

SOFTWARE REQUIREMENTS ENGINEERING

LECTURE # 3

PROBLEM ANALYSIS

17th April, 2013

Engr. Ali Javed

Instructor Information

2

- ❑ Course Instructor: **Engr. Ali Javed**
Assistant Professor
Department of Software Engineering
U.E.T Taxila
 - ✓ Email: ali.javed@uettaxila.edu.pk
 - ✓ Website: <http://web.uettaxila.edu.pk/uet/UETsub/perSites/mySite.asp?frmEmail=ali.javed@uettaxila.edu.pk>
 - ✓ Contact No: +92-51-9047747
 - ✓ Office hours:
 - **Monday, 09:00 - 11:00, Office # 7 S.E.D**

- ❑ Lab Instructor's: Engr. Asra, Engr. Sobia

Course Information

3

- ❑ **Course Name: Software Requirements Engineering**
- ❑ **Course Code: SE-203**
- ❑ **CMS Link:** <http://web.uettaxila.edu.pk/CMS/SP2013/seSREbs/>

4

Presentation Outline

- ❑ The Software Team
- ❑ Problem Analysis
- ❑ 5 Steps of Problem Analysis
 - ✓ Gain Agreement on the Problem Definition
 - ✓ Understand the Root Causes
 - ✓ Identify the Stakeholder and Users
 - ✓ Define the Solution System Boundary
 - ✓ Identify the constraints to be imposed on the solution

5

The Software Team



The Software Team

6

- ❑ Effective requirements management can be accomplished only by an effective software team.
- ❑ Requirements management touches every team member in different ways.
- ❑ Effective requirements management requires mastering six team skills.



Team Members Have Different Skills

7

- ❑ One of the most interesting things about teams is that individual team members have different skills, that's what makes a team a team.
- ❑ In the software team, we hope that
 - ❑ Some players have proven their ability to work with the customers effectively,
 - ❑ Some have software programming abilities, and that
 - ❑ others have testing abilities.
 - ❑ Still other team players will need design and architecture abilities.
 - ❑ Many more skills are required as well.



Requisite Team Skills for Effective Requirements Management

8

- ❑ six team skills that are necessary for a modern software team to successfully address the requirements challenge are mentioned here
 - ✓ **Team Skill 1, Analyzing the Problem**
 - ✓ **Team Skill 2, Understanding User and Stakeholder Needs**
 - ✓ **Team Skill 3, Defining the System**
 - ✓ **Team Skill 4, Managing Scope**
 - ✓ **Team Skill 5, Refining the System Definition**
 - ✓ **Team Skill 6, Building the Right System**



