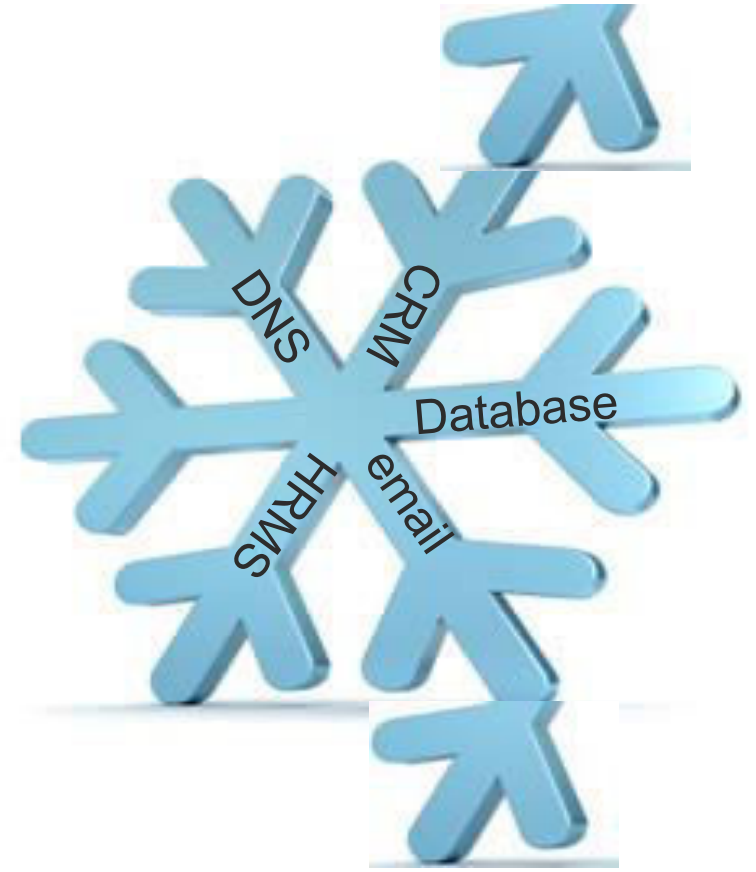


# **IT601 – System and Network Administration**

**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**





➤ **When managing many unique machines it becomes increasingly important to maintain an inventory of machines**

➤ **The inventory should document**

- 
- 
- 
- 
- 

➤ **Use automated means**

- 
- 

➤ **Inventory applications**

- **Always be on the lookout for opportunities to reduce the number of variations in platforms or technologies being supported.**
- **Discourage gratuitous variations by taking advantage of the fact that people lean toward defaults.**
- **Make right easy:**
  - 
  - 
  - 
  -

➤ **While it sounds efficient to customize each machine to the exact needs of the service it provides, the result tends to be an unmanageable mess.**

➤ **Classic example of a local optimization that results in a global deoptimization.**

■

■

# **IT601 – System and Network Administration**

**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**

➤ The next strategy is to buy computing resources in bulk and allocate fractions of it as needed

➤ One way to do this is through virtualization.

■

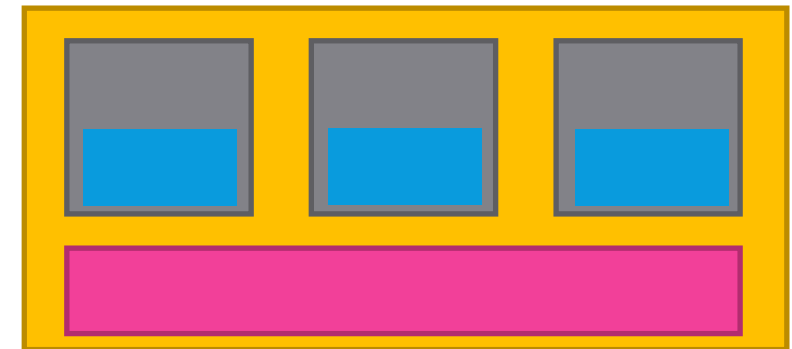
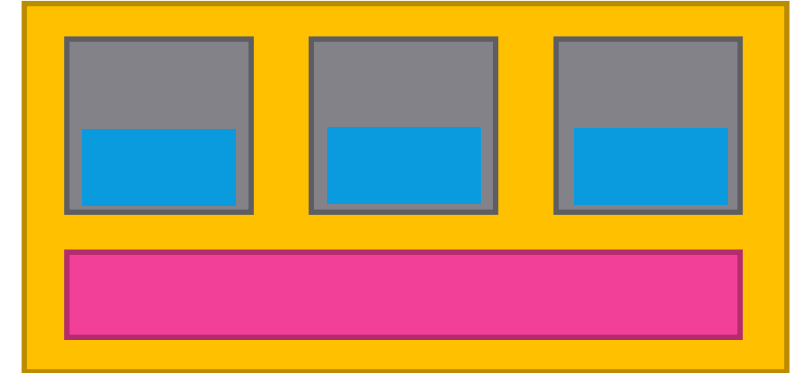
■

➤ Buying an individual machine has a large overhead.

■

■

■



➤ **VMs can also be resized.**



➤ **Virtualization improves computing efficiency.**



- Sharing a large physical machine's power among many smaller virtual machines helps reduce stranded capacity, without getting into the “all eggs in one basket” trap.



## ➤ Independence



## ➤ Resource isolation



- Processes running on one VM can't access the resources of another VM.



## ➤ Granular security

- 

- 

- 

## ➤ Reduced dependency hell

-



➤ There should be limits in place so that customers can't overload the system



they're being moved.

