Real-Time and Distributed System Design

Many of the popular design "methodologies" were developed as design techniques for applications programs, operating systems and utility programs. These methods support concepts such as hierarchical decomposition, modularity, information hiding and data abstractions.

These design concepts are important, for real-time and distributed systems, but they do not explicitly incorporate methods to deal with performance characteristics or problems of network configuration and communication protocols.

Distributed System Design

According to Franta, a distributed system consists of a collection of nearly autonomous processors that communicate to achieve a coherent computing system. Each processor possesses a private memory, and processors communicate through an interconnection network.

Major issues to be addressed in designing a distributed system include specifying the topology of the communication network, establishing rules for accessing the shared communication channel allocating application processing functions to processing functions to processing nodes in the network and establishing rules for process communication and synchronization. The design of distributed systems is further complicated by the need to allocate network functionality between hardware and software components of the network.

Real time system design

By definition, real-time systems must provide specified amounts of computation with in fixed time intervals. Real-time systems typically sense and control external devices, respond to external events, and share processing time between multiple tasks processing demands are both cyclic and event-driven in nature.

A real-time network for process control may consist of several minicomputers and micro computers connected to one or more large processors. Each small processor may be connected to a cluster of real-time devices.

Finally concludes that the traditional considerations hierarchy, information hiding and modularity are important concepts for the design of real-time systems. However these concepts are typically applied to the individual components of real-time systems. Higher level issues of networking, performance and reliability must be analyzed and designed before the component nodes or processes are developed.