CHAPTER 2

THEORY

2.1 Information

By definition from the Merriam- Webster dictionary, information can be defined as the knowledge obtained from investigation, study or instruction. Specifically, in computing, it can be also defined as a signal or character representing data (something as a message, experimental data, or a picture) which justifies change in construct that represents physical or mental experience or another construct. Thus information can be more simply defined as, stimuli that have meaning in some context for its receiver. When information is entered into and stored in a computer, it is generally referred to as data. After processing (such as formatting and printing), output data can again be perceived as information.

2.1.1 Qualities of Good Information

In accordance with the online ACCA Training Course Notes forum, there are several characteristics of a Good Quality Information. They are interrelated to each other. They are as follows:

Accurate

Information should be fair and free from bias. It should not have any arithmetical and grammatical errors. Information comes directly or in written form likely to be more reliable than it comes from indirectly (from hands to hands) or verbally which can be later retracted.

Complete

Accuracy of information is just not enough. It should also be complete which means facts and figures should not be missing or concealed. Telling the truth but not wholly is of no use.

Cost-beneficial

Information should be analyzed for its benefits against the cost of obtaining it. It business context, it is not worthwhile to spend money on information that even cannot recover its costs leading to loss each time that information is obtained. In other contexts, such as hospitals it would be useful to get information even it has no financial benefits due to the nature of the business and expectations of society from it.

User-targeted

Information should be communicated in the style, format, detail and complexity which address the needs of users of the information. Example senior managers need brief reports which enable them to understand the position and performance of the business at a glance, while operational managers need detailed information which enable them to make day to day decisions.

Relevant

Information should be communicated to the right person. It means person which has some control over decisions expected to come out from obtaining the information.

Authoritative

Information should come from reliable source. It depends on qualifications and experience and past performance of the person communicating the information.

Timely

Information should be communicated in time so that receiver of the information has enough time to decide appropriate actions based on the information received. Information which communicates details of the past events earlier in time is of less importance than recently issued information like newspapers. What is timely information depends on situation to situation. Selection of appropriate channel of communication is key skill to achieve.

Easy to Use

Information should be understandable to the users. Style, sentence structure and jargons should be used keeping the receiver in mind. If report is targeted to new-comer in the field, then it should explain technical jargons used in the report.

2.2 System

A system, in general, is a set of interrelated components, with clearly defined boundary, working together to achieve common set of objectives. In computing, it will be the interaction of both the hardware and software components of a computer that, receives inputs as data, and process them to give result as outputs.

2.2.1 Features of a System

A system may have following features:

1. **Adaptability:** some systems are adaptive to the exterior environment, while some systems are non-adaptive to the external environment.
2. **Limitation:** every system has pre-defined limits or boundaries within which it operates. This limits or boundaries can be defined by law or current state of technology.

2.2.3 Components of a System

Every system comprises of basic components which in a co-ordination formulate a system. These are as follows:

1. Input elements

The information entered into a system. For instance raw data input to the computer system.

2. Process

Any specific treatment defined in the system to be performed on the data entered into the system, for instance, computation, analysis, application of any model.

3. Output elements

The results given by the system after the process has been performed on the data being input to the system.

4. Control mechanism

Every system is expected to generate some sort of standardized output. Hence actual output needs to be compared with what it is supposed to generate. This comparison of actual with expected output is done with the help of control mechanism.

5. Feedback system

Once the control mechanism has been devised, it needs to a reporting mechanism, which should respond with a corrective action, if required.

6. Objectives

We just mentioned that a control mechanism should compare actual output with expected/Ideal output. But before this is being done, there needs to be a list of specific objectives which define expected output.

2.2.4 Types of Systems

1. Conceptual Systems

They are theoretical and explanatory in the nature. Provide the much needed clarification. Provide theoretical framework for which there may or may not be any real life counterpart. E.g. of such systems can be philosophy, theology etc.

2. Empirical Systems

Are very practical, specific and also very operational in the nature. It can be based on the conceptual system. Examination system and surgery act as very good examples of the empirical systems.

3. Open Systems

Involve continuous interaction with the environment. So exchanges the information, material, energy with the environment. It is open and also self-organizing in the nature. It is also adoptive or adaptive to the changing environment as it is flexible.