## Benefits of Live Migration

- 
- 
- 
- 
- 

## ➤ Shared Storage

- - a SAN for storage of the VM's disks.
-



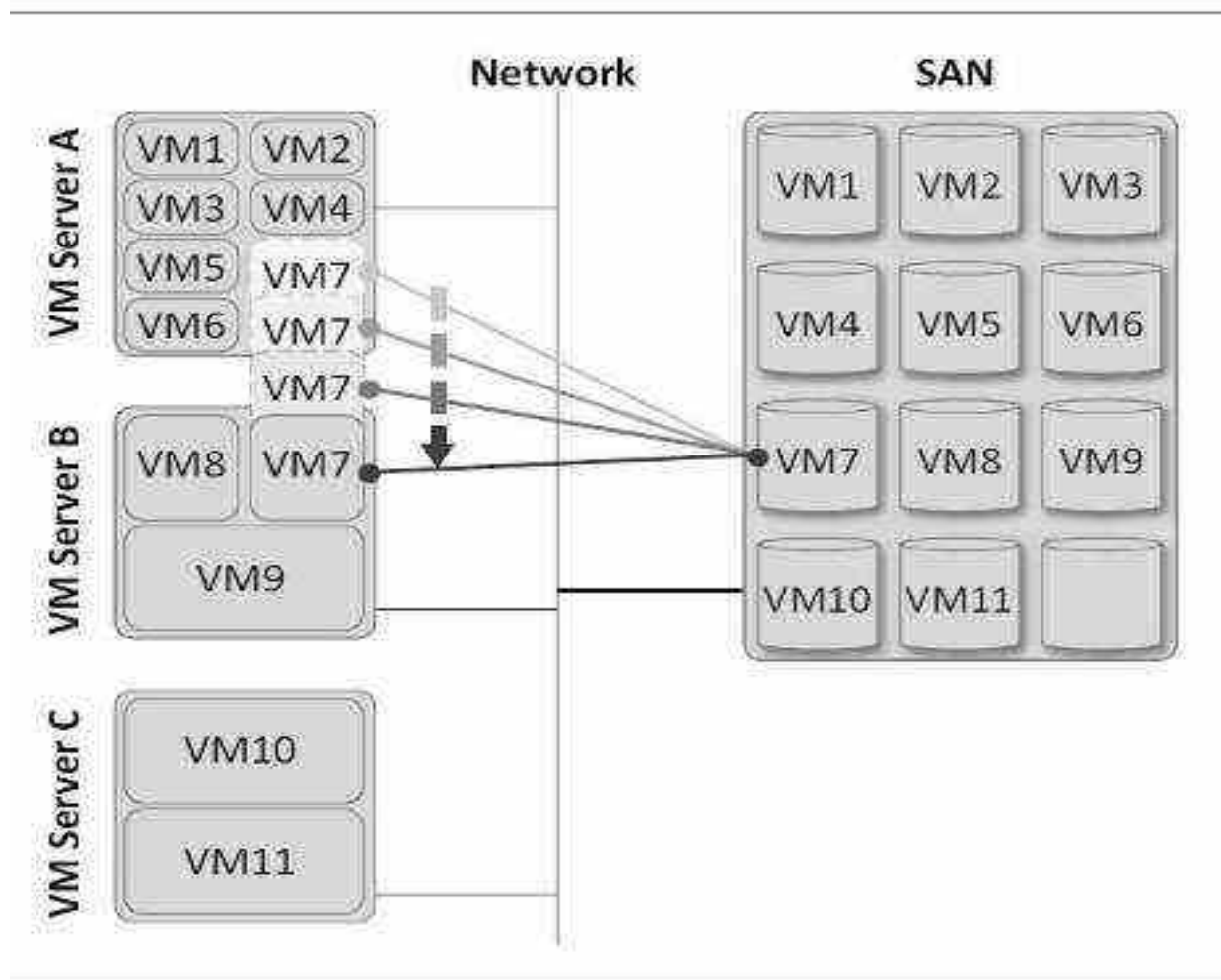


Figure 13.2: Migrating VM7 between servers

# **IT601 – System and Network Administration**

**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**

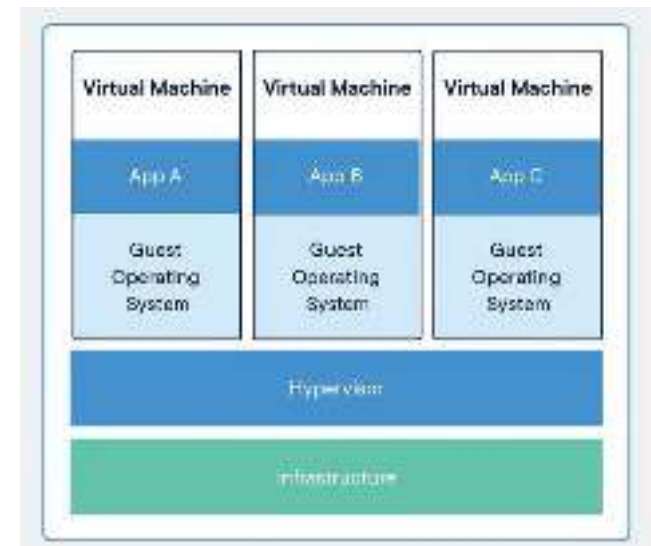
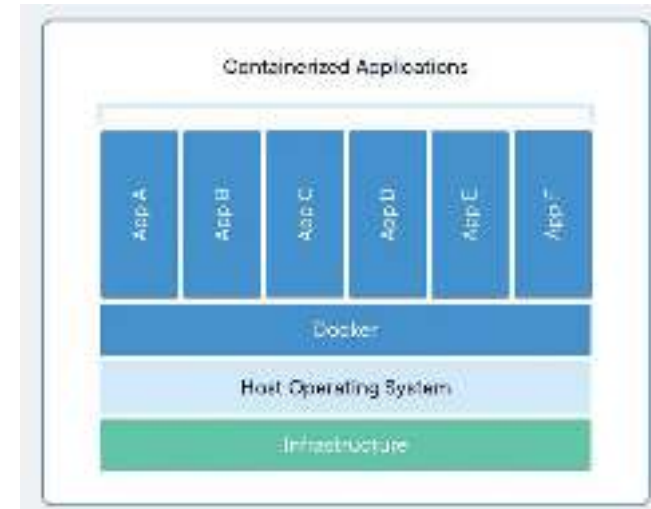