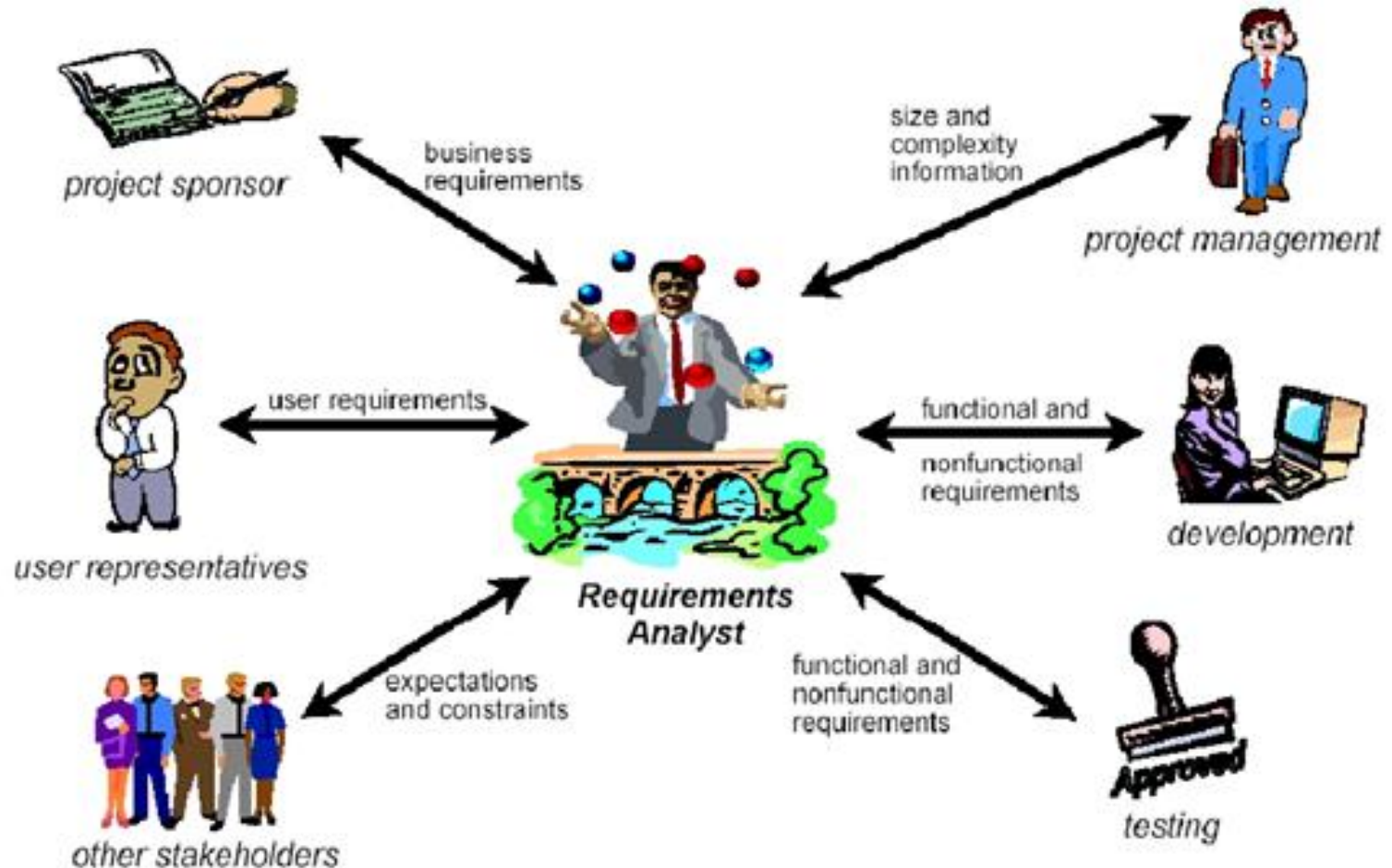
Requirements Engineer / Analyst [2]

9

- Requirements Engineer/ IT Business Analyst is charged with working with the project stakeholders and end users to elicit, understand, analyze, and document the requirements for a system in order to solve a given business problem.
- Other common titles for this role are: Requirements Analyst, Functional Architect, Business Systems Analyst, Business Analyst (generic term), etc.
- *Requirements analyst* is a project role, not necessarily a job title. One or more dedicated specialists can perform the role, or it may be assigned to any of a number of team members: the project manager, product manager, developer e.t.c
- Nevertheless, a talented analyst can make the difference between project success or failure.

Requirements Engineer / Analyst [2]

10



Tasks of Requirements Engineer /Analyst [2]

11

- Define Business Needs.