4. Closed System

Rejects any exchange with the environment. It is rigid in nature and is not at all amenable to the change. It is also self-contained. It is somewhat isolated in the nature. Is having a well-defined boundary. Is not at all adaptive in the nature

5. Natural Systems

Such Systems exist and also abound in the nature. They are also not at all the results of the human endeavors. Rivers, mountains, minerals etc. are the major examples of the natural Systems.

6. Artificial Systems

They are manufactured (man-made). Examples of such Systems are dams, canals, roads, machines, factories etc.

7. Probabilistic Systems

Based on the predictability of the behavior and the outcome of a test.

8. Deterministic Systems

In such Systems, the interaction of the elements is known. As the behavior of the elements is pre-determined, it becomes possible to work upon the reaction well in the advance.

2.3 Information Systems

According to the online webopedia, it described Information systems or IS as: short for ***I****nformation* ***S****ystems* or ***I****nformation* ***S****ervices*. For many companies, *IS* is the name of the department responsible for computers, networking and data management. Other companies refer to the department as *IT (Information Technology)* and *MIS (Management Information Services). Information systems* support different types of decisions at different levels of the organizational hierarchy. Major types of Information systems include structural databases and information management software that can include the following;

2.3.1 Components of Information Systems

Morley and Parker (2010) define information system as a discipline that is formed from elements of business and computer science and is developing to form a separate area of scientific study.

It has been stated that “healthcare information systems and healthcare processes are closely entwined with one another. Health care processes require the use of data and information and they also produce or create information” (Wager et al, 2009, p.65)

Three basic components of system are explained by Bagad (2010) as input, process/transformation and output. In information system inputs are data that are going to be transformed. The process component of an information system transforms input into an output. Output is considered to be the final product of a system. In case of an information system, an output would be obtaining necessary information in a desired format (Currie, 2009).

Explanations of all of the components of information system are offered by Stair et al (2008) in the following manner:

Table 2.1 The Components of Information Systems

Components of information system	**Definitions**
Data	Input the system takes to produce information
Hardware	A computer and its peripheral equipment: input, output and storage devices; hardware also includes data communication equipment

Software	Sets of instructions that tell the computer how to take data in, how to process it, how to display information, and how to store data and information
Telecommunications	Hardware and software that facilitates fast transmission and reception of text, pictures, sound, and animation in the form of electronic data
People	Information systems professionals and users who analyse organisational information needs, design and construct information systems, write computer programs, operate the hardware, and maintain software
Procedures	Rules for achieving optimal and secure operations in data processing; procedures include priorities in dispensing software applications and security measures

Source: Stair, R.M, Reynolds, G & Reynolds, G.W. (2008) "Fundamentals of Information Systems" fifth edition, Cengage Learning p.102

Moreover, Hoyt et al (2008) and Carter (2010), discuss the role of intranet, i.e. a local communications network within organization that is implemented on the basis of World

Wide Web software in terms of effectively sharing information and knowledge within organizations. As Hoyt et al (2008) argue the advantages of intranet include information sharing in a fast manner, facilitating communication between members of the organization in a cost-effective way.

2.4 Understanding Website

According to the online webopedia, web site is defined as: A site (location) on the World Wide Web. Each Web site contains a home page, which is the first document users see when they enter the site. The site might also contain additional documents and files. Each site is owned and managed by an individual, company or organization. Furthermore emphasized by techterm.com; a website, or Web site, is not the same thing as a Web page. Though the two terms are often used interchangeably, they should not be. So what's the difference? To put it simply, a Web site is a collection of Web pages. For example, Amazon.com is a Web site, but there are millions of Web pages that make up the site. Knowing the difference between these two terms can save you a lot of embarrassment.

2.4.1 Web Design

Taken from the techterms.com online forum, it states: Web design is the process of creating websites. It encompasses several different aspects, including webpage layout, content production, and graphic design. While the terms web design and web development are often used interchangeably, web design is technically a subset of the broader category of web development.

Websites are created using a markup language called HTML. Web designers build webpages using HTML tags that define the content and metadata of each page. The layout and appearance of the elements within a webpage are typically defined usingCSS, or cascading style sheets. Therefore, most websites include a combination of HTML and CSS that defines how each page will appear in a browser.

Some web designers prefer to hand code pages (typing HTML and CSS from scratch), while others use a "WYSIWYG" editor like Adobe Dreamweaver. This type of editor provides a visual interface for designing the webpage layout and the software automatically generates the corresponding HTML and CSS code. Another popular way to design websites is with a content management system like WordPress or Joomla. These services provide different website templates that can be used as a starting point for a new website. Webmasters can then add content and customize the layout using a web-based interface.