- 
- 

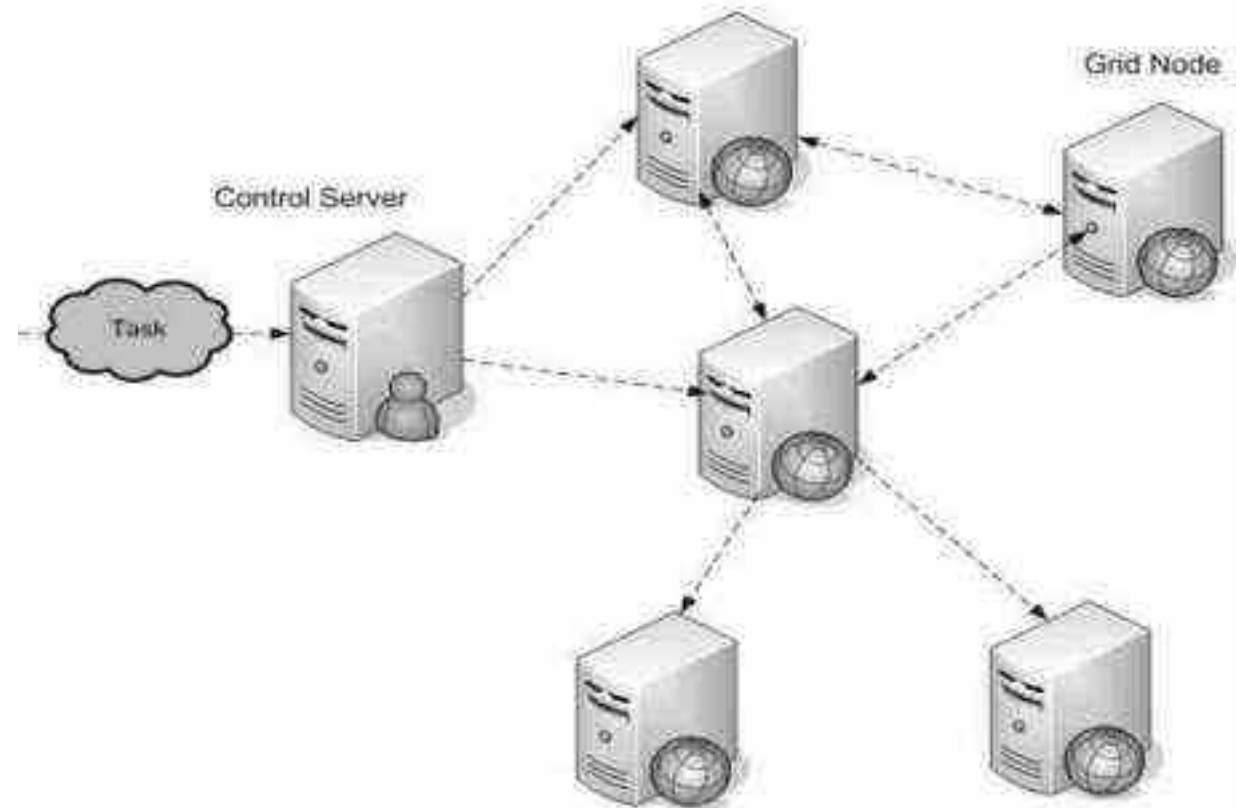


# **IT601 – System and Network Administration**

## **Week # 7**



**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**





# **IT601 – System and Network Administration**



**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**



- There are no cables to connect; the blade's connector conveys

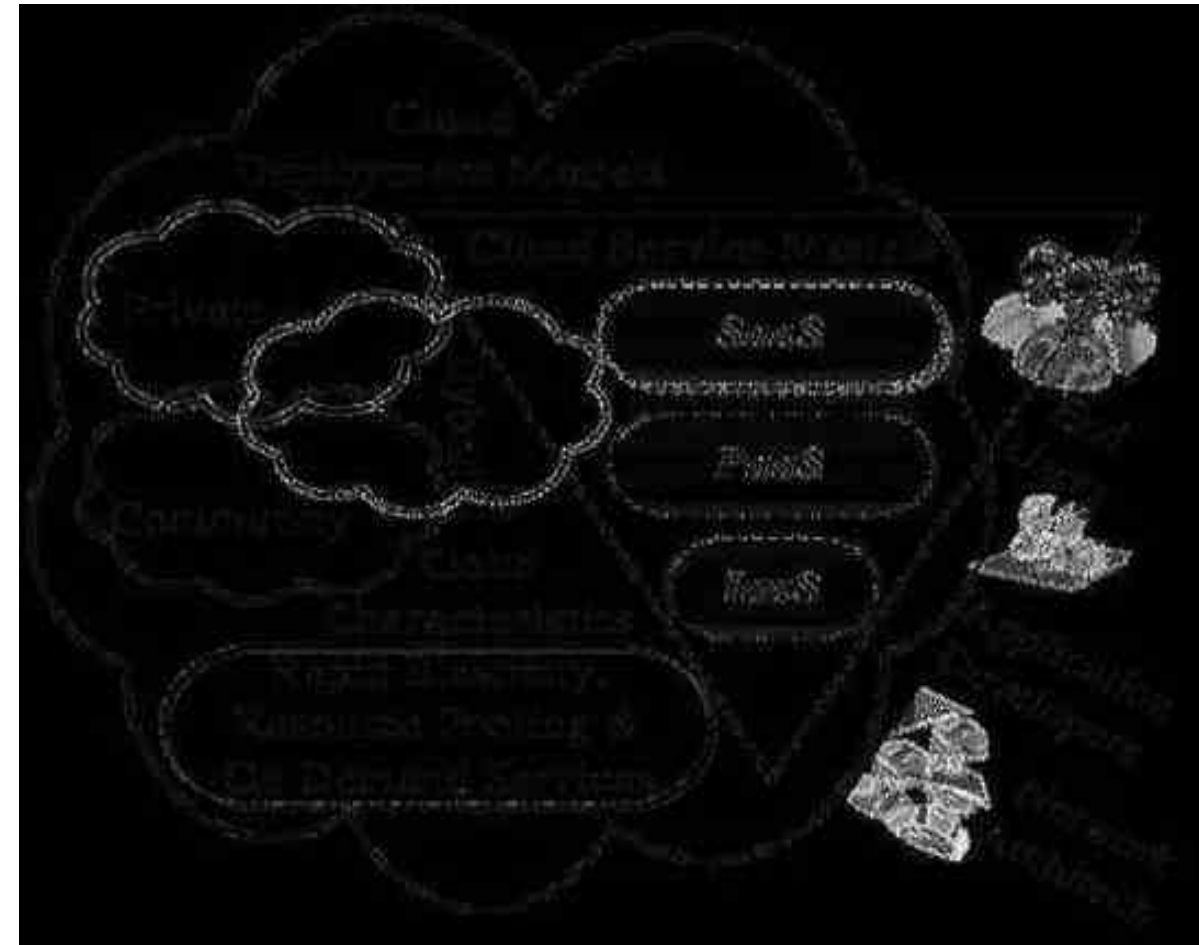


# **IT601 – System and Network Administration**



**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**

on someone else's system.