- Identify Project Stakeholders and user classes



- Elicit Requirements



- Analyze Requirements



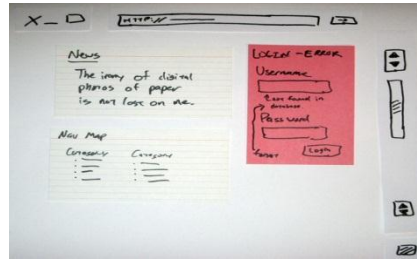
- Write Specifications

[Elicit : Analyze : Document] Repeat

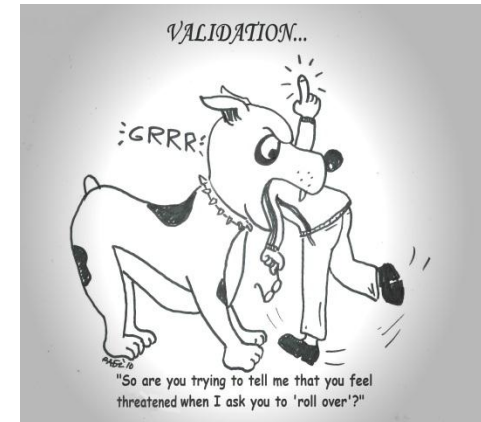
Tasks of Requirements Engineer /Analyst [2]

12

- Model the Requirements



- Lead Validation



- Facilitate Prioritization



- Manage Requirements

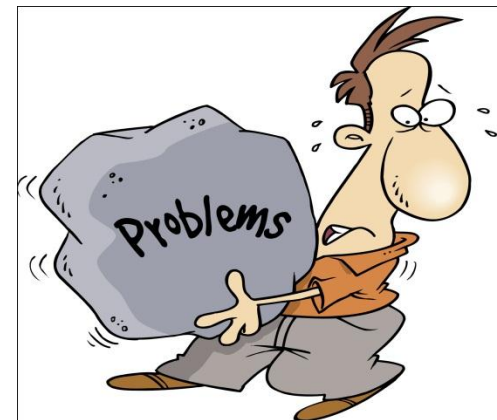




Problem Analysis [1]

13

- ❑ Problems and opportunities are just flip sides of the same coin; your problem is my opportunity.
- ❑ **Problem analysis is the process of understanding real-world problems and user's needs and proposing solutions to meet those needs.**
- ❑ Problem domain must be analyzed and understood, explore a variety of solution domains.
- ❑ Find the optimal solution for the problem among the variety of solutions.
- ❑ In order to be able to do problem analysis, we should know what a problem is, **a problem can be defined as the difference between things as perceived and things as desired.**

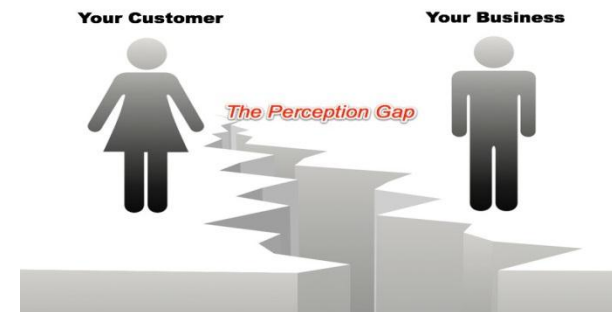




Problem Analysis [1]

14

- Sometimes, the simplest solution is a workaround, or revised business process, rather than a new system.
 - ✓ Changing the user's desire or perception may be the most cost-effective approach to address a problem.
 - ✓ Practical experience shows many examples where changing the perception led to the highest-quality, fastest, and cheapest solutions available
 - ✓ As problem solvers, it is recommended to explore these alternative solutions before leaping into a new system solution.
- However, when these alternative activities fail to reduce this gap, then we have to actively change the distance between perception and desire by defining and implementing new systems



Steps of Problem Analysis [1]

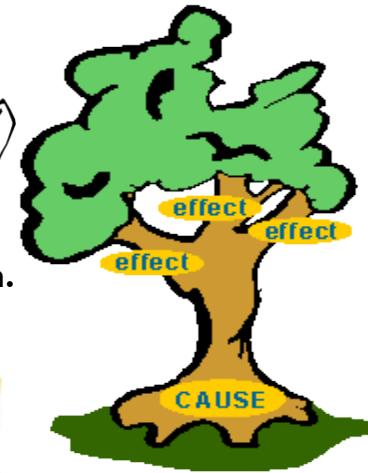
15

- The goal of problem analysis is to gain a better understanding, before development begins, of the problem being solved.

