Network Working Group
Request for Comments: 1583
Obsoletes: 1247
Category: Standards Track

J. Moy
Proteon, Inc.
March 1994

OSPF Version 2

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Abstract

This memo documents version 2 of the OSPF protocol. OSPF is a link-state routing protocol. It is designed to be run internal to a single Autonomous System. Each OSPF router maintains an identical database describing the Autonomous System's topology. From this database, a routing table is calculated by constructing a shortest-path tree.

OSPF recalculates routes quickly in the face of topological changes, utilizing a minimum of routing protocol traffic. OSPF provides support for equal-cost multipath. Separate routes can be calculated for each IP Type of Service. An area routing capability is provided, enabling an additional level of routing protection and a reduction in routing protocol traffic. In addition, all OSPF routing protocol exchanges are authenticated.

OSPF Version 2 was originally documented in RFC 1247. The differences between RFC 1247 and this memo are explained in Appendix E. The differences consist of bug fixes and clarifications, and are backward-compatible in nature. Implementations of RFC 1247 and of this memo will interoperate.

Please send comments to ospf@gated.cornell.edu.

Contents

1 Introduction

This document is a specification of the Open Shortest Path First (OSPF) TCP/IP internet routing protocol. OSPF is classified as an Interior Gateway Protocol (IGP). This means that it distributes routing information between routers belonging to a single Autonomous System. The OSPF protocol is based on link-state or SPF technology. This is a departure from the Bellman-Ford base used by traditional TCP/IP internet routing protocols.

The OSPF protocol was developed by the OSPF working group of the Internet Engineering Task Force. It has been designed expressly for the TCP/IP internet environment, including explicit support for IP subnetting, TOS-based routing and the tagging of externally-derived routing information. OSPF also provides for the authentication of routing updates, and utilizes IP multicast when sending/receiving the updates. In addition, much work has been done to produce a protocol that responds quickly to topology changes, yet involves small amounts of routing protocol traffic.

The author would like to thank Fred Baker, Jeffrey Burgan, Rob Coltun, Dino Farinacci, Vince Fuller, Phanindra Jujjavarapu, Milo Medin, Kannan Varadhan and the rest of the OSPF working group for the ideas and support they have given to this project.

1.1 Protocol overview

OSPF routes IP packets based solely on the destination IP address and IP Type of Service found in the IP packet header. IP packets are routed “as is” – they are not encapsulated in any further protocol headers as they transit the Autonomous System. OSPF is a dynamic routing protocol. It quickly detects topological changes in the AS (such as router interface failures) and calculates new loop-free routes after a period of convergence. This period of convergence is short and involves a minimum of routing traffic.

In a link-state routing protocol, each router maintains a database describing the Autonomous System’s topology. Each participating router has an identical database. Each individual piece of this database is a particular router’s local state (e.g., the router’s usable interfaces and reachable neighbors). The router distributes its local state throughout the Autonomous System by flooding.

All routers run the exact same algorithm, in parallel. From the topological database, each router constructs a tree of shortest paths with itself as root. This shortest-path tree gives the route to each destination in the Autonomous System. Externally derived routing information appears on the tree as leaves.

OSPF calculates separate routes for each Type of Service (TOS). When several equal-cost routes to a destination exist, traffic is distributed equally among them. The cost of a route is described by a single dimensionless metric.

OSPF allows sets of networks to be grouped together. Such a grouping is called an area. The topology of an area is hidden from the rest of the Autonomous System. This information hiding enables a significant reduction in routing traffic. Also, routing within the area is determined only by the area’s own topology, lending the area protection from bad routing data. An area is a generalization of an IP subnetted network.

OSPF enables the flexible configuration of IP subnets. Each route distributed by OSPF has a destination and mask. Two different subnets of the same IP network number may have different sizes (i.e., different masks). This is commonly referred to as variable length subnetting. A packet is routed to the best (i.e., longest or most specific) match. Host routes are considered to be subnets whose masks are “all ones” (`0xffffffff`).

All OSPF protocol exchanges are authenticated. This means that only trusted routers can participate in the Autonomous System’s routing. A variety of authentication schemes can be used; a single authentication scheme is configured for each area. This enables some areas to use much stricter authentication than others.

Externally derived routing data (e.g., routes learned from the Exterior Gateway Protocol (EGP)) is passed transparently throughout the Autonomous System. This externally derived data is kept separate from the OSPF protocol’s link state data. Each external route can also be tagged by the advertising router, enabling the passing of additional information between routers on the boundaries of the Autonomous System.

1.2 Definitions of commonly used terms

This section provides definitions for terms that have a specific meaning to the OSPF protocol and that are used throughout the text. The reader unfamiliar with the Internet Protocol Suite is referred to [RS-85-153] for an introduction to IP.

Router A level three Internet Protocol packet switch. Formerly called a gateway in much of the IP literature.

Autonomous System A group of routers exchanging routing information via a common routing protocol. Abbreviated as AS.

Interior Gateway Protocol The routing protocol spoken by the routers belonging to an Autonomous system. Abbreviated as IGP. Each Autonomous System has a single IGP. Separate Autonomous Systems may be running different IGPs.

Router ID A 32-bit number assigned to each router running the OSPF protocol. This number uniquely identifies the router within an Autonomous System.

Network In this memo, an IP network/subnet/supernet. It is possible for one physical network to be assigned multiple IP network/subnet numbers. We consider these to be separate networks. Point-to-point physical networks are an exception - they are considered a single network no matter how many (if any at all) IP network/subnet numbers are assigned to them.

Network mask A 32-bit number indicating the range of IP addresses residing on a single IP network/subnet/supernet. This specification displays network masks as hexadecimal numbers. For example, the network mask for a class C IP network is displayed as `0xffffff00`. Such a mask is often displayed elsewhere in the literature as `255.255.255.0`.

Multi-access networks Those physical networks that support the attachment of multiple (more than two) routers. Each pair of routers on such a network is assumed to be able to communicate directly (e.g., multi-drop networks are excluded).

Interface The connection between a router and one of its attached networks. An interface has state information associated with it, which is obtained from the underlying lower level protocols and the routing protocol itself. An interface to a network has associated with it a single IP address and mask (unless the network is an unnumbered point-to-point network). An interface is sometimes also referred to as a link.

Neighboring routers Two routers that have interfaces to a common network. On multi-access networks, neighbors are dynamically discovered by OSPF's Hello Protocol.

Adjacency A relationship formed between selected neighboring routers for the purpose of exchanging routing information. Not every pair of neighboring routers become adjacent.

Link state advertisement Describes the local state of a router or network. This includes the state of the router's interfaces and adjacencies. Each link state advertisement is flooded throughout the routing domain. The collected link state advertisements of all routers and networks forms the protocol's topological database.

Hello Protocol The part of the OSPF protocol used to establish and maintain neighbor relationships. On multi-access networks the Hello Protocol can also dynamically discover neighboring routers.

Designated Router Each multi-access network that has at least two attached routers has a Designated Router. The Designated Router generates a link state advertisement for the multi-access network and has other special responsibilities in the running of the protocol. The Designated Router is elected by the Hello Protocol.

The Designated Router concept enables a reduction in the number of adjacencies required on a multi-access network. This in turn reduces the amount of routing protocol traffic and the size of the topological database.

Lower-level protocols The underlying network access protocols that provide services to the Internet Protocol and in turn the OSPF protocol. Examples of these are the X.25 packet and frame levels for X.25 PDNs, and the ethernet data link layer for ethernets.

1.3 Brief history of link-state routing technology

OSPF is a link-state routing protocol. Such protocols are also referred to in the literature as SPF-based or distributed-database protocols. This section gives a brief description of the developments in link-state technology that have influenced the OSPF protocol.

The first link-state routing protocol was developed for use in the ARPANET packet switching network. This protocol is described in [McQuillan]. It has formed the starting point for all other link-state protocols. The homogeneous Arpanet environment, i.e., single-vendor packet switches connected by synchronous serial lines, simplified the design and implementation of the original protocol.

Modifications to this protocol were proposed in [Perlman]. These modifications dealt with increasing the fault tolerance of the routing protocol through, among other things, adding a checksum to the link state advertisements (thereby detecting database corruption). The paper also included means for reducing the routing traffic overhead in a link-state protocol. This was accomplished by introducing mechanisms which enabled the interval between link state advertisement originations to be increased by an order of magnitude.

A link-state algorithm has also been proposed for use as an ISO IS-IS routing protocol. This protocol is described in [DEC]. The protocol includes methods for data and routing traffic reduction when operating over broadcast networks. This is accomplished by election of a Designated Router for each broadcast network, which then originates a link state advertisement for the network.

The OSPF subcommittee of the IETF has extended this work in developing the OSPF protocol. The Designated Router concept has been greatly enhanced to further reduce the amount of routing traffic required. Multicast capabilities are utilized for additional routing bandwidth reduction. An area routing scheme has been developed enabling information hiding/protection/reduction. Finally, the algorithm has been modified for efficient operation in the TCP/IP internets.

1.4 Organization of this document

The first three sections of this specification give a general overview of the protocol's capabilities and functions. Sections 4-16 explain the protocol's mechanisms in detail. Packet formats, protocol constants and configuration items are specified in the appendices.

Labels such as HelloInterval encountered in the text refer to protocol constants. They may or may not be configurable. The architectural constants are explained in Appendix B. The configurable constants are explained in Appendix C.

The detailed specification of the protocol is presented in terms of data structures. This is done in order to make the explanation more precise. Implementations of the protocol are required to support the functionality described, but need not use the precise data structures that appear in this memo.

2 The Topological Database

The Autonomous System's topological database describes a directed graph. The vertices of the graph consist of routers and networks. A graph edge connects two routers when they are attached via a physical point-to-point network. An edge connecting a router to a network indicates that the router has an interface on the network.

The vertices of the graph can be further typed according to function. Only some of these types carry transit data traffic; that is, traffic that is neither locally originated nor locally destined. Vertices that can carry transit traffic are indicated on the graph by having both incoming and outgoing edges.

Vertex type	*Vertex name*	*Transit?*
1	Router	yes
2	Network	yes
3	Stub network	no

Table 1: OSPF vertex types.

OSPF supports the following types of physical networks:

Point-to-point networks A network that joins a single pair of routers. A 56Kb serial line is an example of a point-to-point network.

Broadcast networks Networks supporting many (more than two) attached routers, together with the capability to address a single physical message to all of the attached routers (broadcast). Neighboring routers are discovered dynamically on these nets using OSPF's Hello Protocol. The Hello Protocol itself takes advantage of the broadcast capability. The protocol makes further use of multicast capabilities, if they exist. An ethernet is an example of a broadcast network.

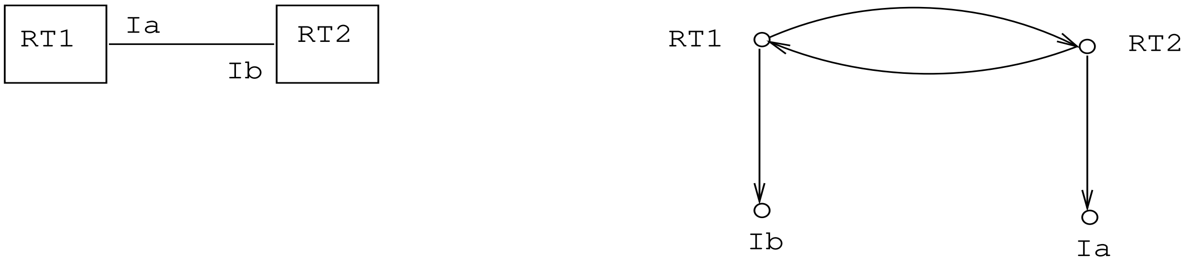
Non-broadcast networks Networks supporting many (more than two) attached routers, but having no broadcast capability. Neighboring routers are also discovered on these nets using OSPF's Hello Protocol. However, due to the lack of broadcast capability, some configuration information is necessary for the correct operation of the Hello Protocol. On these networks, OSPF protocol packets that are normally multicast need to be sent to each neighboring router, in turn. An X.25 Public Data Network (PDN) is an example of a non-broadcast network.

The neighborhood of each network node in the graph depends on whether the network has multi-access capabilities (either broadcast or non-broadcast) and, if so, the number of routers having an interface to the network. The three cases are depicted in Figure 1. Rectangles indicate routers. Circles and oblongs indicate multi-access networks. Router names are prefixed with the letters RT and network names with the letter N. Router interface names are prefixed by the letter I. Lines between routers indicate point-to-point networks. The left side of the figure shows a network with its connected routers, with the resulting graph shown on the right.

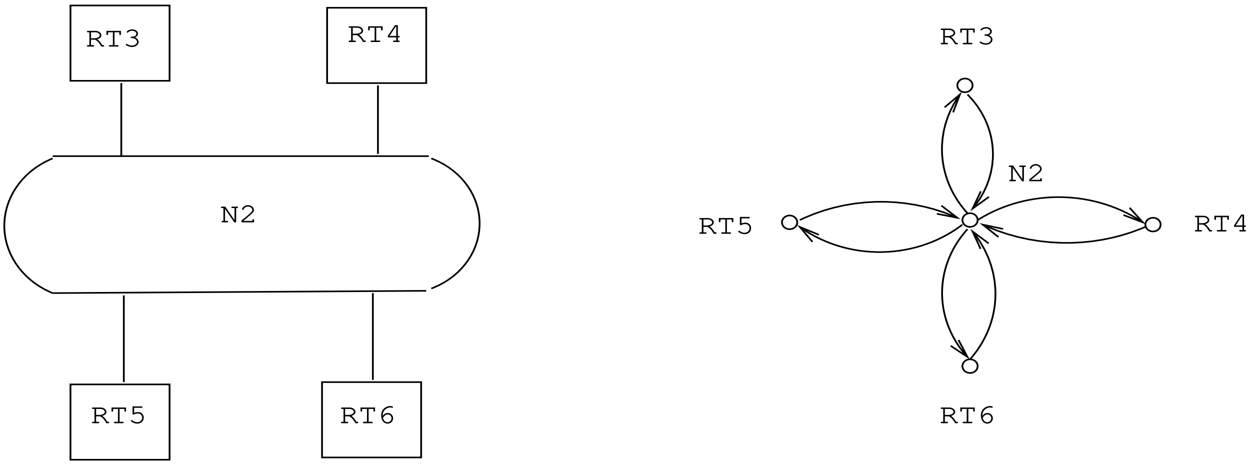
Two routers joined by a point-to-point network are represented in the directed graph as being directly connected by a pair of edges, one in each direction. Interfaces to physical point-to-point networks need not be assigned IP addresses. Such a point-to-point network is called unnumbered. The graphical representation of point-to-point networks is designed so that unnumbered networks can be supported naturally. When interface addresses exist, they are modelled as stub routes. Note that each router would then have a stub connection to the other router's interface address (see Figure 1).

When multiple routers are attached to a multi-access network, the directed graph shows all routers bidirectionally connected to the network vertex (again, see Figure 1). If only a single router is attached to a multi-access network, the network will appear in the directed graph as a stub connection.

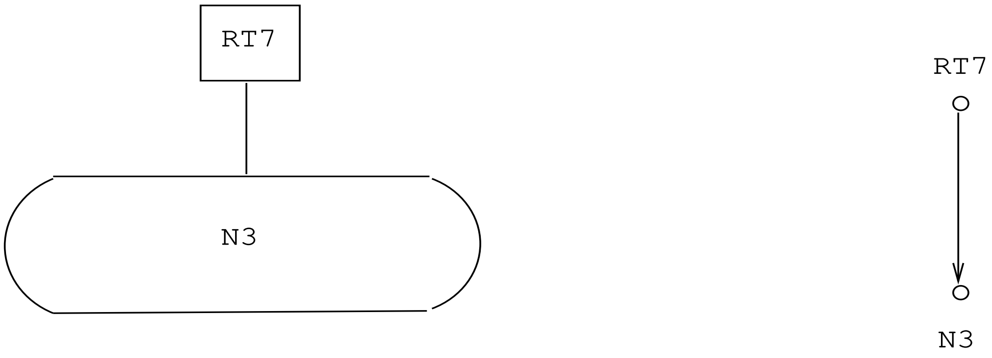
Each network (stub or transit) in the graph has an IP address and associated network mask. The mask indicates the number of nodes on the network. Hosts attached directly to routers (referred to as host routes) appear on the graph as stub networks. The network mask for a host route is always `0xffffffff`, which indicates the presence of a single node.



Physical point-to-point networks



Multi-access networks



Stub multi-access networks

Figure 1: Network map components

Figure 2 shows a sample map of an Autonomous System. The rectangle labelled H1 indicates a host, which has a SLIP connection to Router RT12. Router RT12 is therefore advertising a host route. Lines between routers indicate physical point-to-point networks. The only point-to-point network that has been assigned interface addresses is the one joining Routers RT6 and RT10. Routers RT5 and RT7 have EGP connections to other Autonomous Systems. A set of EGP-learned routes have been displayed for both of these routers.

A cost is associated with the output side of each router interface. This cost is configurable by the system administrator. The lower the cost, the more likely the interface is to be used to forward data traffic. Costs are also associated with the externally derived routing data (e.g., the EGP-learned routes).

The directed graph resulting from the map in Figure 2 is depicted in Figure 3. Arcs are labelled with the cost of the corresponding router output interface. Arcs having no labelled cost have a cost of 0. Note that arcs leading from networks to routers always have cost 0; they are significant nonetheless. Note also that the externally derived routing data appears on the graph as stubs.

The topological database (or what has been referred to above as the directed graph) is pieced together from link state advertisements generated by the routers. The neighborhood of each transit vertex is represented in a single, separate link state advertisement. Figure 4 shows graphically the link state representation of the two kinds of transit vertices: routers and multi-access networks. Router RT12 has an interface to two broadcast networks and a SLIP line to a host. Network N6 is a broadcast network with three attached routers. The cost of all links from Network N6 to its attached routers is 0. Note that the link state advertisement for Network N6 is actually generated by one of the attached routers: the router that has been elected Designated Router for the network.

2.1 The shortest-path tree

When no OSPF areas are configured, each router in the Autonomous System has an identical topological database, leading to an identical graphical representation. A router generates its routing table from this graph by calculating a tree of shortest paths with the router itself as root. Obviously, the shortest-path tree depends on the router doing the calculation. The shortest-path tree for Router RT6 in our example is depicted in Figure 5.

The tree gives the entire route to any destination network or host. However, only the next hop to the destination is used in the forwarding process. Note also that the best route to any router has also been calculated. For the processing of external data, we note the next hop and distance to any router advertising external routes. The resulting routing table for Router RT6 is pictured in Table 2. Note that there is a separate route for each end of a numbered serial line (in this case, the serial line between Routers RT6 and RT10).

Routes to networks belonging to other AS'es (such as N12) appear as dashed lines on the shortest path tree in Figure 5. Use of this externally derived routing information is considered in the next section.

2.2 Use of external routing information

After the tree is created the external routing information is examined. This external routing information may originate from another routing protocol such as EGP, or be statically configured (static routes). Default routes can also be included as part of the Autonomous System's external routing information.

External routing information is flooded unaltered throughout the AS. In our example, all the routers in the Autonomous System know that Router RT7 has two external routes, with metrics 2 and 9.

OSPF supports two types of external metrics. Type 1 external metrics are equivalent to the link state metric. Type 2 external metrics are greater than the cost of any path internal to the AS. Use of Type 2 external metrics assumes that routing between AS'es is the major cost of routing a packet, and eliminates the need for conversion of external costs to internal link state metrics.

