Requirements Analysis, *Modeling* and Specification

- Requirements Analysis, Modeling and Specification
- Problem
- Carving the Solution Space
- Prioritizing Requirements
What is a Model?
Requirements Analysis, Modeling & Specification

A continuous loop

Problem

Problem Elicitation

Text, Notes, Knowledge

Problem Specification

Diagram, Charts, Tables

Problem Specification

Diagram, Charts, Tables

Diagram, Charts, Tables

Modelling

(often informal models)

(concrete models)

(reqs. spec. models)

mediating representations
It is more important to understand the problem than the solution. [Albert Einstein]

“A problem unstated is a problem unsolved”
Douglas Ross, 1977

From Sam Supakkul’s presentation
What is a problem?
Is storm coming a problem?

©Lawrence Chung
Yes, if we want to have outdoor fun

Problem
No, if we’re dying for it
Problem or not depending on the goal we have

Problem

Blessing
Problem = TO-BE – AS-IS

A problem can be defined as the difference between things as *they are now* and things as *they are desired*.

It is more important to understand the problem than the solution. [Albert Einstein]
A Problem Analysis Roadmap

Enterprise/Business Problem

Enterprise/Business problem defined

- Identify stakeholders for problem.
- Root cause analysis.

Actual problem identified and defined

- Understood problem in the context of business goals.
- Identify constraints on the system/project

Problem validated/adjusted

- Gain agreement on the problem def. wrt. root causes.
- Consider alternatives & choose the best solution(s) to meet the goals.

Best solution identified

- Define the solution system boundary
- Reassess that the solution idea is the best solution.

Elicit Requirements

- Expand stakeholder list for solution.

Establish common vocabulary => Glossary with a Domain Model
One Man's Ceiling is Another Man's Floor!

One Man's Problem is Another Man's Solution!

One Man's Floor is Another Man's Ceiling!

One Man's Solution is Another Man's Problem!

©Lawrence Chung
What Is the Problem Behind the Problem?

Fishbone Diagram Techniques

The perceived business problem.

Customers are dissatisfied with our service.

List contributing causes to the identified problem. Keep asking “Why?” (expand each rib).
Problem Analysis – Validating a Solution

We need ATMs.

- The perceived solution to some ill-defined problem.

List the reasons why the solution is the right solution. Keep asking “Why?” (expand each rib).

What is the problem of your project? Why is your solution the right solution?
Representation of problems
fault tree diagram

Hot Water Heater Explodes

Pressure Relief Valve Fails

Relief Valve

Frozen

Relief Valve Disch Line Plugged

RV Line Frozen

RV Line Crimped

Pipe Plug Installed in line

High Water Temperature

Temp Regulator Fails

High Temp Cut-off Fails

AIID Symbol

OR Symbol

Transfer Symbol

http://www.qualityassembling.com/resources/problem_solving/images/fault_tree.gif
Problem Interdependency Graph (PIG)
But what if there are too many problems?

Pareto principle

In economics
The original observation was in connection with income and wealth. Pareto noticed that 80% of Italy's wealth was owned by 20% of the population.[4] He then carried out surveys on a variety of other countries and found to his surprise that a similar distribution applied. [Wikipedia]
Pareto effect Analysis

Contributing Causes

- Banking at Night
- More banking locations
- Banking at airport
- Queues too long
- Privacy while banking
- Other Reasons

% Contribution

©Lawrence Chung
Focus on Largest Contributors - *Pareto’s Law*

Vital-few-trivial-many; 80-20 rule
Priority; criticality

Rank in order. Use the 80-20 Rule to focus on the top contributing causes to address the greatest portion of the problem.