1. Gain agreement on the problem definition.



2. Understand the root causes—the problem behind the problem.

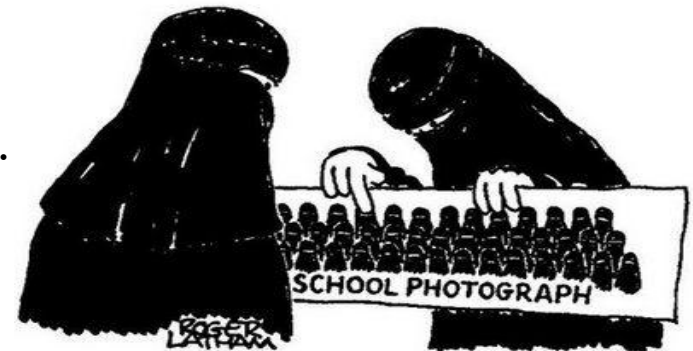


3. Identify the stakeholders and the users.



4. Define the solution system boundary.

5. Identify the constraints to be imposed on the solution.



"That's me, there"

Gain Agreement on the problem definition

16

- ❑ The first step is to gain agreement on the definition of the problem to be solved.
- ❑ One of the simplest ways to gain this agreement is to simply write the problem down and see whether everyone agrees.
- ❑ It is often helpful to understand some of the benefits of a proposed solution
- ❑ Having the user describe the benefits provides additional contextual background on the real problem.



The problem Statement

17

- You may find it helpful to write your problem down in a standardized format (Table 1).

Table 1: Problem Statement Format

Problem	Description
The Problem of	Describe the problem.
Affects	Identify stakeholders affected by the problem.
The result of which	Describe the impact of this problem on stakeholders and business activity.
Benefits of	Indicate the proposed solution and list a few key benefits.

Step 2: Understand the Root Causes

18

- Once you have an understanding of the larger problem, your team can use a variety of techniques to gain an understanding of its causes.

- One such technique is root cause analysis, which is a systematic way of uncovering the root, or underlying, cause of an identified problem.

- For example,
 - ✓ consider a real-world example: a company, **GoodsAreUs**, manufactures and sells a variety of inexpensive, miscellaneous items for home and personal use.

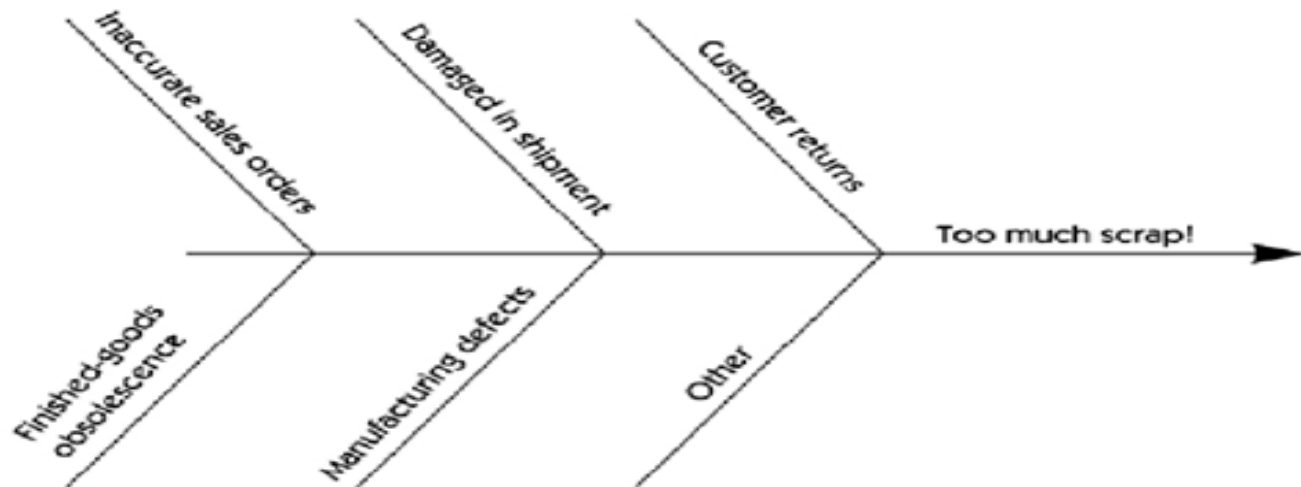
 - ✓ As the company addresses the problem of insufficient profitability, it uses total quality management (TQM) techniques for problem solving