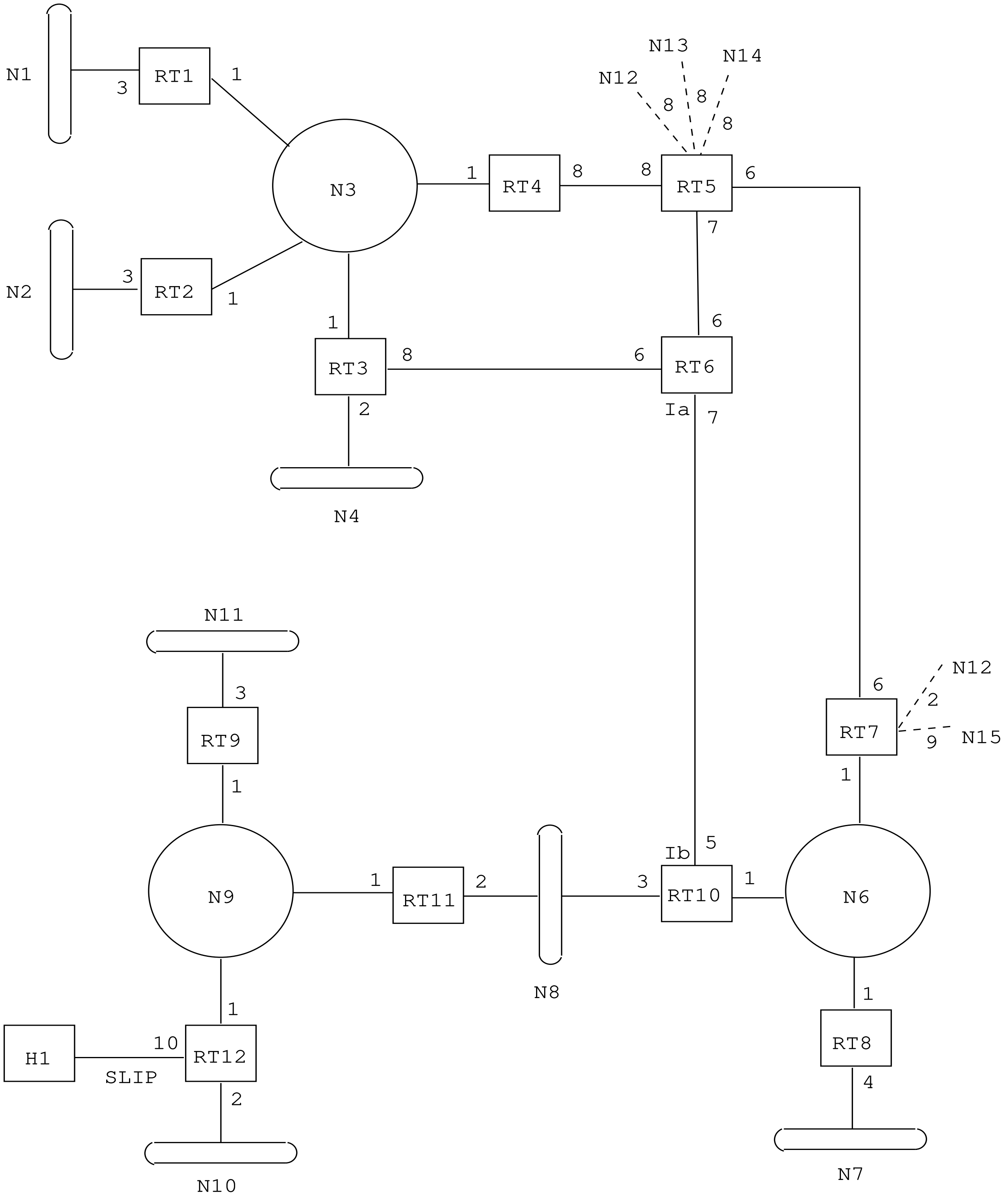


Figure 2: A sample Autonomous System

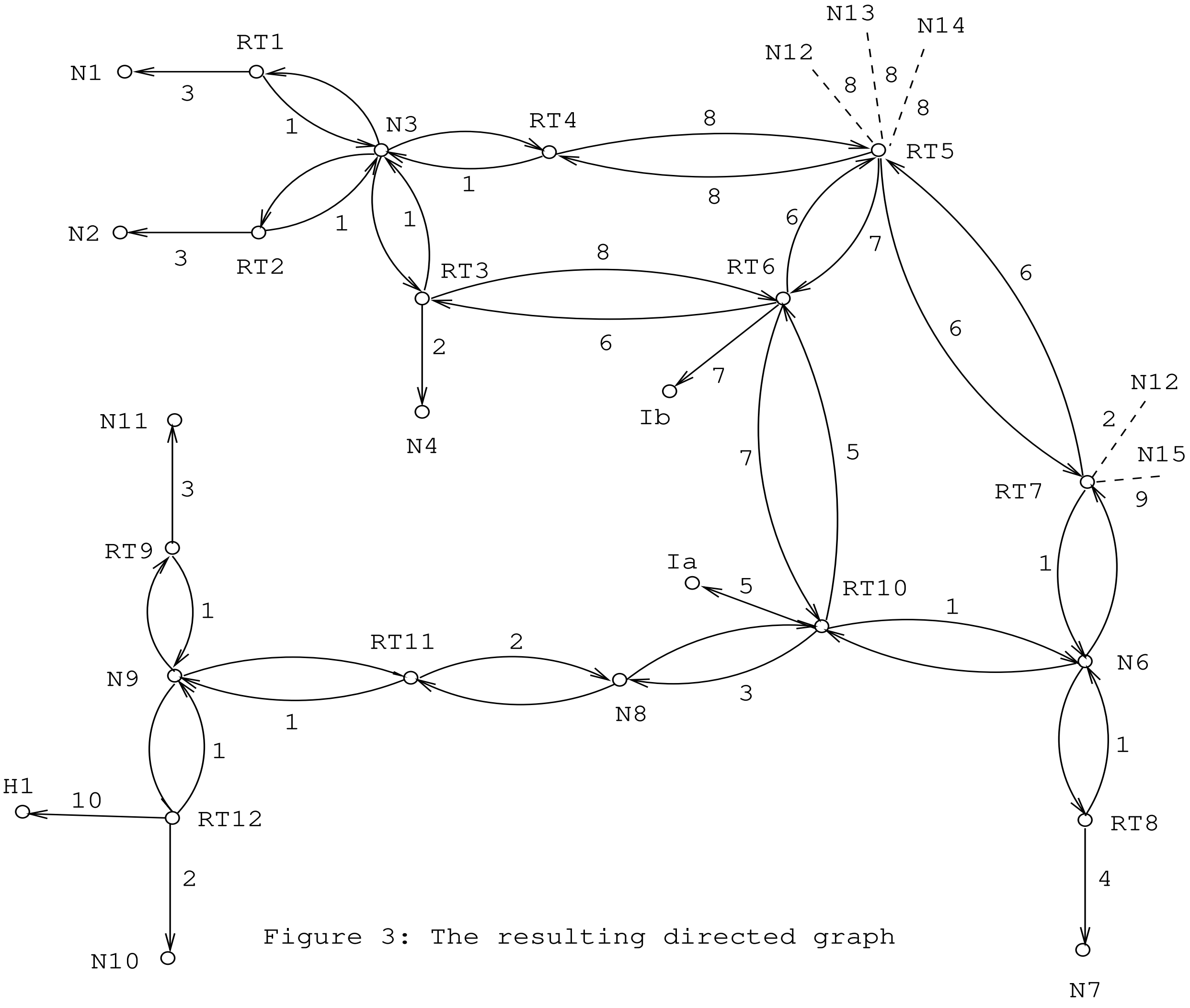
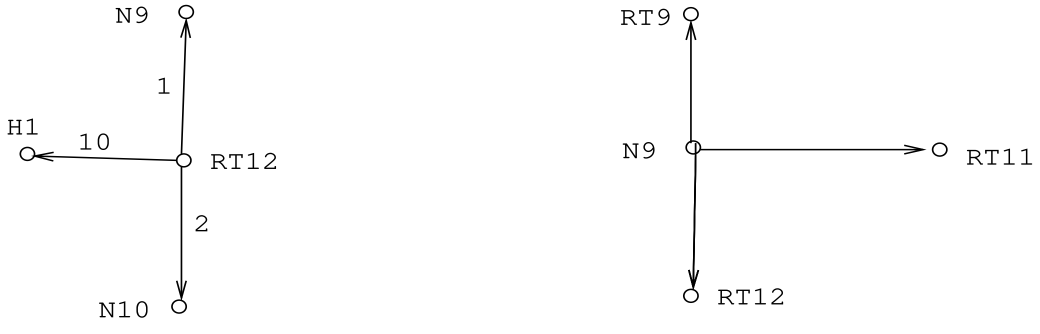


Figure 3: The resulting directed graph



A router links advertisement

A network links advertisement

Figure 4: Individual link state components

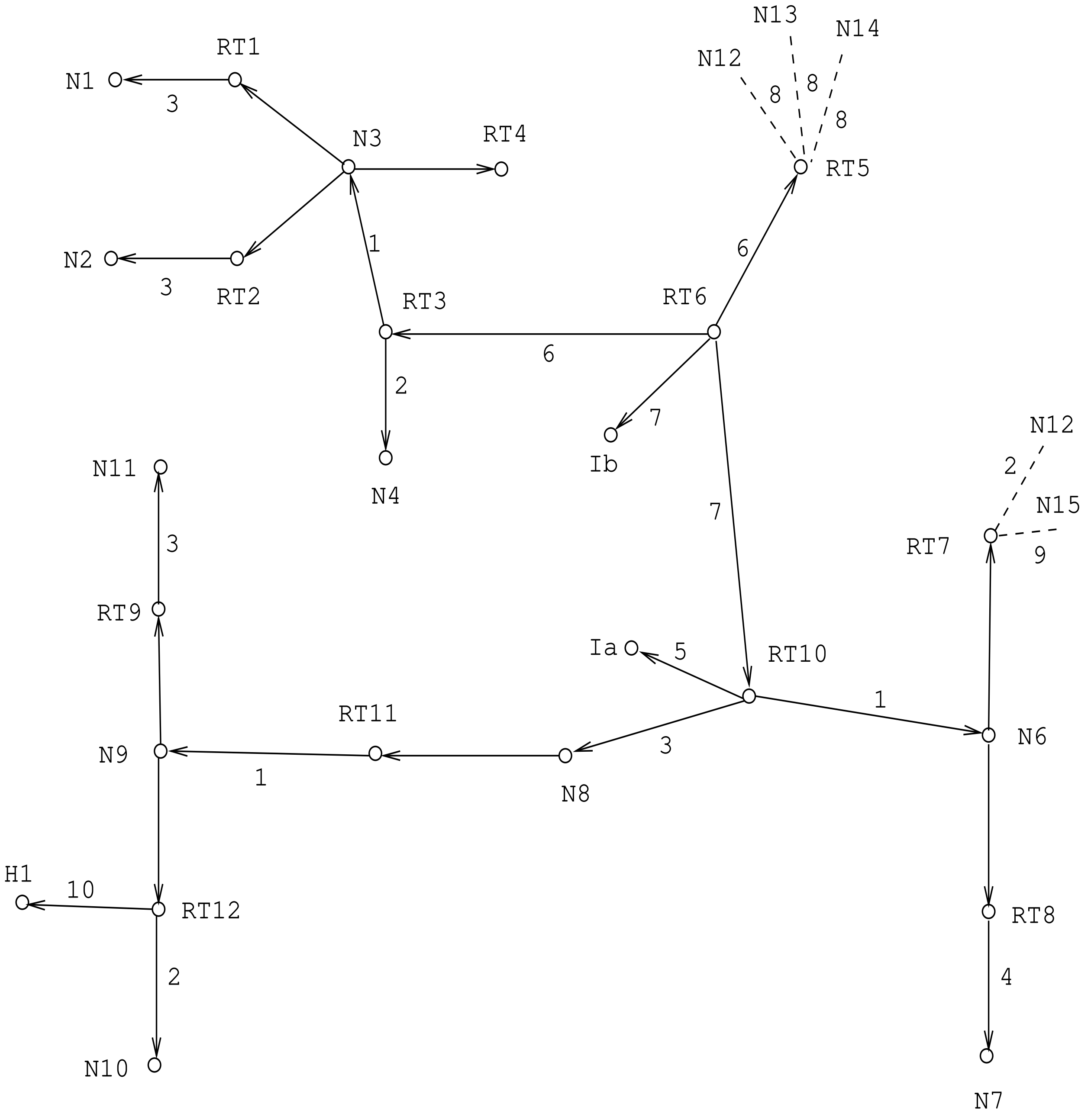


Figure 5: The SPF tree for router RT6

Destination	*Next Hop*	*Distance*
N1	RT3	10
N2	RT3	10
N3	RT3	7
N4	RT3	8
Ib	*	7
Ia	RT10	12
N6	RT10	8
N7	RT10	12
N8	RT10	10
N9	RT10	11
N10	RT10	13
N11	RT10	14
H1	RT10	21
RT5	RT5	6
RT7	RT10	8

Table 2: The portion of Router RT6's routing table listing local destinations.

As an example of Type 1 external metric processing, suppose that the routers RT7 and RT5 in Figure 2 are advertising Type 1 external metrics. For each external route, the distance from Router RT6 is calculated as the sum of the external route's cost and the distance from Router RT6 to the advertising router. For every external destination, the router advertising the shortest route is discovered, and the next hop to the advertising router becomes the next hop to the destination.

Both Router RT5 and RT7 are advertising an external route to destination Network N12. Router RT7 is preferred since it is advertising N12 at a distance of 10 (8+2) to Router RT6, which is better than Router RT5's 14 (6+8). Table 3 shows the entries that are added to the routing table when external routes are examined:

Destination	*Next Hop*	*Distance*
N12	RT10	10
N13	RT5	14
N14	RT5	14
N15	RT10	17

Table 3: The portion of Router RT6's routing table listing external destinations.

Processing of Type 2 external metrics is simpler. The AS boundary router advertising the smallest external metric is chosen, regardless of the internal distance to the AS boundary router. Suppose in our example both Router RT5 and Router RT7 were advertising Type 2 external routes. Then all traffic destined for Network N12 would be forwarded to Router RT7, since $2 < 8$. When several equal-cost Type 2 routes exist, the internal distance to the advertising routers is used to break the tie.

Both Type 1 and Type 2 external metrics can be present in the AS at the same time. In that event, Type 1 external metrics always take precedence.

This section has assumed that packets destined for external destinations are always routed through the advertising AS boundary router. This is not always desirable. For example, suppose in Figure 2 there is an additional router attached to Network N6, called Router RTX. Suppose further that RTX does not participate in OSPF routing, but does exchange EGP information with the AS boundary router RT7. Then, Router RT7 would end up advertising OSPF external routes for all destinations that should be routed to RTX. An extra hop will sometimes be introduced if packets for these destinations need always be routed first to Router RT7 (the advertising router).

To deal with this situation, the OSPF protocol allows an AS boundary router to specify a “forwarding address” in its external advertisements. In the above example, Router RT7 would specify RTX’s IP address as the “forwarding address” for all those destinations whose packets should be routed directly to RTX.

The “forwarding address” has one other application. It enables routers in the Autonomous System’s interior to function as “route servers”. For example, in Figure 2 the router RT6 could become a route server, gaining external routing information through a combination of static configuration and external routing protocols. RT6 would then start advertising itself as an AS boundary router, and would originate a collection of OSPF external advertisements. In each external advertisement, Router RT6 would specify the correct Autonomous System exit point to use for the destination through appropriate setting of the advertisement’s “forwarding address” field.

2.3 Equal-cost multipath

The above discussion has been simplified by considering only a single route to any destination. In reality, if multiple equal-cost routes to a destination exist, they are all discovered and used. This requires no conceptual changes to the algorithm, and its discussion is postponed until we consider the tree-building process in more detail.

With equal cost multipath, a router potentially has several available next hops towards any given destination.

2.4 TOS-based routing

OSPF can calculate a separate set of routes for each IP Type of Service. This means that, for any destination, there can potentially be multiple routing table entries, one for each IP TOS. The IP TOS values are represented in OSPF exactly as they appear in the IP packet header.

Up to this point, all examples shown have assumed that routes do not vary on TOS. In order to differentiate routes based on TOS, separate interface costs can be configured for each TOS. For example, in Figure 2 there could be multiple costs (one for each TOS) listed for each interface. A cost for TOS 0 must always be specified.

When interface costs vary based on TOS, a separate shortest path tree is calculated for each TOS (see Section 2.1). In addition, external costs can vary based on TOS. For example, in Figure 2 Router RT7 could advertise a separate type 1 external metric for each TOS. Then, when calculating the TOS X distance to Network N15 the cost of the shortest TOS X path to RT7 would be added to the TOS X cost advertised by RT7 for Network N15 (see Section 2.2).

All OSPF implementations must be capable of calculating routes based on TOS. However, OSPF routers can be configured to route all packets on the TOS 0 path (see Appendix C), eliminating the need to calculate non-zero TOS paths. This can be used to conserve routing table space and processing resources in the router. These TOS-0-only routers can be mixed with routers that do route based on TOS. TOS-0-only routers will be avoided as much as possible when forwarding traffic requesting a non-zero TOS.

It may be the case that no path exists for some non-zero TOS, even if the router is calculating non-zero TOS paths. In that case, packets requesting that non-zero TOS are routed along the TOS 0 path (see Section 11.1).

3 Splitting the AS into Areas

OSPF allows collections of contiguous networks and hosts to be grouped together. Such a group, together with the routers having interfaces to any one of the included networks, is called an area. Each area runs a separate copy of the basic link-state routing algorithm. This means that each area has its own topological database and corresponding graph, as explained in the previous section.

The topology of an area is invisible from the outside of the area. Conversely, routers internal to a given area know nothing of the detailed topology external to the area. This isolation of knowledge enables the protocol to effect a marked reduction in routing traffic as compared to treating the entire Autonomous System as a single link-state domain.

With the introduction of areas, it is no longer true that all routers in the AS have an identical topological database. A router actually has a separate topological database for each area it is connected to. (Routers connected to multiple areas are called area border routers). Two routers belonging to the same area have, for that area, identical area topological databases.

Routing in the Autonomous System takes place on two levels, depending on whether the source and destination of a packet reside in the same area (intra-area routing is used) or different areas (inter-area routing is used). In intra-area routing, the packet is routed solely on information obtained within the area; no routing information obtained from outside the area can be used. This protects intra-area routing from the injection of bad routing information. We discuss inter-area routing in Section 3.2.

3.1 The backbone of the Autonomous System

The backbone consists of those networks not contained in any area, their attached routers, and those routers that belong to multiple areas. The backbone must be contiguous.

It is possible to define areas in such a way that the backbone is no longer contiguous. In this case the system administrator must restore backbone connectivity by configuring virtual links.

Virtual links can be configured between any two backbone routers that have an interface to a common non-backbone area. Virtual links belong to the backbone. The protocol treats two routers joined by a virtual link as if they were connected by an unnumbered point-to-point network. On the graph of the backbone, two such routers are joined by arcs whose costs are the intra-area distances between the two routers. The routing protocol traffic that flows along the virtual link uses intra-area routing only.

The backbone is responsible for distributing routing information between areas. The backbone itself has all of the properties of an area. The topology of the backbone is invisible to each of the areas, while the backbone itself knows nothing of the topology of the areas.

3.2 Inter-area routing

When routing a packet between two areas the backbone is used. The path that the packet will travel can be broken up into three contiguous pieces: an intra-area path from the source to an area border router, a backbone path between the source and destination areas, and then another intra-area path to the destination. The algorithm finds the set of such paths that have the smallest cost.

Looking at this another way, inter-area routing can be pictured as forcing a star configuration on the Autonomous System, with the backbone as hub and each of the areas as spokes.

The topology of the backbone dictates the backbone paths used between areas. The topology of the backbone can be enhanced by adding virtual links. This gives the system administrator some control over the routes taken by inter-area traffic.

The correct area border router to use as the packet exits the source area is chosen in exactly the same way routers advertising external routes are chosen. Each area border router in an area summarizes for the area its cost to all networks external to the area. After the SPF tree is calculated for the area, routes to all other networks are calculated by examining the summaries of the area border routers.

3.3 Classification of routers

Before the introduction of areas, the only OSPF routers having a specialized function were those advertising external routing information, such as Router RT5 in Figure 2. When the AS is split into OSPF areas, the routers are further divided according to function into the following four overlapping categories:

Internal routers A router with all directly connected networks belonging to the same area. Routers with only backbone interfaces also belong to this category. These routers run a single copy of the basic routing algorithm.

Area border routers A router that attaches to multiple areas. Area border routers run multiple copies of the basic algorithm, one copy for each attached area and an additional copy for the backbone. Area border routers condense the topological information of their attached areas for distribution to the backbone. The backbone in turn distributes the information to the other areas.

Backbone routers A router that has an interface to the backbone. This includes all routers that interface to more than one area (i.e., area border routers). However, backbone routers do not have to be area border routers. Routers with all interfaces connected to the backbone are considered to be internal routers.

AS boundary routers A router that exchanges routing information with routers belonging to other Autonomous Systems. Such a router has AS external routes that are advertised throughout the Autonomous System. The path to each AS boundary router is known by every router in the AS. This classification is completely independent of the previous classifications: AS boundary routers may be internal or area border routers, and may or may not participate in the backbone.

3.4 A sample area configuration

Figure 6 shows a sample area configuration. The first area consists of networks N1-N4, along with their attached routers RT1-RT4. The second area consists of networks N6-N8, along with their attached routers RT7, RT8, RT10 and RT11. The third area consists of networks N9-N11 and Host H1, along with their attached routers RT9, RT11 and RT12. The third area has been configured so that networks N9-N11 and Host H1 will all be grouped into a single route, when advertised external to the area (see Section 3.5 for more details).

In Figure 6, Routers RT1, RT2, RT5, RT6, RT8, RT9 and RT12 are internal routers. Routers RT3, RT4, RT7, RT10 and RT11 are area border routers. Finally, as before, Routers RT5 and RT7 are AS boundary routers.

Figure 7 shows the resulting topological database for the Area 1. The figure completely describes that area's intra-area routing. It also shows the complete view of the internet for the two internal routers RT1 and RT2. It is the job of the area border routers, RT3 and RT4, to advertise into Area 1 the distances to all destinations external to the area. These are indicated in Figure 7 by the dashed stub routes. Also, RT3 and RT4 must advertise into Area 1 the location of the AS boundary routers RT5 and RT7. Finally, external advertisements from RT5 and RT7 are flooded throughout the entire AS, and in particular throughout Area 1. These advertisements are included in Area 1's database, and yield routes to networks N12-N15.

