**Analysis Phase**

In the analysis phase, the systems analyst works extensively with the business users of the new system to understand their needs from the new system.

The basic process of analysis involves three steps:

■Understand the existing situation (the as-is system).

■ Identify improvements.

■ Define requirements for the new system (the to-be system).

The final deliverable of the analysis phase is the system proposal, which compiles the detailed requirements definition statement, use cases, process models, and data model together with a revised feasibility analysis and work plan.

We can think of the analysis phase as a combo of four steps:

1. Requirement determination
2. Use case analysis
3. Process modeling (DFD)
4. Data modeling (ERD)
5. Requirement Determination:

In many ways, determining requirements is the single most critical aspect of the entire SDLC. Therefore, analysts should devote considerable attention to the work performed in the analysis phase. It is here that the major elements of the system first begin to emerge. If the requirements are later found to be incorrect or incomplete, significant rework may be needed, adding substantial time and cost to the project.

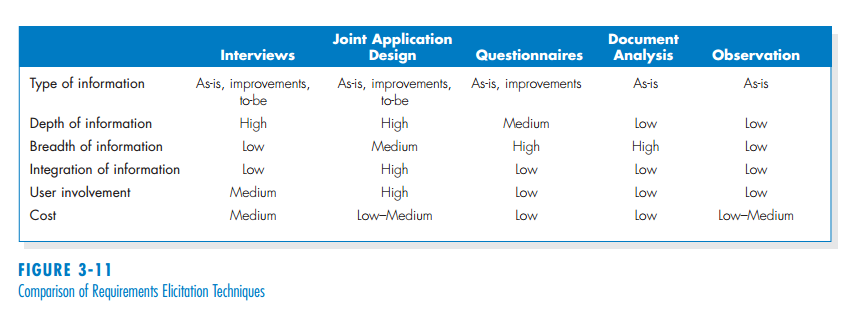
During requirements determination, the to-be system concept is easy to change because little work has been done yet. As the system moves through the subsequent SDLC phases (design and implementation), it becomes harder and harder to return to requirements determination and make major changes because of all of the rework that is involved.

Requirements determination is performed to transform the system request’s highlevel statement of business requirements into a more detailed, precise list of what the new system must do to provide the needed value to the business.

**Requirements elicitation techniques:**

We focus on the five most commonly used requirements elicitation techniques:

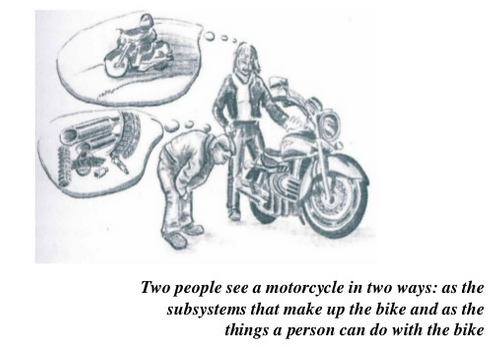
* Interviews
* JAD sessions
* questionnaires
* document analysis
* observation



1. Use case analysis:

Use case is a pictorial representation of requirement analysis which is easier to understand. DFD and ERDs’ are very useful for technical people but these easily confuse general users.

Different users look the same system differently. For example: a motorcycle mechanic sees the different parts and subsystems that make up the motorcycle whereas a bike racer would think what things he would be able to do with the bike.



**SEE SLIDE: REQUIREMENT MODELLING FOR DETAILED USE-CASE MODELLING**