# ISSUES IN NETWORK ANALYSIS AND DESIGN: A CRITICAL EXAMINATION

**Someswar Kesh, University of Central Missouri, kesh@ucmo.edu**
**Sam Ramanujan, University of Central Missouri, ramanujan@ucmo.edu**

## ABSTRACT

*Communication networks now play a critical role in our lives. The failure of these networks can cause serious financial and other losses in our lives. Therefore, it is imperative that communication networks are properly planned and designed. In this paper, we use the Network Development Life Cycle (NDLC) to facilitate a critical review of some of the important aspects of network analysis and logical design. For the analysis of network requirements we discuss how actors can be identified, the methodologies available for information elicitation, identifying the business as well as technical requirements and constraints, and flow analysis. For the logical design of networks we discuss two critical issues, viz., the hierarchical design of networks and IP addressing design issues. It is expected that the critical examination of these issues will spur further research in network analysis and design. This research is also important from a pragmatic as well as a pedagogical standpoint because practitioners and educators can use various concepts discussed in this paper.*

**Keywords:** Network Analysis, Network Design, Flow Analysis, Hierarchical Network Design

## INTRODUCTION

Communication networks are an integral part of our lives. At every facet of our lives we encounter communication networks; whether it is doctors reviewing patient data over a network or songs being downloaded to MP3 players, it is impossible to even think of life without communication networks. Because of this all pervasive role of communication networks, we want to ensure that these networks offer high performance, reliability, and security. At the same time, these networks are increasing greatly in size and complexity. Therefore the days of planning and designing a network on the back of an envelope is gone and the time has come for formal planning, analysis and design. Research on developing enterprise network planning and design has underscored the importance of this topic [2,8]. Many authors have proposed the use of the Network Development Life Cycle (NDLC) or a variant of it as a formal basis for network development [4]. This paper critically examines some of the issues involved at the analysis and design stages of the NDLC and discusses available solutions and tools. Researchers will find the issues useful because it can lead to further examination and research on those issues. Network planners and designers will find it useful because many of these concepts have direct applications. Furthermore, the paper will be useful for pedagogy in an advanced network analysis and design course.

## THE NETWORK DEVELOPMENT LIFE CYCLE

It is well known and recognized that in response to the need for an organized and systematic methodology for software development the Systems Development Life Cycle (SDLC) was born. NDLC has been developed from similar needs. Therefore it is not surprising that the steps between SDLC and NDLC are essentially the same. However, what is accomplished within those steps is completely different between SDLC and NDLC. Both SDLC as well as NDLC use decomposition and modularity as the fundamental principles. There may be some differences between how various authors have described the NDLC, though they are very similar. The NDLC steps can be categorized as: [7]

- Analysis of Network Requirements
- Logical Design of the Network
- Physical Design of the Network
- Implementation
- Maintenance (Optimization, Monitoring and Management).

Cisco uses a similar classification with the steps as; Plan, Design, Implement Operate, Optimize and Retire (PDIOO) [7]. The step that Cisco adds to the NDLC is retire. It may be worthwhile to add this step to the NDLC because retiring a network may involve significant cost and planning. In the next section, we discuss issues related to the analysis of network requirements and the logical design of the network. We have limited ourselves to the first two steps primarily because of space considerations.

## ANALYSIS OF NETWORK REQUIREMENTS

Some authors have classified the analysis of network requirements into two groups; requirements analysis and flow analysis [6]**.** Requirements analysis is the process of collecting user's requirements while flow analysis groups the data flows into various types known as flows. Major issues in the requirements analysis stage are:

1. Who are the actors at this stage (no pun intended!) of the life cycle?
2. Methodology for information elicitation
3. What information should be collected?

Major issues in flow analysis are identifying the various flows in a network and being able to classify the flows. These are discussed next.