Routers RT3 and RT4 must also summarize Area 1's topology for distribution to the backbone. Their backbone advertisements are shown in Table 4. These summaries show which networks are contained in Area 1 (i.e., Networks N1-N4), and the distance to these networks from the routers RT3 and RT4 respectively.

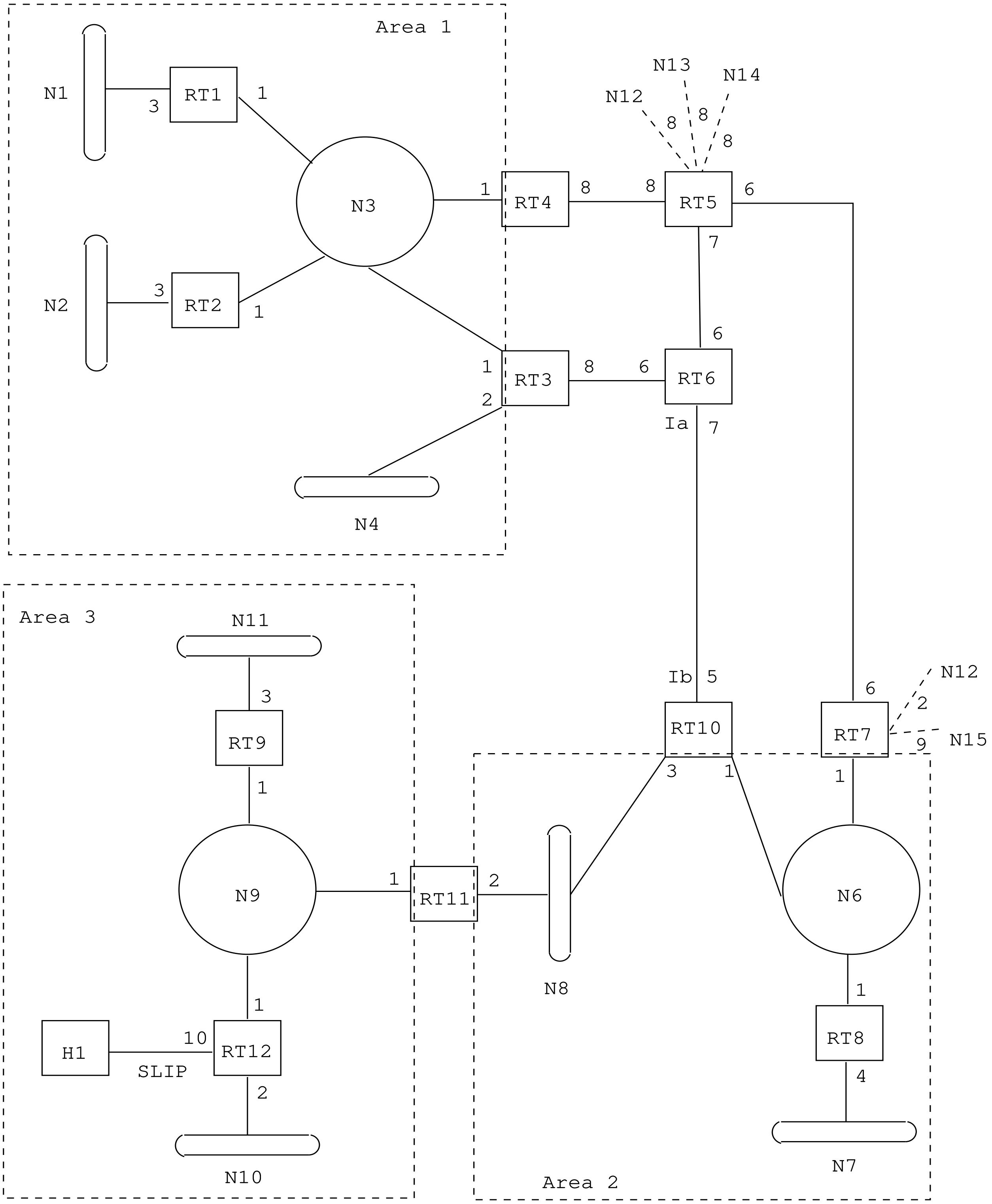


Figure 6: A sample OSPF area configuration

Network	*RT3 adv.*	*RT4 adv.*
N1	4	4
N2	4	4
N3	1	1
N4	2	3

Table 4: Networks advertised to the backbone by routers RT3 and RT4.

The topological database for the backbone is shown in Figure 8. The set of routers pictured are the backbone routers. Router RT11 is a backbone router because it belongs to two areas. In order to make the backbone connected, a virtual link has been configured between routers R10 and R11.

Again, Routers RT3, RT4, RT7, RT10 and RT11 are area border routers. As Routers RT3 and RT4 did above, they have condensed the routing information of their attached areas for distribution via the backbone; these are the dashed stubs that appear in Figure 8. Remember that the third area has been configured to condense Networks N9-N11 and Host H1 into a single route. This yields a single dashed line for networks N9-N11 and Host H1 in Figure 8. Routers RT5 and RT7 are AS boundary routers; their externally derived information also appears on the graph in Figure 8 as stubs.

The backbone enables the exchange of summary information between area border routers. Every area border router hears the area summaries from all other area border routers. It then forms a picture of the distance to all networks outside of its area by examining the collected advertisements, and adding in the backbone distance to each advertising router.

Again using Routers RT3 and RT4 as an example, the procedure goes as follows: They first calculate the SPF tree for the backbone. This gives the distances to all other area border routers. Also noted are the distances to networks (Ia and Ib) and AS boundary routers (RT5 and RT7) that belong to the backbone. This calculation is shown in Table 5.

Area border router	*dist from RT3*	*dist from RT4*
to RT3	*	21
to RT4	22	*
to RT7	20	14
to RT10	15	22
to RT11	18	25
to Ia	20	27
to Ib	15	22
to RT5	14	8
to RT7	20	14

Table 5: Backbone distances calculated by Routers RT3 and RT4.

Next, by looking at the area summaries from these area border routers, RT3 and RT4 can determine the distance to all networks outside their area. These distances are then advertised internally to the area by RT3 and RT4. The

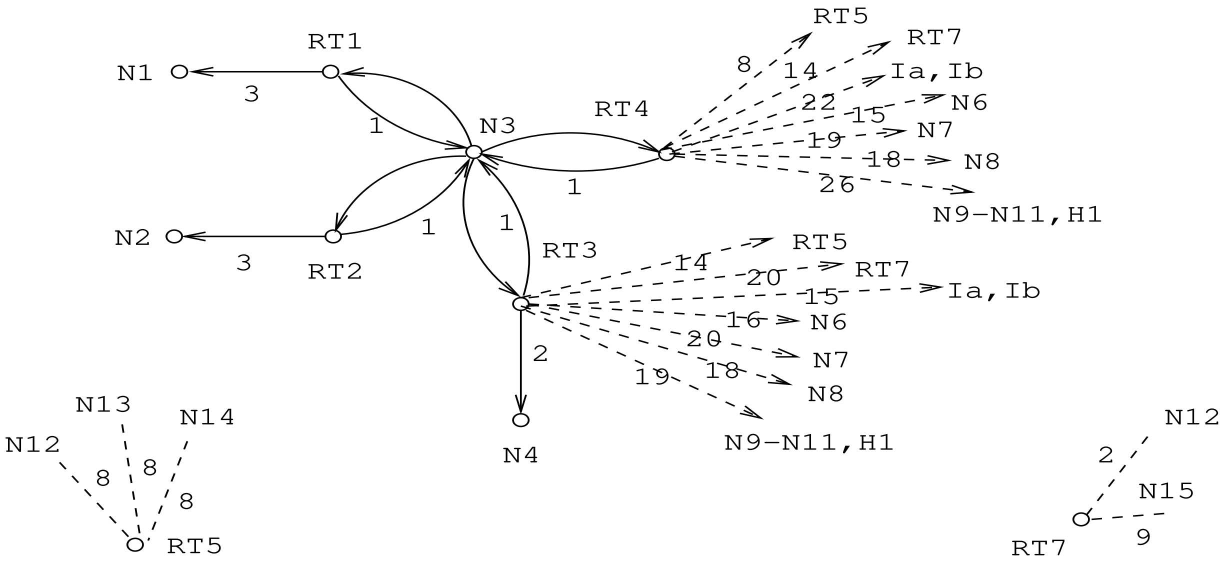


Figure 7: Area 1's Database

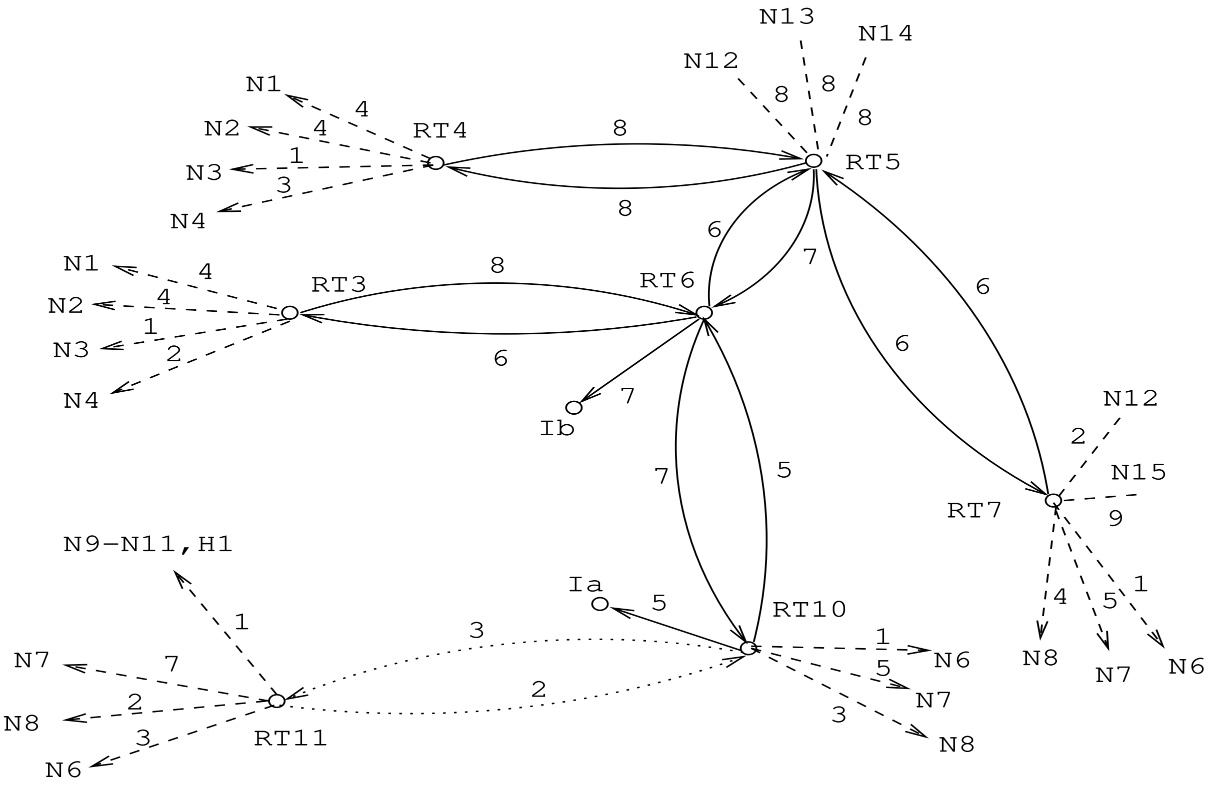


Figure 8: The backbone database

advertisements that Router RT3 and RT4 will make into Area 1 are shown in Table 6. Note that Table 6 assumes that an area range has been configured for the backbone which groups Ia and Ib into a single advertisement.

Destination	*RT3 adv.*	*RT4 adv.*
Ia,Ib	15	22
N6	16	15
N7	20	19
N8	18	18
N9-N11,H1	19	26
RT5	14	8
RT7	20	14

Table 6: Destinations advertised into Area 1 by Routers RT3 and RT4.

The information imported into Area 1 by Routers RT3 and RT4 enables an internal router, such as RT1, to choose an area border router intelligently. Router RT1 would use RT4 for traffic to Network N6, RT3 for traffic to Network N10, and would load share between the two for traffic to Network N8.

Router RT1 can also determine in this manner the shortest path to the AS boundary routers RT5 and RT7. Then, by looking at RT5 and RT7’s external advertisements, Router RT1 can decide between RT5 or RT7 when sending to a destination in another Autonomous System (one of the networks N12-N15).

Note that a failure of the line between Routers RT6 and RT10 will cause the backbone to become disconnected. Configuring a virtual link between Routers RT7 and RT10 will give the backbone more connectivity and more resistance to such failures. Also, a virtual link between RT7 and RT10 would allow a much shorter path between the third area (containing N9) and the router RT7, which is advertising a good route to external Network N12.

3.5 IP subnetting support

OSPF attaches an IP address mask to each advertised route. The mask indicates the range of addresses being described by the particular route. For example, a summary advertisement for the destination `128.185.0.0` with a mask of `0xffff0000` actually is describing a single route to the collection of destinations `128.185.0.0 - 128.185.255.255`. Similarly, host routes are always advertised with a mask of `0xffffffff`, indicating the presence of only a single destination.

Including the mask with each advertised destination enables the implementation of what is commonly referred to as variable-length subnetting. This means that a single IP class A, B, or C network number can be broken up into many subnets of various sizes. For example, the network `128.185.0.0` could be broken up into 62 variable-sized subnets: 15 subnets of size 4K, 15 subnets of size 256, and 32 subnets of size 8. Table 7 shows some of the resulting network addresses together with their masks:

There are many possible ways of dividing up a class A, B, and C network into variable sized subnets. The precise procedure for doing so is beyond the scope of this specification. This specification however establishes the following guideline: When an IP packet is forwarded, it is always forwarded to the network that is the best match for the packet’s destination. Here best match is synonymous with the longest or most specific match. For example, the default route with destination of `0.0.0.0` and mask `0x00000000` is always a match for every IP destination. Yet it is always less specific than any other match. Subnet masks must be assigned so that the best match for any IP destination is unambiguous.

Network address	*IP address mask*	*Subnet size*
`128.185.16.0`	`0xfffff000`	4K
`128.185.1.0`	`0xffffff00`	256
`128.185.0.8`	`0xfffffff8`	8

Table 7: Some sample subnet sizes.

The OSPF area concept is modelled after an IP subnetted network. OSPF areas have been loosely defined to be a collection of networks. In actuality, an OSPF area is specified to be a list of address ranges (see Section C.2 for more details). Each address range is defined as an [address,mask] pair. Many separate networks may then be contained in a single address range, just as a subnetted network is composed of many separate subnets. Area border routers then summarize the area contents (for distribution to the backbone) by advertising a single route for each address range. The cost of the route is the minimum cost to any of the networks falling in the specified range.

For example, an IP subnetted network can be configured as a single OSPF area. In that case, the area would be defined as a single address range: a class A, B, or C network number along with its natural IP mask. Inside the area, any number of variable sized subnets could be defined. External to the area, a single route for the entire subnetted network would be distributed, hiding even the fact that the network is subnetted at all. The cost of this route is the minimum of the set of costs to the component subnets.

3.6 Supporting stub areas

In some Autonomous Systems, the majority of the topological database may consist of AS external advertisements. An OSPF AS external advertisement is usually flooded throughout the entire AS. However, OSPF allows certain areas to be configured as “stub areas”. AS external advertisements are not flooded into/throughout stub areas; routing to AS external destinations in these areas is based on a (per-area) default only. This reduces the topological database size, and therefore the memory requirements, for a stub area’s internal routers.

In order to take advantage of the OSPF stub area support, default routing must be used in the stub area. This is accomplished as follows. One or more of the stub area’s area border routers must advertise a default route into the stub area via summary link advertisements. These summary defaults are flooded throughout the stub area, but no further. (For this reason these defaults pertain only to the particular stub area). These summary default routes will match any destination that is not explicitly reachable by an intra-area or inter-area path (i.e., AS external destinations).

An area can be configured as stub when there is a single exit point from the area, or when the choice of exit point need not be made on a per-external-destination basis. For example, Area 3 in Figure 6 could be configured as a stub area, because all external traffic must travel though its single area border router RT11. If Area 3 were configured as a stub, Router RT11 would advertise a default route for distribution inside Area 3 (in a summary link advertisement), instead of flooding the AS external advertisements for networks N12-N15 into/throughout the area.

The OSPF protocol ensures that all routers belonging to an area agree on whether the area has been configured as a stub. This guarantees that no confusion will arise in the flooding of AS external advertisements.

There are a couple of restrictions on the use of stub areas. Virtual links cannot be configured through stub areas. In addition, AS boundary routers cannot be placed internal to stub areas.

3.7 Partitions of areas

OSPF does not actively attempt to repair area partitions. When an area becomes partitioned, each component simply becomes a separate area. The backbone then performs routing between the new areas. Some destinations reachable via intra-area routing before the partition will now require inter-area routing.

In the previous section, an area was described as a list of address ranges. Any particular address range must still be completely contained in a single component of the area partition. This has to do with the way the area contents are summarized to the backbone. Also, the backbone itself must not partition. If it does, parts of the Autonomous System will become unreachable. Backbone partitions can be repaired by configuring virtual links (see Section 15).

Another way to think about area partitions is to look at the Autonomous System graph that was introduced in Section 2. Area IDs can be viewed as colors for the graph's edges.[1] Each edge of the graph connects to a network, or is itself a point-to-point network. In either case, the edge is colored with the network's Area ID.

A group of edges, all having the same color, and interconnected by vertices, represents an area. If the topology of the Autonomous System is intact, the graph will have several regions of color, each color being a distinct Area ID.

When the AS topology changes, one of the areas may become partitioned. The graph of the AS will then have multiple regions of the same color (Area ID). The routing in the Autonomous System will continue to function as long as these regions of same color are connected by the single backbone region.

---

[1] The graph's vertices represent either routers, transit networks, or stub networks. Since routers may belong to multiple areas, it is not possible to color the graph's vertices.

4 Functional Summary

A separate copy of OSPF's basic routing algorithm runs in each area. Routers having interfaces to multiple areas run multiple copies of the algorithm. A brief summary of the routing algorithm follows.

When a router starts, it first initializes the routing protocol data structures. The router then waits for indications from the lower-level protocols that its interfaces are functional.

A router then uses the OSPF's Hello Protocol to acquire neighbors. The router sends Hello packets to its neighbors, and in turn receives their Hello packets. On broadcast and point-to-point networks, the router dynamically detects its neighboring routers by sending its Hello packets to the multicast address AllSPFRouters. On non-broadcast networks, some configuration information is necessary in order to discover neighbors. On all multi-access networks (broadcast or non-broadcast), the Hello Protocol also elects a Designated router for the network.

The router will attempt to form adjacencies with some of its newly acquired neighbors. Topological databases are synchronized between pairs of adjacent routers. On multi-access networks, the Designated Router determines which routers should become adjacent.

Adjacencies control the distribution of routing protocol packets. Routing protocol packets are sent and received only on adjacencies. In particular, distribution of topological database updates proceeds along adjacencies.

A router periodically advertises its state, which is also called link state. Link state is also advertised when a router's state changes. A router's adjacencies are reflected in the contents of its link state advertisements. This relationship between adjacencies and link state allows the protocol to detect dead routers in a timely fashion.

Link state advertisements are flooded throughout the area. The flooding algorithm is reliable, ensuring that all routers in an area have exactly the same topological database. This database consists of the collection of link state advertisements received from each router belonging to the area. From this database each router calculates a shortest-path tree, with itself as root. This shortest-path tree in turn yields a routing table for the protocol.

4.1 Inter-area routing

The previous section described the operation of the protocol within a single area. For intra-area routing, no other routing information is pertinent. In order to be able to route to destinations outside of the area, the area border routers inject additional routing information into the area. This additional information is a distillation of the rest of the Autonomous System's topology.

This distillation is accomplished as follows: Each area border router is by definition connected to the backbone. Each area border router summarizes the topology of its attached areas for transmission on the backbone, and hence to all other area border routers. An area border router then has complete topological information concerning the backbone, and the area summaries from each of the other area border routers. From this information, the router calculates paths to all destinations not contained in its attached areas. The router then advertises these paths into its attached areas. This enables the area's internal routers to pick the best exit router when forwarding traffic to destinations in other areas.

4.2 AS external routes

Routers that have information regarding other Autonomous Systems can flood this information throughout the AS. This external routing information is distributed verbatim to every participating router. There is one exception: external routing information is not flooded into "stub" areas (see Section 3.6).

To utilize external routing information, the path to all routers advertising external information must be known throughout the AS (excepting the stub areas). For that reason, the locations of these AS boundary routers are summarized by the (non-stub) area border routers.

4.3 Routing protocol packets

The OSPF protocol runs directly over IP, using IP protocol 89. OSPF does not provide any explicit fragmentation/reassembly support. When fragmentation is necessary, IP fragmentation/reassembly is used. OSPF protocol packets have been designed so that large protocol packets can generally be split into several smaller protocol packets. This practice is recommended; IP fragmentation should be avoided whenever possible.

