The choice of design depends on the nature of the system, its requirements, and the preferences of the development team. Often, a combination of these approaches is used to create robust, efficient, and maintainable software systems.

1. Functional Decomposition:

Description: Functional Decomposition is a top-down design approach that breaks down a system into smaller, more manageable functions or modules. Each module represents a specific task or function of the system. In a functional decomposition the intended function is decomposed into a number of subfunctions that each solve part of the problem. These subfunctions may be further decomposed into yet more primitive functions, and so on.

Purpose; This method helps in organizing and understanding the system’s functionality and can make it easier to manage and maintain.

2. Data Flow Design:

Description: Data Flow Design focuses on the flow of data within the system and how it is processed from input to output. It emphasizes the transformation of data from one form to another. It uses data flow diagrams.

Purpose; Helps understand how data enters he system, is processed and results in the desired output.

3. Design Based on Data Structures:

Description: This design approach revolves around organizing and structuring the data within the system. It focuses on defining the data structures that will be used to store and manage the data efficiently. It emphasizes the selection and implementation of appropriate data structures like arrays, lists, trees, graphs, etc.

Purpose; Helps in optimizing system performance by choosing the most suitable data structures for the task.