 - ✓ Based on this experience, the company quickly focused on its cost of nonconformance, which is the cost of all the things that go wrong and produce waste, scrap, and other excess costs.

Step 2: Understand the Root Causes

19

- ✓ This cost includes rework, scrap, customer dissatisfaction, employee turnover, and other factors that are negative-value activities.
- ✓ **Production waste, or "scrap,"** was found to be one of the largest contributors after quantification of its cost of non quality
- ✓ This problem of excess scrap, then, is the next problem the company is trying to solve since it directly affects the larger problem of the cost of nonconformance, which in turn affects profitability
- ✓ TQM teaches us the use of the fishbone diagram (see Figure 1) to identify the problems behind the problem. Each source that contributes towards failure are listed as one of the "bones" on the diagram.



Step 2: Understand the Root Causes

20

- ❑ OK, so how do you determine the root causes? In many cases, it's a simple matter of asking the people directly involved what they think the root cause is.
- ❑ If the problem is more serious then it may be necessary to perform a detailed investigation of each contributing problem and to quantify its individual impact.
- ❑ This could vary from perhaps simple brainstorming by participants to a small data collection project or, potentially, to a more detailed experiment.



Addressing the Root Cause

21

- let's look at the problem analysis sequence that got us here
- A further fishbone analysis could then be used to determine what specific types of errors contribute to the inaccurate sales order problem.
- This new, more detailed data can then be used to define the features of the software system to address those errors.



Figure 3: Pareto Chart of Root Causes

Addressing the Root Cause

22

- We might like to fix all of the root causes on the "bones" of the diagram.
- But data shows that a number of root causes are simply not worth fixing, as the cost of the fix exceeds the cost of the problem.
- How do you know which ones to fix? You must determine the contribution, of each root cause. The results of this investigation can be plotted as a Pareto chart [3] or a simple histogram [4] that visually exposes the real culprits.