While HTML and CSS are used to design the look and feel of a website, images must be created separately. Therefore, graphic design may overlap with web design, since graphic designers often create images for use on the Web. Some graphics programs like Adobe Photoshop even include a "Save for Web…" option that provides an easy way to export images in a format optimized for web publishing.

2.5 Understanding Database

Efficient data management typically requires the use of a computer database. A **database** is a shared, integrated computer structure that stores a collection of:

- End-user data, that is, raw facts of interest to the end user.

- **Metadata**, or data about data, through which the end-user data are integrated and managed.

The metadata provide a description of the data characteristics and the set of relationships that links the data found within the database. For example, the metadata component stores information such as the name of each data element, the type of values (numeric, dates, or text) stored on each data element, whether or not the data element can be left empty, and so on. The metadata provide information that complements and expands the value and use of the data. In short, metadata present a more complete picture of the data in the database. Given the characteristics of metadata, you might hear a database described as a "collection of *self-describing* data."

A **database management system** (**DBMS**) is a collection of programs that manages the database structure and controls access to the data stored in the database. In a sense, a database resembles a very well-organized electronic filing cabinet in which powerful software, known as a *database management system*, helps manage the cabinet's contents.

2.6 Understanding Tourism Management System (Destination Management System)

Tourism Management System or mostly known as Destination Management System (DMS). As stated by Frew, A.J. and Horan, P (2007) *Destination Website Effectiveness*, in the New Mind online, it defined it as: "Destination Management System are systems that consolidate and distribute a comprehensive range of tourism products through a variety of channels and platforms, generally catering for a specific region, and supporting the activities of a destination management organization within that region. DMS attempt to utilize a customer centric approach in order to manage and market the

destination as a holistic entity, typically providing strong destination related information, real-time reservations, and destination management tools and paying particular attention to supporting small and independent tourism suppliers."

2.7 UML (Unified Modelling Language) Tools

By techopedia definition, Unified Modeling language (UML) is a standardized modeling language enabling developers to specify, visualize, construct and document artifacts of a software system. Thus, UML makes these artifacts scalable, secure and robust in execution. UML is an important aspect involved in object-oriented software development. It uses graphic notation to create visual models of software systems.

2.7.1 Structural Diagrams

The structural diagrams represent the static aspect of the system. These static aspects represent those parts of a diagram which forms the main structure and therefore stable.

1. **Use Case Diagram**

   A use case is the specification of a set of actions performed by a system, which yields an observable result that is, typically, of value for one or more actors or other stakeholders of the system. A use case is a usage of the system that provides an observable and (usually) meaningful result. The use-case documentation (diagrams and/or text) should delineate the series of steps that take place during the interaction and include different ways that this interaction could play out. Thus external entities are referred to as actors, and in this case, tourist is the actor.

2. **Activity Diagram**

In UML, an activity diagram is used to display the sequence of activities. Activity diagrams show the workflow from a start point to the finish point detailing the many decision paths that exist in the progression of events contained in the activity. They may be used to detail situations where parallel processing may occur in the execution of some activities. Activity diagrams are useful for business modelling where they are used for detailing the processes involved in business activities.

3. **Sequence Diagram**

A Sequence diagram is an interaction diagram that shows how processes operate with one another and what is their order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interaction arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with the use case realization in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

4. **Class Diagram**

Class diagrams are widely used to describe the types of objects in a system and their relationships. Class diagrams model class structure and contents using design elements such as classes, packages and objects. Class diagrams describe three different perspectives when designing a system,

conceptual, specification, and implementation. These perspectives become evident as the diagram is created and help solidify the design. Classes are composed of three things: a name, attributes, and operations. The association relationship is the most common relationship in a class diagram. The association shows the relationship between instances of classes. Class diagrams are used in nearly all Object Oriented software designs. Use them to describe the Classes of the system and their relationships to each other.

5. Object Diagram

An object diagram is used to show the existence of objects and their relationships in the logical design of a system. Stated another way, an object diagram represents a snapshot in time of an otherwise transitory stream of events over a certain configuration of objects. Object diagrams are thus prototypical—each one represents the structural relationships that may occur among a given set of class instances. In this sense, a single object diagram represents a view of the object structure of a system. During analysis, object diagrams are often used to indicate the semantics of primary and secondary scenarios that provide a trace of the system's behavior. During design, object diagrams are often used to illustrate the semantics of mechanisms in the logical design of a system. Regardless of the development phase, object diagrams present concrete examples that assist in the visualization of the associated class diagrams. Two essential elements of the object diagrams are object and their relationships.

