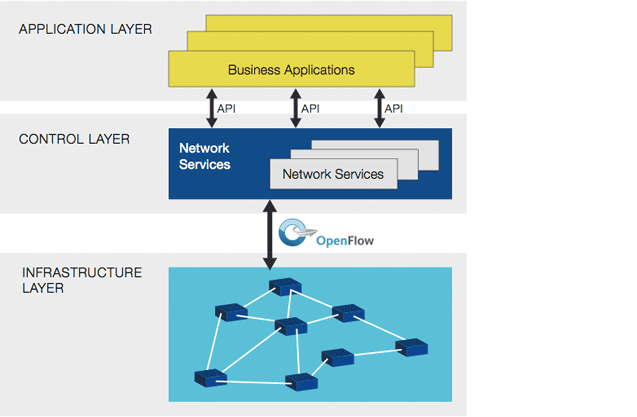
**SOFTWARE DEFINED NETWORKING**

Software-Defined Networking (SDN) is an emerging field in networks that is dynamic, manageable, cost-effective, and adaptable, making it ideal for the high-bandwidth, dynamic nature of today's applications. Unlike the usual networks, SDN favors modularity which increases pliability of controlling the entire network using a controller. SDN reduces the intricacy of the network devices, that the acknowledgement of the various protocol standards becomes redundant. SDN employs the process of decoupling the control plane and the data plane. The separation of the functionalities of the two planes makes the network more transparent. This decoupling mechanism paves way to high speed transmissions in the network and thereby improves the performance of the network .SDN introduces the programmability feature to the control plane in order to manage the network devices such as routers and switches in the network for efficient data delivery.



The SDN is:

* **Directly programmable**: Network control is directly programmable because it is decoupled from forwarding functions.
* **Agile**: Abstracting control from forwarding lets administrators dynamically adjust network-wide traffic flow to meet changing needs.
* **Centrally managed**: Network intelligence is (logically) centralized in software-based SDN controllers that maintain a global view of the network, which appears to applications and policy engines as a single, logical switch.
* **Programmatically configured**: SDN lets network managers configure, manage, secure, and optimize network resources very quickly via dynamic, automated SDN programs, which they can write themselves because the programs do not depend on proprietary software.
* **Open standards-based and vendor-neutral**: When implemented through open standards, SDN simplifies network design and operation because instructions are provided by SDN controllers instead of multiple, vendor-specific devices and protocols.

**Applications:**

* One application of SDN is the [infrastructure as a service](https://en.wikipedia.org/wiki/Infrastructure_as_a_service) (IaaS).
* SDN is an important element in the functioning of so-called The New IP, which is a highly virtualized, software-driven Internet Protocol network using software-defined networking, network functions virtualization, open platforms, and an open-source development model.
* Other uses of SDN in enterprise or carrier managed network services (MNS) address the traditional and geo-distributed campus network. These environments were always challenged by the complexities of additions, changes, mergers, acquisitions, and movement of users.
* Based on SDN principles, it is expected that these identity and policy management challenges could be addressed using global definitions decoupled from the physical interfaces of the network infrastructure. On the other hand, existing infrastructure of potentially thousands of switches and routers can remain intact.

**SIMULATION TOOLS – MININET**

Mininet is a Network Emulator. It runs a collection of end-hosts, switches, routers, and links on a single Linux kernel. It uses lightweight virtualization to make a single system look like a complete network, running the same kernel, system, and user code.

It is economical, customizable network tool which is considered to be one of the most used tools in OpenFlow. It is also helpful in connecting with real time networks based on network stack to process packets. Based on the virtual host emulation, the controller application program can run in the same or different machine in an OpenFlow network with Mininet.

Mininet is useful for interactive development, testing, and demos, especially those using OpenFlow and SDN. OpenFlow-based network controllers prototyped in Mininet can usually be transferred to hardware with minimal changes for full line-rate execution.

Mininet creates virtual networks using process-based virtualization and network namespaces - features that are available in recent Linux kernels. In Mininet, hosts are emulated as bash  processes running in a network namespace, so any code that would normally run on a Linux server (like a web server or client program) should run just fine within a Mininet "Host". The Mininet "Host" will have its own private network interface and can only see its own processes. Switches in Mininet are software-based switches like Open vSwitch or the OpenFlow reference switch. Links are virtual ethernet pairs, which live in the Linux kernel and connect our emulated switches to emulated hosts (processes).

In short, Mininet's virtual hosts, switches, links, and controllers are the real thing – they are just created using software rather than hardware – and for the most part their behavior is similar to discrete hardware elements. It is usually possible to create a Mininet network that resembles a hardware network, or a hardware network that resembles a Mininet network, and to run the same binary code and applications on either platform.