Lecture 6: Starting points for elicitation

- Boundaries
  - Scoping the problem
- Stakeholders
  - Identifying the problem owners
- Goals
  - Identifying the success criteria
- Scenarios
  - Using concrete examples to understand the problem

Where do we start?

- Identify the problem
  - what is the objective of the project?
  - the "vision" of those who are pushing for it?
    - e.g., "Scheduling meetings is too costly right now"
- Scope the problem
  - given the vision, how much do we tackle?
    - e.g. "Build a system that schedules meetings", …or…
    - e.g. "Build a system that maintains people's calendars" …or…
- Identify solution scenarios
  - given the problem, what is the appropriate business process for solving it?
    - e.g. "Anyone who wants to schedule a meeting goes to the secretary, gives details and the secretary handles the rest", …or…
- Scope the solution
  - Given a business process, what parts should be automated, and how?
    - e.g. "Computer takes in scheduling request details, outputs a solution" …or…
    - e.g. "Solution arrived at interactively by secretary and computer" …or…

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 does the problem manifest itself? (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 (dumb?) questions
    - Recognise the domain expertise of the people you talk to

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

Identifying the Problem

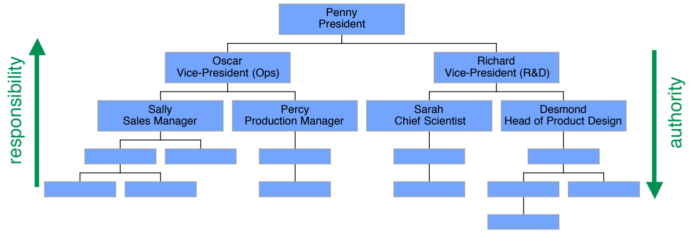
- Vague problem stated by the customer:
  - E.g. university textbook store:
    - Manager wants to computerize the book order forms filled out by instructors;
  - E.g. A large insurance company:
    - Claims manager wants to cut down the average time it takes to process an insurance claim from 2 months to 2 weeks
  - E.g. A telecommunications company:
    - CIO wants to integrate the billing system with customer record systems of several affiliates, so there is only one billing system...
  - E.g. Large Government Aerospace Agency:
    - The president wants to send a manned mission to Mars by the the year 2020
- Often you only see symptoms rather than causes:
  - E.g. "Ontario patients needing X-ray scans have to wait for months"
  - The long wait is the symptom, not the problem. The problem may be:
    - Shortage of X-ray machines;
    - Shortage of trained staff;
    - Shortage of doctors to process the data
    - Inefficient scheduling procedures

Stakeholders

→ **Stakeholder analysis:**
- Identify all the people who must be consulted during information acquisition

→ **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"
  - Wants to get best value for money invested!

Finding stakeholders: The Org Chart



→ **Organization charts show**
- Areas of responsibility (flows upwards)
- Lines of authority (delegated downwards)

→ **A useful tool for figuring out where the stakeholders are**
- …but remember that most activities involve connections that cross the org chart

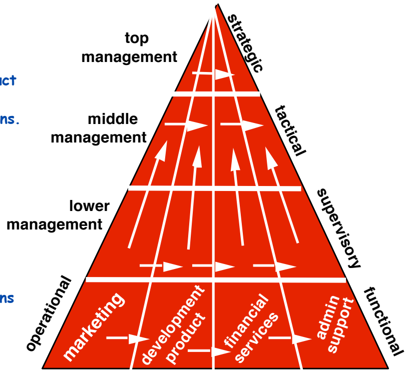
Levels of authority

→ **Top management**
- establishes goals
- does long-range planning
- determines new market & product developments
- decides on mergers & acquisitions.

→ **Middle management**
- sets objectives
- allocates & controls resources
- does planning
- measures performance

→ **Lower management**
- supervises day-to-day operations
- takes corrective action when necessary.

→ **Operational level**
- performs day-to-day operations



Identifying Stakeholders' Goals

Source: *Adapted from Anton, 1996.*

→ **Approach**
- Focus on *why* a system is required
- 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 refinements and alternatives

→ **Advantages**
- Reasonably intuitive
- Explicit declaration of goals provides sound basis for conflict resolution

→ **Disadvantages**
- Captures a static picture - what if goals change over time?
- Can regress forever up (or down) the goal hierarchy

Goal Modeling

→ (Hard) Goals:
- Describe functions that must be carried out. E.g.
  - Satisfaction goals
  - Information goals