Routing protocol packets should always be sent with the IP TOS field set to 0. If at all possible, routing protocol packets should be given preference over regular IP data traffic, both when being sent and received. As an aid to accomplishing this, OSPF protocol packets should have their IP precedence field set to the value Internetwork Control (see [RFC 791]).

All OSPF protocol packets share a common protocol header that is described in Appendix A. The OSPF packet types are listed below in Table 8. Their formats are also described in Appendix A.

Type	*Packet name*	*Protocol function*
1	Hello	Discover/maintain neighbors
2	Database Description	Summarize database contents
3	Link State Request	Database download
4	Link State Update	Database update
5	Link State Ack	Flooding acknowledgment

Table 8: OSPF packet types.

OSPF's Hello protocol uses Hello packets to discover and maintain neighbor relationships. The Database Description and Link State Request packets are used in the forming of adjacencies. OSPF's reliable update mechanism is implemented by the Link State Update and Link State Acknowledgment packets.

Each Link State Update packet carries a set of new link state advertisements one hop further away from their point of origination. A single Link State Update packet may contain the link state advertisements of several routers. Each advertisement is tagged with the ID of the originating router and a checksum of its link state contents. The five different types of OSPF link state advertisements are listed below in Table 9.

As mentioned above, OSPF routing packets (with the exception of Hellos) are sent only over adjacencies. Note that this means that all OSPF protocol packets travel a single IP hop, except those that are sent over virtual adjacencies. The IP source address of an OSPF protocol packet is one end of a router adjacency, and the IP destination address is either the other end of the adjacency or an IP multicast address.

4.4 Basic implementation requirements

An implementation of OSPF requires the following pieces of system support:

Timers Two different kind of timers are required. The first kind, called single shot timers, fire once and cause a protocol event to be processed. The second kind, called interval timers, fire at continuous intervals. These are used for the sending of packets at regular intervals. A good example of this is the regular broadcast of Hello packets (on broadcast networks). The granularity of both kinds of timers is one second.

Interval timers should be implemented to avoid drift. In some router implementations, packet processing can affect timer execution. When multiple routers are attached to a single network, all doing broadcasts, this can lead to the synchronization of routing packets (which should be avoided). If timers cannot be implemented to avoid drift, small random amounts should be added to/subtracted from the timer interval at each firing.

LS type	*Advertisement name*	*Advertisement description*
1	Router links advertisements	Originated by all routers. This advertisement describes the collected states of the router's interfaces to an area. Flooded throughout a single area only.
2	Network links advertisements	Originated for multi-access networks by the Designated Router. This advertisement contains the list of routers connected to the network. Flooded throughout a single area only.
3,4	Summary link advertisements	Originated by area border routers, and flooded throughout the advertisement's associated area. Each summary link advertisement describes a route to a destination outside the area, yet still inside the AS (i.e., an inter-area route). Type 3 advertisements describe routes to networks. Type 4 advertisements describe routes to AS boundary routers.
5	AS external link advertisements	Originated by AS boundary routers, and flooded throughout the AS. Each AS external link advertisement describes a route to a destination in another Autonomous System. Default routes for the AS can also be described by AS external link advertisements.

Table 9: OSPF link state advertisements.

IP multicast Certain OSPF packets take the form of IP multicast datagrams. Support for receiving and sending IP multicast datagrams, along with the appropriate lower-level protocol support, is required. The IP multicast datagrams used by OSPF never travel more than one hop. For this reason, the ability to forward IP multicast datagrams is not required. For information on IP multicast, see [RFC 1112].

Variable-length subnet support The router's IP protocol support must include the ability to divide a single IP class A, B, or C network number into many subnets of various sizes. This is commonly called variable-length subnetting; see Section 3.5 for details.

IP supernetting support The router's IP protocol support must include the ability to aggregate contiguous collections of IP class A, B, and C networks into larger quantities called supernets. Supernetting has been proposed as one way to improve the scaling of IP routing in the worldwide Internet. For more information on IP supernetting, see [RFC 1519].

Lower-level protocol support The lower level protocols referred to here are the network access protocols, such as the Ethernet data link layer. Indications must be passed from these protocols to OSPF as the network interface goes up and down. For example, on an ethernet it would be valuable to know when the ethernet transceiver cable becomes unplugged.

Non-broadcast lower-level protocol support Remember that non-broadcast networks are multi-access networks such as a X.25 PDN. On these networks, the Hello Protocol can be aided by providing an indication to OSPF when an attempt is made to send a packet to a dead or non-existent router. For example, on an X.25 PDN a dead neighboring router may be indicated by the reception of a X.25 clear with an appropriate cause and diagnostic, and this information would be passed to OSPF.

List manipulation primitives Much of the OSPF functionality is described in terms of its operation on lists of link state advertisements. For example, the collection of advertisements that will be retransmitted to an adjacent router until acknowledged are described as a list. Any particular advertisement may be on many such lists. An OSPF implementation needs to be able to manipulate these lists, adding and deleting constituent advertisements as necessary.

Tasking support Certain procedures described in this specification invoke other procedures. At times, these other procedures should be executed in-line, that is, before the current procedure is finished. This is indicated in the text by instructions to *execute* a procedure. At other times, the other procedures are to be executed only when the current procedure has finished. This is indicated by instructions to *schedule* a task.

4.5 Optional OSPF capabilities

The OSPF protocol defines several optional capabilities. A router indicates the optional capabilities that it supports in its OSPF Hello packets, Database Description packets and in its link state advertisements. This enables routers supporting a mix of optional capabilities to coexist in a single Autonomous System.

Some capabilities must be supported by all routers attached to a specific area. In this case, a router will not accept a neighbor's Hello Packet unless there is a match in reported capabilities (i.e., a capability mismatch prevents a neighbor relationship from forming). An example of this is the ExternalRoutingCapability (see below).

Other capabilities can be negotiated during the Database Exchange process. This is accomplished by specifying the optional capabilities in Database Description packets. A capability mismatch with a neighbor in this case will result in only a subset of link state advertisements being exchanged between the two neighbors.

The routing table build process can also be affected by the presence/absence of optional capabilities. For example, since the optional capabilities are reported in link state advertisements, routers incapable of certain functions can be avoided when building the shortest path tree. An example of this is the TOS routing capability (see below).

The current OSPF optional capabilities are listed below. See Section A.2 for more information.

ExternalRoutingCapability Entire OSPF areas can be configured as "stubs" (see Section 3.6). AS external advertisements will not be flooded into stub areas. This capability is represented by the **E-bit** in the OSPF options field (see Section A.2). In order to ensure consistent configuration of stub areas, all routers interfacing to such an area must have the **E-bit** clear in their Hello packets (see Sections 9.5 and 10.5).

TOS capability All OSPF implementations must be able to calculate separate routes based on IP Type of Service. However, to save routing table space and processing resources, an OSPF router can be configured to ignore TOS when forwarding packets. In this case, the router calculates routes for TOS 0 only. This capability is represented by the **T-bit** in the OSPF options field (see Section A.2). TOS-capable routers will attempt to avoid non-TOS-capable routers when calculating non-zero TOS paths.

5 Protocol Data Structures

The OSPF protocol is described in this specification in terms of its operation on various protocol data structures. The following list comprises the top-level OSPF data structures. Any initialization that needs to be done is noted. OSPF areas, interfaces and neighbors also have associated data structures that are described later in this specification.

Router ID A 32-bit number that uniquely identifies this router in the AS. One possible implementation strategy would be to use the smallest IP interface address belonging to the router. If a router's OSPF Router ID is changed, the router's OSPF software should be restarted before the new Router ID takes effect. Before restarting in order to change its Router ID, the router should flush its self-originated link state advertisements from the routing domain (see Section 14.1), or they will persist for up to MaxAge minutes.

Area structures Each one of the areas to which the router is connected has its own data structure. This data structure describes the working of the basic algorithm. Remember that each area runs a separate copy of the basic algorithm.

Backbone (area) structure The basic algorithm operates on the backbone as if it were an area. For this reason the backbone is represented as an area structure.

Virtual links configured The virtual links configured with this router as one endpoint. In order to have configured virtual links, the router itself must be an area border router. Virtual links are identified by the Router ID of the other endpoint – which is another area border router. These two endpoint routers must be attached to a common area, called the virtual link's Transit area. Virtual links are part of the backbone, and behave as if they were unnumbered point-to-point networks between the two routers. A virtual link uses the intra-area routing of its Transit area to forward packets. Virtual links are brought up and down through the building of the shortest-path trees for the Transit area.

List of external routes These are routes to destinations external to the Autonomous System, that have been gained either through direct experience with another routing protocol (such as EGP), or through configuration information, or through a combination of the two (e.g., dynamic external information to be advertised by OSPF with configured metric). Any router having these external routes is called an AS boundary router. These routes are advertised by the router into the OSPF routing domain via AS external link advertisements.

List of AS external link advertisements Part of the topological database. These have originated from the AS boundary routers. They comprise routes to destinations external to the Autonomous System. Note that, if the router is itself an AS boundary router, some of these AS external link advertisements have been self-originated.

The routing table Derived from the topological database. Each destination that the router can forward to is represented by a cost and a set of paths. A path is described by its type and next hop. For more information, see Section 11.

TOS capability This item indicates whether the router will calculate separate routes based on TOS. This is a configurable parameter. For more information, see Sections 4.5 and 16.9.

Figure 9 shows the collection of data structures present in a typical router. The router pictured is RT10, from the map in Figure 6. Note that Router RT10 has a virtual link configured to Router RT11, with Area 2 as the link's Transit area. This is indicated by the dashed line in Figure 9. When the virtual link becomes active, through the building of the shortest path tree for Area 2, it becomes an interface to the backbone (see the two backbone interfaces depicted in Figure 9).

6 The Area Data Structure

The area data structure contains all the information used to run the basic routing algorithm. Each area maintains its own topological database. A network belongs to a single area, and a router interface connects to a single area. Each router adjacency also belongs to a single area.

The OSPF backbone has all the properties of an area. For that reason it is also represented by an area data structure. Note that some items in the structure apply differently to the backbone than to non-backbone areas.

The area topological (or link state) database consists of the collection of router links, network links and summary link advertisements that have originated from the area's routers. This information is flooded throughout a single area only. The list of AS external link advertisements (see Section 5) is also considered to be part of each area's topological database.

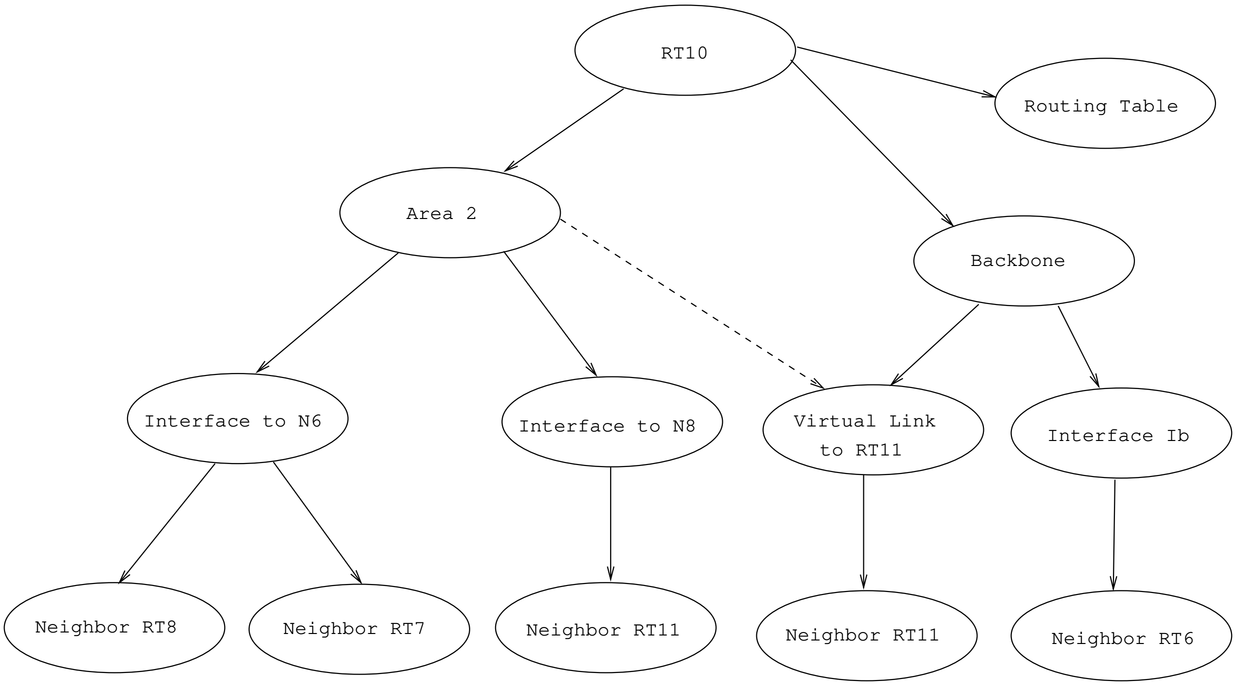


Figure 9: Router RT10's Data Structures

Area ID A 32-bit number identifying the area. `0.0.0.0` is reserved for the Area ID of the backbone. If assigning subnetted networks as separate areas, the IP network number could be used as the Area ID.

List of component address ranges The address ranges that define the area. Each address range is specified by an [address,mask] pair and a status indication of either Advertise or DoNotAdvertise (see Section 12.4.3). Each network is then assigned to an area depending on the address range that it falls into (specified address ranges are not allowed to overlap). As an example, if an IP subnetted network is to be its own separate OSPF area, the area is defined to consist of a single address range - an IP network number with its natural (class A, B or C) mask.

Associated router interfaces This router's interfaces connecting to the area. A router interface belongs to one and only one area (or the backbone). For the backbone structure this list includes all the virtual links. A virtual link is identified by the Router ID of its other endpoint; its cost is the cost of the shortest intra-area path through the Transit area that exists between the two routers.

List of router links advertisements A router links advertisement is generated by each router in the area. It describes the state of the router's interfaces to the area.

List of network links advertisements One network links advertisement is generated for each transit multi-access network in the area. A network links advertisement describes the set of routers currently connected to the network.

List of summary link advertisements Summary link advertisements originate from the area's area border routers. They describe routes to destinations internal to the Autonomous System, yet external to the area.

Shortest-path tree The shortest-path tree for the area, with this router itself as root. Derived from the collected router links and network links advertisements by the Dijkstra algorithm (see Section 16.1).

AuType The type of authentication used for this area. Authentication types are defined in Appendix D. All OSPF packet exchanges are authenticated. Different authentication schemes may be used in different areas.

TransitCapability Set to TRUE if and only if there are one or more active virtual links using the area as a Transit area. Equivalently, this parameter indicates whether the area can carry data traffic that neither originates nor terminates in the area itself. This parameter is calculated when the area's shortest-path tree is built (see Section 16.1), and is used as an input to a subsequent step of the routing table build process (see Section 16.3).

ExternalRoutingCapability Whether AS external advertisements will be flooded into/throughout the area. This is a configurable parameter. If AS external advertisements are excluded from the area, the area is called a "stub". Internal to stub areas, routing to AS external destinations will be based solely on a default summary route. The backbone cannot be configured as a stub area. Also, virtual links cannot be configured through stub areas. For more information, see Section 3.6.

StubDefaultCost If the area has been configured as a stub area, and the router itself is an area border router, then the StubDefaultCost indicates the cost of the default summary link that the router should advertise into the area. There can be a separate cost configured for each IP TOS. See Section 12.4.3 for more information.

Unless otherwise specified, the remaining sections of this document refer to the operation of the protocol in a single area.

7 Bringing Up Adjacencies

OSPF creates adjacencies between neighboring routers for the purpose of exchanging routing information. Not every two neighboring routers will become adjacent. This section covers the generalities involved in creating adjacencies. For further details consult Section 10.

7.1 The Hello Protocol

The Hello Protocol is responsible for establishing and maintaining neighbor relationships. It also ensures that communication between neighbors is bidirectional. Hello packets are sent periodically out all router interfaces. Bidirectional communication is indicated when the router sees itself listed in the neighbor's Hello Packet.

On multi-access networks, the Hello Protocol elects a Designated Router for the network. Among other things, the Designated Router controls what adjacencies will be formed over the network (see below).

The Hello Protocol works differently on broadcast networks, as compared to non-broadcast networks. On broadcast networks, each router advertises itself by periodically multicasting Hello Packets. This allows neighbors to be discovered dynamically. These Hello Packets contain the router's view of the Designated Router's identity, and the list of routers whose Hello Packets have been seen recently.

On non-broadcast networks some configuration information is necessary for the operation of the Hello Protocol. Each router that may potentially become Designated Router has a list of all other routers attached to the network. A router, having Designated Router potential, sends Hello Packets to all other potential Designated Routers when its interface to the non-broadcast network first becomes operational. This is an attempt to find the Designated Router for the network. If the router itself is elected Designated Router, it begins sending Hello Packets to all other routers attached to the network.

After a neighbor has been discovered, bidirectional communication ensured, and (if on a multi-access network) a Designated Router elected, a decision is made regarding whether or not an adjacency should be formed with the neighbor (see Section 10.4). An attempt is always made to establish adjacencies over point-to-point networks and virtual links. The first step in bringing up an adjacency is to synchronize the neighbors' topological databases. This is covered in the next section.

7.2 The Synchronization of Databases

In a link-state routing algorithm, it is very important for all routers' topological databases to stay synchronized. OSPF simplifies this by requiring only adjacent routers to remain synchronized. The synchronization process begins as soon as the routers attempt to bring up the adjacency. Each router describes its database by sending a sequence of Database Description packets to its neighbor. Each Database Description Packet describes a set of link state advertisements belonging to the router's database. When the neighbor sees a link state advertisement that is more recent than its own database copy, it makes a note that this newer advertisement should be requested.

This sending and receiving of Database Description packets is called the “Database Exchange Process”. During this process, the two routers form a master/slave relationship. Each Database Description Packet has a sequence number. Database Description Packets sent by the master (polls) are acknowledged by the slave through echoing of the sequence number. Both polls and their responses contain summaries of link state data. The master is the only one allowed to retransmit Database Description Packets. It does so only at fixed intervals, the length of which is the configured constant RxmtInterval.

Each Database Description contains an indication that there are more packets to follow — the M-bit. The Database Exchange Process is over when a router has received and sent Database Description Packets with the M-bit off.

During and after the Database Exchange Process, each router has a list of those link state advertisements for which the neighbor has more up-to-date instances. These advertisements are requested in Link State Request Packets. Link State Request packets that are not satisfied are retransmitted at fixed intervals of time RxmtInterval. When the Database Description Process has completed and all Link State Requests have been satisfied, the databases are deemed synchronized and the routers are marked fully adjacent. At this time the adjacency is fully functional and is advertised in the two routers' link state advertisements.

The adjacency is used by the flooding procedure as soon as the Database Exchange Process begins. This simplifies database synchronization, and guarantees that it finishes in a predictable period of time.

7.3 The Designated Router

Every multi-access network has a Designated Router. The Designated Router performs two main functions for the routing protocol:

- The Designated Router originates a network links advertisement on behalf of the network. This advertisement lists the set of routers (including the Designated Router itself) currently attached to the network. The Link State ID for this advertisement (see Section 12.1.4) is the IP interface address of the Designated Router. The IP network number can then be obtained by using the subnet/network mask.
- The Designated Router becomes adjacent to all other routers on the network. Since the link state databases are synchronized across adjacencies (through adjacency bring-up and then the flooding procedure), the Designated Router plays a central part in the synchronization process.

The Designated Router is elected by the Hello Protocol. A router's Hello Packet contains its Router Priority, which is configurable on a per-interface basis. In general, when a router's interface to a network first becomes functional, it checks to see whether there is currently a Designated Router for the network. If there is, it accepts that Designated Router, regardless of its Router Priority. (This makes it harder to predict the identity of the Designated Router, but ensures that the Designated Router changes less often. See below.) Otherwise, the router itself becomes Designated Router if it has the highest Router Priority on the network. A more detailed (and more accurate) description of Designated Router election is presented in Section 9.4.

The Designated Router is the endpoint of many adjacencies. In order to optimize the flooding procedure on broadcast networks, the Designated Router multicasts its Link State Update Packets to the address AllSPFRouters, rather than sending separate packets over each adjacency.

Section 2 of this document discusses the directed graph representation of an area. Router nodes are labelled with their Router ID. Multi-access network nodes are actually labelled with the IP address of their Designated Router. It follows that when the Designated Router changes, it appears as if the network node on the graph is replaced by an entirely new node. This will cause the network and all its attached routers to originate new link state advertisements. Until the topological databases again converge, some temporary loss of connectivity may result. This may result in ICMP unreachable messages being sent in response to data traffic. For that reason, the Designated Router should change only infrequently. Router Priorities should be configured so that the most dependable router on a network eventually becomes Designated Router.