### Requirements Analysis

Identification of Actors

The actors are people who will be affected by the network, either directly or indirectly. It seems that identification of the actors at this stage will be simple. After all, we need to collect the requirements of the users from the users themselves! This may be an easy task when the network is being designed and developed for a very small number of users, but for a large network either users or user groups will have to be identified. A useful tool for identifying users can be the value-chain analysis [5] in which first the value chain of the organization is identified. Once that is done, each group that caters to the value chain can be identified. The value-chain approach provides a systematic and structured approach for identifying the necessary actors, without leaving out anyone. This includes employees within the organization as well as vendors who are external to the organization because they are either users of the network or their network interfaces with the network of the organization. A vendor is a typical example for an external agent while the manufacturing department may be an example of an internal agent. All business components as well as people from various hierarchies should be included to ensure comprehensiveness. The director of a respective unit can be given the responsibility for collecting information from that department. Also higher level management should be included because they have an overview of the organization.

Methodology for Information Elicitation

In software development many methods have been suggested for collecting information. Methodologies like Joint Application Design (JAD) may be specially useful for information elicitation because it allows the actors to meet together, typically in a location away from their place of work, where the only purpose is to gather information [3]. At the end of the JAD session for the network analysis, the deliverables are the details of the current network and the details of the replacement network.

Information to be Collected

Primarily, two broad categories of information should be collected from the users. These are: [7]

- Business Requirements and Constraints
- Technical Requirements and Constraints

In the business requirements the analyst should identify the primary business goal of the networking project and any secondary goals that may exist. The goals should be evaluated against the organization's missions and goals. It may also be helpful to understand the goals in the context of the customer's competition. At this time, the analyst may also collect information regarding the customer's organizational structure.

Other important issues related to the customer's business requirements relate to the critical success factors of the project. The customer should clearly define and articulate the intended results of the project and what the customer will consider to be a success. Once the business requirements are known, the analyst should try to grasp the scope of the project. The customer should also explain the corporation's policies regarding selecting and dealing with vendors. The analyst may also specify a tentative schedule for the project. Some of the constraints can be evaluated more easily than others. For example, budgetary constraints may be specified clearly and easily, within certain parameters. However, political constraints may be far more difficult to understand, yet may have a significant impact on the success of the project. It may be up to the network analyst to sense the political environment of the organization.

In order to specify the technical requirements, two components will have to be identified. First the applications and data that will use the network will have to be specified and then the expected behavioral parameters of the applications will have to be specified. In some cases, the requirements for the network and the applications will be the same. For example, users may specify an application reliability of 99.5%, while that may be the same for the network as a whole. In some cases, these numbers may be different. Typically, to specify the network parameter, the most restrictive parameter will have to be chosen. For example, if two applications are running on a network, one with a reliability requirement of 99.0% and another with a reliability requirement of 99.5%, the network should have a reliability requirement of at least 99.5%. Also, some simplifying assumptions can be made at this stage [6]. In the only best-effort delivery assumption, no particular application will have to be singled out and will not need high performance service. It also allows the designer to base the design on a generic capacity plan instead of a service plan. If however, the focus is on the highest priority plans, then the design will focus only on certain applications. However, for certain parameters, the focus on highest priority applications will benefit all applications. Other technical performance information that should be collected at this stage are: [1,7].

Scalability: Scalability refers to the future growth of the network. Some businesses can expand rapid growth and may expect to double in one year. Other businesses may expect to grow only at a moderate pace. Some others may expect to remain steady over the next few years. With the increase in businesses, the network may need to grow. This growth may be only in terms of numbers or other considerations like expand businesses geographically.

Adaptability: Adaptability is somewhat related to scalability. While scalability issues are related to changes in size, adaptability refers to how quickly the network can change with changes in user needs. For example, if the users demand mobile computing, can a current system without mobile computing be changed quickly enough to adopt mobile computing?

Availability: Many times this is a crucial concern for users of the network. This refers to the percentage of time the network will be available to the users. Availability refers to the percentage up time of the system for a specified time period. Many users now expect availability of the system to be greater than 99%. Availability can be specified as a percentage of time. Availability can also be specified as a mean time between failure (MTBF) and the meant time to repair. In this case availability can be specified as (MTBF)/(MTBF + MTTR). If the MTBF is 3000 hours and the MTTR is 2 hours, then the availability is (3000/3002)x 100 = 99.94 percent. The MTBF and MTBR is an estimate from the statistical data for MTBF and MTBR.