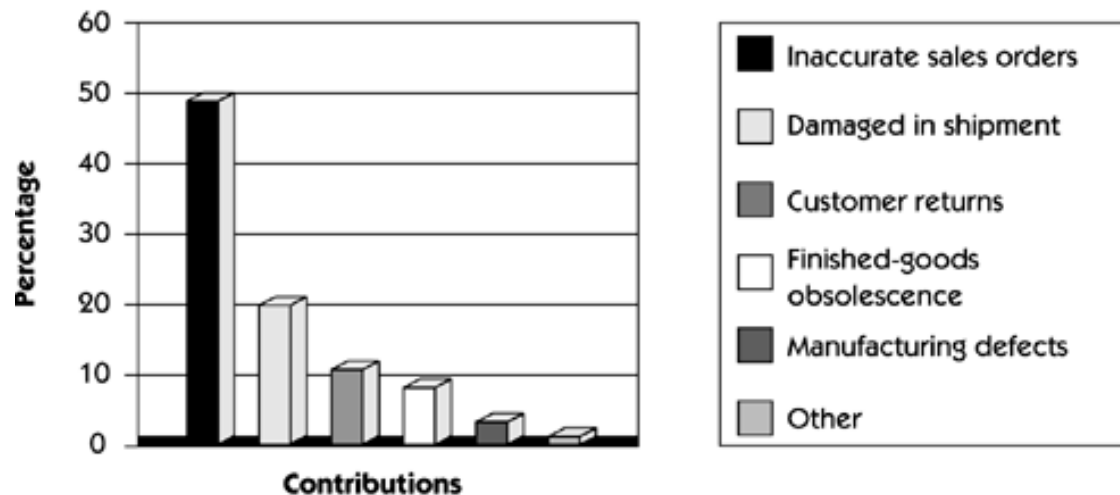


Figure 4: Pareto Chart of Root Causes

Addressing the Root Cause

23

- Once we have identified inaccurate sales orders as a root cause of a problem worth solving, we can create a problem statement for the sales order entry problem, as seen in Table 2

Table 2: Sales Order Problem Statement

Problem	Description
The Problem of	Inaccuracies in sales orders.
Affects	Sales order personnel, customers, manufacturing, shipping, and Customer service.
The result of which	Increased scrap, excessive handling costs, customer Dissatisfaction and decreased profitability
Benefits of	That creates a new system to address the problem include Increased accuracy of sales orders at point of entry Improved reporting of sales data to management Ultimately, higher profitability

- Once written, the problem statement can be circulated to the stakeholders for comment and feedback.

Addressing the Root Cause

24

□ Example 2

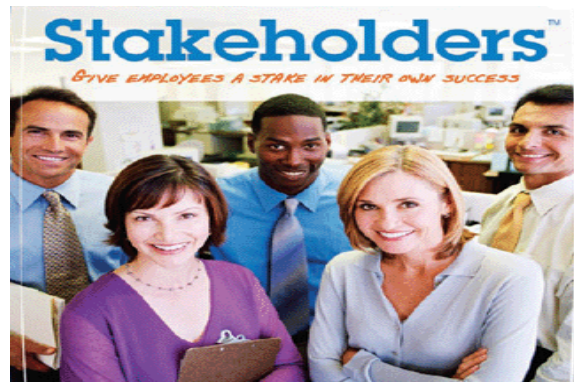
Table 3: Attendance System Problem Statement

Problem	Description
The Problem of	Student Attendance Management.
Affects	Teacher. Students. Attendance Department.
The result of which	Teacher-- Has to take some precious time and energy out of the lecture and take attendance. Student-- If misses calling the attendance then it's a absent mark for him. Attendance Department-- The department has to print hundreds of pages on daily bases to maintain attendance. After the teacher marks attendance the attendance sheet is taken to department and attendance is manually entered
Benefits of	Totally automates the process of taking attendance. Reduces the role of attendance department in printing attendance sheet and then manually entering the attendance record.

Step 3: Identify the Stakeholders and the Users

25

- Understanding the needs of the users and other stakeholders is a key factor in developing an effective solution.
- Effectively solving any complex problem typically involves satisfying the needs of a diverse group of stakeholders.
- Many stakeholders are users of the system, and their needs are easy to focus on because they will be directly involved with system use.
- However, some stakeholders are only indirect users of the system or are affected only by the business outcomes that the system influences.
- For example, they include the people and organizations involved in the development of the system, the subcontractors, the customer's customer, e.t.c
- Non user stakeholder needs must also be identified and addressed.



Step 3: Identify the Stakeholders and the Users

26

The following questions can be helpful in this process.

- ❑ **Who are the users of the system?**
- ❑ **Who is the customer (economic buyer) for the system?**
- ❑ **Who else will be affected by the outputs the system produces?**
- ❑ **Who will evaluate and approve the system when it is delivered and deployed?**
- ❑ **Are there any other internal or external users of the system whose needs must be addressed?**
- ❑ **Who will maintain the new system?**
- ❑ **Is there anyone else who cares?**



Step 4: Define the Solution System Boundary

27

- Once the problem statement is agreed to and the users and stakeholders are identified, we can turn our attention to defining a system that can be deployed to address the problem.
- In so doing, we enter an important transition state wherein we have to keep two things in mind:
 - ✓ an understanding of the problem and
 - ✓ the considerations of a potential solution.
- The next important step is to determine the boundaries of the solution system. The system boundary defines the border between the solution and the real world that surrounds the solution (Figure 4).
- Information, in the form of inputs and outputs, is passed back and forth from the system to users.



