CHAPTER 1

# Understanding High Availability of IP and MPLS Networks

Until recently, many service providers maintained and operated separate legacy circuit-switched and packet-switched networks. Traditionally, voice services have been offered over circuit-switched networks, commonly known as *Public Switched Telephone Networks* (PSTN). On the other hand, connectivity between enterprises for *virtual private network* (VPN) data applications has been provided over packet-switched networks such as Frame Relay (FR) and *Asynchronous Transfer Mode* (ATM). Of late, many service providers are migrating legacy Layer 2 and Layer 3 services to converged *Multiprotocol Label Switching* (MPLS)-enabled IP networks.[1] This migration toward a common multiservice IP/MPLS network is driven by the necessity to reduce the *capital expenditure* (capex) and *operational expenses* (opex) of both building and operating separate network infrastructures.

This chapter describes major sources of network failures and provides an overview of techniques that are commonly used to improve availability of IP/MPLS networks. In particular, this chapter outlines mechanisms for reducing network downtime due to control-plane failures.

## Reliability and Availability of Converged Networks

For service providers, maintaining highly reliable and revenue-generating legacy service offerings is extremely important. So as much as possible, they are interested in migrating legacy services on to IP/MPLS infrastructures without cannibalizing revenue from these services. During migration, they also try to keep network downtime to a minimum (for example, in the order of a few minutes per year) to keep the cost of network outages in check. For example, a 1-minute network outage that affects 100 customers could cost a service provider several hundred thousand dollars.[2] Therefore, it is not surprising to know that network reliability and availability rank among the top concerns of the most service providers. In short, high availability of IP/MPLS networks is a prerequisite to offer reliable and profitable carrier-class services. A well-designed network element, such as a router, facilitates the building of highly available networks and reduces the capex and opex associated with redundant network infrastructures.

# Defining Key Terms

Before proceeding further, it would be useful to define some key terms.

## Availability and Unavailability

The phrase "availability of a system such as a router or network" denotes the probability (with values in the 0.0 to 1.0 range such as 0.1, 0.2, and so forth) that the system or network can be used when needed. Alternatively, the phrase describes the fraction of the time that the service is available. As a benchmark, carrier-class network equipment requires availability in the range of five-nines (0.99999), which means the equipment is available for service 99.999 percent of the time.

The term *unavailability* is defined as the probability that a system or network is not available when needed, or as the fraction of the time service is not available. An alternative and often more convenient expression (because of its additive properties) for unavailability is *downtime per year*. Downtime in units of minutes per year is obtained through multiplication of unavailability values by minutes in a year (365 days in a year times 24 hours in a day times 60 minutes in an hour). Service providers commonly use yet another expression for unavailability, especially when referring to voice calls. This term is *defects per million* (DPM). DPM measures the number of defective units (or number of failed call attempts) out of a sample size of one million units (1,000,000 call attempts). DPM is obtained by multiplying unavailability by 1,000,000. From these definitions, it follows that 0.99999 availability is equivalent to 0.00001 unavailability, 5.256 downtime per year, or 10 DPM.

## Reliability and Its Relationship to Availability

The phrase "reliability of a system or network" is defined as the probability that the system or network will perform its intended function without failure over a given period of time. A commonly used measure of reliability is known as *mean time between failures* (MTBF), which is the average expected time between failures. A service outage caused by a failure is represented as mean time to repair (MTTR). That is the average time expected to be required to restore a system from a failure. MTTR includes time required for failure detection, fault diagnosis, and actual repair. Availability is related to MTBF and MTTR as follows:

$$\text{Availability} = \text{MTBF}/(\text{MTBF} + \text{MTTR})$$

This relationship shows that increasing MTBF and decreasing MTTR improves availability. This means that the availability of a router can be improved by increasing the reliability of its hardware and software components. Similarly, improving the reliability of its constituent elements such as routers, switches, and transport facilities can enhance the availability of a network.

In general, reliability is just one of several factors that can influence the availability of a system. For example, in addition to reliability of constituent network elements, network availability is strongly influenced by the fault-tolerance capability of the network elements, as described in the following section.

## Fault Tolerance and Its Effect on Availability

Fault tolerance describes the characteristics of a system or component that is designed in such a way that, in the event of a component failure, a backup or "redundant" component immediately can take its place with no loss of service. Fault tolerance can be provided via software, hardware, or combination of the two. The switch between the failing component and the backup component is opaque to the outside world—from the view outside the system, no failure has occurred.

