

Structured Analysis

Chapter 12 Pressman

Chapter 10 Schach

Introduction

- Software development is recognized a time consuming and difficult endeavor.
- The most difficult part may be in understanding what the system is to do and not how it will be done.
- There are many different techniques to help the developer to understand the problem and to aid in the development of the software requirement document.

Methods

- **Various requirements analysis and specification methods are used:**
 - **Structured Analysis—Developed in the mid-to late-70's by Constantine, DeMarco, Yourdon, Gane, Sarsen, and others; still the most popular and widely used method**

Methods

- **Object Oriented Analysis**—A family of methods currently being developed by Jacobson, Coad, Yourdon, Rumbaugh, Schlaer, Mellor, Wirfs-Brock, Booch, and many others; this method has become the new standard

- **Structured System Analysis/ Structured Design (SSA/SD)**
 - **Most widely used and sophisticated of the function-oriented analysis and design methods**
 - **Created in the mid-1970's, it has evolved and been refined by many people since:**

- Newer and older versions are in use
- Extensions for special domains (real-time systems) are widely used
 - **Supported by most CASE tools**
 - **Documented in many popular books**
 - **Influenced other design methods**

– Notations

- **statement of purpose**
 - **context diagrams**
 - **Data flow diagrams, including context diagrams**
 - **Pseudocode (minispecs or P-specs)**
 - **Entity relationship diagrams**
 - **Data dictionary**
 - **Event lists**
 - **State transition diagrams**
-

- **Structured Analysis is the activity of deriving a structured model of the requirements for a system.**
- **The major goal of structured analysis and structured design is to improve software quality and reduce risk of failure.**
- **Behind this primary goal lies a secondary goal, the need to obtain a more clear and more complete specification, as well as documentation for the system.**

- **Therefore, SA tackles reliability, flexibility, maintainability and effectiveness goals by making visible and precise models that show analysis information.**

1) A Statement of Purpose

- This is going to be the 'what' and not the 'how' of the system. It is important that this is defined correctly.
- If it is not, the SASD technique will produce a brilliantly maintainable solution - but to the wrong problem.
- The statement should not be more than a single paragraph, otherwise it is an indication of too much detail.
- It is the role of the rest of the environmental and behavioral model to fill in this extra detail.

- **Context Diagram :**

- A context Diagram is the top-level of a Data Flow Diagram. It is designed to show the system in perspective to the rest of the world.
 - The context diagram shows the application as a single black box with interfaces to the external sources and interfaces to the users of the application's data.
 - As the context diagram is pretty much the first step, the developer should spend a fair amount of time in its creation to make sure no critical information is missed
-

- **Event List**

- The event list is a listing of all external events to the system or a listing of events to which the system must respond.
- So which comes first, the event list or the context diagram? It really doesn't matter, as long as both units are produced and that they are consistent with each other.
- The listing does not have to be in any certain order.