Figure 4: The inputs/system/outputs relationship

Step 4: Define the Solution System Boundary

28

- We divide the world in two:
 - ✓ Our System
 - ✓ Things that interact with our system
- Let's identify the "things that interact with our system" generically as "actors on our system."
- Once we understand the concept of an actor, we can illustrate a system boundary as shown in Figure 5

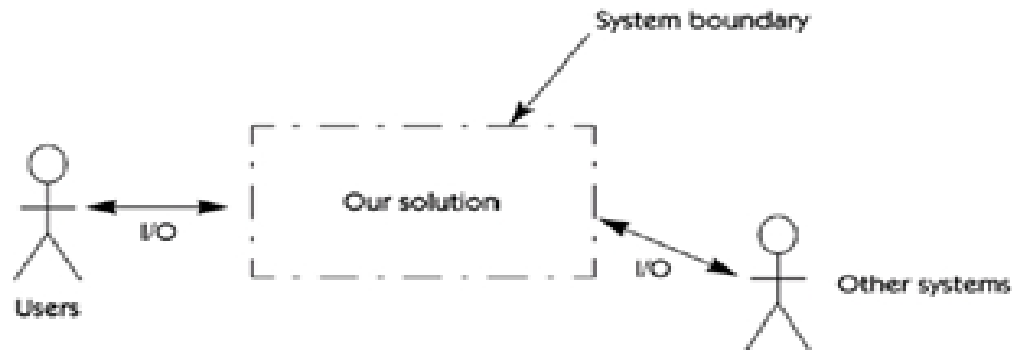
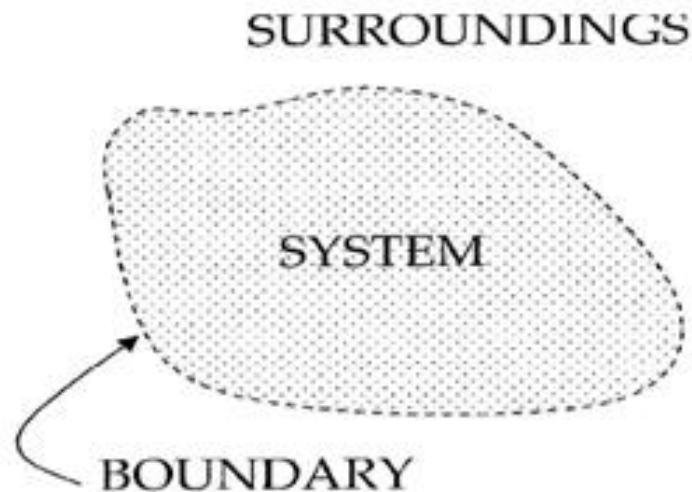


Figure 5: The inputs/system/outputs relationship

Step 4: Define the Solution System Boundary

29

- ❑ How do we find these actors? Here are some helpful questions to ask.
 - ✓ Who will supply, use, or remove information from the system?
 - ✓ Who will operate the system?
 - ✓ Who will perform any system maintenance?
 - ✓ Where will the system be used?
 - ✓ Where does the system get its information?
 - ✓ What other external systems will interact with the system?



Step 4: Define the Solution System Boundary

30

- After identifying the actors, the analyst can now create a "system perspective," a block diagram that describes the boundaries of the system, the users, and other interfaces. Figure 6 provides a system perspective for the new sales order system.
- The dotted line illustrates the system boundary for the proposed solution. The diagram shows that in order to solve our problem we will have to both develop a new system and modify some elements of the existing system (legacy system).