A network is said to be fault tolerant or survivable if it can maintain or restore an acceptable level of service performance during network failures. Network-level fault tolerance relies on software or hardware to quickly detect the failure and switch to a known backup path/link. The backup paths may be provided at multiple transport layers, including *wavelength-division multiplexing* (WDM), Synchronous Optical Network/Synchronous Digital Hierarchy (SONET/SDH), and MPLS.

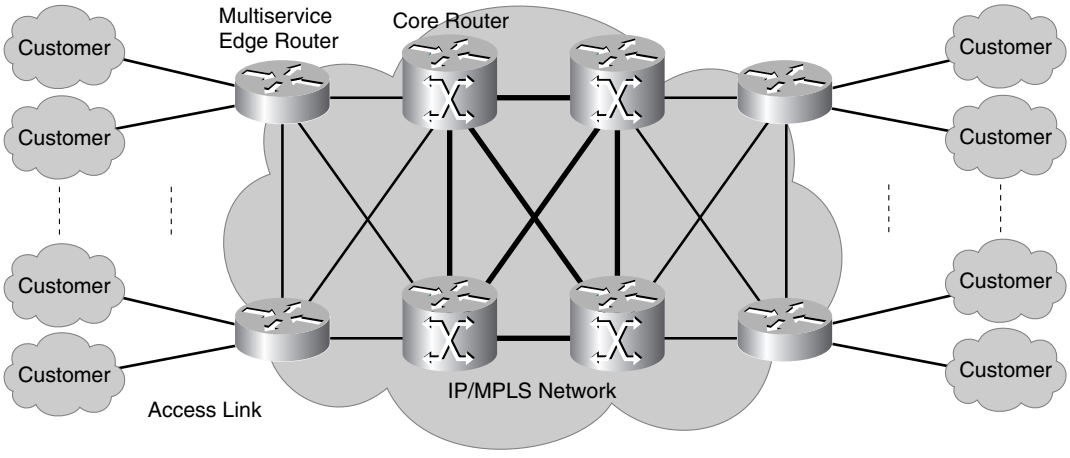
As described in the previous section, improving MTBF can increase overall system availability. However, by using redundant components, one can reduce system downtime by orders of magnitude and get closer to the carrier-class goal of five-nines availability while keeping the MTBF and MTTR the same. The effectiveness of a redundancy scheme depends on its switchover success rate (the probability of a successful switchover from active to standby component when the active component fails). Generally, it is difficult to achieve a perfect (100 percent) switchover success rate. In practice, a redundancy scheme that can achieve a 99 percent or better switchover success rate is considered a good design.

To summarize, redundancy is one of the key building blocks for improving high availability. Redundancy not only prevents equipment failures from causing service outages, it also can provide a means for in-service planned maintenance and upgrade activities.

# MPLS Network Components

An MPLS-based network consists of routers and switches interconnected via transport facilities such as fiber links (see Figure 1-1). Customers connect to the backbone (core) network through multiservice edge (MSE) routers. The backbone comprises the core routers that provide high-speed transport and connectivity between the MSE routers. An MSE router contains different types of line cards and physical interfaces to provide Layer 2 and Layer 3 services, including ATM, FR, Ethernet, and IP/MPLS VPNs.