Reliability: The reliability of a system is related to availability but is a broader concept. Reliability also includes the concept of performance. Therefore, the level of service provided should be high and consistent.

Capacity: The capacity of a system is the rate at which it can transfer data. Capacity is measured either by bandwidth, which is the theoretical capacity and throughput, which is the actual capacity. For example, in a 10Mbps network, the capacity is 10 Mbps. This is the rate that will be achieved under ideal conditions as well as when the network is not busy. If the network gets busy the actual data transfer rate will be far lower, approximately one-third the capacity. This is the throughput of the network.

Delay: Delay is the total time taken to transmit a unit of information from the source to the destination. The data unit can be a frame, cell or any other data packet. The delay is caused at each node of the network. From a user's standpoint, delay causes lack of productivity, loss of business and in general irritation. However, some applications require delay to be within a certain time period for functioning.

Foremost amongst these are the real time applications that have extremely strict timing requirements. Examples of these are systems that control robots. Non real-time applications have delay varying delay requirements based on the application.

Timeliness: Users want to perform their tasks within a certain time period. For example, users may want to download a certain file or complete a certain transaction within an expected time. This is the timeliness of the system. Timeliness is related to delay and latency, because as the delay or latency in the network goes up, timeliness suffers. However, delay and latency are not the only factors that affect timeliness. Performance of the application programs also will affect the timeliness. For example, if we have two database management systems running on the same network, and we try to execute the same queries on these two systems the time taken will be different.

Network Management and Security: The kind of network management that will be deployed is also extremely important. Different types of network management systems require varying amount of bandwidth. What network management protocol is being used, whether in-band or out-of-band signaling will be used are examples of information needed for network management. What kind of monitoring is done, whether traps or polling will be used are other examples of the kind of information needed. Users should specify both system and application security requirements. Security requirements can be classified into categories, high, medium, low etc.

Budget: It is important to know the budget (or have a good idea of the budget) from the outset. Many grand designs have failed because eventually it was realized that enough money was not available for the project.

### Flow Analysis

Once the basic requirements are known, they can further be defined based on flows. Flow analysis allows the network designer to segment and group network data flow based on common characteristics. It also allows the network designer to estimate the volume of traffic flow on each segment of the network. If we can group multiple applications with common characteristics between any two points A and point B, then we have defined a flow. The common characteristic can be, the source and sink (which in this case is A and B), similar protocols (like TCP/IP), other characteristics like "best-effort" services etc. or characteristics like delay and latency that was discussed in the previous section. The data sources are where data is generated and the data sink is where data is received. A host can be either a data source or a sink or both. Because dumb-terminals are rarely used these days, most hosts would typically be both a source and a sink. Data sources are represented by a dot, whereas data sinks are represented by a cross.

Three types of flows have been described. These are; individual flow, composite flow and backbone flow. The individual flow is the flow for a single session or application. Individual flows can be combined with other flows that have similar characteristics. If the individual flow does not have specified characteristics, then it will not have to be combined with other flows. When individual flows are combined, they make a composite flow. For composite flows, only the best effort delivery case will have to be considered, because that will satisfy the requirements of the other individual flows. The flow characteristics of individual flows will determine if the flow can be combined into a composite flow. Essentially then the designer has an idea of how much flow will be going from the source to the sink. This can then form the basis for determining the capacity of the network. When the network becomes hierarchical, there can be many such composite flows into one or more backbone networks. The combination of these composite flows makes up the backbone flow. Flow maps can also be developed using what is described as the well-known flow models. McCabe [6] considers the following well-known flow models:

- Peer-to-peer
- Client-server
- Cooperative computing
- Distributed Computing

In peer-to-peer computing, each computer is of equal status. Therefore each computer acts both as a source and a sink for every other computer in the network. The flow diagram in this case becomes two way arrows between each pair of computers. In the client server model the clients make requests and the servers respond. Normally, the client request is small compared to the response of the server. The server acts primarily as a data source and the client acts primarily as a sink. In the cooperative computing flow model, apart from the interaction between the servers and the clients, additional hierarchy may be imposed. In this case, the flow between the client and server remains as before. The multiple servers in this case, operate in a peer-to-peer fashion and the flow diagram between the servers and the manager is similar to that of the peer-to-peer setting.