7.4 The Backup Designated Router

In order to make the transition to a new Designated Router smoother, there is a Backup Designated Router for each multi-access network. The Backup Designated Router is also adjacent to all routers on the network, and becomes Designated Router when the previous Designated Router fails. If there were no Backup Designated Router, when a new Designated Router became necessary, new adjacencies would have to be formed between the new Designated Router and all other routers attached to the network. Part of the adjacency forming process is the synchronizing of topological databases, which can potentially take quite a long time. During this time, the network would not be available for transit data traffic. The Backup Designated obviates the need to form these adjacencies, since they already exist. This means the period of disruption in transit traffic lasts only as long as it takes to flood the new link state advertisements (which announce the new Designated Router).

The Backup Designated Router does not generate a network links advertisement for the network. (If it did, the transition to a new Designated Router would be even faster. However, this is a tradeoff between database size and speed of convergence when the Designated Router disappears.)

The Backup Designated Router is also elected by the Hello Protocol. Each Hello Packet has a field that specifies the Backup Designated Router for the network.

In some steps of the flooding procedure, the Backup Designated Router plays a passive role, letting the Designated Router do more of the work. This cuts down on the amount of local routing traffic. See Section 13.3 for more information.

7.5 The graph of adjacencies

An adjacency is bound to the network that the two routers have in common. If two routers have multiple networks in common, they may have multiple adjacencies between them.

One can picture the collection of adjacencies on a network as forming an undirected graph. The vertices consist of routers, with an edge joining two routers if they are adjacent. The graph of adjacencies describes the flow of routing protocol packets, and in particular Link State Update Packets, through the Autonomous System.

Two graphs are possible, depending on whether the common network is multi-access. On physical point-to-point networks (and virtual links), the two routers joined by the network will be adjacent after their databases have been synchronized. On multi-access networks, both the Designated Router and the Backup Designated Router are adjacent to all other routers attached to the network, and these account for all adjacencies.

These graphs are shown in Figure 10. It is assumed that Router RT7 has become the Designated Router, and Router RT3 the Backup Designated Router, for the network N2. The Backup Designated Router performs a lesser function during the flooding procedure than the Designated Router (see Section 13.3). This is the reason for the dashed lines connecting the Backup Designated Router RT3.

8 Protocol Packet Processing

This section discusses the general processing of OSPF routing protocol packets. It is very important that the router topological databases remain synchronized. For this reason, routing protocol packets should get preferential treatment over ordinary data packets, both in sending and receiving.

Routing protocol packets are sent along adjacencies only (with the exception of Hello packets, which are used to discover the adjacencies). This means that all routing protocol packets travel a single IP hop, except those sent over virtual links.

All routing protocol packets begin with a standard header. The sections below give the details on how to fill in and verify this standard header. Then, for each packet type, the section is listed that gives more details on that particular packet type's processing.

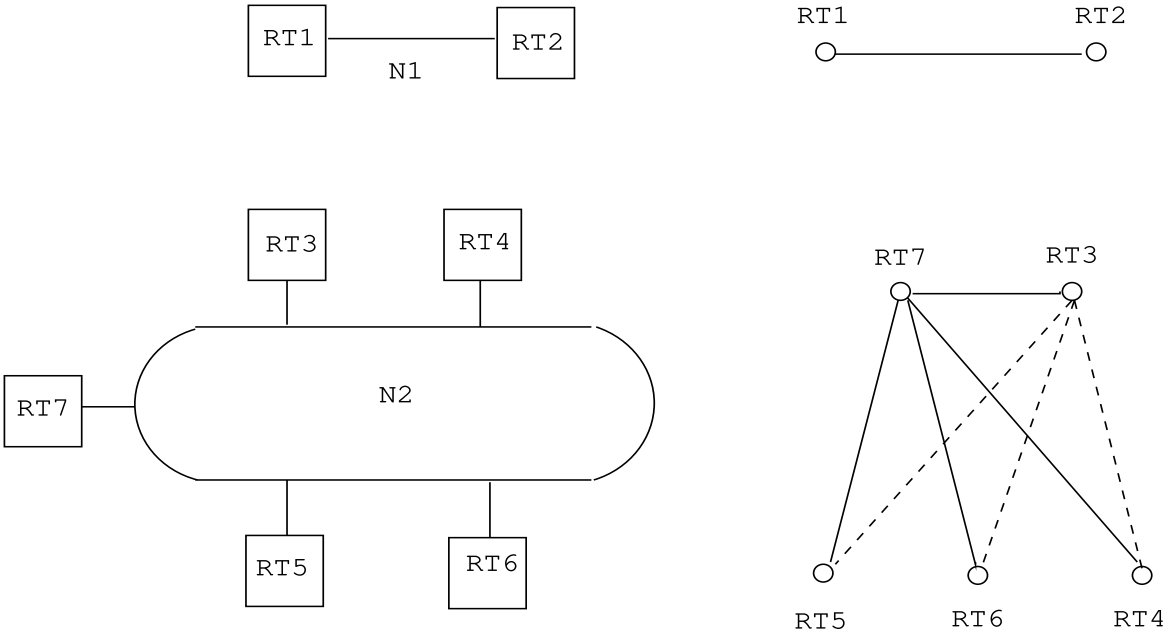
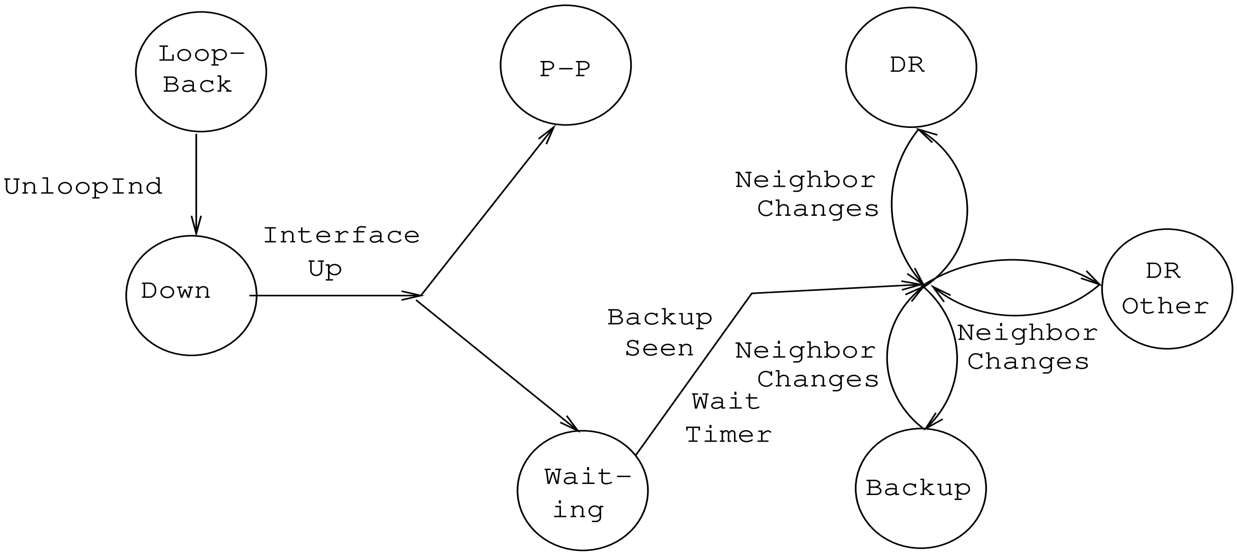


Figure 10: The graph of adjacencies



* InterfaceDown forces Down state
* LoopInd forces Loopback state

Figure 11: Interface state changes

8.1 Sending protocol packets

When a router sends a routing protocol packet, it fills in the fields of the standard OSPF packet header as follows. For more details on the header format consult Section A.3.1:

Version # Set to 2, the version number of the protocol as documented in this specification.

Packet type The type of OSPF packet, such as Link state Update or Hello Packet.

Packet length The length of the entire OSPF packet in bytes, including the standard OSPF packet header.

Router ID The identity of the router itself (who is originating the packet).

Area ID The OSPF area that the packet is being sent into.

Checksum The standard IP 16-bit one's complement checksum of the entire OSPF packet, excluding the 64-bit authentication field. This checksum should be calculated before handing the packet to the appropriate authentication procedure.

AuType and Authentication Each OSPF packet exchange is authenticated. Authentication types are assigned by the protocol and documented in Appendix D. A different authentication scheme can be used for each OSPF area. The 64-bit authentication field is set by the appropriate authentication procedure (determined by AuType). This procedure should be the last called when forming the packet to be sent. The setting of the authentication field is determined by the packet contents and the authentication key (which is configurable on a per-interface basis).

The IP destination address for the packet is selected as follows. On physical point-to-point networks, the IP destination is always set to the address AllSPFRouters. On all other network types (including virtual links), the majority of OSPF packets are sent as unicasts, i.e., sent directly to the other end of the adjacency. In this case, the IP destination is just the **Neighbor IP address** associated with the other end of the adjacency (see Section 10). The only packets not sent as unicasts are on broadcast networks; on these networks Hello packets are sent to the multicast destination AllSPFRouters, the Designated Router and its Backup send both Link State Update Packets and Link State Acknowledgment Packets to the multicast address AllSPFRouters, while all other routers send both their Link State Update and Link State Acknowledgment Packets to the multicast address AllDRouters.

Retransmissions of Link State Update packets are ALWAYS sent as unicasts.

The IP source address should be set to the IP address of the sending interface. Interfaces to unnumbered point-to-point networks have no associated IP address. On these interfaces, the IP source should be set to any of the other IP addresses belonging to the router. For this reason, there must be at least one IP address assigned to the router. [2] Note that, for most purposes, virtual links act precisely the same as unnumbered point-to-point networks. However, each virtual link does have an **IP interface address** (discovered during the routing table build process) which is used as the IP source when sending packets over the virtual link.

For more information on the format of specific OSPF packet types, consult the sections listed in Table 10.

---

[2] It is possible for all of a router's interfaces to be unnumbered point-to-point links. In this case, an IP address must be assigned to the router. This address will then be advertised in the router's router links advertisement as a host route.

Type	*Packet name*	*detailed section (transmit)*
1	Hello	Section 9.5
2	Database description	Section 10.8
3	Link state request	Section 10.9
4	Link state update	Section 13.3
5	Link state ack	Section 13.5

Table 10: Sections describing OSPF protocol packet transmission.

8.2 Receiving protocol packets

Whenever a protocol packet is received by the router it is marked with the interface it was received on. For routers that have virtual links configured, it may not be immediately obvious which interface to associate the packet with. For example, consider the router RT11 depicted in Figure 6. If RT11 receives an OSPF protocol packet on its interface to Network N8, it may want to associate the packet with the interface to Area 2, or with the virtual link to Router RT10 (which is part of the backbone). In the following, we assume that the packet is initially associated with the non-virtual link. [3]

In order for the packet to be accepted at the IP level, it must pass a number of tests, even before the packet is passed to OSPF for processing:

- The IP checksum must be correct.
- The packet's IP destination address must be the IP address of the receiving interface, or one of the IP multicast addresses AllSPFRouters or AllDRouters.
- The IP protocol specified must be OSPF (89).
- Locally originated packets should not be passed on to OSPF. That is, the source IP address should be examined to make sure this is not a multicast packet that the router itself generated.

Next, the OSPF packet header is verified. The fields specified in the header must match those configured for the receiving interface. If they do not, the packet should be discarded:

- The version number field must specify protocol version 2.
- The 16-bit one's complement checksum of the OSPF packet's contents must be verified. Remember that the 64-bit authentication field must be excluded from the checksum calculation.
- The Area ID found in the OSPF header must be verified. If both of the following cases fail, the packet should be discarded. The Area ID specified in the header must either:
    1. *Match the Area ID of the receiving interface*. In this case, the packet has been sent over a single hop. Therefore, the packet's IP source address must be on the same network as the receiving interface. This can be determined by comparing the packet's IP source address to the interface's IP address, after masking both addresses with the interface mask. This comparison should not be performed on point-to-point networks. On point-to-point networks, the interface addresses of each end of the link are assigned independently, if they are assigned at all.

---

[3] Note that in these cases both interfaces, the non-virtual and the virtual, would have the same IP address.

2. *Indicate the backbone.* In this case, the packet has been sent over a virtual link. The receiving router must be an area border router, and the Router ID specified in the packet (the source router) must be the other end of a configured virtual link. The receiving interface must also attach to the virtual link's configured Transit area. If all of these checks succeed, the packet is accepted and is from now on associated with the virtual link (and the backbone area).

- Packets whose IP destination is AllDRouters should only be accepted if the state of the receiving interface is **DR** or **Backup** (see Section 9.1).
- The AuType specified in the packet must match the AuType specified for the associated area.

Next, the packet must be authenticated. This depends on the AuType specified (see Appendix D). The authentication procedure may use an Authentication key, which can be configured on a per-interface basis. If the authentication fails, the packet should be discarded.

If the packet type is Hello, it should then be further processed by the Hello Protocol (see Section 10.5). All other packet types are sent/received only on adjacencies. This means that the packet must have been sent by one of the router's active neighbors. If the receiving interface is a multi-access network (either broadcast or non-broadcast) the sender is identified by the IP source address found in the packet's IP header. If the receiving interface is a point-to-point link or a virtual link, the sender is identified by the Router ID (source router) found in the packet's OSPF header. The data structure associated with the receiving interface contains the list of active neighbors. Packets not matching any active neighbor are discarded.

At this point all received protocol packets are associated with an active neighbor. For the further input processing of specific packet types, consult the sections listed in Table 11.

Type	*Packet name*	*detailed section (receive)*
1	Hello	Section 10.5
2	Database description	Section 10.6
3	Link state request	Section 10.7
4	Link state update	Section 13
5	Link state ack	Section 13.7

Table 11: Sections describing OSPF protocol packet reception.

9 The Interface Data Structure

An OSPF interface is the connection between a router and a network. There is a single OSPF interface structure for each attached network; each interface structure has at most one IP interface address (see below). The support for multiple addresses on a single network is a matter for future consideration.

An OSPF interface can be considered to belong to the area that contains the attached network. All routing protocol packets originated by the router over this interface are labelled with the interface's **Area ID**. One or more router adjacencies may develop over an interface. A router's link state advertisements reflect the state of its interfaces and their associated adjacencies.

The following data items are associated with an interface. Note that a number of these items are actually configuration for the attached network; those items must be the same for all routers connected to the network.

Type The kind of network to which the interface attaches. Its value is either broadcast, non-broadcast yet still multi-access, point-to-point or virtual link.

State The functional level of an interface. State determines whether or not full adjacencies are allowed to form over the interface. State is also reflected in the router's link state advertisements.

IP interface address The IP address associated with the interface. This appears as the IP source address in all routing protocol packets originated over this interface. Interfaces to unnumbered point-to-point networks do not have an associated IP address.

IP interface mask Also referred to as the subnet mask, this indicates the portion of the IP interface address that identifies the attached network. Masking the IP interface address with the IP interface mask yields the IP network number of the attached network. On point-to-point networks and virtual links, the IP interface mask is not defined. On these networks, the link itself is not assigned an IP network number, and so the addresses of each side of the link are assigned independently, if they are assigned at all.

Area ID The Area ID of the area to which the attached network belongs. All routing protocol packets originating from the interface are labelled with this Area ID.

HelloInterval The length of time, in seconds, between the Hello packets that the router sends on the interface. Advertised in Hello packets sent out this interface.

RouterDeadInterval The number of seconds before the router's neighbors will declare it down, when they stop hearing the router's Hello Packets. Advertised in Hello packets sent out this interface.

InfTransDelay The estimated number of seconds it takes to transmit a Link State Update Packet over this interface. Link state advertisements contained in the Link State Update packet will have their age incremented by this amount before transmission. This value should take into account transmission and propagation delays; it must be greater than zero.

Router Priority An 8-bit unsigned integer. When two routers attached to a network both attempt to become Designated Router, the one with the highest Router Priority takes precedence. A router whose Router Priority is set to 0 is ineligible to become Designated Router on the attached network. Advertised in Hello packets sent out this interface.

Hello Timer An interval timer that causes the interface to send a Hello packet. This timer fires every HelloInterval seconds. Note that on non-broadcast networks a separate Hello packet is sent to each qualified neighbor.

Wait Timer A single shot timer that causes the interface to exit the Waiting state, and as a consequence select a Designated Router on the network. The length of the timer is RouterDeadInterval seconds.

List of neighboring routers The other routers attached to this network. On multi-access networks, this list is formed by the Hello Protocol. Adjacencies will be formed to some of these neighbors. The set of adjacent neighbors can be determined by an examination of all of the neighbors' states.

Designated Router The Designated Router selected for the attached network. The Designated Router is selected on all multi-access networks by the Hello Protocol. Two pieces of identification are kept for the Designated Router: its Router ID and its IP interface address on the network. The Designated Router advertises link state for the network; this network link state advertisement is labelled with the Designated Router's IP address. The Designated Router is initialized to `0.0.0.0`, which indicates the lack of a Designated Router.

Backup Designated Router The Backup Designated Router is also selected on all multi-access networks by the Hello Protocol. All routers on the attached network become adjacent to both the Designated Router and the

Backup Designated Router. The Backup Designated Router becomes Designated Router when the current Designated Router fails. The Backup Designated Router is initialized to `0.0.0.0`, indicating the lack of a Backup Designated Router.

Interface output cost(s) The cost of sending a data packet on the interface, expressed in the link state metric. This is advertised as the link cost for this interface in the router links advertisement. There may be a separate cost for each IP Type of Service. The cost of an interface must be greater than zero.

RxmtInterval The number of seconds between link state advertisement retransmissions, for adjacencies belonging to this interface. Also used when retransmitting Database Description and Link State Request Packets.

Authentication key This configured data allows the authentication procedure to generate and/or verify the Authentication field in the OSPF header. The Authentication key can be configured on a per-interface basis. For example, if the AuType indicates simple password, the Authentication key would be a 64-bit password. This key would be inserted directly into the OSPF header when originating routing protocol packets, and there could be a separate password for each network.

9.1 Interface states

The various states that router interfaces may attain is documented in this section. The states are listed in order of progressing functionality. For example, the inoperative state is listed first, followed by a list of intermediate states before the final, fully functional state is achieved. The specification makes use of this ordering by sometimes making references such as "those interfaces in state greater than X".

Figure 11 shows the graph of interface state changes. The arcs of the graph are labelled with the event causing the state change. These events are documented in Section 9.2. The interface state machine is described in more detail in Section 9.3.

Down This is the initial interface state. In this state, the lower-level protocols have indicated that the interface is unusable. No protocol traffic at all will be sent or received on such a interface. In this state, interface parameters should be set to their initial values. All interface timers should be disabled, and there should be no adjacencies associated with the interface.

Loopback In this state, the router's interface to the network is looped back. The interface may be looped back in hardware or software. The interface will be unavailable for regular data traffic. However, it may still be desirable to gain information on the quality of this interface, either through sending ICMP pings to the interface or through something like a bit error test. For this reason, IP packets may still be addressed to an interface in Loopback state. To facilitate this, such interfaces are advertised in router links advertisements as single host routes, whose destination is the IP interface address. [4]

Waiting In this state, the router is trying to determine the identity of the (Backup) Designated Router for the network. To do this, the router monitors the Hello Packets it receives. The router is not allowed to elect a Backup Designated Router nor a Designated Router until it transitions out of Waiting state. This prevents unnecessary changes of (Backup) Designated Router.

Point-to-point In this state, the interface is operational, and connects either to a physical point-to-point network or to a virtual link. Upon entering this state, the router attempts to form an adjacency with the neighboring router. Hello Packets are sent to the neighbor every HelloInterval seconds.

---

[4] Note that no host route is generated for, and no IP packets can be addressed to, interfaces to unnumbered point-to-point networks. This is regardless of such an interface's state.

DR Other The interface is to a multi-access network on which another router has been selected to be the Designated Router. In this state, the router itself has not been selected Backup Designated Router either. The router forms adjacencies to both the Designated Router and the Backup Designated Router (if they exist).

Backup In this state, the router itself is the Backup Designated Router on the attached network. It will be promoted to Designated Router when the present Designated Router fails. The router establishes adjacencies to all other routers attached to the network. The Backup Designated Router performs slightly different functions during the Flooding Procedure, as compared to the Designated Router (see Section 13.3). See Section 7.4 for more details on the functions performed by the Backup Designated Router.

DR In this state, this router itself is the Designated Router on the attached network. Adjacencies are established to all other routers attached to the network. The router must also originate a network links advertisement for the network node. The advertisement will contain links to all routers (including the Designated Router itself) attached to the network. See Section 7.3 for more details on the functions performed by the Designated Router.

9.2 Events causing interface state changes

State changes can be effected by a number of events. These events are pictured as the labelled arcs in Figure 11. The label definitions are listed below. For a detailed explanation of the effect of these events on OSPF protocol operation, consult Section 9.3.

InterfaceUp Lower-level protocols have indicated that the network interface is operational. This enables the interface to transition out of Down state. On virtual links, the interface operational indication is actually a result of the shortest path calculation (see Section 16.7).

WaitTimer The Wait Timer has fired, indicating the end of the waiting period that is required before electing a (Backup) Designated Router.

BackupSeen The router has detected the existence or non-existence of a Backup Designated Router for the network. This is done in one of two ways. First, an Hello Packet may be received from a neighbor claiming to be itself the Backup Designated Router. Alternatively, an Hello Packet may be received from a neighbor claiming to be itself the Designated Router, and indicating that there is no Backup Designated Router. In either case there must be bidirectional communication with the neighbor, i.e., the router must also appear in the neighbor's Hello Packet. This event signals an end to the Waiting state.