**Figure 1-1** *Converged IP/MPLS Network Architecture*

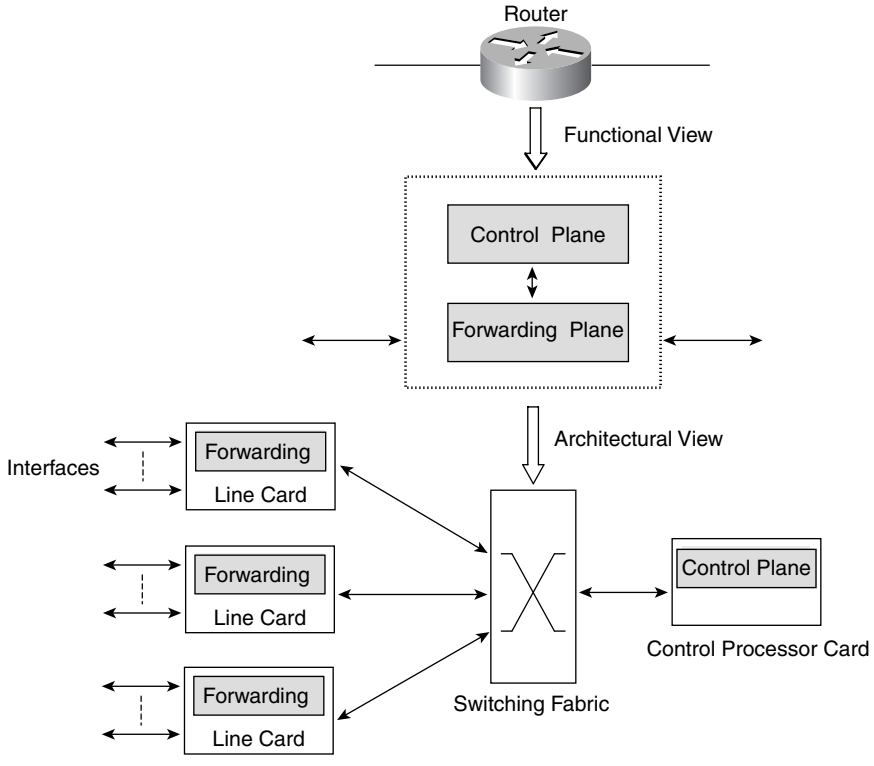



In the incoming direction, line cards receive packets from external interfaces and forward them to the switching fabric (see Figure 1-2). In the outgoing direction, line cards receive packets from the switching fabric and forward them to the outgoing interfaces. The switching fabric, the heart of the router, is used for switching packets between line cards. The IP/MPLS control-plane software, the brain of a router, resides in the control processor card. The phrase *IP/MPLS control plane* refers to the set of tasks performed by IP routing and MPLS signaling protocols. IP routing protocols are used to advertise network topology, exchange routing information, and calculate forwarding paths between routers within (intra) and between (inter) network routing domains. Examples of IP routing protocols include *Open Shortest Path First* (OSPF), *Intermediate System-to-Intermediate System* (IS-IS), and *Border Gateway Protocol* (BGP). MPLS signaling protocols are used to establish, maintain, and release *label-switched paths* (LSP). Examples of MPLS signaling protocols include BGP, *Label Distribution Protocol* (LDP), and *Resource Reservation Protocol* (RSVP). The IP control plane may also contain tunneling protocols such as Layer 2 Tunneling Protocol (L2TP) and Generic Routing Encapsulation (GRE), but these protocols are not covered in this book.

Because redundant network elements add to the overall network cost, service providers typically employ different levels and types of fault tolerance in the edge and core network. For example, the core network is generally designed to protect against core router failures through mesh connectivity. This allows alternative paths to be quickly established and used in the face of a failure. In the core, additional routers and links are used to provide fault tolerance. In contrast, on the edge, often thousands of customers are connected through a single router, and the edge router usually represents a single point of failure. The edge router is what most service providers consider the most vulnerable point of their network after the core is protected. On the edge,

instead of using additional routers and links as in the core, redundancy within the edge router via redundant control processor cards, redundant line cards, and redundant links (such as SONET/SDH Automatic Protection Switching [APS]) are commonly used to provide fault tolerance.

**Figure 1-2** *Functional Components of Router Architecture*




In summary, service (to a customer) downtime can result from failure of the access port, edge links, the edge router, backbone transport facilities, or the core routers. Generally, the core network offers a higher level of fault tolerance than the edge network. The edge router is an important network element because it routes traffic to/from multiple customers to the core network. Therefore, improving the availability of edge routers is extremely important. In short, service providers are looking for truly edge-to-edge reliability, and this includes all of the edge routers as well as the core routers.

# Network and Service Outages

A service is the set of tasks performed by the network upon a request from the user such as a voice call, Internet access, e-mail, and so forth. A *service outage* is the users' inability to request a new service or to continue to use an existing service because the service is either no longer available or it is impaired. As discussed previously, availability of a network strongly depends on the frequency of service outages and the recovery time for each outage. A *network outage* is the loss of network resources, including routers, switches, and transport facilities, because of the following:

- Complete or partial failure of hardware and software components
- Power outages
- Scheduled maintenance such as software or hardware upgrades
- Operational errors such as configuration errors
- Acts of nature such as floods, tornadoes, and earthquakes

## Planned and Unplanned Outages

Each network outage can be broadly categorized as either "unplanned" or "planned." An unplanned network outage occurs because of unforeseen failures of network elements. These failures include faults internal to a router's hardware/software components such as control-plane software crashes, line cards, link transceivers, and the power supply or faults external to the router such as fiber cuts, loss of power in a carrier facility, and so forth. A planned network outage occurs when a network element such as router is taken out of service because of scheduled events (for example, a software upgrade).

