2 Background: Service Oriented Network Architectures

Most of the issues in the Internet arise because of inflexibility and rigidness attributes of the network architecture, which is built upon a protocol stack. The problem that is faced by the Internet is that it is hard to integrate new functionalities in it and to remove existing functionalities from it. The reason is that protocols and layers are tightly coupled between themselves as well as within each other. In addition, they are also coupled with the applications. This problem is not limited to specific protocols and mechanisms. However, it is an architectural issue.

Similar problems were seen in software engineering which has evolved to manage complexities (e.g. maintenance, integration of new functionalities, time and task management) of development process, which has direct effects in terms such as of cost, quality and development time. That is why, for designing a new software architecture for the Internet core, the principles and techniques from software engineering can be applied.

Software engineering has evolved from structural programming to service oriented programming. The design of a future network architecture can benefit from software engineering techniques to make network architecture
more flexible and easy to maintain rather than having an ossified architecture (e.g. Internet).

The Service Oriented Network Architecture (SONATE) [82][97], a clean slate network architecture, applies the principles of Service Oriented Architecture (SOA) to communication systems.

Services are the essential elements of a SOA. The protocol stack of the Internet has also been developed considering services.

2.1 Layering in Protocol Stacks

To reduce complexity and promote modularity, the protocol stack has been organized as layers. The International Organization for Standardization (ISO) had specified seven layers for the Open System Interconnection (OSI) model namely physical, data link, network, transport, session, presentation and application [144]. The TCP/IP model has 5 layers as it integrates all of the functionalities of the session, presentation and application layers of the OSI model into one layer called application layer [113]. Each layer provides services to its upper layer and consumes services from its lower layer.

An example scenario is shown in Figure 2.1 where a user started browsing Internet using the WLAN connection of his laptop. In this case, the browser sends the request to the server using Hyper Text Transfer Protocol (HTTP). The packet is then sent to the TCP protocol of the transport layer which encapsulates the HTTP packet and wraps its with its own header and trailer around it. The transport layer then sends the packet to the IP protocol of the network layer which does the same, i.e., encapsulates the TCP packet coming from transport layer and adds its own header and trailer. The packet is then sent to its lower layer so called link layer. The IEEE 802.11 protocol