What are in the 20% of the problem of your project?
Requirements Prioritization


Given n requirements,
- Create n x n matrix
- Compare each pair
  - $\text{entry (i, j)} =$
    - 1 if i and j are of equal value
    - 3 if i is slightly more preferred than j
    - 5 if i is strongly more preferred than j
    - 7 if i is very strongly more preferred than j
    - 9 if i is extremely more preferred than j
  - $\text{entry (j, i)} = \frac{1}{\text{entry (i, j)}}$
- Estimate the eigenvalues
  - Calculate the sum of each column
  - Divide each entry by the sum of its column
  - Calculate the sum of each row
  - Divide each row sum by n

This gives a value for each requirement based on estimated percentage of total value of the project.
Requirements Prioritization

An Analytic Hierarchy Process (AHP) Approach

Example

<table>
<thead>
<tr>
<th></th>
<th>r1</th>
<th>r2</th>
<th>r3</th>
<th>r4</th>
</tr>
</thead>
<tbody>
<tr>
<td>r1</td>
<td>1</td>
<td>1/3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>r2</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>r3</td>
<td>1/2</td>
<td>1/5</td>
<td>1</td>
<td>1/3</td>
</tr>
<tr>
<td>r4</td>
<td>1/4</td>
<td>1/3</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

**normalize columns**

<table>
<thead>
<tr>
<th></th>
<th>r1</th>
<th>r2</th>
<th>r3</th>
<th>r4</th>
</tr>
</thead>
<tbody>
<tr>
<td>r1</td>
<td>0.21</td>
<td>0.18</td>
<td>0.18</td>
<td>0.48</td>
</tr>
<tr>
<td>r2</td>
<td>0.63</td>
<td>0.54</td>
<td>0.45</td>
<td>0.36</td>
</tr>
<tr>
<td>r3</td>
<td>0.11</td>
<td>0.11</td>
<td>0.09</td>
<td>0.04</td>
</tr>
<tr>
<td>r4</td>
<td>0.05</td>
<td>0.18</td>
<td>0.27</td>
<td>0.12</td>
</tr>
</tbody>
</table>

**sum the rows**

<table>
<thead>
<tr>
<th>sum</th>
<th>Sum/4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.05</td>
<td>0.26</td>
</tr>
<tr>
<td>1.98</td>
<td>0.50</td>
</tr>
<tr>
<td>0.34</td>
<td>0.09</td>
</tr>
<tr>
<td>0.62</td>
<td>0.16</td>
</tr>
</tbody>
</table>

*Also should compute the consistency index, since the pairwise comparisons may be inconsistent*
Using "Shall" and Related Words

- "shall" indicates a binding provision, i.e., one that must be implemented by the specification users.
- To state non-binding provisions, use "should" or "may."
- Use "will" to express a declaration of purpose (e.g., "The government will furnish ...") or to express future tense
• Appendix
Exercise: Student Application Processing System

(AS-IS) (Problem) (TO-BE)

To be continued later ...
What is a problem?

Undesirable situation

causes

hurts

Goal

A phenomenon or a situation that hurts our goals

Phenomenon

©Lawrence Chung
Representation of problems
fault tree diagram

- Clear relationships between siblings
  - AND/OR

- No relationship with
  - Goals
  - Alternatives
Carving the Product Space

Requirements represent a compromise.

User Needs

- Wants

Environmental

Economic

Political

Feasibility

User Wants

Customer Needs

Admissible Product Space Solution

Developer Perspective

(acceptable)

(acceptable)

Laws & Standards

(acceptable)

(acceptable)

(acceptable)

(acceptable)

(acceptable)

Technology Risks

(acceptable)

(acceptable)

(acceptable)

(acceptable)

(acceptable)

(acceptable)
Carving the Product Space

Requirements represent a compromise.

Example

E.g., Wired/Wireless phone & PCS

Admissible Product Space Solution

User Needs
- Ordinary
  - Wants: all in one device (phone, pager, TV, computer)
  - Needs: cellular + wired service
  - Wants: video phone + pager + teleconferencing
  - Needs: cellular + wired service + ID

Customer Needs
- Sprint
  - Wants: 2 yr, $2 billion, across US
guarantee w. CDMA only
  - Needs: 3yr, $3B, MANs, CDMA + FDMA

Developer Perspective
- (acceptable)
- (unacceptable)

Technology Risks
- (acceptable)
- (unacceptable)
  - Unacceptable: 95% guarantee w. CDMA only
  - only 7-layer
  - Acceptable: 90% guarantee w. CDMA + FDMA

(acceptable)
- Laws & Standards
  - Unacceptable: towers/cells every 100 meters
  - Acceptable: close to hexagonal, but special BSs in subway, dense areas
  - Unacceptable: use any frequency -> hearing aid
  - Acceptable: ~ Zoning law

Varying degrees of acceptable/unacceptable solutions
D, S achieves R to solve P in D

What next?