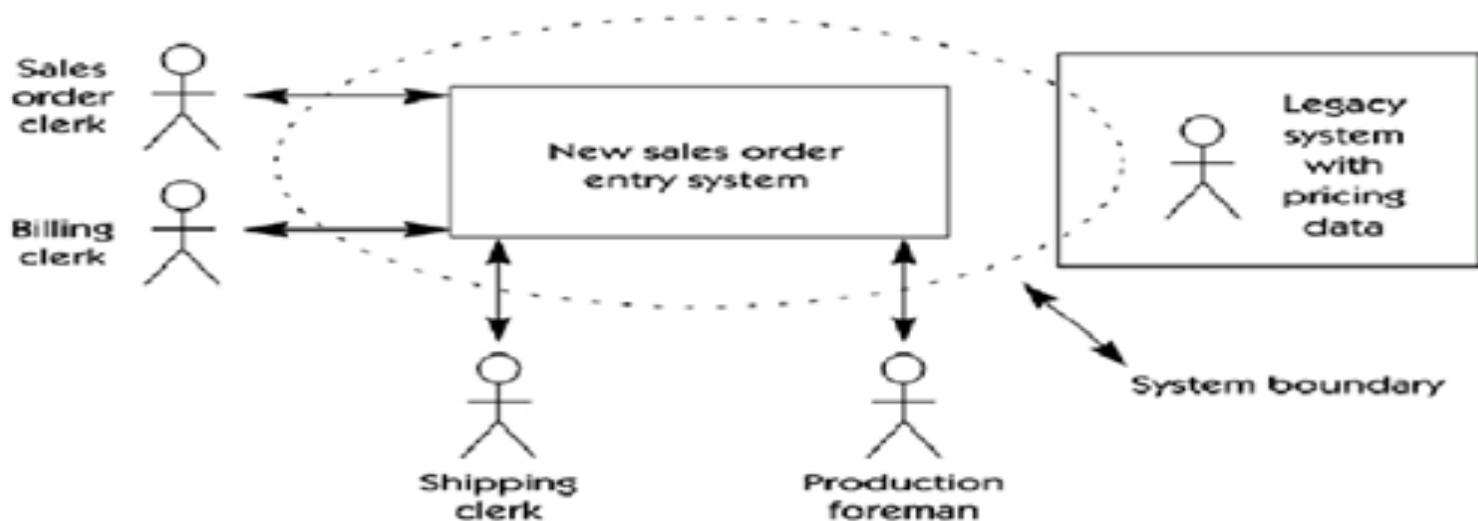


Figure 5: System Perspective

Step 5: Identify the Constraints to Be Imposed on the Solution

31

- ❑ Constraints are restrictions on the degrees of freedom we have in providing a solution.
- ❑ Each constraint has the potential to severely restrict our ability to deliver a solution as we visualize it. Therefore, each constraint must be carefully considered as part of the planning process.
- ❑ A variety of sources of constraints must be considered.
- ❑ These constraints may be given to us before we even begin or we may have to actively elicit them.



I KNOW, WE HAVE GOT THE
"SHOOT-AT-SIGHT" ORDERS...
BUT, WHAT DO WE SHOOT WITH?

Step 5: Identify the Constraints to Be Imposed on the Solution

32

- Some potential sources of constraints are listed below
 - ✓ Economics
 - ✓ Politics
 - ✓ Technology
 - ✓ Systems
 - ✓ Environment
 - ✓ Schedule & Resources

References

1. Managing Software Requirements: A Use Case Approach, Second Edition
By Dean Leffingwell, Don Widrig, Addison- Wesley
2. http://www.processimpact.com/articles/be_analyst.pdf
3. http://en.wikipedia.org/wiki/Pareto_chart
4. <http://en.wikipedia.org/wiki/Histogram>

For any query Feel Free to ask



34