6. Component Diagram

Component diagrams illustrate the pieces of software, embedded controllers, etc., that will make up a system. A component diagram has a higher level of abstraction than a Class Diagram - usually a component is implemented by one or more classes (or objects) at runtime. They are building blocks so a component can eventually encompass a large portion of a system. Components are similar in practice to package diagrams, as they define boundaries and are used to group elements into logical structures. The difference between package diagrams and component diagrams is that Component Diagrams offer a more semantically rich grouping mechanism. With component diagrams all of the model elements are private, whereas package diagrams only display public items.

7. Deployment Diagram

A deployment diagram is used to show the allocation of artifacts to nodes in the physical design of a system. A single deployment diagram represents a view into the artifact structure of a system. During development, we use deployment diagrams to indicate the physical collection of nodes that serve as the platform for execution of our system. A deployment diagram shows how the finished system will be deployed on one or more machines. A deployment diagram can include all sorts of features such as machines, processes, files and dependencies. Deployment diagrams indicate how the software is to be installed across systems.

2.7.2 UML Architecture

Any real world system is used by different users. The users can be developers, testers, business people, analysts and many more. So before designing a system the architecture is made with different perspectives in mind. The most important part is to visualize the system from different viewer.s perspective. The better we understand the better we make the system.

UML plays an important role in defining different perspectives of a system. These perspectives are: Design, Implementation, Process and Deployment.

And the center is the **Use Case** view which connects all these four. A **Use case** represents the functionality of the system. So the other perspectives are connected with use case.

- **Design** of a system consists of classes, interfaces and collaboration. UML provides class diagram, object diagram to support this.

- **Implementation** defines the components assembled together to make a complete physical system. UML component diagram is used to support implementation perspective.

- **Process** defines the flow of the system. So the same elements as used in *Design* are also used to support this perspective.

- **Deployment** represents the physical nodes of the system that forms the hardware. UML deployment diagram is used to support this perspective.

2.8 Software Support Used

The software used to support this system are:

2.8.1 PHP

Taken from PHP.net online, it states; “instead of lots of commands to output HTML (as seen in C or Perl), PHP pages contain HTML with embedded code that does "something". The PHP code is enclosed in special start and end processing instructions <?php and ?> that allow you to jump into and out of "PHP mode." What distinguishes PHP from something like client-side JavaScript is that the code is executed on the server, generating HTML which is then sent to the client. The client would receive the results of running that script, but would not know what the underlying code was. You can even configure your web server to process all your HTML files with PHP, and then there's really no way that users can tell what you have up your sleeve. PHP is mainly focused on server-side scripting, so you can do anything any other CGI program can do, such as collect form data, generate dynamic page content, or send and receive cookies. But PHP can do much more.”

There are three main areas where PHP scripts are used.

- Server-side scripting. This is the most traditional and main target field for PHP. You need three things to make this work. The PHP parser (CGI or server module), a web server and a web browser. You need to run the web server, with a connected PHP installation. You can access the PHP program output with a web browser, viewing the PHP page through the server. All these can run on your home machine if you are just

experimenting with PHP programming. See the installation instructions section for more information.

- Command line scripting. You can make a PHP script to run it without any server or browser. You only need the PHP parser to use it this way. This type of usage is ideal for scripts regularly executed using cron (on *nix or Linux) or Task Scheduler (on Windows). These scripts can also be used for simple text processing tasks. See the section about Command line usage of PHP for more information.
- Writing desktop applications. PHP is probably not the very best language to create a desktop application with a graphical user interface, but if you know PHP very well, and would like to use some advanced PHP features in your client-side applications you can also use PHP-GTK to write such programs. You also have the ability to write cross-platform applications this way. PHP-GTK is an extension to PHP, not available in the main distribution.

2.8.2 HTML

According to yourhtmlsource.com: “HTML is a computer language devised to allow website creation. These websites can then be viewed by anyone else connected to the Internet. It is relatively **easy to learn**, with the basics being accessible to most people in one sitting; and quite **powerful** in what it allows you to create. It is constantly undergoing revision and evolution to meet the demands

and requirements of the growing Internet audience under the direction of the **W3C**, the organization charged with designing and maintaining the language."

