OBJECT-ORIENTED MODELING AND DESIGN

System Conception

CONTENTS:

1. Devising a System Concept
2. Elaborating a Concept
3. Preparing a Problem Statement
4. Chapter Summary.

System Conception deals with the genesis of an application. Some people, who understand both business needs and technology, think of an idea for an application. Developers then explore the idea to understand the needs and devise possible solutions. Purpose of System Conception is to defer the details and understand the big picture…...what need does the proposed system meet, can it be developed at a reasonable cost, and will the demand for the result justify the cost of building it..?

1. Devising a System Concept.

- Most ideas for a new system are extension of existing ideas.
- Eg: HR dept may have a DB of employee benefit choices and require that a clerk enter changes.
- An obvious extension is to allow employees to view and enter their own changes.
- Many issues :(Security, reliability, privacy ...)
- But.. New idea is a straightforward extension of an existing concept.
- Eg2: online auction system. which is the automation of the existing system which is running presently manually .

Some ways to find new system concepts:

- New Functionality- Add functionality to an existing system.
- Streamlining- remove restriction or generalize the way a system works.
- Simplification- Let ordinary persons perform tasks previously assigned to specialists.
- Automation- automate manual processes
- Integration- Combine functionality from different systems.

- Analogies- Look for analogies in other problem domains and see if they have useful ideas.
- Globalization-Travel to other countries and observe their cultural and business practices.

Following is the summary of system conception highlighting some essential points:

- System conception (requirement)
- Analysis
  - Domain analysis
  - Application analysis
- System Design
- Class diagram
- Implementation

Focus of OOMD

- Testing, training, deployment, maintenance

Requirement is a big deal

- 37% of software project failure is attributed to requirement [Larman2002]
- Gaps of understanding between stakeholders, esp., clients + users vs. pm + developers.
- *System conception* looks to close the gaps through disciplined development of requirement document.

E

2. Elaborate a concept

A good system concept must answer the following questions.

- Who is the application for ?
- What problems will it Solve?

- Where will it be used ?
- When is it needed?
- Why is it needed ?
- How will it work

Who is the application for?

- Understand the stakeholders of the system;
- Usually two important ones are:
  Financial sponsor (client)
  End Users.

What problems will it Solve?

- Bound the size of effort and scope of system
- Determine what feature's in and what's out.

Where will it be needed?

- Characterize the environment the system will be used, e.g.,
  - Mission-critical?
  - Experimental?
  - Enhancements to existing system?
- For commercial products, characterize the customer base.

When is it needed?

- Feasible time, T_f
  - The time in which the system can be developed within the constraints of cost and available resource
- Required time, T_r
  - The time that the system is needed to meet the business goals.
- If ($T\_r < T\_f$), work with technologists and business experts to trim the system

Why is it needed?

- Prepare a business case
  - Financial justification, including
    - cost,
    - tangible benefits,
    - intangible benefits,
    - risk, and
    - alternatives (why build when you can buy?)
  - For a commercial product, estimate the number of units you can sell and determine a reasonable price.
    - Revenue must cover the cost

How will it work?

- Investigate feasibility of the problem
- Build prototype, if it helps clarifying a concept or removing a technological risk.

CASE STUDY

ATM Case Study (Automated Teller machine)

System Concept for an automated teller machine.

"Develop software so that customers can access a bank's computer and carry out their own financial transactions without the mediation of a bank employee"

- Who is the application for?

  The Vendor building the software (assume its we)

- What problems will it Solve?

  ATM is built to serve both the bank (automation) and the customer (ubiquitous)

- Where will it be used ?
  ATM are available at many stores, sporting events and other locations throughout the world

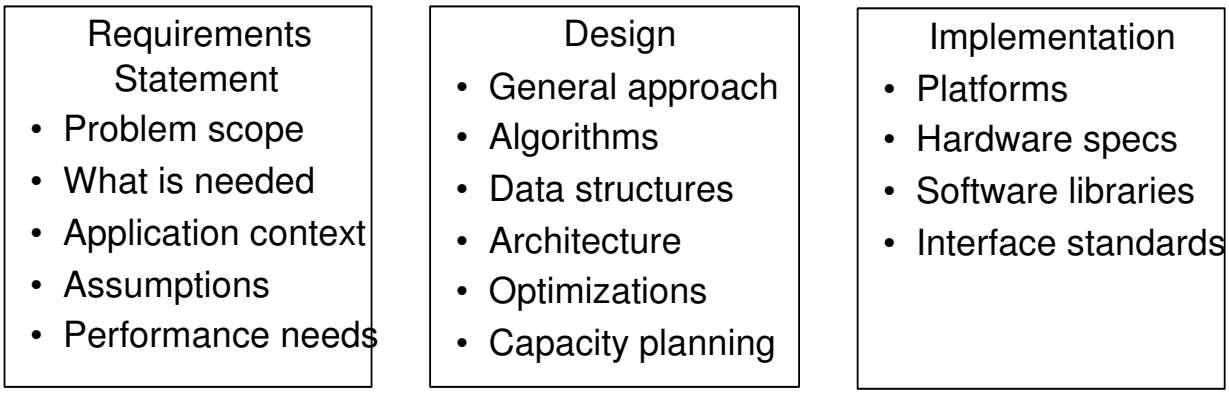
- When is it needed?
  From an economic perspective, it is desirable to minimize the investment, maximize the revenue, and realize revenue as soon as possible. (OOMD Helps)

- Why is it needed ?

  A novel product could outflank competitors and lead to premium pricing.Here the case study is taken for explanation.