- demand nature is the most exciting because they won't have to deal with



# Cloud Computing's Cost Benefits



# **IT601 – System and Network Administration**

**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**



**COM**

**ERP**

**CBM**

**ACM**

**PMS**

**CMS**

**ECOM**

**HRM**

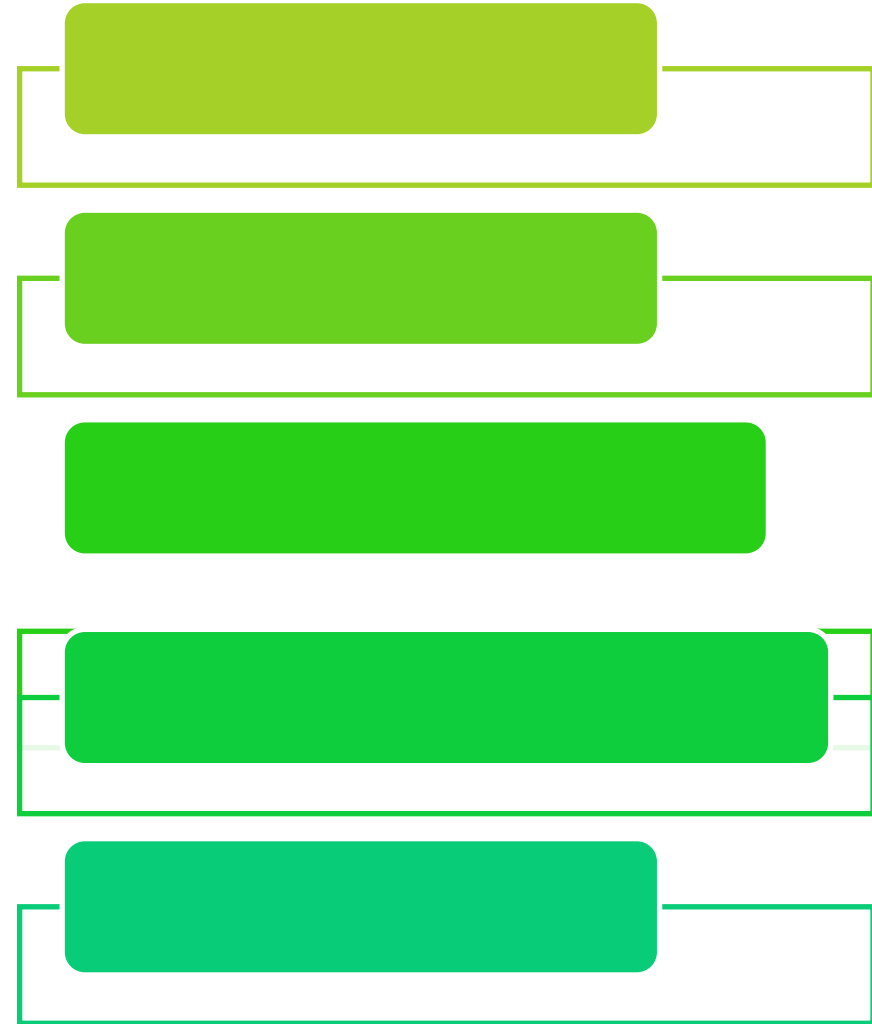
- 

- 

webbrowsing devices, they don't need traditional computers and can

- 

Internet connections, HTML5's ability to create interactive applications, and



# **IT601 – System and Network Administration**

**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**

# **IT601 – System and Network Administration**

**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**

➤ An **appliance** is a device designed specifically for a particular task.

➤ **Examples**

- Toasters make toast.
- Blenders blend.

➤ **General Purpose Vs. Specialized**

- We could build servers using general-purpose hardware with specific software packages
- However, there are significant benefits in using a device designed to do one task very well.

➤ **The computing area also has appliances:**

- 
- 
- 
- 
- 

➤ **The first appliance was the dedicated network router.**

- 

can easily add extra interfaces to our VAX and do the same thing?”

➤ **Common Opinion**

- 

-

➤ **A server appliance brings lot of things together in one box.**

- 
- 
- 
- 
- 

➤ **It's a lot of work! Appliances do all this for you right out of the box.**





that can't be found elsewhere.