## Main Causes of Network Outages

What are the main causes of network outages? As it turns out, several culprits contribute to network downtime. According to a University of Michigan one-year reliability study of IP core routers conducted in a regional IP service provider network, router interface downtime averaged about 955 minutes per year, which translates to an interface availability of only 0.998.[3] As a reference point, a carrier-class router is expected to have a downtime of only 5.2 minutes per year. The same study indicated the following percentages of causes for total network downtime:

- 23 percent for router failure (software/hardware faults, denial-of-service attack)
- 32 percent for link failures (fiber cuts, network congestion)
- 36 percent for router maintenance (software and hardware upgrade, configuration errors)
- The remaining 9 percent for other miscellaneous reasons

According to another study, router software failures are the single biggest (25 percent) cause of all router outages.[4] Moreover, within software-related outages, router control-plane failure is the biggest (60 percent) cause of software failures. The following section provides a brief overview of various node- and network-level fault-tolerance approaches that can help to improve network availability.

# Design Strategies for Network Survivability

The reliability and availability of an IP/MPLS network can be examined from two interrelated viewpoints: service and network views. The *service view* deals with satisfying customer expectations such as availability of service and other *service-level agreements* (SLA). The *network view* deals with reducing network equipment and operation costs. Because the main task of a network is to provide user services, the reliability and availability requirements for the network are driven by the service view. An effective network design seeks to satisfy service reliability and availability objectives at the minimum network equipment (capex) and operational (opex) cost.

A packet-switched network consists of interconnected network elements, including routers, switches, and transport links. Network availability depends on the reliability and availability of its network elements. In particular, fault tolerance of router hardware and software components is crucial to deliver user services with negotiated SLAs. A carrier-class router is typically expected to satisfy requirements such as the following:

- No single hardware fault should result in a loss or degradation of user traffic or a loss of control-plane and management functions.
- System downtime should be less than 5.256 minutes per year.
- Line cards, switching fabric, and control processor cards should be redundant with capability to monitor standby cards.
- The control-plane software/hardware module should not be a single point of failure, and the service (forwarding plane) should not be disrupted due to failure of the control plane.
- The router should be capable of service recovery from link/node failures.

Generally, these carrier-class availability requirements are satisfied using a combination of node- and network-level fault-tolerance techniques, as described in the sections that follow.

## Mitigating Node-Level Unplanned Hardware-Related Outages

One of the most effective techniques for reducing unplanned hardware-related downtime in a router is the use of redundant hardware components, including line cards, switching fabric,

control processor cards, and physical interfaces. Three types of redundancy schemes are commonly used for this purpose:

- **One-for-N (1:N)**—There is one standby component for every N active component.
- **One-for-one (1:1)**—There is a standby component for each active component.
- **One-plus-one (1+1)**—This is similar to the one-for-one scheme except that in the case of one-plus-one, traffic is transmitted simultaneously on both active and standby components. (Traffic is generally ignored on the standby.) An example of one-plus-one redundancy is the 1+1 SONET/SDH APS scheme that avoids loss of data traffic caused by link failure.

A detailed discussion of component redundancy architectures is beyond the scope of this book.

## Mitigating Node-Level Unplanned Software-Related Outages

It is apparent that reliability and stability of router hardware and software are absolutely crucial for building reliable and available IP/MPLS networks. As discussed previously, routers use redundant switching fabric, control processor cards, line cards, and interfaces to achieve node-level hardware fault tolerance. Although most routers usually have adequate hardware-component redundancy coverage, the control-plane software still remains a weak link and a prime cause of router failures.

The two most important constituents of the router software are IP and MPLS control-plane protocols. The IP control-plane component consists of IP routing protocols such as OSPF, IS-IS, and BGP, which exchange network topology information and thus help build the IP forwarding state. The MPLS control-plane component is composed of signaling protocols such as LDP, RSVP-TE, and BGP. Label-switching routers (LSR) use information provided by IP/MPLS control-plane components to construct the MPLS forwarding state. The IP forwarding state is used to transfer IP packets from an incoming port of the router to an outgoing port using a destination IP address. In contrast, the MPLS forwarding state is used for moving packets from input to output ports based on label information.

IP and MPLS forwarding tables are collectively referred to as the *forwarding plane*. Because of the time-critical nature of packet-forwarding operations, the forwarding-plane functions are typically distributed on line cards to enhance forwarding performance. In contrast, control-plane tasks are relatively less time critical and therefore often reside on the central control processor card. Because control-plane protocols constitute router intelligence, the control processor serves as host to the router's brain. Because of the pivotal importance of the control-plane functions to the router operation, a control processor is normally protected against failure through 1:1 (active and standby) redundancy.

