Science

Joint/Rapid Application Development

→ **JAD & RAD Principles:**

- ↳ **Group Dynamics**
 - ↳ one-to-one or group interview formats replaced with workshops
- ↳ **Visual Aids**
 - ↳ Use lots of visualization media, ranging from wall charts to large monitors or graphical interfaces
- ↳ **Organized, Rational Process**
 - ↳ Using techniques such as brainstorming and top-down analysis to structure the elicitation and analysis process
- ↳ **WYSIWYG Documentation Approach**
 - ↳ each JAD session results in a document which is easy to understand and is created and agreed upon during the session

→ **Notes:**

- ↳ Choose workshop participants carefully
 - ↳ they should be the best people possible representing various stakeholder groups
- ↳ Workshop should last 3-5 days.
 - ↳ Must turn a group of participants into a team - this takes 1-2 days.
 - ↳ Session leader makes sure each step has been completed thoroughly.
 - ↳ Session leader steps in when there are differences of opinion - "open issues".
 - ↳ Meeting room should be well-equipped for presentations, recording etc.

© 2000-2003, Steve Easterbrook 19

University of Toronto Department of Computer Science

Goal-based Approaches

→ **Approach**

- ↳ Focus on why systems are constructed
- ↳ Express the 'why' as a set of stakeholder goals
- ↳ Use goal refinement to arrive at specific requirements
- ↳ **Goal analysis**
 - ↳ document, organize and classify goals
- ↳ **Goal evolution**
 - ↳ refine, elaborate, and operationalize goals
- ↳ Goal hierarchies show **refinement** and **obstacle** relationships between goals

→ **Advantages**

- ↳ Reasonably intuitive
- ↳ Explicit declaration of goals provides sound basis for conflict resolution

→ **Disadvantages**

- ↳ Hard to cope with evolution of goals
- ↳ Can regress forever up (or down) the goal hierarchy

© 2000-2003, Steve Easterbrook Source: Adapted from Anton, 1996. 20

University of Toronto Department of Computer Science

Using a goal-based approach

→ **Goals**

- ↳ high level objectives of the business or organisation

→ **Requirements**

- ↳ specify how a goal is to be accomplished by the new system

→ **Types**

- ↳ Achievement goals
- ↳ Maintenance goals
- ↳ Soft goals

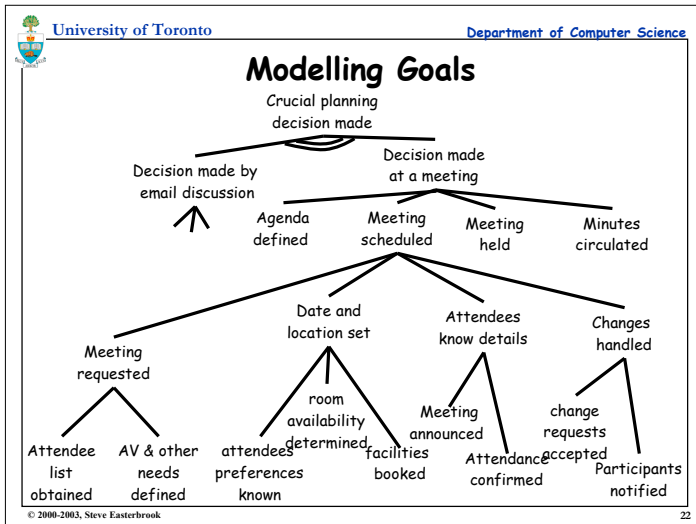
→ **Obstacles & constraints**

- ↳ Obstacles are behaviors that prevent achievement of a given goal
- ↳ Constraints are conditions on the achievement of goals

→ **Tips**

- ↳ Multiple sources yield better goals
- ↳ Associate stakeholders with each goal (reveals viewpoints and conflict)
- ↳ Use scenarios to explore how goals can be met
- ↳ Explicit consideration of obstacles helps to elicit exceptions

© 2000-2003, Steve Easterbrook Source: Adapted from Anton, 1996. 21



University of Toronto Department of Computer Science

Goal Analysis

→ **Goal Elaboration:**

- ↳ "Why" questions explore higher goals (context)
- ↳ "How" questions explore lower goals (operations)
- ↳ "How else" questions explore alternatives

→ **Dependency Analysis:**

- ↳ Precedence ordering - must achieve goals in a particular order
- ↳ Obligation - achieving one goal requires achievement of another
- ↳ Thwarting - achieving one goal prevents achievement of another

→ **Obstacle Analysis:**

- ↳ Can this goal be obstructed, if so how?
- ↳ What are the consequences of obstructing it?