that allow end users to “cd back in time,” thus eliminating many

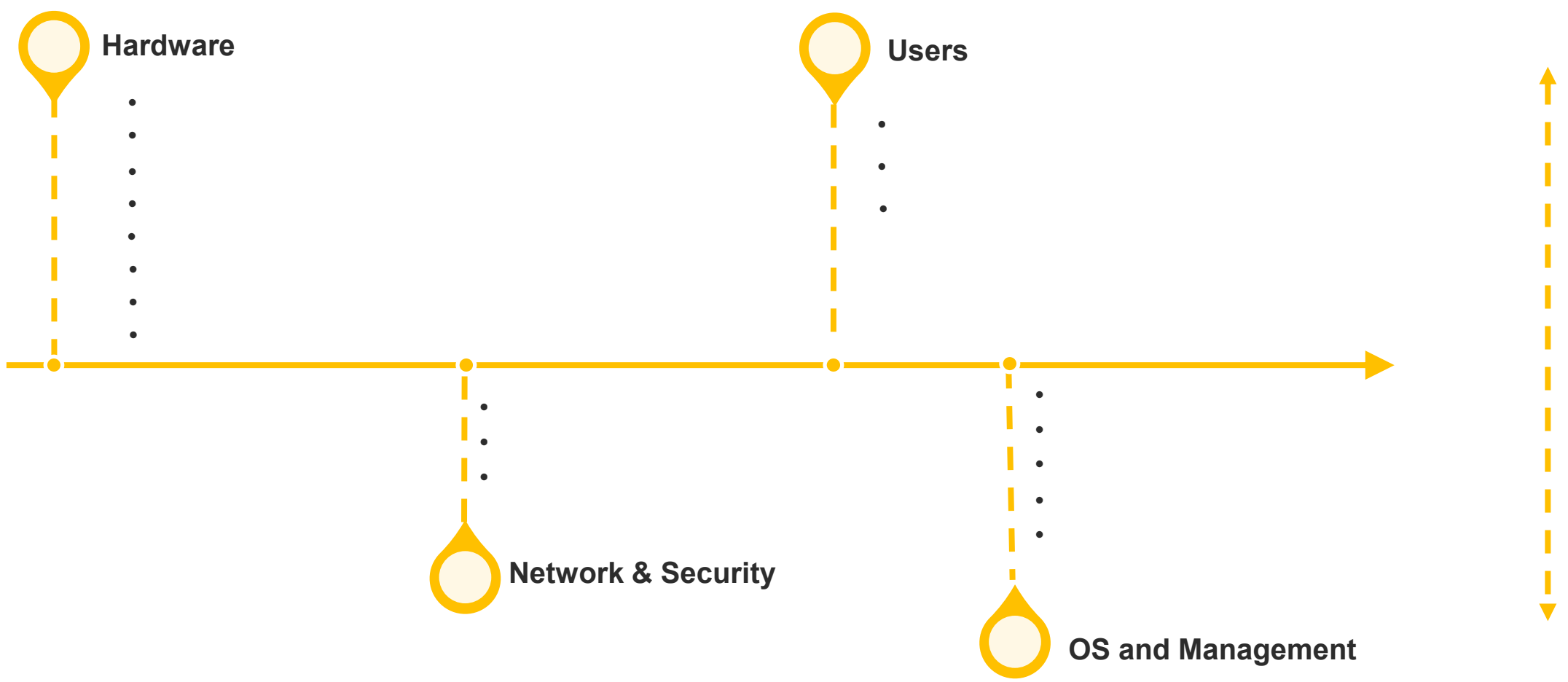


# **IT601 – System and Network Administration**

**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**

# **IT601 – System and Network Administration**

**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**



- **Levels of Redundancy**

- N+0, N+1, ...,
- 

- **Data Integrity**

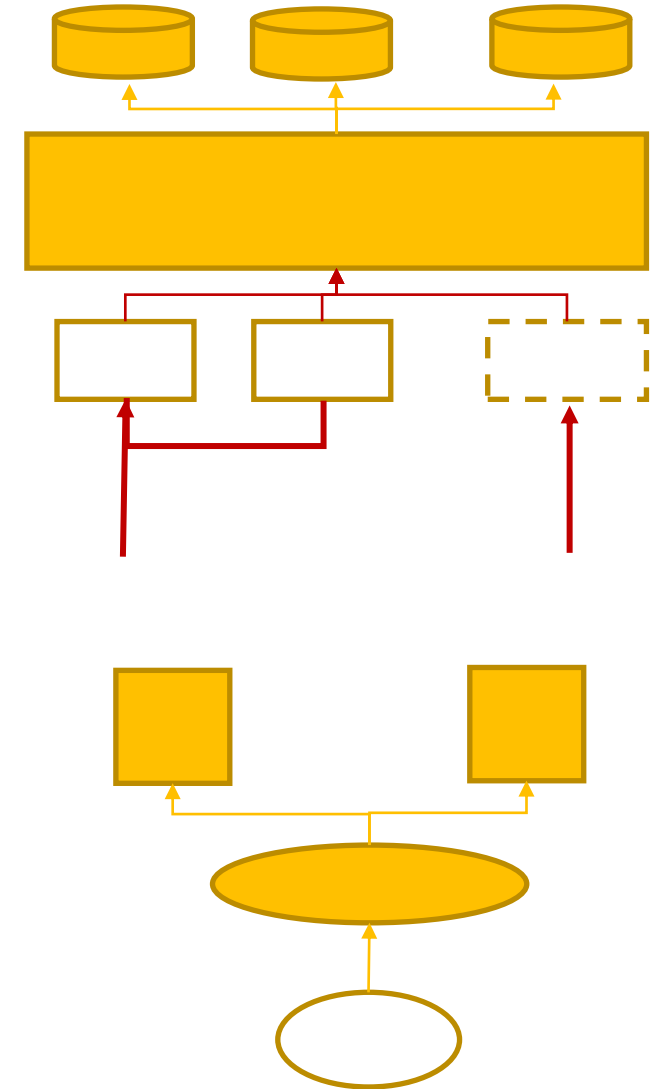


- **Hot-Swap Components**



- **Servers Should Be in Computer Rooms**

Google's GFS, Hadoop's HDFS, Cassandra





- access the machine's console and
- system's console when the machine is in a bad state or otherwise  
Accessing a basic BIOS configuration, CTRL-ALT-DEL keys



## ➤ Two Types of Remote management techniques



➤ **Integrated Out-of-Band Management**

- 

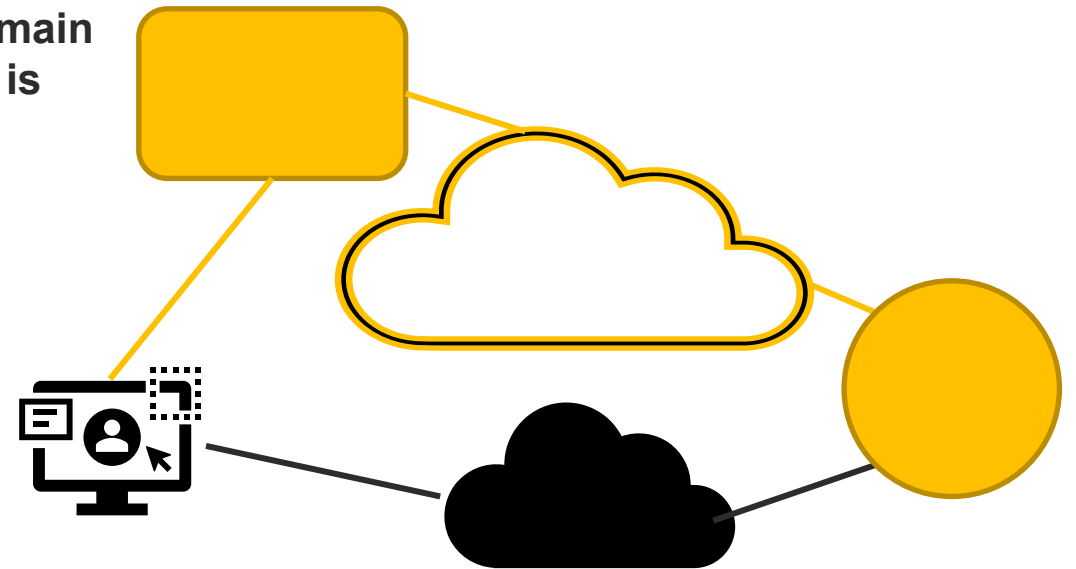
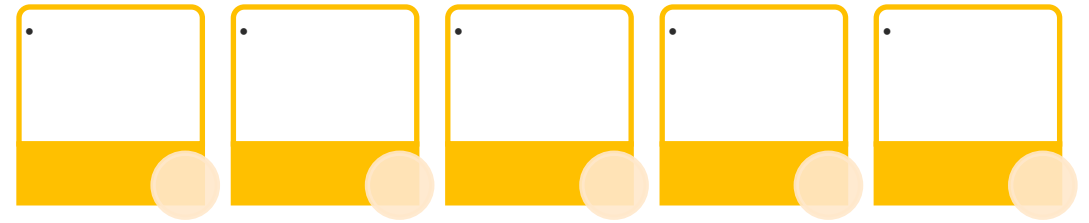
➤ **OOB interface requires an IP Address, and remote system console is accessed via browser or client.**

- 

➤ **The remote management systems must be isolated from the main network and must be accessible even when the main system is down or powered off**

➤ **Security Aspects**

- 
- 
- 
- 



➤

- **Remote Power Cycle**

- 

- **Remote Console with IP-KVM**

- 

- 

- 



- **Remote Console with Serial Consoles**

- 

- 

- 

- 



- It is common for servers to have a separate NIC that is connected to a dedicated administrative network. Separating administrative traffic from normal service traffic has a number of benefits.