The existing control-plane software restart and switchover behavior in routers is disruptive and therefore undesirable. When a router detects a software/hardware failure in the active control processor, it switches over to the standby and, in this process, not only restarts its control software but also resets the forwarding plane in the line cards. The end result of this behavior means disruption of data forwarding and the accompanied service outage. Consider, for

example, the restart of an IP control-plane protocol such as OSPF or IS-IS. When OSPF or IS-IS restarts, the failing router's *interior gateway protocol* (IGP) neighbors detect this restart and originate LSAs or LSPs to omit links to the restarting router. Upon receiving new LSAs or LSPs, the nonrestarting routers recompute their paths to avoid the restarting router. This shows that the original IP control-plane restart behavior causes unnecessary disruption of traffic in the restarting router, generates extra IGP control traffic, and triggers costly *shortest path first* (SPF) recomputations in nonrestarting routers. Similarly, when the MPLS control plane restarts, LDPs withdraw labels that were advertised prior to this failure. Once again, this behavior results in disruption of the MPLS forwarding. In short, one can say that control-plane restart causes instability throughout the network.

This description clearly shows that the original IP/MPLS control-plane restart behavior is totally unacceptable, particularly when you consider the fact that service providers are deploying more and more IP/MPLS networks to deliver legacy services and customers are expecting a better or comparable level of reliability and availability. Therefore, disruption of the IP/MPLS forwarding plane must be reduced to an absolute minimum. The next section outlines some approaches to achieve this goal.

## Reducing Downtime Related to Unplanned Control-Plane Restart

Several types of software redundancy schemes enable you to reduce router downtime resulting from unplanned control-plane failures. One such approach (similar to the 1:1 hardware redundancy scheme) is to instantiate two identical copies of control-plane software on active and standby control processors. The two instances execute independently without any inter-instance communication, and both instances send/receive identical control packets. For example, in the incoming direction, control packets are replicated and passed on to both instances. In the outgoing direction, control packets from the standby instance are discarded.

A second scheme (a variant of the previous approach) is to instantiate two identical copies of the control plane. The two instances execute in complete lock step using inter-instance communication. When a control packet is received, it is processed in identical fashion and at the exact same instant by the active and the standby instance.

A third approach is to instantiate two copies of the control plane on active and standby control processors. The active instance executes, whereas the inactive instance does not. However, the standby instance maintains partial state of the active instance by receiving state synchronization messages. For example, the active instance of an IP routing protocol establishes sessions, exchanges routing information with peers, and helps build and maintain routing/forwarding tables. In contrast, the inactive standby instance does not exchange routing information with external peers. After switchover, the standby instance takes over, reestablishes peer sessions, and resynchronizes its state information. Another variant of the third approach maintains complete state on the standby and can switch over without having to reestablish sessions from the point of view of the neighbors. However, this variant is less scalable because it requires preservation of the complete state.

Table 1-1 describes the advantages and disadvantages of each approach.

**Table 1-1** *Advantages and Disadvantages of Software Redundancy Approaches*

| Approach | Advantages | Disadvantages |
|---|---|---|
| **Approach 1**<br>Instantiate two identical copies of the control-plane software on active and standby control processors. The two instances execute independently. | Control-plane failure and resulting switchover hidden from neighbors<br>No requirement for changes to IP/MPLS control-plane protocols | Extra processing burden for replicating control packets<br>Necessity to start both instances at the same time<br>Restriction of precluding software upgrades and downgrades |
| **Approach 2**<br>Instantiate two identical copies of the control-plane software on active and standby control processors. The two instances execute in lock step using inter-instance communication. | Fast recovery time<br>No necessity for protocol changes | Design complexity of synchronizing state<br>Requirement to run two instances synchronously<br>Restriction of precluding software upgrades and downgrades |
| **Approach 3**<br>Instantiate two copies of the control plane on active and standby control processors. The standby instance maintains partial state. | Allows the restarting router to continue to forward across the control-plane recovery<br>Allows for software upgrades | Necessity to reestablish sessions and recover the control-plane state information after the restart<br>Needs protocol extensions |

The main strength of the first approach is that the control-plane failure and resulting switchover is hidden from neighbors and therefore does not require any changes to IP/MPLS control-plane protocols. However, this approach has two big drawbacks: extra processing burden for replicating control packets, and the necessity to start both instances at the same time. The second drawback is very restrictive because it precludes software upgrades and downgrades. The key advantages of the second approach are fast recovery time and absence of necessity to make protocol changes. The main disadvantages of this approach are design complexity to synchronize state and the requirement to run two instances synchronously. The latter requirement implies that, like the first approach, the second approach does not allow software upgrades. The main disadvantage of the third approach is the necessity to reestablish sessions and recover control-plane state information after the restart. This requires IP/MPLS protocol extensions and support from neighbors in maintaining their forwarding state while the restarting router comes back.