The definition of HTML is **HyperText Markup Language**.

- *HyperText* is the method by which you move around on the web — by clicking on special text called **hyperlinks** which bring you to the next page. The fact that it is *hyper* just means it is not linear — i.e. you can go to any place on the Internet whenever you want by clicking on links — there is no set order to do things in.
- *Markup* is what **HTML tags** do to the text inside them. They mark it as a certain type of text (*italicised* text, for example).
- HTML is a *Language*, as it has code-words and syntax like any other language.

HTML consists of a series of short codes typed into a text-file by the site author — these are the tags. The text is then saved as a html file, and viewed through a browser, like Internet Explorer or Netscape Navigator. This browser reads the file and translates the text into a visible form, hopefully rendering the page as the author had intended. Writing your own HTML entails using tags correctly to create your vision. You can use anything from a rudimentary text-editor to a powerful graphical editor to create HTML pages.

2.8.3 MySQL Database

According to its online website, dev.mysql.com, it defines it as:

- **MySQL is a database management system**

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

- **MySQL databases are relational**

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required or optional, and "pointers" between different tables. The database enforces these rules, so that with a well-designed database, your application never sees inconsistent, duplicate, orphan, out-of-date, or missing data.

The SQL part of "MySQL" stands for "Structured Query Language". SQL is the most common standardized language used to access databases. Depending on your programming environment, you might enter SQL directly (for example, to generate reports), embed SQL statements into code written in another language, or use a language-specific API that hides the SQL syntax.

- **MySQL software is Open Source.**

Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License), http://www.fsf.org/licenses/, to define what you may and may not do with the software in different situations.

2.8.4 Networking

Networking is when a set of devices (often referred to as nodes) connected by communication links. A node can be a computer, printer, or any other device capable of sending and/or receiving data generated by other nodes on the network.

The term physical topology refers to the way in which a network is laid out physically. Two or more devices connect to a link; two or more links form a topology. The topology of a network is the geometric representation of the relationship of all the links and linking devices (usually called nodes) to one another. There are four basic topologies possible.

- Mesh: A mesh offers several advantages over other network topologies. First, the use of dedicated links guarantees that each connection can carry its own data load, thus eliminating the traffic problems that can occur when links must be shared by multiple devices. Secondly, it is robust. If one link becomes unstable, it does not incapacitate the entire system. Third, there is advantage of privacy or security. When every message travels along the dedicated line, only the intended recipient sees it.

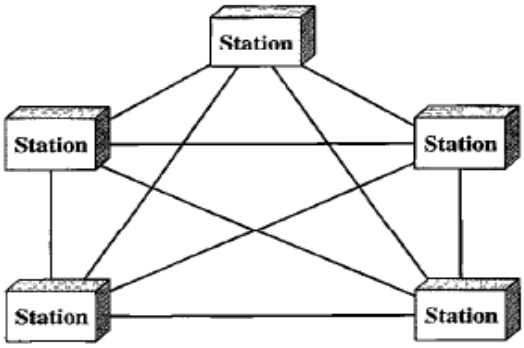


Figure 2.1 Mesh Topology

(Source: whatis.techtarget.com, Margaret Rouse-"Network Topology")

- Star Topology: In this topology, each device has a dedicated point-to-point link only to a central controller, usually called a hub. The device are not directly linked to one another. Unlike mesh topology, a star topology does not allow direct traffic between devices. The controller, which then relays the data to the other connected device.

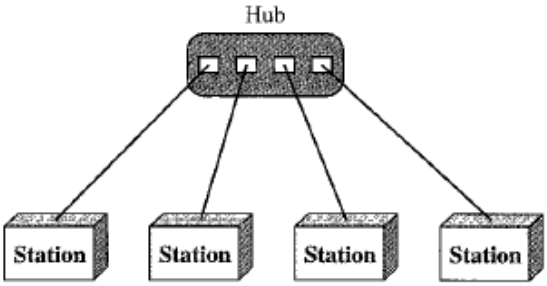


Figure 2.2 Star Topology

(Source: whatis.techtarget.com, Margaret Rouse-"Network Topology")

- Bus Topology: A bus topology is multipoint. One long cable acts as a backbone to link all the devices in a network. Nodes are connected to the bus cable by drop lines and taps. A drop line is a connection running between the device and the main cable. A tap is a connector that either

splices into the main cable or punctures the sheathing of a cable to create a contact with the metallic core. As a signal travels along the backbone, some of its energy is transformed into heat. Thus, it becomes weaker and weaker as it travels farther and farther. For this reason there is a limit on the number of taps a bus can support and on the distance between those taps.