- 

- 



- Characteristics of administrative networks

| <ul style="list-style-type: none"><li>•</li><li>•</li><li>•</li></ul> | <ul style="list-style-type: none"><li>•</li><li>•</li></ul> | <ul style="list-style-type: none"><li>•</li></ul> | <ul style="list-style-type: none"><li>•</li></ul> | <ul style="list-style-type: none"><li>•</li><li>•</li><li>•</li></ul> | <ul style="list-style-type: none"><li>•</li><li>•</li></ul> |
|-----------------------------------------------------------------------|-------------------------------------------------------------|---------------------------------------------------|---------------------------------------------------|-----------------------------------------------------------------------|-------------------------------------------------------------|

## ➤ Vendor SLA



### ▪ Response time



### ▪ Critical Servers



### ▪ Non-Critical Servers





## ➤ Tracking Service Contracts

- **Machines not covered in maintenance contract**

- 
- 

- **Improvement Strategies**



- 
- 



- 



- 
- 

## ➤ Cross-Shipping

- 
- -
-



# **IT601 – System and Network Administration**

**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**

# **IT601 – System and Network Administration**

## **Week # 8**

**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**

- **Services are the applications that customers need and use.**
- **Different components that make an application work includes**
  - 
  - 
  -
- **A service may have direct customers,**
  - 
  -
- **The fundamental purpose of System Administration is to provide services**
  - 
  - 
  -
- **A service begins life as a set of requirements.**
  - 
  - 
  - 
  -

➤ **The launch of a service is not the end of this process, but it is really just the beginning.**

■

■



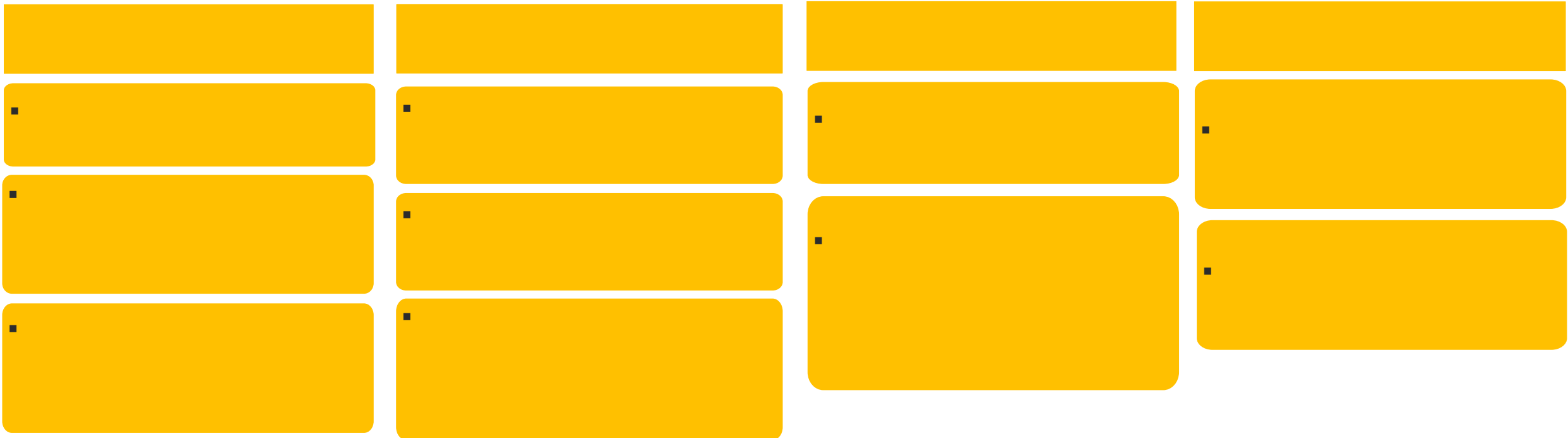
■



➤ **Generally, the computing environment and processes depends on the type of services being offered.**

- 
- 
- A large organization's fundamental services are run like an in

➤ **Services can be categorized into following categories**



➤ Together they form the company's critical path The company cannot function without them.

# **IT601 – System and Network Administration**

**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**

➤ The determination of requirements for a services follows two steps.

— —

1 - Starting with a Kick-Off Meeting

- 
- 
- ❖
- .
- ❖

▪ What to expect from Kick off Meeting?

- 
- 
- 
- 
- 
- 

▪ Benefits of Kick off Meeting?

- 
- 
-

## 2 - Gathering Written Requirements



### ➤ Goal of Requirements



### ➤ Serve as checklist for other steps

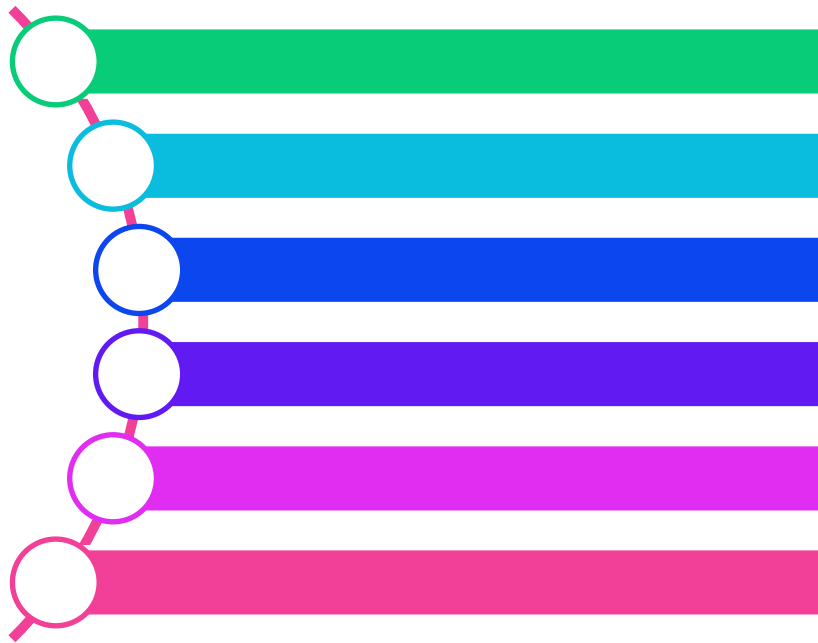


## 2 - Gathering Written Requirements

- Requirements are written down. They are not simply agreed to in a verbal discussion, tracked on a dry-erase board in your office, or kept in your head.

- 

### ➤ Benefits



### ➤ What to avoid



## ➤ Guidelines

- **Define terminology early in the document. Getting agreement to the ontology that will be used is very important.**



- **Annotate the importance of each requirement.**

☐ Marking each as “must have,” “desired,” or “optional/nice to have” communicates intent.



- The requirements can be classified into following types



**A - Describing Features**

**B - Questions to Ask**

**C - Service Level Agreements**

**D- Handling Difficult Requests**

# A - Describing Features

## ➤ Record the “what,” not the “how.”



### ❑ Example

It is better to record a requirement such as “the user should be able to send email” than “there should be a button on the left that the user clicks to send email.” The latter assumes a lot about the interface. Why should the designers

## ➤ Do not proscribe particular technology

# B - Questions to Ask

- **Ask how, why, where, and when, as if you are a journalist conducting an interview.**
- **Example Questions**
  - 
  - 
  - 
  -
- **Determine how large the customer base for this service will be and what sort of performance they will expect from it.**
- **Another way to record requirements is through use cases that tell a story. As long as the story can be completed by the final design, the use case is sufficiently covered.**
  - In Agile methodology, the format is “As a <type of user>, I want <some goal> so that <some reason>.”
  - For example, one might specify: “As a payroll clerk, I want to be able to enter future payroll adjustments so that I don’t need to do them all on the day they take effect.”  
Another example: “As a user, I want to indicate which folders should not be backed up so that my backup drive isn’t filled up with things I don’t need saved.”