The third approach, analogous to other two approaches, allows the restarting router to continue to forward across the control-plane recovery. However, unlike the other two schemes, the third approach allows software upgrades, which is a big plus toward achieving the carrier-class availability goals.

Cisco IOS architecture is likely to adopt the third approach and its variants to provide fault-tolerant control-plane software architecture on routers. For example, in the core and on the core side of the edge routers where scalability is extremely important, the third approach can be used because it requires preserving partial control-plane state. In contrast, on the customer side of edge routers where scalability is generally not much of an issue, a variant of the third approach (completely stateful) can be used.

## Stateful Switchover and Nonstop Forwarding

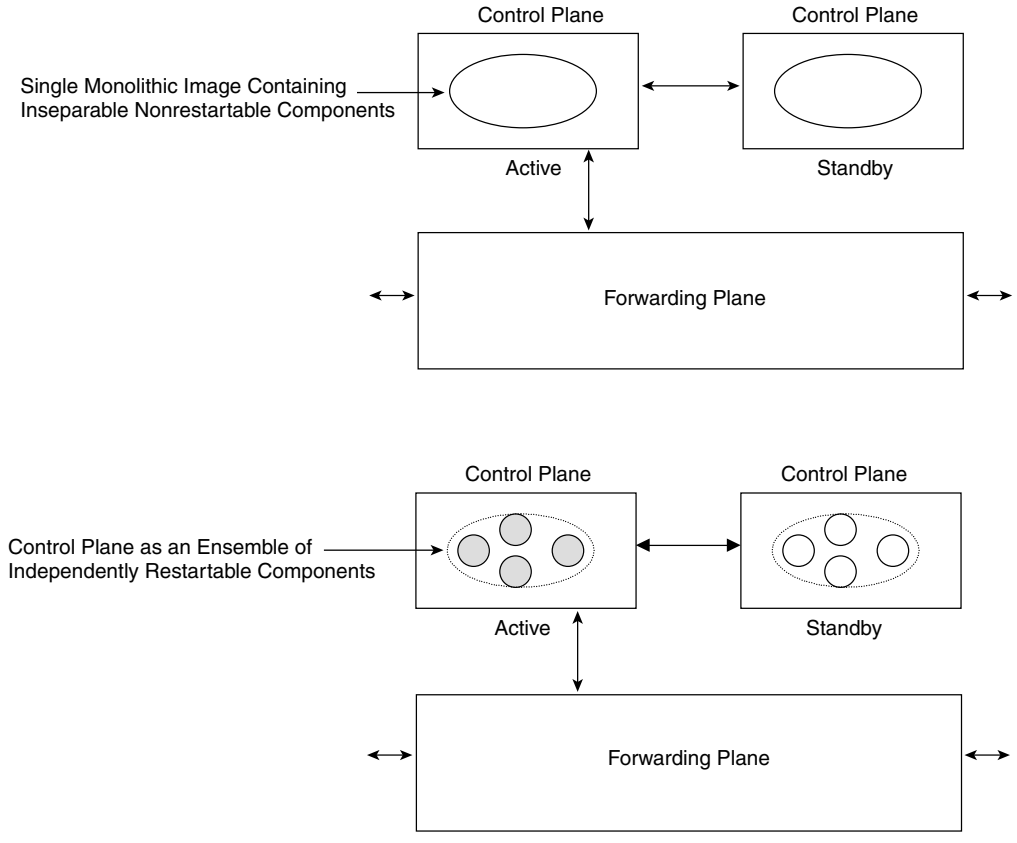
The combination of separation of control- and forwarding-plane components, 1:1 redundant control processors, and fault-tolerant control-plane software allows Cisco IOS architecture to make an automatic nondisruptive stateful control-plane switchover upon detection of software/hardware failures in the active control processor. In IOS the term *stateful switchover* (SSO) refers to the aforementioned control-plane redundancy framework that enables nondisruptive automatic SSO of the control plane upon detection of hardware/software failure. The term *nonstop forwarding* (NSF) refers to the capability of a router to continue to forward while its control plane recovers from a fault. NSF requires separation of control- and forwarding-plane functions.