The flow pattern is more complex in a distributed computing model. In distributed computing, the computing is shared. If the computing is evenly shared, the flow becomes similar to peer-to-peer computing. If an additional task manager is present for managing the distributed computing process, the task manager may have a greater degree of communication and server as the primary data source.

Once the flows are identified, flow boundaries will have to be established. The flow boundary separates either individual or composite flows from backbone flows. Therefore the flow boundary identifies the point of concentration of the individual or composite flows. Examples of typical flow boundaries are the separation between the LAN and WAN or between MAN and WAN etc.

Flow distributions show the where the backbone flows are distributed. In some cases, flows remain in a single area, for example, within a single LAN. In other cases, it will cross the flow boundaries. When many flows cross the flow boundary, a background flow is created. To determine how much traffic crosses the flow boundary and to determine the flow distribution, two approaches may be used. The first is to use a rule of thumb, like the 80-20 rule which states that 80% of the traffic will remain within the flow boundary and 20% of the traffic will cross the flow boundary. Given the emergence of the web, a high volume of traffic crosses the flow boundary. In that case, the second approach will be to rely on traffic statistics collected from previous studies.

We now need to know how to integrate the flows into composite and backbone flows. To achieve this, flowspecs should be used. Flowspecs are classified into three categories; unitary, two-part and multi-part flowspec. Unitary flowspecs are those that use only best effort flows, no other types of flows are specified. Two-part flowspecs use both best-effort and specified flows while multipart flowspecs provides details on individual flows. In an unitary flowspec the capacity calculation is based on sum of the capacity requirements of the individual flows. Because of the best effort service, only the total capacity (or bandwidth) is calculated. This is given by:

Bandwidth = ∑(individual flow bandwidth)

In case of the two-part flowspec, the bandwidth required for the best effort is calculated as before. However, for the specified environment, apart from the bandwidth the reliability and delay characteristics will have to be calculated as well. In a multipart flowspec, there are requirements for guaranteed environment as well. The parameters for guaranteed requirements; capacity, reliability and delay will have to listed separately. The network then will have to be designed in a manner consistent with the guaranteed requirements. The two-part and unitary flowspec capacities and other parameters can then be developed as before. Once a flow analysis has been performed, the network analyst/designer can now focus on the logical design of the network.

## LOGICAL DESIGN

### Hierarchical Design

Once the flow analysis has been performed, it is critical to group servers and clients into individual LANs, decide on the placement of switches and then routers and how they can be integrated with each other. At this stage, it is strongly recommended that network designers consider using the hierarchical model for the following reasons:

- Hierarchical models provide greater modularity than non-hierarchical models.
- Hierarchical models provide greater security and better performance by blocking broadcast traffic.

Hierarchical networks are faster because the roles of the routers are limited and therefore the routers are not burdened by unnecessary processing. A three layer hierarchical model is widely recommended [7]. The topmost layer of the hierarchical model consists of high speed routers and is known as the core layer. The goal of the routers at this stage is to optimize packet throughput. The second layer is known as the distribution layer and consists of routers with policies. The third layer is known as the access layer and consists of access routers, switches, hubs etc. This layer can provide access to the corporate internetwork. A challenge for network designers and planners is to maintain the strict hierarchy that has been used and implemented. For example, if the network needs to grow, additional switches and routers with policies may be added to the access layer that violates the principles of a strict hierarchical design. The principles of hierarchical network design should be avoided only when absolutely necessary. For example, in case of international communications a limited number of lines and routers may be available. In that case, a pragmatic design consideration will be to use the available lines and routers, even though that might violate some of the principles of hierarchical design.