- The requirements document should include the service level agreement (SLA), or at least some general specifications that can be used to build an SLA.
  -

| Enumerate Services | Categories Problems | Escalation Process | Penalties | Resources | Customer Needs   | Future Needs |
|--------------------|---------------------|--------------------|-----------|-----------|------------------|--------------|
|                    |                     |                    |           |           | customers' needs |              |

- 
- The SLA creation process is a forum for the SAs to understand the customers' expectations and to set them appropriately, so that the customers understand what is and isn't possible and why.
- meet customers' and their own expectations and to help keep them on track.

- **Requirements gathering is a collaborative process.**
- **The ultimate goal is to find the middle ground.**



## ➤ **Examples**

- 

- 

that's in the thousands of dollars.

- 

## ➤ **Your job is to educate as much as it is to record features.**

- Don't become upset when a customer asks for something technically

- 

- Rather than say "no," focus on the need, not the technology.

# **IT601 – System and Network Administration**



**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**

- A project plan generally has three elements.



- Flexible Vs Inflexible Features



- ☐ Management Buy-in is important





➤ **System Observability means the ability of a manageable system that we can see what it is doing.**

- 
- 
- A system may runs itself, or that a company's goal may be to build a system that it doesn't need maintenance.
  -

➤ **A system which is not observable, we cannot manage it, fix it, or scale it**

➤ **Basic Tools of Visibility**

- **System Logging**

- 
- 

- **System Monitoring**

- Ability to monitor not just the system 's up or down, but also internal resources, timing, bandwidth, disk space usage
- 
-

- **Services should be remotely managed.** If you have to be on the console of a machine to do rudimentary maintenance and configuration, each task will be more time consuming.

- 

- **Two basic reasons for remote servers**

- **Specialized Premises**

- **Geographic Redundancy**

- **Consider high latency over long-distance connections**

- 

—

➤ If a service is successful, more people will want to use it. Over time it will need to scale to handle more users, more transactions, or more capacity.

➤ There are two ways services generally scale.

- **Scale-up**

- 

- **Scale-out**

-

➤ **A system must have a reasonable way to upgrades.**

- 
- 

➤ **Two types of upgrades.**

- 
- 

➤ **Ideally, it should be possible to test a new software release outside of production and upgrade the service without interrupting it.**

➤ **new software releases**

- 
- New security holes are discovered constantly, and may not even be in the vendor's code but in a library
- over time. There's always room for improvement.

## ➤ Upgrades should be a Process

- 
- The process should integrate into the organization's software patching automation.
- 
- - 
  - 
  -

## ➤ Test environment

- 
- 

## ➤ Production assurance test (PAT) or user acceptance test (UAT)

- 
-

- **The better a service fits into the existing IT environment, the easier it is to adopt it in your environment. Integration issues are reduced, and less training or skill development is required.**

- 

- **For a small project, it is reasonable to make it an operational requirement that the service fit into the existing OS infrastructure, directory service, and so on. Larger projects require more flexibility.**

- 

- **Environmental Factors**

- 
- 
- 
- 
- 

- 
- 
- 
- 
-

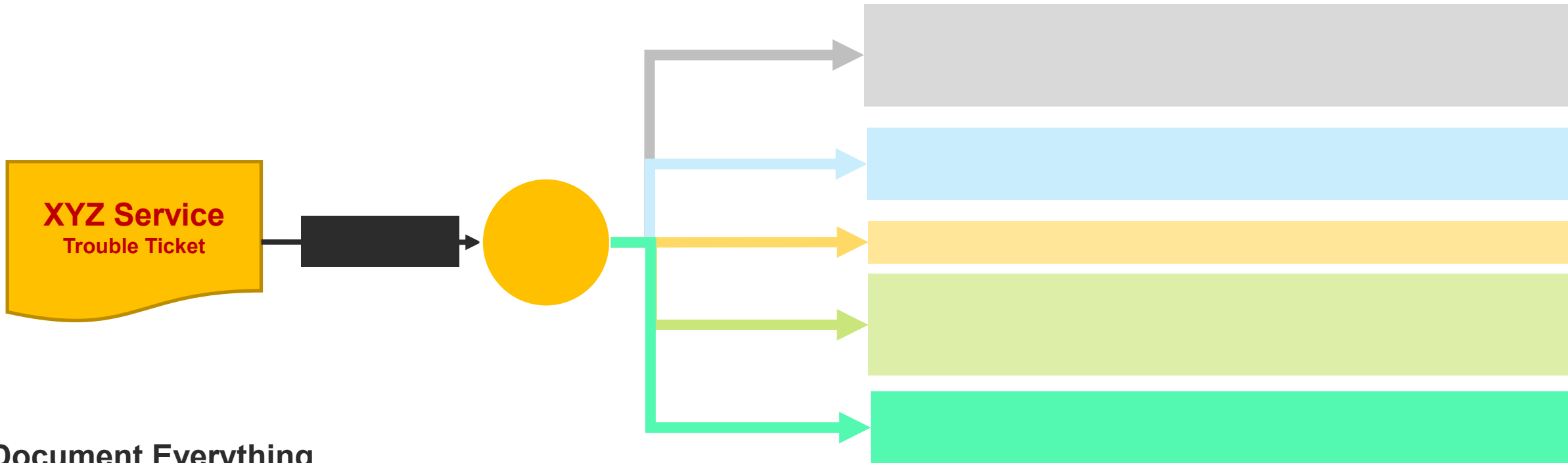
- A key component in the success of a new service is support processes.

- 

- Problems arise as service goes into production

- 

- 



- Document Everything

-

➤ **Most services will include standard service requests that people can raise.**

■

➤ **A checklist for building requirements might look like this:**

---

- What should people be able to request?

---

- How do people raise these requests?

---

- Which changes need to be made to the service request system to support these new service requests?

---

- Which approvals are required for each of these service requests?

---

- Who will process the requests?

---

- Which access and authorizations will that team need?

---

- Which runbooks does that team need to correctly service the requests?