Domain/World/Enterprise/Business

D, Problem => Requirements => Specification
D, S achieves R to solve P in D

Define Boundaries for the Enterprise/Business and the Solution

Consider Application Processing

More on this in Enterprise Modeling
\[ D, S \text{ achieves } R \text{ to satisfy/satisfice } G \text{ (solve } P \text{) in } D \]

\[ M^0, \text{Prog}^0 \models S^0; S^0, D^0 \models R^0; R^0, D^0 \models G; (G \models \neg P) \lor (G \models \neg \neg P) \]

**Exercise**

**An Application Processing System**

**D**: include a functional model, a workflow model, an informational model, a BM

**P**: include complaints (both external and internal), weaknesses, etc.

**G**: include wants and needs countering **P** – both hard and soft

**R**: include an interaction model between **D** and **S**

**S**: include a functional model, an informational model, a behavioral model

**Requirements should contain nothing but information about the environment.**
Modeling is Everywhere

- Problem Elicitation
  - exploratory, brain-storming, open-ended thinking
  - elaboration of unclear goals and needs
  - identification of sources, views, needs & wants

- Problem Analysis

  ![Diagram of Detect Defects and Resolve Defects]

  - determination of "real" users (<= identification of sources)
    - (e.g., stratification during sampling, questionnaires & interviewing)
    - customers, tellers, other employees
  - clarification of goals (e.g., Goal-directed approach)
  - detecting differences in views and integrating them, & recording rationale
  - resolving a mismash of wants and needs
  - prioritization of defects
  - understanding (all constraints on the) solutions and evaluating them
  - risk analysis (e.g., scenarios)

- Problem Specification
  - choose formal notations
  - create a formal model of the requirements

---

Wicked Problem

- process of understanding real-world problems, how they relate to stakeholder needs, and proposing solutions to meet those needs.

Modeling is NOT Perfect
[adapted from Jackson, 1995, p124-5]

- There will always be phenomena in the model that are not present in the application domain
- There will always be phenomena in the application domain that are not in the model
- Perfecting the model is not always a good use of your time
### Stakeholders in the Vision Document

**Stakeholder** - An individual who is materially affected by the outcome of the system or the project(s) producing the system.

<table>
<thead>
<tr>
<th>Stakeholder</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td>Representative</td>
<td>Kelly Hansen</td>
</tr>
<tr>
<td>Description</td>
<td>User</td>
</tr>
<tr>
<td><strong>Type</strong></td>
<td>The Registrar is typically a college-educated professional with full computer skills. The Registrar is trained and experienced with the use of the current batch-oriented registration.</td>
</tr>
<tr>
<td><strong>Responsibilities</strong></td>
<td>The Registrar is responsible for administering course registration for each school term. This includes supervising administrative and data entry personnel.</td>
</tr>
<tr>
<td><strong>Success Criteria</strong></td>
<td>The registrar’s primary responsibility will be maintaining student and professor databases, and opening/closing courses to registration. The registrar’s office will also be required to perform….</td>
</tr>
<tr>
<td><strong>Involvement</strong></td>
<td>The registrar’s primary responsibility will be maintaining student and professor databases, and opening/closing courses to registration. The registrar’s office will also be required to perform…..</td>
</tr>
<tr>
<td><strong>Deliverables</strong></td>
<td>Management reviewer – especially related to functionality and usability of features required by the Registrar staff.</td>
</tr>
<tr>
<td><strong>Comments/Concerns</strong></td>
<td>None</td>
</tr>
</tbody>
</table>

©Lawrence Chung
An externally observable service by which the system directly fulfills one or more stakeholder requests.

specifies, from a black-box perspective, how the solution interacts with the outside world.

Carving the Product Space

Problem

Problem Space

The system to be built

Solution Space

Traceability

Needs

Features

Software Requirements

Test Procedures

Design

User Doc
Carving the Product Space

Requirements exist at many levels of abstraction, possibly with different terminology.

- Stakeholder **Needs**
- Product or System **Features**
- Software **Requirements**
  - Design Spec
  - Test Procedures
  - Documentation Plans