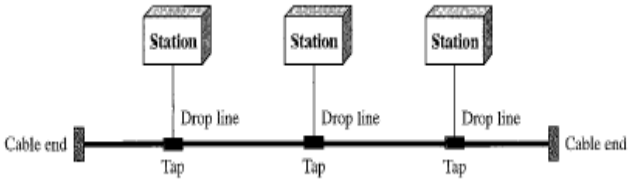


Figure 2.3 Bus Topology

(Source: whatis.techtarget.com, Margaret Rouse-"Network Topology")

- Ring Topology: Each device has a dedicated point-to-point connection with only two devices on either side of it. A signal is passed along the ring in one direction, from device to device, until it reaches its destination. Each device in the ring incorporates a repeater. When a device receives a signal intended for another device, its repeater regenerates the bits and passes them along.

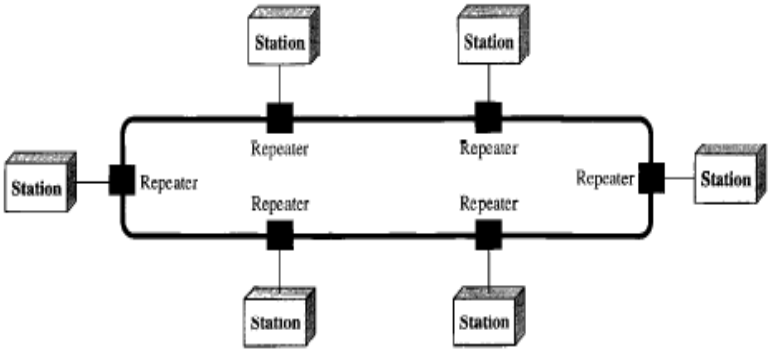


Figure 2.4 Ring Topology

(Source: whatis.techtarget.com, Margaret Rouse-"Network Topology")

- Hybrid Topology: A network can be hybrid. For example, we can have a main star topology with each branch connecting several stations in a bus topology as shown in the figure

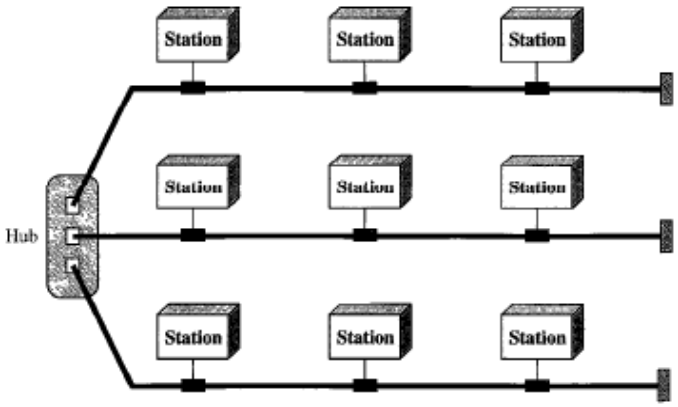


Figure 2.5 Hybrid Topology

(Source: whatis.techtarget.com, Margaret Rouse-"Network Topology")

Today when speaking of networks, one generally referred to two primary categories: local-area networks and wide-area networks. The category into which a network falls is determined by its size. A LAN normally covers an area less than 2 miles:

a WAN can be worldwide. Networks of a size in between are normally referred to as metropolitan area networks and span tens of miles.

- A **local area network (LAN)** is usually privately owned and links devices in a single office, building or campus. Depending on the needs of an organization and the type of technology used, LAN can be simple as tow PCs and a printer in someone's home office; or it can extend throughout a company and include audio and video peripherals. Currently LAN size is limited to a few kilometers.
- A **wide area network (WAN)** provides long-distance transmission of data, image, audio, and video information over large geographic areas that may comprise a country, a continent, or even the whole world. A WAN can be complex as the backbones that connect the Internet or as simple as a dial-up line that connects a home computer to the Internet.
- A **metropolitan area network (MAN)** is a network with a size between a LAN and a WAN. It normally covers the area inside a town or a city. It is designed for customers who need a high-speed connectivity, normally to the Internet, and have endpoints spread over a city or part of city.