---

- To what extent can, or should, these requests be self

---

➤ **The answers to these questions will provide some new requirements.**

■

## ➤ Failures cannot be eliminated.



➤ **A new service should be built around an architecture that uses open standards and open protocols.**

➤ **Open Vs Closed Service Architectures**

➤ **Open Architecture**

- 

- 

➤ **Closed Architecture**

- 

- 

- 

➤ **What is difference between a Protocol Versus a Product?**

## ➤ **Standard Vs Extended Products**

- That's not very customer oriented

## ➤ **Benefits Of deploying open products**

- 
- 
- 
- 
- 
- don't require gateways to the rest of the world.

## ➤ **A better way is to select protocols based on open standards, permitting users and operations to select their own software.**

- 

-

# **IT601 – System and Network Administration**

**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**

**IT601 – System and Network Administration**

**Week # 9 (Part 1)**

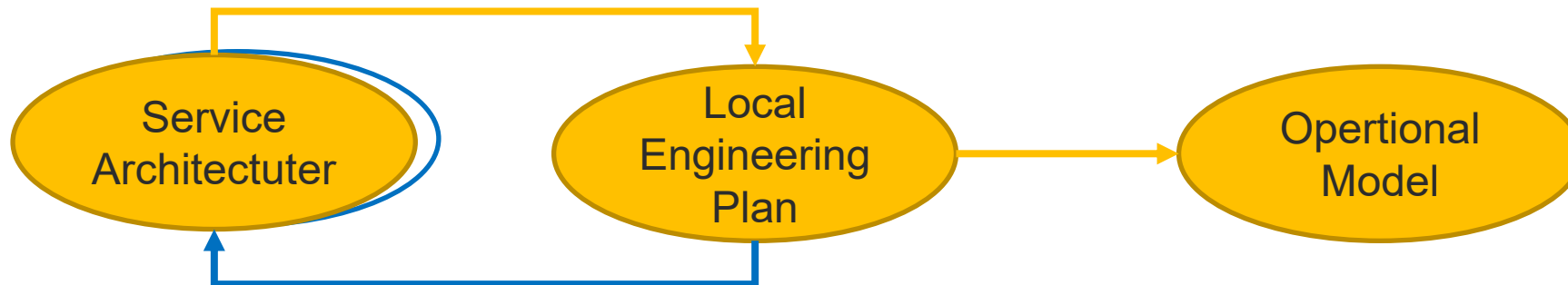
# Service Planning and Design

Arif Husen

**Department of Computer Science and Information Technology,  
Virtual University of Pakistan**

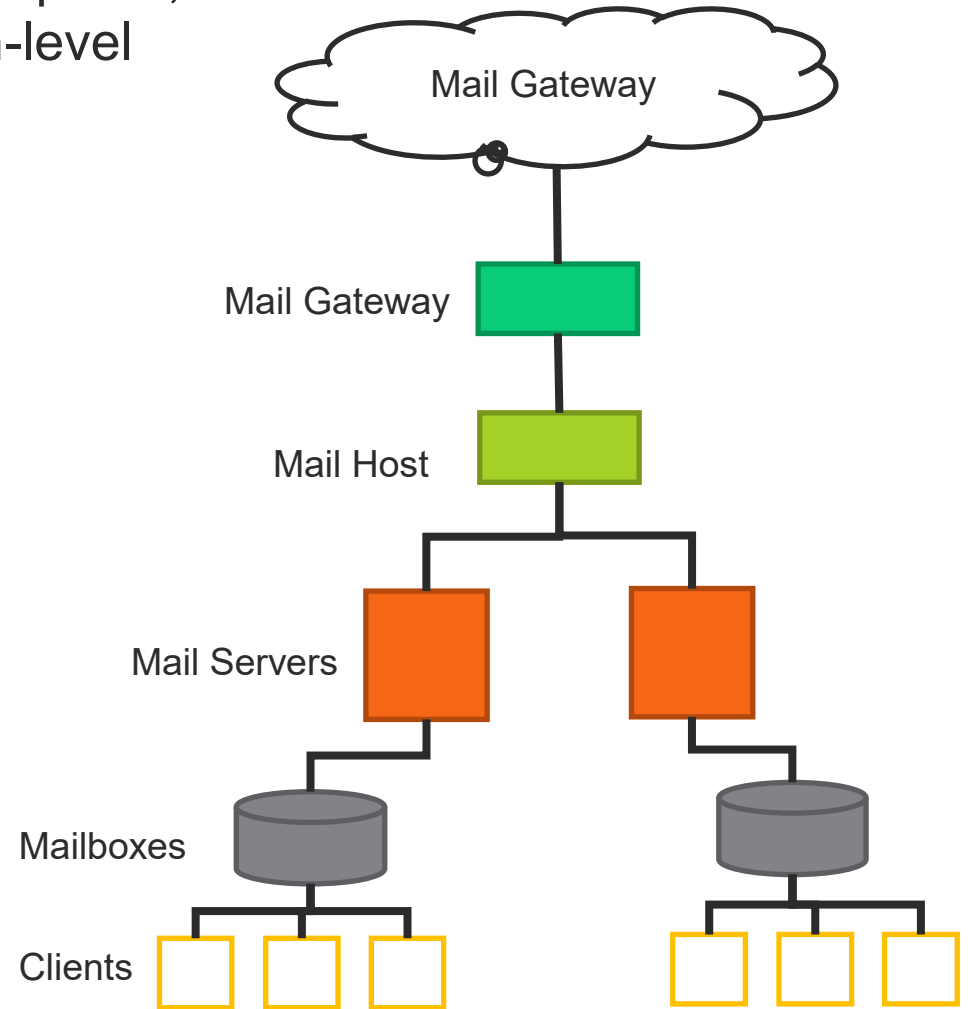
# Designing a Service

- **Service engineering involves designing how a product will be configured and used in your environment. With a simple service, it is a matter of installing the software.**
- **But most services are not simple. In such a case, we must design the service architecture and local engineering plans.**
- **The service design includes the following plans. All these must be defined and documented.**
  - The service architecture
  - The local engineering plans
  - The operational model or support model



# Service Architecture

- The service architecture is the high-level design. It describes how the parts will work together, which resources are required, and so on. Service Architecture is also known as High-level description
- It Includes:
  - Service Type
  - Sites and Clusters
  - Type of Machines, Technologies, Topologies and Storage, Networks
  - IP addressing Design



# Local Engineering Plan

Local Engineering Plan is also known as Low Level Design description

It Includes:

- Specific Component Levels Plan
- Product Ordering and Product Codes
- Cable, Mounting Brackets and Connectors their types
- Detailed IP addressing Hierarchy

# Operational Plan

Operational Plan is also known as Service Support Plan.

It Includes:

- Operational level agreement (OLA)
- The service level agreement (SLA) for the service
- The supports the various components of the service

# Guidelines for Service Designing



Simplicity



Vendor-Certified Designs



Dependency Engineering



Decoupling Hostname from Service Name



Support