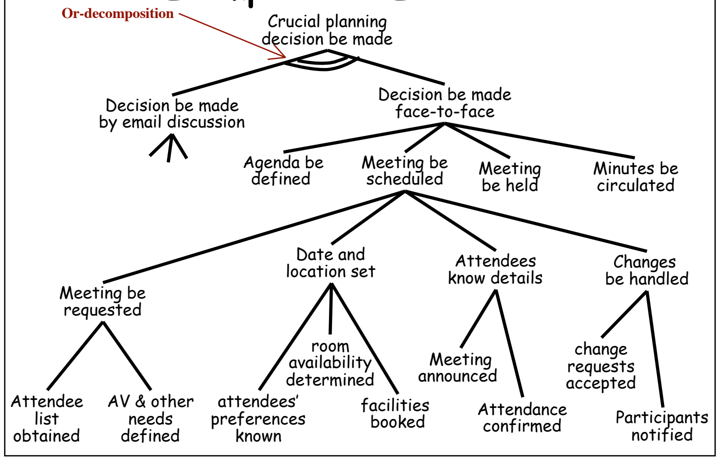
→ Softgoals:
- Cannot really be fully satisfied. E.g.
  - Accuracy
  - Performance
  - Security
  - …

→ Also classified temporally:
- Achieve/Cease goals
  - Reach some desired state eventually
- Maintain/Avoid goals
  - Keep some property invariant
- Optimize
  - A criterion for selecting behaviours

→ Agents:
- Owners of goals
- Choice of when to ascribe goals to agents:
  - Identify agents first, and then their goals
  - Identify goals first, and then allocate them to agents during operationalization

→ Modelling 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

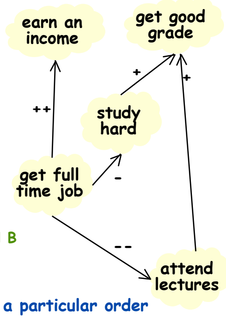
Example Goal Elaboration



Goal Analysis

→ Goal Elaboration:
- "Why" questions explore higher goals (context)
- "How" questions explore lower goals (operations)
- "How else" questions explore alternatives

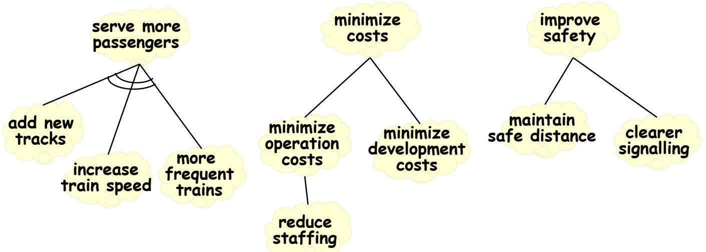
→ Relationships between goals:
- One goal helps achieve another (+)
- One goal hurts achievement of another (-)
- One goal makes another (++)
  - Achievement of goal A guarantees achievement of goal B
- One goal breaks another (--)
  - Achievement of goal A prevents achievement of goal B
- Precedence ordering – if goals must be achieved in a particular order



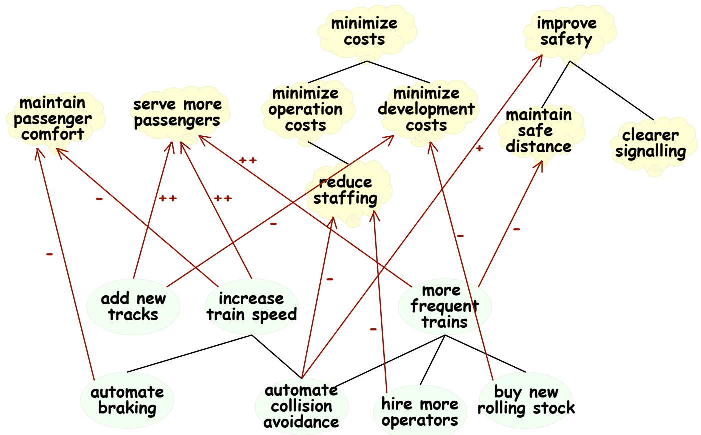
→ Obstacle Analysis:
- Can this goal be obstructed, if so how?
- What are the consequences of obstructing it?

Softgoals

→ Some goals can never be fully satisfied
- Treat these as softgoals
  - E.g. "system be easy to use"; "access be secure"
  - Also known as 'non-functional requirements'; 'quality requirements'
- Will look for things that contribute to satisficing the softgoals
- E.g. for a train system:



Softgoals as selection criteria



Scenarios

Source: *Adapted from Dardenne, 1993.*

➜ 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
- Hard to check for completeness

Example Scenario

<u>Title: Successful meeting scheduled using messaging option</u>

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?

Summary

➜ Scoping is important

- What is scope of problem should you tackle?
- What is the scope of the desired solution?

➜ Ask Who and Why questions

- Who are the key stakeholders?
- Why do they want this problem solved?
- Analyze their goals.

➜ Ask How questions

- How is each goal satisfied?
- How might a new system improve things?
- Develop scenarios.