## Reducing Unplanned Downtime Using Component-Level Modularity and Restartability

The SSO discussion in the preceding section assumes that control-plane software executes as one or more inseparable components (or processes) that are sharing critical data structures. In that case, because various control-plane components are inseparable and incapable of restarting individually, failure in one component leads to failure of all other components and necessitates control-plane switchover. Therefore, nonrestartable components require stateful redundancy schemes and switchovers to recover from failures (see Figure 1-3).

A software component is said to be *restartable* if it is capable of recovering from fatal runtime errors. In a redundant system, a restartable component that can correctly recover from failures should not require a switchover. However, when a restartable component fails to restart correctly, it should cause a switchover to the standby to recover from failures.

**Figure 1-3** *Nonrestartable and Restartable Control-Plane Components*




Hence, the component-level restartability complements and extends the SSO approach by providing additional fault-tolerance coverage. In the component-level restartability approach, following system initialization, a system management module instantiates all control-plane components and monitors their health. Upon detecting failure of a control-plane component, the system manager restarts the failed component without disturbing other components or requiring control-plane switchover. After having restarted, the process recovers its preserved state information and resumes normal operation.

It is worth noting that unlike SSO, the process-level restartability approach can be used to improve control-plane availability of routers with single as well as redundant control processors. In a nonredundant control processor scenario, the component restartability-based approach allows a router to recover from unplanned control-plane software component failures. In a redundant control processor case, a combination of SSO and component-level restartability helps improve the overall fault isolation, reliability, and availability of the control plane. In the latter case, for example, a router can recover from minor software component-level faults

without requiring switchover to the standby control processor and yet use SSO to recover from major software or hardware faults.

In summary, component restartability is not a remedy for all failure scenarios. For example, if critical data structures of the restartable component are damaged, that component will fail to restart. Complete switchover with redundant hardware components and no shared data (no shared memory) offers a higher level of fault tolerance. You should view these approaches as occurring within a spectrum of approaches to improve high availability, with their own attendant benefits and costs.

In the remainder of this book, it is assumed that the control-plane software runs as a single image containing inseparable nonrestartable components. Discussion of approaches for improving control-plane reliability and availability using component-level modularity and restartability is beyond the scope of this book.

## Mitigating Node-Level Planned Outages

As discussed previously, planned events such as software upgrades are a big contributor to network downtime. To deliver carrier-class services, unplanned and planned outages must be reduced. The downtime due to planned events is reduced using an in-service software-upgrade mechanism that allows upgrading the router software without service disruption.

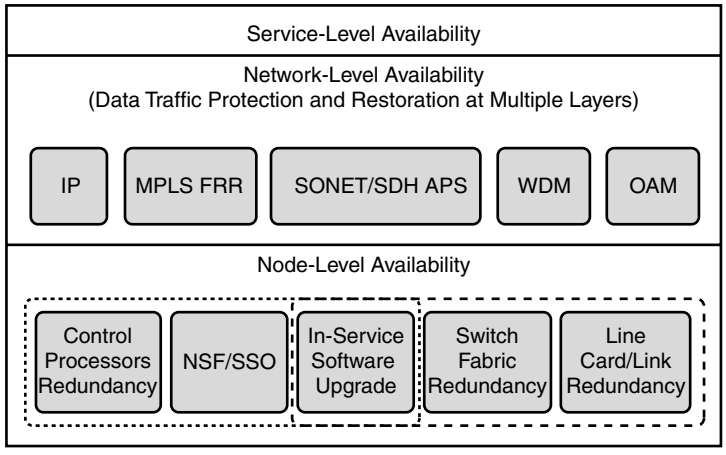
## Mitigating Network Outages Against Link and Node Failures

So far this chapter has discussed strategies for reducing downtime caused by unplanned control-plane restart and planned router operations such as software upgrades. Another significant contribution to network downtime comes from link failures. The impact of transport path failures is mitigated by using multilayer protection/rerouting schemes such as SONET/SDH APS and emerging MPLS-based methods such as *FastReRoute* (FRR). With the success of MPLS deployments, the use of MPLS-based recovery schemes is also growing to provide LSP-level protection against link/node failures. Although SONET/SDH-based protection is widely deployed, protection at the lower transport layers is very coarse and can be very wasteful and expensive. In contrast, MPLS-based recovery can provide much finer granularity and presents an efficient, attractive, and complementary alternative to SONET/SDH-based protection.

## Mitigating Network Outages via Effective Operation and Maintenance Mechanisms

As service providers move more and more revenue-generating services onto converged IP/MPLS networks, effective MPLS *operation and maintenance* (OAM) mechanisms become an absolute necessity to deliver carrier-class services. This is because service providers rely on robust OAM tools to quickly identify and remove network faults, reduce service downtime, and maintain a high level of network availability. A layered view of IP/MPLS availability architecture is depicted in Figure 1-4.