NeighborChange There has been a change in the set of bidirectional neighbors associated with the interface. The (Backup) Designated Router needs to be recalculated. The following neighbor changes lead to the **NeighborChange** event. For an explanation of neighbor states, see Section 10.1.

- Bidirectional communication has been established to a neighbor. In other words, the state of the neighbor has transitioned to **2-Way** or higher.
- There is no longer bidirectional communication with a neighbor. In other words, the state of the neighbor has transitioned to **Init** or lower.
- One of the bidirectional neighbors is newly declaring itself as either Designated Router or Backup Designated Router. This is detected through examination of that neighbor's Hello Packets.
- One of the bidirectional neighbors is no longer declaring itself as Designated Router, or is no longer declaring itself as Backup Designated Router. This is again detected through examination of that neighbor's Hello Packets.

- The advertised **Router Priority** for a bidirectional neighbor has changed. This is again detected through examination of that neighbor's Hello Packets.

LoopInd An indication has been received that the interface is now looped back to itself. This indication can be received either from network management or from the lower level protocols.

UnloopInd An indication has been received that the interface is no longer looped back. As with the **LoopInd** event, this indication can be received either from network management or from the lower level protocols.

InterfaceDown Lower-level protocols indicate that this interface is no longer functional. No matter what the current interface state is, the new interface state will be Down.

9.3 The Interface state machine

A detailed description of the interface state changes follows. Each state change is invoked by an event (Section 9.2). This event may produce different effects, depending on the current state of the interface. For this reason, the state machine below is organized by current interface state and received event. Each entry in the state machine describes the resulting new interface state and the required set of additional actions.

When an interface's state changes, it may be necessary to originate a new router links advertisement. See Section 12.4 for more details.

Some of the required actions below involve generating events for the neighbor state machine. For example, when an interface becomes inoperative, all neighbor connections associated with the interface must be destroyed. For more information on the neighbor state machine, see Section 10.3.

State(s): **Down**
Event: **InterfaceUp**
New state: Depends upon action routine
Action: Start the interval **Hello Timer**, enabling the periodic sending of Hello packets out the interface. If the attached network is a physical point-to-point network or virtual link, the interface state transitions to **Point-to-Point**. Else, if the router is not eligible to become Designated Router the interface state transitions to **DR Other**.

Otherwise, the attached network is multi-access and the router is eligible to become Designated Router. In this case, in an attempt to discover the attached network's Designated Router the interface state is set to **Waiting** and the single shot **Wait Timer** is started. If in addition the attached network is non-broadcast, examine the configured list of neighbors for this interface and generate the neighbor event **Start** for each neighbor that is also eligible to become Designated Router.

State(s): **Waiting**
Event: **BackupSeen**
New state: Depends upon action routine.
Action: Calculate the attached network's Backup Designated Router and Designated Router, as shown in Section 9.4. As a result of this calculation, the new state of the interface will be either **DR Other**, **Backup** or **DR**.

State(s): **Waiting**
Event: **WaitTimer**

New state: Depends upon action routine.
Action: Calculate the attached network's Backup Designated Router and Designated Router, as shown in Section 9.4. As a result of this calculation, the new state of the interface will be either **DR Other**, **Backup** or **DR**.

State(s): **DR Other**, **Backup** or **DR**
Event: **NeighborChange**
New state: Depends upon action routine.
Action: Recalculate the attached network's Backup Designated Router and Designated Router, as shown in Section 9.4. As a result of this calculation, the new state of the interface will be either **DR Other**, **Backup** or **DR**.

State(s): Any State
Event: **InterfaceDown**
New state: **Down**
Action: All interface variables are reset, and interface timers disabled. Also, all neighbor connections associated with the interface are destroyed. This is done by generating the event **KillNbr** on all associated neighbors (see Section 10.2).

State(s): Any State
Event: **LoopInd**
New state: **Loopback**
Action: Since this interface is no longer connected to the attached network the actions associated with the above **InterfaceDown** event are executed.

State(s): **Loopback**
Event: **UnloopInd**
New state: **Down**
Action: No actions are necessary. For example, the interface variables have already been reset upon entering the **Loopback** state. Note that reception of an **InterfaceUp** event is necessary before the interface again becomes fully functional.

9.4 Electing the Designated Router

This section describes the algorithm used for calculating a network's Designated Router and Backup Designated Router. This algorithm is invoked by the Interface state machine. The initial time a router runs the election algorithm for a network, the network's Designated Router and Backup Designated Router are initialized to `0.0.0.0`. This indicates the lack of both a Designated Router and a Backup Designated Router.

The Designated Router election algorithm proceeds as follows: Call the router doing the calculation Router X. The list of neighbors attached to the network and having established bidirectional communication with Router X is examined. This list is precisely the collection of Router X's neighbors (on this network) whose state is greater than or equal to **2-Way** (see Section 10.1). Router X itself is also considered to be on the list. Discard all routers from the list that are ineligible to become Designated Router. (Routers having Router Priority of 0 are ineligible to become Designated Router.) The following steps are then executed, considering only those routers that remain on the list:

1. Note the current values for the network's Designated Router and Backup Designated Router. This is used later for comparison purposes.
2. Calculate the new Backup Designated Router for the network as follows. Only those routers on the list that have not declared themselves to be Designated Router are eligible to become Backup Designated Router. If one or more of these routers have declared themselves Backup Designated Router (i.e., they are currently listing themselves as Backup Designated Router, but not as Designated Router, in their Hello Packets) the one having highest Router Priority is declared to be Backup Designated Router. In case of a tie, the one having the highest Router ID is chosen. If no routers have declared themselves Backup Designated Router, choose the router having highest Router Priority, (again excluding those routers who have declared themselves Designated Router), and again use the Router ID to break ties.
3. Calculate the new Designated Router for the network as follows. If one or more of the routers have declared themselves Designated Router (i.e., they are currently listing themselves as Designated Router in their Hello Packets) the one having highest Router Priority is declared to be Designated Router. In case of a tie, the one having the highest Router ID is chosen. If no routers have declared themselves Designated Router, assign the Designated Router to be the same as the newly elected Backup Designated Router.
4. If Router X is now newly the Designated Router or newly the Backup Designated Router, or is now no longer the Designated Router or no longer the Backup Designated Router, repeat steps 2 and 3, and then proceed to step 5. For example, if Router X is now the Designated Router, when step 2 is repeated X will no longer be eligible for Backup Designated Router election. Among other things, this will ensure that no router will declare itself both Backup Designated Router and Designated Router. [5]
5. As a result of these calculations, the router itself may now be Designated Router or Backup Designated Router. See Sections 7.3 and 7.4 for the additional duties this would entail. The router's interface state should be set accordingly. If the router itself is now Designated Router, the new interface state is **DR**. If the router itself is now Backup Designated Router, the new interface state is **Backup**. Otherwise, the new interface state is **DR Other**.
6. If the attached network is non-broadcast, and the router itself has just become either Designated Router or Backup Designated Router, it must start sending Hello Packets to those neighbors that are not eligible to become Designated Router (see Section 9.5.1). This is done by invoking the neighbor event **Start** for each neighbor having a **Router Priority** of 0.
7. If the above calculations have caused the identity of either the Designated Router or Backup Designated Router to change, the set of adjacencies associated with this interface will need to be modified. Some adjacencies may need to be formed, and others may need to be broken. To accomplish this, invoke the event **AdjOK?** on all neighbors whose state is at least **2-Way**. This will cause their eligibility for adjacency to be reexamined (see Sections 10.3 and 10.4).

The reason behind the election algorithm's complexity is the desire for an orderly transition from Backup Designated Router to Designated Router, when the current Designated Router fails. This orderly transition is ensured through the introduction of hysteresis: no new Backup Designated Router can be chosen until the old Backup accepts its new Designated Router responsibilities.

---

[5] It is instructive to see what happens when the Designated Router for the network crashes. Call the Designated Router for the network RT1, and the Backup Designated Router RT2. If Router RT1 crashes (or maybe its interface to the network dies), the other routers on the network will detect RT1's absence within RouterDeadInterval seconds. All routers may not detect this at precisely the same time; the routers that detect RT1's absence before RT2 does will, for a time, select RT2 to be both Designated Router and Backup Designated Router. When RT2 detects that RT1 is gone it will move itself to Designated Router. At this time, the remaining router having highest Router Priority will be selected as Backup Designated Router.

The above procedure may elect the same router to be both Designated Router and Backup Designated Router, although that router will never be the calculating router (Router X) itself. The elected Designated Router may not be the router having the highest Router Priority, nor will the Backup Designated Router necessarily have the second highest Router Priority. If Router X is not itself eligible to become Designated Router, it is possible that neither a Backup Designated Router nor a Designated Router will be selected in the above procedure. Note also that if Router X is the only attached router that is eligible to become Designated Router, it will select itself as Designated Router and there will be no Backup Designated Router for the network.

9.5 Sending Hello packets

Hello packets are sent out each functioning router interface. They are used to discover and maintain neighbor relationships. [6] On multi-access networks, Hello Packets are also used to elect the Designated Router and Backup Designated Router, and in that way determine what adjacencies should be formed.

The format of an Hello packet is detailed in Section A.3.2. The Hello Packet contains the router's **Router Priority** (used in choosing the Designated Router), and the interval between Hello Packets sent out the interface (**HelloInterval**). The Hello Packet also indicates how often a neighbor must be heard from to remain active (**RouterDeadInterval**). Both HelloInterval and RouterDeadInterval must be the same for all routers attached to a common network. The Hello packet also contains the IP address mask of the attached network (**Network Mask**). On unnumbered point-to-point networks and on virtual links this field should be set to `0.0.0.0`.

The Hello packet's **Options** field describes the router's optional OSPF capabilities. There are currently two optional capabilities defined (see Sections 4.5 and A.2). The **T-bit** of the **Options** field should be set if the router is capable of calculating separate routes for each IP TOS. The **E-bit** should be set if and only if the attached area is capable of processing AS external advertisements (i.e., it is not a stub area). If the **E-bit** is set incorrectly the neighboring routers will refuse to accept the Hello Packet (see Section 10.5). The rest of the Hello Packet's **Options** field should be set to zero.

In order to ensure two-way communication between adjacent routers, the Hello packet contains the list of all routers from which Hello Packets have been seen recently. The Hello packet also contains the router's current choice for Designated Router and Backup Designated Router. A value of `0.0.0.0` in these fields means that one has not yet been selected.

On broadcast networks and physical point-to-point networks, Hello packets are sent every HelloInterval seconds to the IP multicast address AllSPFRouters. On virtual links, Hello packets are sent as unicasts (addressed directly to the other end of the virtual link) every HelloInterval seconds. On non-broadcast networks, the sending of Hello packets is more complicated. This will be covered in the next section.

9.5.1 Sending Hello packets on non-broadcast networks

Static configuration information is necessary in order for the Hello Protocol to function on non-broadcast networks (see Section C.5). Every attached router which is eligible to become Designated Router has a configured list of all of its neighbors on the network. Each listed neighbor is labelled with its Designated Router eligibility.

The interface state must be at least **Waiting** for any Hello Packets to be sent. Hello Packets are then sent directly (as unicasts) to some subset of a router's neighbors. Sometimes an Hello Packet is sent periodically on a timer; at other times it is sent as a response to a received Hello Packet. A router's hello-sending behavior varies depending on whether the router itself is eligible to become Designated Router.

---

[6] On point-to-point networks, the lower level protocols indicate whether the neighbor is up and running. Likewise, existence of the neighbor on virtual links is indicated by the routing table calculation. However, in both these cases, the Hello Protocol is still used. This ensures that communication between the neighbors is bidirectional, and that each of the neighbors has a functioning routing protocol layer.

If the router is eligible to become Designated Router, it must periodically send Hello Packets to all neighbors that are also eligible. In addition, if the router is itself the Designated Router or Backup Designated Router, it must also send periodic Hello Packets to all other neighbors. This means that any two eligible routers are always exchanging Hello Packets, which is necessary for the correct operation of the Designated Router election algorithm. To minimize the number of Hello Packets sent, the number of eligible routers on a non-broadcast network should be kept small.

If the router is not eligible to become Designated Router, it must periodically send Hello Packets to both the Designated Router and the Backup Designated Router (if they exist). It must also send an Hello Packet in reply to an Hello Packet received from any eligible neighbor (other than the current Designated Router and Backup Designated Router). This is needed to establish an initial bidirectional relationship with any potential Designated Router.

When sending Hello packets periodically to any neighbor, the interval between Hello Packets is determined by the neighbor's state. If the neighbor is in state **Down**, Hello Packets are sent every PollInterval seconds. Otherwise, Hello Packets are sent every HelloInterval seconds.

10 The Neighbor Data Structure

An OSPF router converses with its neighboring routers. Each separate conversation is described by a “neighbor data structure”. Each conversation is bound to a particular OSPF router interface, and is identified either by the neighboring router's OSPF Router ID or by its Neighbor IP address (see below). Thus if the OSPF router and another router have multiple attached networks in common, multiple conversations ensue, each described by a unique neighbor data structure. Each separate conversation is loosely referred to in the text as being a separate “neighbor”.

The neighbor data structure contains all information pertinent to the forming or formed adjacency between the two neighbors. (However, remember that not all neighbors become adjacent.) An adjacency can be viewed as a highly developed conversation between two routers.

State The functional level of the neighbor conversation. This is described in more detail in Section 10.1.

Inactivity Timer A single shot timer whose firing indicates that no Hello Packet has been seen from this neighbor recently. The length of the timer is RouterDeadInterval seconds.

Master/Slave When the two neighbors are exchanging databases, they form a master/slave relationship. The master sends the first Database Description Packet, and is the only part that is allowed to retransmit. The slave can only respond to the master's Database Description Packets. The master/slave relationship is negotiated in state **ExStart**.

DD Sequence Number A 32-bit number identifying individual Database Description packets. When the neighbor state ExStart is entered, the DD sequence number should be set to a value not previously seen by the neighboring router. One possible scheme is to use the machine's time of day counter. The DD sequence number is then incremented by the master with each new Database Description packet sent. The slave's DD sequence number indicates the last packet received from the master. Only one packet is allowed outstanding at a time.

Neighbor ID The OSPF Router ID of the neighboring router. The Neighbor ID is learned when Hello packets are received from the neighbor, or is configured if this is a virtual adjacency (see Section C.4).

Neighbor Priority The Router Priority of the neighboring router. Contained in the neighbor's Hello packets, this item is used when selecting the Designated Router for the attached network.

Neighbor IP address The IP address of the neighboring router's interface to the attached network. Used as the Destination IP address when protocol packets are sent as unicasts along this adjacency. Also used in router links advertisements as the **Link ID** for the attached network if the neighboring router is selected to be Designated Router (see Section 12.4.1). The Neighbor IP address is learned when Hello packets are received from the neighbor. For virtual links, the Neighbor IP address is learned during the routing table build process (see Section 15).

Neighbor Options The optional OSPF capabilities supported by the neighbor. Learned during the Database Exchange process (see Section 10.6). The neighbor's optional OSPF capabilities are also listed in its Hello packets. This enables received Hello Packets to be rejected (i.e., neighbor relationships will not even start to form) if there is a mismatch in certain crucial OSPF capabilities (see Section 10.5). The optional OSPF capabilities are documented in Section 4.5.

Neighbor's Designated Router The neighbor's idea of the Designated Router. If this is the neighbor itself, this is important in the local calculation of the Designated Router. Defined only on multi-access networks.

Neighbor's Backup Designated Router The neighbor's idea of the Backup Designated Router. If this is the neighbor itself, this is important in the local calculation of the Backup Designated Router. Defined only on multi-access networks.

The next set of variables are lists of link state advertisements. These lists describe subsets of the area topological database. There can be five distinct types of link state advertisements in an area topological database: router links, network links, and Type 3 and 4 summary links (all stored in the area data structure), and AS external links (stored in the global data structure).

Link state retransmission list The list of link state advertisements that have been flooded but not acknowledged on this adjacency. These will be retransmitted at intervals until they are acknowledged, or until the adjacency is destroyed.

Database summary list The complete list of link state advertisements that make up the area topological database, at the moment the neighbor goes into Database Exchange state. This list is sent to the neighbor in Database Description packets.

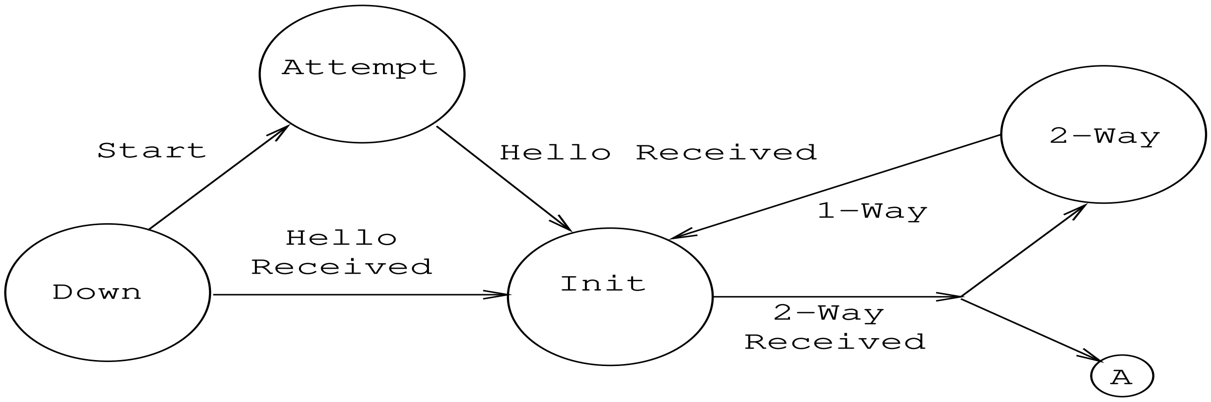
Link state request list The list of link state advertisements that need to be received from this neighbor in order to synchronize the two neighbors' topological databases. This list is created as Database Description packets are received, and is then sent to the neighbor in Link State Request packets. The list is depleted as appropriate Link State Update packets are received.

10.1 Neighbor states

The state of a neighbor (really, the state of a conversation being held with a neighboring router) is documented in the following sections. The states are listed in order of progressing functionality. For example, the inoperative state is listed first, followed by a list of intermediate states before the final, fully functional state is achieved. The specification makes use of this ordering by sometimes making references such as "those neighbors/adjacencies in state greater than X". Figures 12 and 13 show the graph of neighbor state changes. The arcs of the graphs are labelled with the event causing the state change. The neighbor events are documented in Section 10.2.

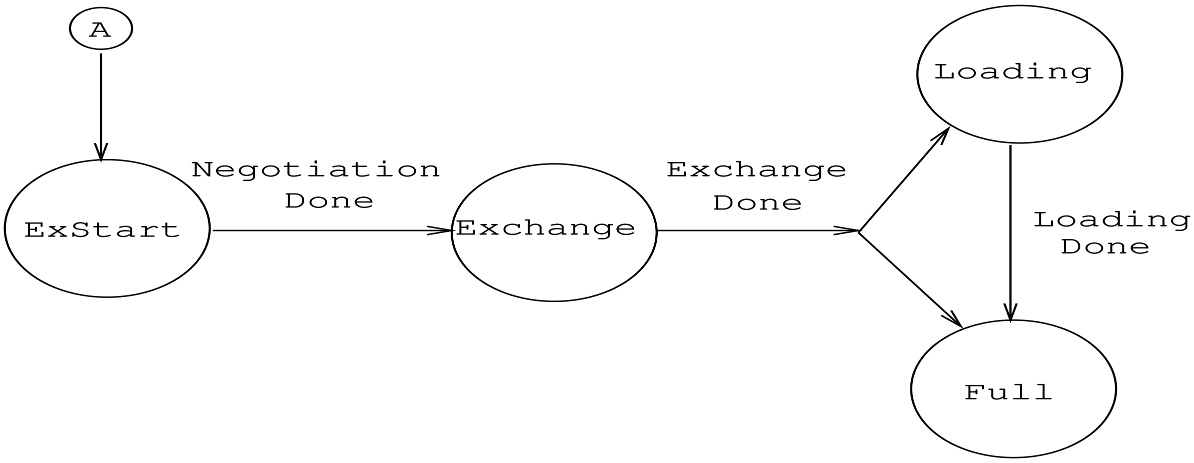
The graph in Figure 12 shows the state changes effected by the Hello Protocol. The Hello Protocol is responsible for neighbor acquisition and maintenance, and for ensuring two way communication between neighbors.