Occasionally flat WAN topologies make sense. For example, in a very small network with only two subnets, the hierarchical design model will be overkill. In some cases, peer computers need to be connected to each other in a mesh network because of the speed and redundancy that mesh networks provide because in a mesh network each host is directly connected to all other hosts in the network. Three supercomputers connected to each other is an example where the use of a mesh network is appropriate. Because the mesh topology is expensive, some have adopted a less expensive solution, viz., a partial-mesh topology.

Irrespective of whether a flat network topology or a hierarchical network topology is adopted, redundancy issues are critical and network planners and designers should consider that issue as part of the design. This includes redundant connections, routers, switches, and hubs as well as servers, power supplies etc.

### IP Address Design Issues

Once a network design layout has been completed, the designers can focus on using appropriate model for IP addressing, since these days IP addressing is certainly going to the addressing scheme of choice. If the addressing scheme is not properly set up, duplicate addresses may result. Moreover, someone implementing a network may not know which addresses to select. DHCP servers may not be properly placed, resulting in slower distribution of IP addresses and consequently a slower network. A systematic approach for designing an IP addressing scheme is known as structured addressing [7,9]. Many important issues need to be resolved for structured addressing. This includes:

- The addressing scheme and hierarchical addressing
- Responsibility for assigning IP addresses

In a hierarchical addressing scheme, the left part of an IP address will typically refer to larger networks or nodes while the right part of an IP address will refer to the smaller networks or nodes contained therein. Subnetting can be considered as a hierarchical addressing scheme where the bits commonly reserved for host addresses are assigned to networks. Under such a scheme, it is recommended that the network addresses be assigned from the left, whereas the host addresses from the right [9]**.** This allows zeros in the middle, and provides tremendous flexibility to the network designers, should the designs change. However, at this stage, it should be decided whether the organization will follow a fixed length subnetting or Classless Inter Domain Routing (CIDR). CIDR is useful with large networks where routing tables becomes very large and it slows down the router. The network designers should also consider whether to use private IP addresses and Network Address Translation (NAT) that not only conserves IP addresses but also provides security.

The responsibility for assigning IP addresses should also be determined at this stage. A central authority should be used to distribute IP addresses. Local network administrators can then follow a supplied scheme to distribute IP addresses as they see fit.

## CONCLUSIONS

The emergence of complex networks has made proper analysis and design of networks extremely important. The Network Development Life Cycle (NDLC) is a structured approach that network analysts and designers can follow. In this paper, we have discussed various issues and tools available for the first two phases of the NDLC, viz. analysis of network requirements and logical design of the network. We have shown that various tools like JAD , that has been successfully used for software development can be used for network analysis and design as well.

We have also shown how tools like flow analysis and hierarchical network design can play an important part and how they can be used for network analysis and design. In the future, it is expected that more research will be done on the NDLC so that communication networks can keep up with the newer challenges of today.

## REFERENCES

1. Doherty, J., Anderson, N., and Maggiora, P.D., (2008), Cisco Networking Simplified, Indianpolis, IN, Cisco Press
2. Drakpoulus, E. (1999). Enterprise Network Planning and Design: Methodology and Application, Computer Communications, Volume 22, Number 4, 340-352.
3. George, J.F. et.al., (2007), Object-Oriented Systems Analysis and Design, Upper Saddle River, New Jersey, Pearson Prentice Hall.
4. Goldamn, J.A., and Rawles, P.T., (2004), Applied Data Communications, Hoboken, New Jersey: John Wiley and Sons, Inc.
5. Laudon, K.C., and Laudon, J.P., (2005), Essentials of Management Information Systems, Upper Saddle River, NJ: Pearson Prentice Hall
6. McCabe, J.D.,(2007), Network Analysis, Architecture and Design, Burlington, MA: Morgan Kaufmann Publishers
7. Oppenheimer, P., (2004) Top-Down Network Design, Indianapolis, IN: Cisco Press
8. Papachristodoulou, A., Antonios, Li, L., and Doyle, J.C., (2004). Methodological Frameworks for Large-scale Network Analysis and Design. ACM SIGCOMM Computer Communications Review 34(3), 7-20.
9. Siyan, K., (2000), Windows 2000 TCP/IP, Indianapolis, Indiana, New Riders.