**Figure 1-4** *Dependence of End-to-End IP/MPLS Service Availability on Node and Network Level Availability*




## Improving Network Security via Fault-Tolerance Mechanisms

Network resources include routers, switches, hardware, software, data stored on line, data in transit over the network, and so forth. Network security refers to the set of measures taken to protect a resource against unauthorized access. For each resource, the key objectives of security are resource availability, data confidentiality (meaning that information is not made available or disclosed to unauthorized individuals or entities), and data integrity (meaning that information has not been modified in an unauthorized manner). Some exploits that might threaten an IP/MPLS network include attacks on control and forwarding planes, sniffing of data packets, denial-of-service (DoS) attacks, and so forth. In a DoS attack, an attacker seeks to disrupt or prevent the use of a service by its legitimate users. A DoS attack might appear in different forms such as taking network devices out of service by overwhelming the target devices with requests for service or modifying their normal behavior. A DoS attack in which the network is overwhelmed with requests for service is also known as a resource-exhaustion DoS attack. Resource-exhaustion DoS attacks can be mounted against any network resource such as forwarding plane, control plane (for example, control processor), link bandwidth, and so forth.

Because the goal of fault-tolerance mechanisms is to protect a system or network against different types of failures by improving its availability, fault-tolerance mechanisms may also be thought of as defensive techniques against malicious security threats. For example, separation of control plane and forwarding plane (as provided in the SSO/NSF framework) can be used to improve security against some attacks. This, for example, might help to limit DoS attacks against a control plane to that particular component only and might allow the forwarding-plane component to continue to function normally.

In general, to offer network services securely and reliably, security and fault-tolerance mechanisms must be built in to IP and MPLS networks. Examples of common defensive techniques against network security threats include data encryption, authentication, packet filtering, firewalls, separation of control and forwarding planes, intrusion detection, intrusion prevention, and so forth.[5] A detailed discussion of network security mechanisms is beyond the scope of this book.

## Scope of the Book

From the discussions in this chapter so far, you know that the design of carrier-class IP/MPLS networks involves reducing both unplanned and planned outages by using a variety of fault-tolerance techniques, including node-level hardware redundancy, control-plane software redundancy, MPLS-layer redundant LSPs, OAM mechanisms, and in-service software upgrades. In short, the reliability and availability of an IP/MPLS network encompasses a broad set of functional areas.

The main purpose of this book is to describe IP/MPLS control-plane fault-tolerance mechanisms that enable you to reduce downtime and improve network availability (by reducing unplanned IP/MPLS control-plane failures). Specifically, this book intends to cover three aspects of the control plane, as follows:

- IP/MPLS forwarding-plane NSF mechanisms that allow a router to continue to forward traffic while its control plane recovers from a failure
- IP/MPLS control-plane restart mechanisms that enable IP/MPLS control-plane components to restart and recover state without disrupting the forwarding plane
- Use of the previous two mechanisms to reduce downtime in the converged IP/MPLS backbone when using MPLS applications such as *traffic engineering* (TE), *Layer 2 VPNs* (L2VPNs), and *Layer 3 VPNs* (L3VPNs).

In the remainder of this book, it is assumed that the control-plane software executes as a single image containing inseparable nonrestartable components. A detailed discussion of process-level modularity and restartability is beyond the scope of this book.

Although for completeness sake fault-tolerance mechanisms such as MPLS FRR, MPLS OAM, and in-service software upgrades are briefly mentioned in a later chapter, a detailed discussion of these mechanisms is also beyond the scope of this book.

## References

[1] Heavy Reading Analysts, "2004 Survey of Carrier Attitudes Toward IP/MPLS Backbones and VPNs," *Heavy Reading Report*, Vol. 2, No. 4, January 2004.

[2] Network Strategy Partners, "Reliable IP Nodes: A Prerequisite to Profitable IP Services," White Paper, November 2002.

[3] Ahuja, A., F. Jahanian, and C. Labovitz, "Experimental Study of Internet Stability and Wide-Area Backbone Failures," Proceeding of 29th International Symposium on Fault-Tolerant Computing, June 1999.

[4] Heywood, P., and M. Reardon, "IP Reliability," *Light Reading Report*, March 2003.

[5] Fang, L., "Security Framework for Provider Provisioned Virtual Private Networks," IETF work in progress, July 2004.