The graph in Figure 13 shows the forming of an adjacency. Not every two neighboring routers become adjacent (see Section 10.4). The adjacency starts to form when the neighbor is in state **ExStart**. After the two routers discover



* Event KillNbr forces Down state
  Event Inactivity Timer forces Down State
  Event LLDown forces Down state

Figure 12: Neighbor state changes (Hello Protocol)



* Event SeqNumberMismatch forces ExStart state
  Event BadLSReq forces ExStart state
  Event 1-Way forces Init state
  Event KillNbr forces Down state
  Event InactivityTimer forces Down state
  Event LLDown forces Down state
  Event AdjOK? leads to adjacency forming/breaking

Figure 13: Neighbor state changes (Database Exchange)

their master/slave status, the state transitions to **Exchange**. At this point the neighbor starts to be used in the flooding procedure, and the two neighboring routers begin synchronizing their databases. When this synchronization is finished, the neighbor is in state **Full** and we say that the two routers are fully adjacent. At this point the adjacency is listed in link state advertisements.

For a more detailed description of neighbor state changes, together with the additional actions involved in each change, see Section 10.3.

Down This is the initial state of a neighbor conversation. It indicates that there has been no recent information received from the neighbor. On non-broadcast networks, Hello packets may still be sent to “Down” neighbors, although at a reduced frequency (see Section 9.5.1).

Attempt This state is only valid for neighbors attached to non-broadcast networks. It indicates that no recent information has been received from the neighbor, but that a more concerted effort should be made to contact the neighbor. This is done by sending the neighbor Hello packets at intervals of HelloInterval (see Section 9.5.1).

Init In this state, an Hello packet has recently been seen from the neighbor. However, bidirectional communication has not yet been established with the neighbor (i.e., the router itself did not appear in the neighbor’s Hello packet). All neighbors in this state (or higher) are listed in the Hello packets sent from the associated interface.

2-Way In this state, communication between the two routers is bidirectional. This has been assured by the operation of the Hello Protocol. This is the most advanced state short of beginning adjacency establishment. The (Backup) Designated Router is selected from the set of neighbors in state **2-Way** or greater.

ExStart This is the first step in creating an adjacency between the two neighboring routers. The goal of this step is to decide which router is the master, and to decide upon the initial DD sequence number. Neighbor conversations in this state or greater are called adjacencies.

Exchange In this state the router is describing its entire link state database by sending Database Description packets to the neighbor. Each Database Description Packet has a DD sequence number, and is explicitly acknowledged. Only one Database Description Packet is allowed outstanding at any one time. In this state, Link State Request Packets may also be sent asking for the neighbor’s more recent advertisements. All adjacencies in **Exchange** state or greater are used by the flooding procedure. In fact, these adjacencies are fully capable of transmitting and receiving all types of OSPF routing protocol packets.

Loading In this state, Link State Request packets are sent to the neighbor asking for the more recent advertisements that have been discovered (but not yet received) in the **Exchange** state.

Full In this state, the neighboring routers are fully adjacent. These adjacencies will now appear in router links and network links advertisements.

10.2 Events causing neighbor state changes

State changes can be effected by a number of events. These events are shown in the labels of the arcs in Figures 12 and 13. The label definitions are as follows:

HelloReceived A Hello packet has been received from a neighbor.

Start This is an indication that Hello Packets should now be sent to the neighbor at intervals of HelloInterval seconds. This event is generated only for neighbors associated with non-broadcast networks.

2-WayReceived Bidirectional communication has been realized between the two neighboring routers. This is indicated by this router seeing itself in the other’s Hello packet.

NegotiationDone The Master/Slave relationship has been negotiated, and DD sequence numbers have been exchanged. This signals the start of the sending/receiving of Database Description packets. For more information on the generation of this event, consult Section 10.8.

ExchangeDone Both routers have successfully transmitted a full sequence of Database Description packets. Each router now knows what parts of its link state database are out of date. For more information on the generation of this event, consult Section 10.8.

BadLSReq A Link State Request has been received for a link state advertisement not contained in the database. This indicates an error in the Database Exchange process.

Loading Done Link State Updates have been received for all out-of-date portions of the database. This is indicated by the **Link state request list** becoming empty after the Database Exchange process has completed.

AdjOK? A decision must be made (again) as to whether an adjacency should be established/maintained with the neighbor. This event will start some adjacencies forming, and destroy others.

The following events cause well developed neighbors to revert to lesser states. Unlike the above events, these events may occur when the neighbor conversation is in any of a number of states.

SeqNumberMismatch A Database Description packet has been received that either a) has an unexpected DD sequence number, b) unexpectedly has the Init bit set or c) has an Options field differing from the last Options field received in a Database Description packet. Any of these conditions indicate that some error has occurred during adjacency establishment.

1-Way An Hello packet has been received from the neighbor, in which this router is not mentioned. This indicates that communication with the neighbor is not bidirectional.

KillNbr This is an indication that all communication with the neighbor is now impossible, forcing the neighbor to revert to Down state.

InactivityTimer The inactivity Timer has fired. This means that no Hello packets have been seen recently from the neighbor. The neighbor reverts to Down state.

LLDown This is an indication from the lower level protocols that the neighbor is now unreachable. For example, on an X.25 network this could be indicated by an X.25 clear indication with appropriate cause and diagnostic fields. This event forces the neighbor into Down state.

10.3 The Neighbor state machine

A detailed description of the neighbor state changes follows. Each state change is invoked by an event (Section 10.2). This event may produce different effects, depending on the current state of the neighbor. For this reason, the state machine below is organized by current neighbor state and received event. Each entry in the state machine describes the resulting new neighbor state and the required set of additional actions.

When a neighbor's state changes, it may be necessary to rerun the Designated Router election algorithm. This is determined by whether the interface **NeighborChange** event is generated (see Section 9.2). Also, if the Interface is in **DR** state (the router is itself Designated Router), changes in neighbor state may cause a new network links advertisement to be originated (see Section 12.4).

When the neighbor state machine needs to invoke the interface state machine, it should be done as a *scheduled* task (see Section 4.4). This simplifies things, by ensuring that neither state machine will be *executed* recursively.

State(s): **Down**
Event: **Start**
New state: **Attempt**
Action: Send an Hello Packet to the neighbor (this neighbor is always associated with a non-broadcast network) and start the Inactivity Timer for the neighbor. The timer's later firing would indicate that communication with the neighbor was not attained.

State(s): **Attempt**
Event: **HelloReceived**
New state: **Init**
Action: Restart the Inactivity Timer for the neighbor, since the neighbor has now been heard from.

State(s): **Down**
Event: **HelloReceived**
New state: **Init**
Action: Start the Inactivity Timer for the neighbor. The timer's later firing would indicate that the neighbor is dead.

State(s): **Init** or greater
Event: **HelloReceived**
New state: No state change.
Action: Restart the Inactivity Timer for the neighbor, since the neighbor has again been heard from.

State(s): **Init**
Event: **2-WayReceived**
New state: Depends upon action routine.
Action: Determine whether an adjacency should be established with the neighbor (see Section 10.4). If not, the new neighbor state is **2-Way**.

Otherwise (an adjacency should be established) the neighbor state transitions to **ExStart**. Upon entering this state, the router increments the DD sequence number for this neighbor. If this is the first time that an adjacency has been attempted, the DD sequence number should be assigned some unique value (like the time of day clock). It then declares itself master (sets the master/slave bit to master), and starts sending Database Description Packets, with the initialize (I), more (M) and master (MS) bits set. This Database Description Packet should be otherwise empty. This Database Description Packet should be retransmitted at intervals of RxmtInterval until the next state is entered (see Section 10.8).

State(s): **ExStart**
Event: **NegotiationDone**
New state: **Exchange**
Action: The router must list the contents of its entire area link state database in the neighbor **Database summary list**. The area link state database consists of the router links,

network links and summary links contained in the area structure, along with the AS external links contained in the global structure. AS external link advertisements are omitted from a virtual neighbor's **Database summary list**. AS external advertisements are omitted from the **Database summary list** if the area has been configured as a stub (see Section 3.6). Advertisements whose age is equal to MaxAge are instead added to the neighbor's **Link state retransmission list**. A summary of the **Database summary list** will be sent to the neighbor in Database Description packets. Each Database Description Packet has a DD sequence number, and is explicitly acknowledged. Only one Database Description Packet is allowed outstanding at any one time. For more detail on the sending and receiving of Database Description packets, see Sections 10.8 and 10.6.

State(s): **Exchange**
Event: **ExchangeDone**
New state: Depends upon action routine.
Action: If the neighbor **Link state request list** is empty, the new neighbor state is **Full**. No other action is required. This is an adjacency's final state.

Otherwise, the new neighbor state is **Loading**. Start (or continue) sending Link State Request packets to the neighbor (see Section 10.9). These are requests for the neighbor's more recent advertisements (which were discovered but not yet received in the **Exchange** state). These advertisements are listed in the **Link state request list** associated with the neighbor.

State(s): **Loading**
Event: **Loading Done**
New state: **Full**
Action: No action required. This is an adjacency's final state.

State(s): **2-Way**
Event: **AdjOK?**
New state: Depends upon action routine.
Action: Determine whether an adjacency should be formed with the neighboring router (see Section 10.4). If not, the neighbor state remains at **2-Way**. Otherwise, transition the neighbor state to **ExStart** and perform the actions associated with the above state machine entry for state **Init** and event **2-WayReceived**.

State(s): **ExStart** or greater
Event: **AdjOK?**
New state: Depends upon action routine.
Action: Determine whether the neighboring router should still be adjacent. If yes, there is no state change and no further action is necessary.

Otherwise, the (possibly partially formed) adjacency must be destroyed. The neighbor state transitions to **2-Way**. The **Link state retransmission list**, **Database summary list** and **Link state request list** are cleared of link state advertisements.

State(s): **Exchange** or greater

Event: **SeqNumberMismatch**
New state: **ExStart**
Action: The (possibly partially formed) adjacency is torn down, and then an attempt is made at reestablishment. The neighbor state first transitions to **ExStart**. The **Link state retransmission list**, **Database summary list** and **Link state request list** are cleared of link state advertisements. Then the router increments the DD sequence number for this neighbor, declares itself master (sets the master/slave bit to master), and starts sending Database Description Packets, with the initialize (I), more (M) and master (MS) bits set. This Database Description Packet should be otherwise empty (see Section 10.8).

State(s): **Exchange** or greater
Event: **BadLSReq**
New state: **ExStart**
Action: The action for event **BadLSReq** is exactly the same as for the neighbor event **SeqNumberMismatch**. The (possibly partially formed) adjacency is torn down, and then an attempt is made at reestablishment. For more information, see the neighbor state machine entry that is invoked when event **SeqNumberMismatch** is generated in state **Exchange** or greater.

State(s): Any state
Event: **KillNbr**
New state: **Down**
Action: The **Link state retransmission list**, **Database summary list** and **Link state request list** are cleared of link state advertisements. Also, the Inactivity Timer is disabled.

State(s): Any state
Event: **LLDown**
New state: **Down**
Action: The **Link state retransmission list**, **Database summary list** and **Link state request list** are cleared of link state advertisements. Also, the Inactivity Timer is disabled.

State(s): Any state
Event: **InactivityTimer**
New state: **Down**
Action: The **Link state retransmission list**, **Database summary list** and **Link state request list** are cleared of link state advertisements.

State(s): **2-Way** or greater
Event: **1-WayReceived**
New state: **Init**
Action: The **Link state retransmission list**, **Database summary list** and **Link state request list** are cleared of link state advertisements.

State(s): **2-Way** or greater

Event: **2-Way received**
New state: No state change.
Action: No action required.

State(s): **Init**
Event: **1-WayReceived**
New state: No state change.
Action: No action required.

10.4 Whether to become adjacent

Adjacencies are established with some subset of the router's neighbors. Routers connected by point-to-point networks and virtual links always become adjacent. On multi-access networks, all routers become adjacent to both the Designated Router and the Backup Designated Router.

The adjacency-forming decision occurs in two places in the neighbor state machine. First, when bidirectional communication is initially established with the neighbor, and secondly, when the identity of the attached network's (Backup) Designated Router changes. If the decision is made to not attempt an adjacency, the state of the neighbor communication stops at **2-Way**.

An adjacency should be established with a bidirectional neighbor when at least one of the following conditions holds:

- The underlying network type is point-to-point
- The underlying network type is virtual link
- The router itself is the Designated Router
- The router itself is the Backup Designated Router
- The neighboring router is the Designated Router
- The neighboring router is the Backup Designated Router

10.5 Receiving Hello Packets

This section explains the detailed processing of a received Hello Packet. (See Section A.3.2 for the format of Hello packets.) The generic input processing of OSPF packets will have checked the validity of the IP header and the OSPF packet header. Next, the values of the **Network Mask**, **HelloInterval**, and **RouterDeadInterval** fields in the received Hello packet must be checked against the values configured for the receiving interface. Any mismatch causes processing to stop and the packet to be dropped. In other words, the above fields are really describing the attached network's configuration. However, there is one exception to the above rule: on point-to-point networks and on virtual links, the **Network Mask** in the received Hello Packet should be ignored.

The receiving interface attaches to a single OSPF area (this could be the backbone). The setting of the **E-bit** found in the Hello Packet's Options field must match this area's ExternalRoutingCapability. If AS external advertisements are not flooded into/throughout the area (i.e, the area is a "stub") the **E-bit** must be clear in received Hello Packets, otherwise the **E-bit** must be set. A mismatch causes processing to stop and the packet to be dropped. The setting of the rest of the bits in the Hello Packet's Options field should be ignored.

At this point, an attempt is made to match the source of the Hello Packet to one of the receiving interface's neighbors. If the receiving interface is a multi-access network (either broadcast or non-broadcast) the source is identified by the IP source address found in the Hello's IP header. If the receiving interface is a point-to-point link or a virtual link, the source is identified by the Router ID found in the Hello's OSPF packet header. The interface's current list of neighbors is contained in the interface's data structure. If a matching neighbor structure cannot be found, (i.e., this is the first time the neighbor has been detected), one is created. The initial state of a newly created neighbor is set to **Down**.

When receiving an Hello Packet from a neighbor on a multi-access network (broadcast or non-broadcast), set the neighbor structure's Neighbor ID equal to the Router ID found in the packet's OSPF header. When receiving an Hello on a point-to-point network (but not on a virtual link) set the neighbor structure's Neighbor IP address to the packet's IP source address.

Now the rest of the Hello Packet is examined, generating events to be given to the neighbor and interface state machines. These state machines are specified either to be *executed* or *scheduled* (see Section 4.4). For example, by specifying below that the neighbor state machine be *executed* in line, several neighbor state transitions may be effected by a single received Hello:

- Each Hello Packet causes the neighbor state machine to be *executed* with the event **HelloReceived**.
- Then the list of neighbors contained in the Hello Packet is examined. If the router itself appears in this list, the neighbor state machine should be *executed* with the event **2-WayReceived**. Otherwise, the neighbor state machine should be executed with the event **1-WayReceived**, and the processing of the packet stops.
- Next, the Hello Packet's Router Priority field is examined. If this field is different than the one previously received from the neighbor, the receiving interface's state machine is *scheduled* with the event **NeighborChange**. In any case, the Router Priority field in the neighbor data structure should be updated accordingly.
- Next the Designated Router field in the Hello Packet is examined. If the neighbor is both declaring itself to be Designated Router (Designated Router field = Neighbor IP address) and the Backup Designated Router field in the packet is equal to `0.0.0.0` and the receiving interface is in state **Waiting**, the receiving interface's state machine is *scheduled* with the event **BackupSeen**. Otherwise, if the neighbor is declaring itself to be Designated Router and it had not previously, or the neighbor is not declaring itself Designated Router where it had previously, the receiving interface's state machine is *scheduled* with the event **NeighborChange**. In any case, the Neighbor's Designated Router item in the neighbor structure is updated accordingly.
- Finally, the Backup Designated Router field in the Hello Packet is examined. If the neighbor is declaring itself to be Backup Designated Router (Backup Designated Router field = Neighbor IP address) and the receiving interface is in state **Waiting**, the receiving interface's state machine is *scheduled* with the event **BackupSeen**. Otherwise, if the neighbor is declaring itself to be Backup Designated Router and it had not previously, or the neighbor is not declaring itself Backup Designated Router where it had previously, the receiving interface's state machine is *scheduled* with the event **NeighborChange**. In any case, the Neighbor's Backup Designated Router item in the neighbor structure is updated accordingly.

On non-broadcast multi-access networks, receipt of an Hello Packet may also cause an Hello Packet to be sent back to the neighbor in response. See Section 9.5.1 for more details.

10.6 Receiving Database Description Packets

This section explains the detailed processing of a received Database Description Packet. The incoming Database Description Packet has already been associated with a neighbor and receiving interface by the generic input packet processing (Section 8.2). The further processing of the Database Description Packet depends on the neighbor state. If the neighbor's state is **Down** or **Attempt** the packet should be ignored. Otherwise, if the state is:

Init The neighbor state machine should be *executed* with the event **2-WayReceived**. This causes an immediate state change to either state **2-Way** or state **ExStart**. If the new state is ExStart, the processing of the current packet should then continue in this new state by falling through to case ExStart below.

2-Way The packet should be ignored. Database Description Packets are used only for the purpose of bringing up adjacencies. [7]

ExStart If the received packet matches one of the following cases, then the neighbor state machine should be *executed* with the event **NegotiationDone** (causing the state to transition to **Exchange**), the packet's Options field should be recorded in the neighbor structure's **Neighbor Options** field and the packet should be accepted as next in sequence and processed further (see below). Otherwise, the packet should be ignored.

- The initialize(I), more (M) and master(MS) bits are set, the contents of the packet are empty, and the neighbor's Router ID is larger than the router's own. In this case the router is now Slave. Set the master/slave bit to slave, and set the DD sequence number to that specified by the master.
- The initialize(I) and master(MS) bits are off, the packet's DD sequence number equals the router's own DD sequence number (indicating acknowledgment) and the neighbor's Router ID is smaller than the router's own. In this case the router is Master.

Exchange If the state of the MS-bit is inconsistent with the master/slave state of the connection, generate the neighbor event **SeqNumberMismatch** and stop processing the packet. Otherwise:

- If the initialize(I) bit is set, generate the neighbor event **SeqNumberMismatch** and stop processing the packet.
- If the packet's Options field indicates a different set of optional OSPF capabilities than were previously received from the neighbor (recorded in the **Neighbor Options** field of the neighbor structure), generate the neighbor event **SeqNumberMismatch** and stop processing the packet.
- If the router is master, and the packet's DD sequence number equals the router's own DD sequence number (this packet is the next in sequence) the packet should be accepted and its contents processed (below).
- If the router is master, and the packet's DD sequence number is one less than the router's DD sequence number, the packet is a duplicate. Duplicates should be discarded by the master.
- If the router is slave, and the packet's DD sequence number is one more than the router's own DD sequence number (this packet is the next in sequence) the packet should be accepted and its contents processed (below).
- If the router is slave, and the packet's DD sequence number is equal to the router's DD sequence number, the packet is a duplicate. The slave must respond to duplicates by repeating the last Database Description packet that it had sent.
- Else, generate the neighbor event **SeqNumberMismatch** and stop processing the packet.

Loading or Full In this state, the router has sent and received an entire sequence of Database Description Packets. The only packets received should be duplicates (see above). In particular, the packet's Options field should match the set of optional OSPF capabilities previously indicated by the neighbor (stored in the neighbor structure's **Neighbor Options** field). Any other packets received, including the reception of a packet with the Initialize(I) bit set, should generate the neighbor event **SeqNumberMismatch**. [8] Duplicates should be

---

[7] When the identity of the Designated Router is changing, it may be quite common for a neighbor in this state to send the router a Database Description packet; this means that there is some momentary disagreement on the Designated Router's identity.

[8] Note that it is possible for a router to resynchronize any of its fully established adjacencies by setting the adjacency's state back to **ExStart**. This will cause the other end of the adjacency to process a **SeqNumberMismatch** event, and therefore to also go back to **ExStart** state.

discarded by the master. The slave must respond to duplicates by repeating the last Database Description packet that it had sent.

When the router accepts a received Database Description Packet as the next in sequence the packet contents are processed as follows. For each link state advertisement listed, the advertisement's LS type is checked for validity. If the **LS type** is unknown (e.g., not one of the LS types 1-5 defined by this specification), or if this is a AS external advertisement (LS type = 5) and the neighbor is associated with a stub area, generate the neighbor event **SeqNumberMismatch** and stop processing the packet. Otherwise, the router looks up the advertisement in its database to see whether it also has an instance of the link state advertisement. If it does not, or if the database copy is less recent (see Section 13.1), the link state advertisement is put on the **Link state request list** so that it can be requested (immediately or at some later time) in Link State Request Packets.

When the router accepts a received Database Description Packet as the next in sequence, it also performs the following actions, depending on whether it is master or slave:

Master Increments the DD sequence number. If the router has already sent its entire sequence of Database Description Packets, and the just accepted packet has the more bit (M) set to 0, the neighbor event **ExchangeDone** is generated. Otherwise, it should send a new Database Description to the slave.

Slave Sets the DD sequence number to the DD sequence number appearing in the received packet. The slave must send a Database Description Packet in reply. If the received packet has the more bit (M) set to 0, and the packet to be sent by the slave will also have the M-bit set to 0, the neighbor event **ExchangeDone** is generated. Note that the slave always generates this event before the master.

10.7 Receiving Link State Request Packets

This section explains the detailed processing of received Link State Request packets. Received Link State Request Packets specify a list of link state advertisements that the neighbor wishes to receive. Link State Request Packets should be accepted when the neighbor is in states **Exchange**, **Loading**, or **Full**. In all other states Link State Request Packets should be ignored.

