

Information engineering

INVESTIGATING AND ANALYSING
SYSTEMS

B Holland

The SDLC Phase 2: Systems Investigation and Analysis

Phase 2 of the SDLC involves determining how the existing system functions and assessing what users would like to see in a new system (system requirements).

An important (but not the only) result of systems analysis and design is application software, software designed to support a specific organizational function or process, such as inventory management, payroll, or market analysis. In addition to application software

Systems Investigation

The systems investigation is a detailed fact-finding exercise that finds out about the organisation's current system and its problems, and determines what users want from the new system.

Systems Investigation

This stage of the study involves:

- *Fact-finding* by means of questionnaires, interviews, observation, reading existing organisational documents (such as handbooks, procedures manuals and standard operational forms, e.g. invoices, order forms, etc.).
- *Fact recording* using use case diagrams (CASE tools – computer –aided software engineering), data flow diagrams (DFD's), decision tables, narrative descriptions, etc.
- *Evaluation*, assessing the strengths and weaknesses of the existing system.

Systems Analysis

The analysis stage is aimed at identifying what the new system will do to overcome the weaknesses identified in the existing system.

This is based on an investigation into user requirements.

The outcome from this investigation/ analysis phase of the SDLC is a document known as the *requirements specification*.

Requirements Specification

The requirements specification describes:

- The organisation's **existing** system – how it works, and the problems associated with it.
- The **proposed** system: what users want from the new system, what it must do.

The Existing System

To discover how the organisation's existing system works, the analyst will use various fact-finding techniques to gather information.

The analyst must discover;

- what data flows through the system (the inputs and outputs)
- what processes the data undergoes, the sources and destinations of the data
- how the data is stored as it flows through the system.

Key Questions

What is being done?

- What separate processes (steps or activities) does the system involve?
- How is it being done?
- By whom?
- How frequently does it occur?
- Is it cyclical/seasonal?
- How great is the volume of transactions?

Key Questions

What triggers each process?

- What are the inputs to the process or activity?
- What data is needed?
- In what form is it received?
- Where/who does it come from?
- How is it stored?
- Is the data accurate?
- Is there any data missing?
- Is there unnecessary data?

Key Questions

Why is the process done?

- What is its purpose/objective?
- What are the results/outputs?
- In what form is the output?
- When is the output required?
- On demand/daily/weekly/monthly?
- Who uses the output from each process?
- What is it used for?

Key Questions

How long does each process take?

- What delays can occur?
- Who does this affect?

Key Questions

How well is each process or task being performed?

- Does a problem exist?
- If so, how serious is it?
- What is the underlying cause?
- What control/security methods are currently used?
- What areas need more control or security?

Key Questions

What are the costs of the system (often intangible)?

- Goods out of stock?
- Delays in production?
- Late delivery?
- Loss of orders?
- Customer dissatisfaction?

The Existing System

The workings of the existing system are usually modelled graphically using use case diagrams, with supporting narratives to explain step-by-step each process and its problems.

The Proposed System

The analyst will also use fact-finding techniques to discover how the existing system can be improved

- what users want from the new system,
- what it must do
- the requirements

A requirement is a new and different capability or feature that must be included in a system.

The Proposed System

These features may include:

- ways of capturing or processing information;
- methods of interacting with the system; a processing activity or result;
- a piece of information displayed or printed as a result of processing;
- a function that can be performed to safeguard data and information, etc.

When specifying requirements, keep the analysis at the general business level.

Do not prematurely express computer alternatives and technology details (*what* the system should do, not *how*).

Requirements

In documenting the system requirements an analyst may identify dozens of unique requirements for a new information system.

To simplify the presentation of requirements and make them more readable, understandable and traceable, requirements are often categorised as:

- *functional* requirements (what the system must do), and
- *non-functional* requirements (how well the system must do it).

Functional Requirements

A functional (or operations) requirement relates to what the system is expected to do.

It is an action of the system and is usually written using an action verb (a doing word), such as 'calculate' or 'capture'.

The following are all examples of functional requirements:

- Record a current account deposit
- Calculate the grade point average for a student
- Capture the account holder identification information.

Non-Functional Requirements

A non-functional requirement is a description of the features, characteristics and attributes of the system, as well as any constraints that may limit the system.

Examples of non-functional requirements include:

- Be user friendly
- Be able to accommodate users at different physical locations

Since these statements do not include a strong action verb, they are considered to be non-functional requirements.

Requirements Specification

During the analysis phase, many inexperienced analysts make the mistake of looking for technical solutions.

One of the most frequently cited errors in new information systems is illustrated in the statement,

'sure the system works, and it is technically impressive, but it just doesn't do what we needed it to do.'

The key in the analysis phase is to determine *what* not *how*.

Analysts are frequently so preoccupied with the technical solution that they inadequately define the users' requirements.

Requirements Specification

It is important to correctly identify requirements because when requirements are wrong, any one of the following could happen (**Whitten et al**):

- The system may cost more than projected.
- The system may be delivered later than promised.
- The system may not meet the users' expectations and that dissatisfaction may cause them not to use it.
- Once in production, the costs of maintaining and enhancing the system may be excessively high.
- The system may be unreliable and prone to errors and downtime.
- The reputation of the IT staff on the development team is tarnished because any failure, regardless of who is at fault, will be perceived as a mistake by the team.

Requirements Specification

Systems requirements are usually documented in a formal way to communicate the requirements to the key stakeholders of the system.

This document usually serves as the contract between the system owners and the development team on what is going to be provided in terms of a new system.

- Description of existing system:
- Description of the organisation
- Outline of existing system (narrative)
- Use case diagram/descriptions
- Proposed system requirements:
- Functional requirements
- Non-functional requirements:
- Schedule/update the project plan