© 2000-2003, Steve Easterbrook 23

University of Toronto Department of Computer Science

Scenarios

→ **Scenarios**

- ↳ Specific sequence of interaction between actor and system
- ↳ Tend to be short (e.g between 3 and 7 steps)
- ↳ May be:
 - positive (i.e. required behavior)
 - negative (i.e. an undesirable interaction)
- ↳ May be indicative (describe current system) or optative (how it should be)

→ **Advantages**

- ↳ Very natural: stakeholders tend to use them spontaneously
 - E.g "suppose I'm admitted to hospital - what happens during my admission?"
 - Typical answer: "You, or the person accompanying you would talk to the person at the admissions desk. You have to show your OHIP card and explain who referred you to the hospital. Then you..." [and so on]
- ↳ Short scenarios very good for quickly illustrating specific interactions

→ **Disadvantages**

- ↳ Lack of structure: need use cases or task models to provide higher level view

© 2000-2003, Steve Easterbrook Source: Adapted from Dardenne, 1993. 24

University of Toronto Department of Computer Science

Example Scenario

Title: Successful meeting scheduled using messaging option
Participants: Alice (initiator, not attending); Bob, Carlo, Daphne (attendees)

Action	Goals satisfied	Obstacles / Problems
Alice requests meeting, specifying participants, timeframe	Meeting requested; Attendee list obtained	What if selected timeframe is infeasible?
AS sends participant requests to Bob, Carlo and Daphne	?	Did we miss a goal?
Bob reads message	Participants informed	Can't detect when messages are read; what happens if Bob reads the message but doesn't reply?
Carlo reads message		
Daphne reads message		
Bob replies with preferences	Attendees preferences known	What if the preferences are mutually exclusive? Should we allow some to be higher priority?
Carlo replies with preferences		
Daphne replies with preferences		
AS schedules meeting	Room availability determined; room booked	
AS notifies Alice, Bob, Carlo, Daphne of time and location	Meeting announced; Attendance Confirmed (?)	How do we know if they've all read the announcement? What if the schedule is no longer convenient for one of them?

© 2000-2003, Steve Easterbrook 25



Task Models & Scenarios

→ Task Models:

- ↳ hierarchical collections of stereotypical activities
- ↳ Subgoals are tasks (or possibly use-cases)
 - Subgoals may occur in sequence, in parallel, or as alternatives.
 - Subgoals may occur periodically or in response to contingencies.

→ Scenarios:

- ↳ are paths through a task model, taking in a specific time-sequence of steps
- ↳ can be used to organize requirements
- ↳ Can include parallelism
 - ...but only one alternative at each choice point.

→ Exceptions

- ↳ are important (often "business critical") variants on the use case.
- ↳ Cannot be modeled as scenarios themselves, as they interact with many concrete executable scenarios.



Use Cases

→ What is a use case?

- ↳ Each different way that an actor interacts with a system is a use case
 - "a description of a sequence of actions that a system performs that yields an observable result of value to a particular actor" [Booch]
 - All the use cases need to be enumerated (or the requirements will not be complete)
- ↳ A description of a set of possible scenarios, with a common purpose
- ↳ Typically written in natural language
- ↳ No internal description of the system; just the interaction.

→ Combining use cases

- ↳ extends/uses

→ Advantages & Disadvantages

- ↳ detailed characterization of all possible interaction with the system
- ↳ helps in drawing system boundary, and scoping the requirements
- ↳ Use cases do not capture domain knowledge!
- ↳ Don't confuse use cases with a precise specification!



Use Cases - Example

Use Case Description

Name: Place Order

Precondition: A valid user has logged into the system.

Description:

1. The use case starts when the customer selects Place Order.
2. The customer enters his or her name and address.
3. If the customer enters only the zip code, the system will supply the city & state.
4. The customer will enter product codes for the desired products.
5. The system will supply a product description and price for each item.
6. The system will keep a running total of items ordered as they are entered.
7. The customer will enter credit card payment information.
8. The customer will select Submit.
9. The system will verify the information, save the order as pending, and forward payment information to the accounting system.
10. When payment is confirmed, the order is marked Confirmed, an order ID is returned to the customer, and the use case ends.

Exceptions:

In step 9, if any information is incorrect, the system will prompt the customer to correct the information.

Postcondition: The order has been saved in the system and marked confirmed.

Use Case Diagram

