Lecture Objectives

• At the completion of this lecture you should:
  – be aware of the range of architectures available to the systems designer
  – Be aware of some of the issues surrounding the different architectures available

System Architecture

• System architecture is concerned with the technical and organisational configuration of the design of an application system
• An architecture defines the technologies to be used by the IS in terms of its data, process interface and network components.
• The architecture must be consistent with
  – budget constraints
  – organisational constraints
  – existing technology
  – requirements identified by analysis

System Architecture Components

• Processing
• Data
• Communication
• These components are interrelated and must be considered in conjunction with each other.

Architecture Components: Processing

• The evolution of computing environments has been largely determined by
  – technological advances in IT
  – the convergence of computing and communications technologies
  – rapidly changing organisational needs in dynamic business environments

Architecture Components: Processing

–Processing can be:
  > Centralised
    – e.g. Host and terminals
  > Distributed:
    – co-operative as in client/server
    – functionally distributed as in regional processing and centralised reporting
Architecture Components: Computers

- **Mainframe** – A large, centralised computer (e.g., an IBM AS400, UNISYS, FUJITSU) normally interfacing large numbers of terminals with large, centralised databases, running many concurrent processes.
- **Server** – A medium-sized computer, normally providing services, particularly shared data resources and printing, to a number of PCs.
- **Terminal** – Usually a PC, but may be a less intelligent device. Typically running a GUI interface and interacting with a single user at a time.

Processing Architecture: Types

- **Host-based** – All processing is performed on one computer system to which are attached unintelligent, “dumb”, character-based terminals. Processing is totally non-distributed.
- **Master-slave** – Slave computers are attached to a master computer and perform application processing functions but only as directed by their master. The minimal distribution is uni-directional from master to slave although slave has some limited local processing capability (e.g., field editing).
- **Shared-device** – PC workstations are attached to a system device, the server, that allows them to share common resources, files, disk and printer. All application processing is performed on the PC and only certain functions are distributed.

Processing Architecture: Types

- **Co-operative Computing**
  - **Client-server** – Workstations become clients as they request servers to process significant parts of the application running on the workstation in addition to sharing resources. The application is now distributed and the client and server cooperate to successfully execute the application. The client initiates interaction but the server enforces control over what services and data the client can request and arbitrates conflict between clients.
  - **Peer-to-peer** – Each participating system is equal and can both request services from and provide services to all other systems. There is no distinction between client and server and in intelligent networks, the server workload can be distributed and shared between participating systems.

Co-operative Computing

- **Advantages:**
  - Exploits the power and relative low cost of desktop computing technology.
  - Increased software portability.
  - Reduction in the application backlog.
  - Allows processing to take place at the source of the data thus reducing network traffic and response times and increasing the capacity and effective throughput of the network.
  - Facilitates the use of GUIs.
  - Encourages open systems as clients and servers are independent of hardware and software platforms.

- **Disadvantages:**
  - Bottlenecks at the server if significant portion of the application resides there.
  - Distributed applications are more complex thus increasing the cost of application development, maintenance and support in terms of the run-time environment and management of the distributed environment.

A communication system:

- Is a mechanism that allows distributed resources to exchange data and control information.
- Can be implemented as totally transparent or may be visible to the user.
- Requires physical connection between interacting nodes.
• Data architecture requires additional information about the essential data model:
  > volumes
  > locations
  > events
  > distribution
• This additional information not only impacts the data but also the processing and the communication choices.

Advantages of Distributed Databases

– Increased system reliability due to redundancy
– Local control of data … helps with improved data integrity and administration
– Lower communication costs by reducing system traffic
– Faster response since most applications use data at a local site

Disadvantages of Distributed Databases

– Software cost and complexity for a distributed environment
– Data integrity …control problems with multiple and dispersed sites
– Slow response if data and software not distributed properly

References