- How will it work?
  We adopt a three-tier architecture

3. Preparing a Problem Statement



- Throughout development, you should distinguish among requirements, design, and implementation.
- Do not Make early design and implementation decisions or you will compromise development?
- Most problem statements are ambiguous, incomplete, or even inconsistent.
- Some requirements are just plain wrong.
- …..The purpose of Analysis is to fully understand the problem and its implications

4. Chapter Summary.

- First Stage of a Project is to devise a new idea.
- This can involve a new system or an improvement to an existing system.
- Before investing survey on the risk factors and the benefits.
- When the system conception stage is complete, write a problem statement that serves as the starting point for analysis.
- Problem statement need not be complete, and it is prone to change, but helps to focus the attention of the project.

Exercise

1. Consider a new antilock braking system for crash avoidance in an automobile. Elaborate the following high level questions and explain your answers.
    a. Who is the application for? Whoa re the stakeholders? Estimate how many persons in your country are potential customers.
    b. Identify three features that should be included and three features that should be omitted.
    c. Identify three systems with which it must work
    d. What are to of the largest risks?

Answer:

Here is elaboration for an antilock braking system for an automobile.

a. An antilock braking system could target the mass market. If the antilock system was inexpensive and safer than current technology, it could be government mandated and installed on all cars. (Further study would be needed to determine what price is "inexpensive" and what would be a "significant" safety improvement.) There would be several stakeholders. Auto customers would expect improved safety and minimal detriment to drivability. Auto manufacturers would want to minimize the cost and quantify the benefit so they could tout the technology in their advertising. The government would be looking for a statistical safety improvement without compromising fuel efficiency. If the new system was inexpensive, worked well, and did not hurt drivability, all car owners could be potential customers. An expensive antilock system could be a premium option on high-end cars.

b. Desirable features would include: effective prevention of brake locking, ability to detect excessive brake wear, and acquisition of data to facilitate auto maintenance. Some undesirable features would be: reduced fuel efficiency, reduced drivability, and greater maintenance complexity.

c. An antilock system must work with the brakes, steering, and automotive electronics.

d. There would be a risk that an antilock braking system could fail leading to an accident and a lawsuit. Also it might be difficult to understand fully how the antilock system would interact with the brakes.

2. Prepare a problem statement, similar to that of the ATM system:
    a. Bridge player
    b. Change making machine
    c. Car cruise control
    d. Electronic typewriter
    e. Spelling checker
    f. Telephone answering machine.

Answer:

a. bridge player. Develop a computerized bridge playing system. The system will support as many as four players. From zero to four of the players will be computer generated. The computer must deal random hands, bid using standard conventions, play any hand, and keep score. All computer generated opponents must be "honest" and not take advantage of any knowledge of the hidden cards of opponents. There should be a setting for level of difficulty which the human user can adjust. The program must have

an excellent, user-friendly interface with high resolution color graphics and quickly determine bids and plays. It is permissible for performance to degrade as the level of play becomes more difficult. The system should keep an optional log in which it records the cards dealt and the bids and plays for a given game. The system should also accept a predefined deal and bid and play list; this is useful for studying fine details of a bridge game.

b. change-making machine. Design the software for a machine that accepts paper currency and returns change. Important design goals in order of importance are: rejection of counterfeit and foreign currency, determination of denomination, correct dispensing of change, software versatility, and low cost. The software may be custom written for a particular microprocessor chip. Versatility refers to the fact that the software must be configurable to a variety of conditions. The software must be easy to reconfigure for international use with different types of currency and different formulas for dispensing change. The software must allow a fee to be imposed for change-making service.

c. car cruise control. Design a cruise control system for an automobile. The control has four buttons: *on/off*, *set*, *coast*, and *resume*. Once the control system is on, the driver accelerates to a desired speed and presses *set*. The speed will be maintained within a fixed tolerance until the driver hits the brake or presses the *off* button.The driver may accelerate above the preset speed by using the accelerator; once the accelerator is released the car will resume the preset speed. If the driver hits the brake, the cruise control is disabled until the driver presses the *resume* button at which time the car will resume the preset speed. If the driver holds the *coast* button, the car will decelerate until the button is released at which point the car's speed becomes the new desired speed. Abrupt changes or oscillations in speed will be avoided.

d. electronic typewriter. Design the software and hardware for an electronic typewriter. The typewriter only has to support the standard *QWERTY* arrangement. Keys that are not letters and numbers can be arranged in the manner that seems most appropriate. Cost of the typewriter is paramount; intentions are to aim for the low end of the marketplace. The power supply need only handle the standard 120 volts of North America. Color of the typewriter is irrelevant—choose an inoffensive color that is inexpensive. The typewriter should be lightweight and easy to assemble; it need not be easy to repair. The typewriter should have a one-line electronic display; the line does not print until the carriage return is pressed. This buffering simplifies correction of minor typing errors.

e. spelling checker. Design the software for a spelling checker. The spelling checker must find incorrect words in a document and suggest corrections for all misspelled words. The spelling checker must use a word dictionary and permit the user to add new

words. The software must run on PCs and integrate with a variety of word processors. It must be memory resident and easily activated with a few keystrokes. The spelling

checker must accept commands from both a mouse and keyboard; keyboard commands should be redefinable by modifying a configuration file. The software must be easy to use and present a polished pull-down menu interface. It is important that the system occupy as little memory as possible.

f. telephone answering machine. Design the software for a telephone answering machine. The software must provide the following services as a minimum: answer the phone after a predetermined number of rings, play a recorded message, record the caller's message, and hang up after a predefined length of recording. The software should support remote dial-in and identification by password to hear any recorded calls. The software should be suitable for burning into ROM. As such it is important that the software be small in size and extremely reliable since it would be very costly to update the equipment once it is in a customer's hands. The software must operate in real-time but early projections are that this goal is easy to meet with modern microprocessors. You may choose any CPU chip for developing the software that you choose, but the wholesale quantity price of the CPU chip must be $10 or less.