Each link state advertisement specified in the Link State Request packet should be located in the router's database, and copied into Link State Update packets for transmission to the neighbor. These link state advertisements should NOT be placed on the **Link state retransmission list** for the neighbor. If a link state advertisement cannot be found in the database, something has gone wrong with the Database Exchange process, and neighbor event **BadLSReq** should be generated.

10.8 Sending Database Description Packets

This section describes how Database Description Packets are sent to a neighbor. The router's optional OSPF capabilities (see Section 4.5) are transmitted to the neighbor in the **Options** field of the Database Description packet. The router should maintain the same set of optional capabilities throughout the Database Exchange and flooding procedures. If for some reason the router's optional capabilities change, the Database Exchange procedure should be restarted by reverting to neighbor state **ExStart**. There are currently two optional capabilities defined. The **T-bit** should be set if and only if the router is capable of calculating separate routes for each IP TOS. The **E-bit** should be set if and only if the attached network belongs to a non-stub area. The rest of the **Options** field should be set to zero.

The sending of Database Description packets depends on the neighbor's state. In state **ExStart** the router sends empty Database Description packets, with the initialize (I), more (M) and master (MS) bits set. These packets are retransmitted every RxmtInterval seconds.

In state **Exchange** the Database Description Packets actually contain summaries of the link state information contained in the router's database. Each link state advertisement in the area's topological database (at the time the neighbor transitions into **Exchange** state) is listed in the neighbor **Database summary list**. When a new Database Description Packet is to be sent, the packet's DD sequence number is incremented, and the (new) top of the **Database summary list** is described by the packet. Items are removed from the **Database summary list** when the previous packet is acknowledged.

In state **Exchange**, the determination of when to send a Database Description packet depends on whether the router is master or slave:

Master Database Description packets are sent when either a) the slave acknowledges the previous Database Description packet by echoing the DD sequence number or b) RxmtInterval seconds elapse without an acknowledgment, in which case the previous Database Description packet is retransmitted.

Slave Database Description packets are sent only in response to Database Description packets received from the master. If the Database Description packet received from the master is new, a new Database Description packet is sent, otherwise the previous Database Description packet is resent.

In states **Loading** and **Full** the slave must resend its last Database Description packet in response to duplicate Database Description packets received from the master. For this reason the slave must wait RouterDeadInterval seconds before freeing the last Database Description packet. Reception of a Database Description packet from the master after this interval will generate a **SeqNumberMismatch** neighbor event.

10.9 Sending Link State Request Packets

In neighbor states **Exchange** or **Loading**, the **Link state request list** contains a list of those link state advertisements that need to be obtained from the neighbor. To request these advertisements, a router sends the neighbor the beginning of the **Link state request list**, packaged in a Link State Request packet.

When the neighbor responds to these requests with the proper Link State Update packet(s), the **Link state request list** is truncated and a new Link State Request packet is sent. This process continues until the Link state request list becomes empty. Unsatisfied Link State Request packets are retransmitted at intervals of RxmtInterval. There should be at most one Link State Request packet outstanding at any one time.

When the **Link state request list** becomes empty, and the neighbor state is **Loading** (i.e., a complete sequence of Database Description packets has been sent to and received from the neighbor), the **Loading Done** neighbor event is generated.

10.10 An Example

Figure 14 shows an example of an adjacency forming. Routers RT1 and RT2 are both connected to a broadcast network. It is assumed that RT2 is the Designated Router for the network, and that RT2 has a higher Router ID than Router RT1.

The neighbor state changes realized by each router are listed on the sides of the figure.

At the beginning of Figure 14, Router RT1's interface to the network becomes operational. It begins sending Hello Packets, although it doesn't know the identity of the Designated Router or of any other neighboring routers. Router RT2 hears this hello (moving the neighbor to **Init** state), and in its next Hello Packet indicates that it is itself the Designated Router and that it has heard Hello Packets from RT1. This in turn causes RT1 to go to state **ExStart**, as it starts to bring up the adjacency.

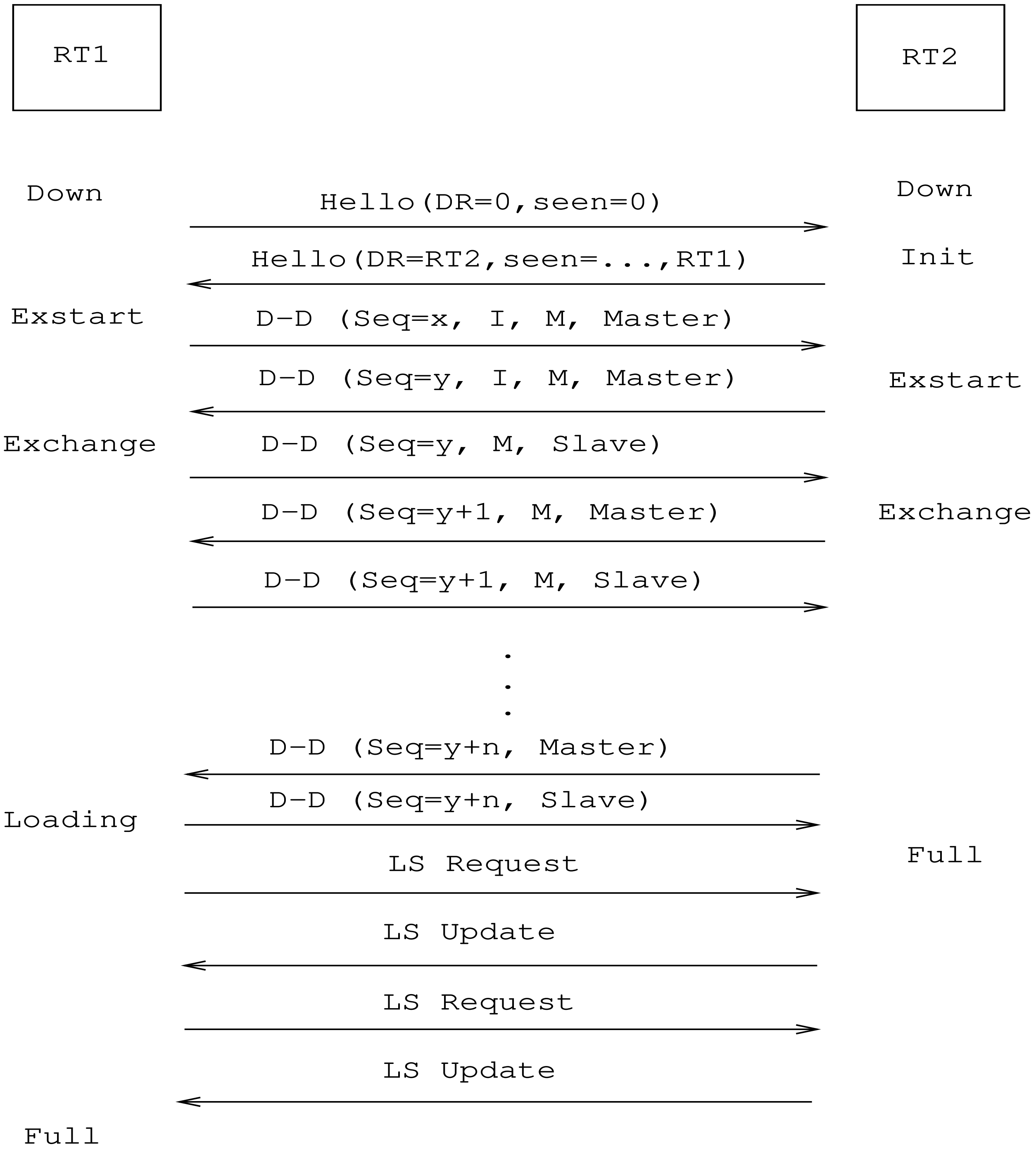


Figure 14: An adjacency bring-up example

RT1 begins by asserting itself as the master. When it sees that RT2 is indeed the master (because of RT2's higher Router ID), RT1 transitions to slave state and adopts its neighbor's DD sequence number. Database Description packets are then exchanged, with polls coming from the master (RT2) and responses from the slave (RT1). This sequence of Database Description Packets ends when both the poll and associated response has the M-bit off.

In this example, it is assumed that RT2 has a completely up to date database. In that case, RT2 goes immediately into **Full** state. RT1 will go into **Full** state after updating the necessary parts of its database. This is done by sending Link State Request Packets, and receiving Link State Update Packets in response. Note that, while RT1 has waited until a complete set of Database Description Packets has been received (from RT2) before sending any Link State Request Packets, this need not be the case. RT1 could have interleaved the sending of Link State Request Packets with the reception of Database Description Packets.

11 The Routing Table Structure

The routing table data structure contains all the information necessary to forward an IP data packet toward its destination. Each routing table entry describes the collection of best paths to a particular destination. When forwarding an IP data packet, the routing table entry providing the best match for the packet's IP destination is located. The matching routing table entry then provides the next hop towards the packet's destination. OSPF also provides for the existence of a default route (Destination ID = DefaultDestination, Address Mask = `0x00000000`). When the default route exists, it matches all IP destinations (although any other matching entry is a better match). Finding the routing table entry that best matches an IP destination is further described in Section 11.1.

There is a single routing table in each router. Two sample routing tables are described in Sections 11.2 and 11.3. The building of the routing table is discussed in Section 16.

The rest of this section defines the fields found in a routing table entry. The first set of fields describes the routing table entry's destination.

Destination Type The destination can be one of three types. Only the first type, Network, is actually used when forwarding IP data traffic. The other destinations are used solely as intermediate steps in the routing table build process.

- **Network** A range of IP addresses, to which IP data traffic may be forwarded. This includes IP networks (class A, B, or C), IP subnets, IP supernets and single IP hosts. The default route also falls in this category.
- **Area border router** Routers that are connected to multiple OSPF areas. Such routers originate summary link advertisements. These routing table entries are used when calculating the inter-area routes (see Section 16.2). These routing table entries may also be associated with configured virtual links.
- **AS boundary router** Routers that originate AS external link advertisements. These routing table entries are used when calculating the AS external routes (see Section 16.4).

Destination ID The destination's identifier or name. This depends on the Destination Type. For networks, the identifier is their associated IP address. For all other types, the identifier is the OSPF Router ID. [9]

Address Mask Only defined for networks. The network's IP address together with its address mask defines a range of IP addresses. For IP subnets, the address mask is referred to as the subnet mask. For host routes, the mask is "all ones" (`0xffffffff`).

---

[9] The address space of IP networks and the address space of OSPF Router IDs may overlap. That is, a network may have an IP address which is identical (when considered as a 32-bit number) to some router's Router ID.

Optional Capabilities When the destination is a router (either an area border router or an AS boundary router) this field indicates the optional OSPF capabilities supported by the destination router. The two optional capabilities currently defined by this specification are the ability to route based on IP TOS and the ability to process AS external link advertisements. For a further discussion of OSPF's optional capabilities, see Section 4.5.

The set of paths to use for a destination may vary based on IP Type of Service and the OSPF area to which the paths belong. This means that there may be multiple routing table entries for the same destination, depending on the values of the next two fields.

Type of Service There can be a separate set of routes for each IP Type of Service. The encoding of TOS in OSPF link state advertisements is described in Section 12.3.

Area This field indicates the area whose link state information has led to the routing table entry's collection of paths. This is called the entry's associated area. For sets of AS external paths, this field is not defined. For destinations of type "area border router", there may be separate sets of paths (and therefore separate routing table entries) associated with each of several areas. This will happen when two area border routers share multiple areas in common. For all other destination types, only the set of paths associated with the best area (the one providing the shortest route) is kept.

The rest of the routing table entry describes the set of paths to the destination. The following fields pertain to the set of paths as a whole. In other words, each one of the paths contained in a routing table entry is of the same path-type and cost (see below).

Path-type There are four possible types of paths used to route traffic to the destination, listed here in order of preference: intra-area, inter-area, type 1 external or type 2 external. Intra-area paths indicate destinations belonging to one of the router's attached areas. Inter-area paths are paths to destinations in other OSPF areas. These are discovered through the examination of received summary link advertisements. AS external paths are paths to destinations external to the AS. These are detected through the examination of received AS external link advertisements.

Cost The link state cost of the path to the destination. For all paths except type 2 external paths this describes the entire path's cost. For Type 2 external paths, this field describes the cost of the portion of the path internal to the AS. This cost is calculated as the sum of the costs of the path's constituent links.

Type 2 cost Only valid for type 2 external paths. For these paths, this field indicates the cost of the path's external portion. This cost has been advertised by an AS boundary router, and is the most significant part of the total path cost. For example, a type 2 external path with type 2 cost of 5 is always preferred over a path with type 2 cost of 10, regardless of the cost of the two paths' internal components.

Link State Origin Valid only for intra-area paths, this field indicates the link state advertisement (router links or network links) that directly references the destination. For example, if the destination is a transit network, this is the transit network's network links advertisement. If the destination is a stub network, this is the router links advertisement for the attached router. The advertisement is discovered during the shortest-path tree calculation (see Section 16.1). Multiple advertisements may reference the destination, however a tie-breaking scheme always reduces the choice to a single advertisement. The Link State Origin field is not used by the OSPF protocol, but it is used by the routing table calculation in OSPF's Multicast routing extensions (MOSPF).

When multiple paths of equal path-type and cost exist to a destination (called elsewhere "equal-cost" paths), they are stored in a single routing table entry. Each one of the "equal-cost" paths is distinguished by the following fields:

Next hop The outgoing router interface to use when forwarding traffic to the destination. On multi-access networks, the next hop also includes the IP address of the next router (if any) in the path towards the destination. This next router will always be one of the adjacent neighbors.

Advertising router Valid only for inter-area and AS external paths. This field indicates the Router ID of the router advertising the summary link or AS external link that led to this path.

11.1 Routing table lookup

When an IP data packet is received, an OSPF router finds the routing table entry that best matches the packet's destination. This routing table entry then provides the outgoing interface and next hop router to use in forwarding the packet. This section describes the process of finding the best matching routing table entry. The process consists of a number of steps, wherein the collection of routing table entries is progressively pruned. In the end, the single routing table entry remaining is the called best match.

Note that the steps described below may fail to produce a best match routing table entry (i.e., all existing routing table entries are pruned for some reason or another). In this case, the packet's IP destination is considered unreachable. Instead of being forwarded, the packet should be dropped and an ICMP destination unreachable message should be returned to the packet's source.

1. Select the complete set of "matching" routing table entries from the routing table. Each routing table entry describes a (set of) path(s) to a range of IP addresses. If the data packet's IP destination falls into an entry's range of IP addresses, the routing table entry is called a match. (It is quite likely that multiple entries will match the data packet. For example, a default route will match all packets.)
2. Suppose that the packet's IP destination falls into one of the router's configured **area address ranges** (see Section 3.5), and that the particular **area address range** is active. This means that there are one or more reachable (by intra-area paths) networks contained in the area address range. The packet's IP destination is then required to belong to one of these constituent networks. For this reason, only matching routing table entries with **path-type** of intra-area are considered (all others are pruned). If no such matching entries exist, the destination is unreachable (see above). Otherwise, skip to step 4.
3. Reduce the set of matching entries to those having the most preferential **path-type** (see Section 11). OSPF has a four level hierarchy of paths. Intra-area paths are the most preferred, followed in order by inter-area, type 1 external and type 2 external paths.
4. Select the remaining routing table entry that provides the longest (most specific) match. Another way of saying this is to choose the remaining entry that specifies the narrowest range of IP addresses. [10] For example, the entry for the address/mask pair of (`128.185.1.0, 0xffffff00`) is more specific than an entry for the pair (`128.185.0.0, 0xffff0000`). The default route is the least specific match, since it matches all destinations.
5. At this point, there may still be multiple routing table entries remaining. Each routing entry will specify the same range of IP addresses, but a different IP Type of Service. Select the routing table entry whose TOS value matches the TOS found in the packet header. If there is no routing table entry for this TOS, select the routing table entry for TOS 0. In other words, packets requesting TOS X are routed along the TOS 0 path if a TOS X path does not exist.

---

[10] It is assumed that, for two different address ranges matching the destination, one range is more specific than the other. Non-contiguous subnet masks can be configured to violate this assumption. Such subnet mask configurations cannot be handled by the OSPF protocol.

11.2 Sample routing table, without areas

Consider the Autonomous System pictured in Figure 2. No OSPF areas have been configured. A single metric is shown per outbound interface, indicating that routes will not vary based on TOS. The calculation of Router RT6's routing table proceeds as described in Section 2.1. The resulting routing table is shown in Table 12. Destination types are abbreviated: Network as "N", area border router as "BR" and AS boundary router as "ASBR".

There are no instances of multiple equal-cost shortest paths in this example. Also, since there are no areas, there are no inter-area paths.

Routers RT5 and RT7 are AS boundary routers. Intra-area routes have been calculated to Routers RT5 and RT7. This allows external routes to be calculated to the destinations advertised by RT5 and RT7 (i.e., Networks N12, N13, N14 and N15). It is assumed all AS external advertisements originated by RT5 and RT7 are advertising type 1 external metrics. This results in type 1 external paths being calculated to destinations N12-N15.

Type	*Dest*	*Area*	*Path Type*	*Cost*	*Next Hop(s)*	*Advertising Router(s)*
N	N1	0	intra-area	10	RT3	*
N	N2	0	intra-area	10	RT3	*
N	N3	0	intra-area	7	RT3	*
N	N4	0	intra-area	8	RT3	*
N	Ib	0	intra-area	7	*	*
N	Ia	0	intra-area	12	RT10	*
N	N6	0	intra-area	8	RT10	*
N	N7	0	intra-area	12	RT10	*
N	N8	0	intra-area	10	RT10	*
N	N9	0	intra-area	11	RT10	*
N	N10	0	intra-area	13	RT10	*
N	N11	0	intra-area	14	RT10	*
N	H1	0	intra-area	21	RT10	*
ASBR	RT5	0	intra-area	6	RT5	*
ASBR	RT7	0	intra-area	8	RT10	*
N	N12	*	type 1 external	10	RT10	RT7
N	N13	*	type 1 external	14	RT5	RT5
N	N14	*	type 1 external	14	RT5	RT5
N	N15	*	type 1 external	17	RT10	RT7

Table 12: The routing table for Router RT6 (no configured areas).

11.3 Sample routing table, with areas

Consider the previous example, this time split into OSPF areas. An OSPF area configuration is pictured in Figure 6. Router RT4's routing table will be described for this area configuration. Router RT4 has a connection to Area 1 and a backbone connection. This causes Router RT4 to view the AS as the concatenation of the two graphs shown in Figures 7 and 8. The resulting routing table is displayed in Table 13.

Again, Routers RT5 and RT7 are AS boundary routers. Routers RT3, RT4, RT7, RT10 and RT11 are area border routers. Note that there are two routing table entries (in this case having identical paths) for Router RT7, in its dual capacities as an area border router and an AS boundary router. Note also that there are two routing entries for the area border router RT3, since it has two areas in common with RT4 (Area 1 and the backbone).

Backbone paths have been calculated to all area border routers (BR). These are used when determining the inter-area routes. Note that all of the inter-area routes are associated with the backbone; this is always the case when the calculating router is itself an area border router. Routing information is condensed at area boundaries. In this example, we assume that Area 3 has been defined so that networks N9-N11 and the host route to H1 are all condensed to a single route when advertised into the backbone (by Router RT11). Note that the cost of this route is the minimum of the set of costs to its individual components.

There is a virtual link configured between Routers RT10 and RT11. Without this configured virtual link, RT11 would be unable to advertise a route for networks N9-N11 and Host H1 into the backbone, and there would not be an entry for these networks in Router RT4's routing table.

In this example there are two equal-cost paths to Network N12. However, they both use the same next hop (Router RT5).

Type	*Dest*	*Area*	*Path Type*	*Cost*	*Next Hop(s)*	*Advertising Router(s)*
N	N1	1	intra-area	4	RT1	*
N	N2	1	intra-area	4	RT2	*
N	N3	1	intra-area	1	*	*
N	N4	1	intra-area	3	RT3	*
BR	RT3	1	intra-area	1	*	*
N	Ib	0	intra-area	22	RT5	*
N	Ia	0	intra-area	27	RT5	*
BR	RT3	0	intra-area	21	RT5	*
BR	RT7	0	intra-area	14	RT5	*
BR	RT10	0	intra-area	22	RT5	*
BR	RT11	0	intra-area	25	RT5	*
ASBR	RT5	0	intra-area	8	*	*
ASBR	RT7	0	intra-area	14	RT5	*
N	N6	0	inter-area	15	RT5	RT7
N	N7	0	inter-area	19	RT5	RT7
N	N8	0	inter-area	18	RT5	RT7
N	N9-N11,H1	0	inter-area	26	RT5	RT11
N	N12	*	type 1 external	16	RT5	RT5,RT7
N	N13	*	type 1 external	16	RT5	RT5
N	N14	*	type 1 external	16	RT5	RT5
N	N15	*	type 1 external	23	RT5	RT7

Table 13: Router RT4's routing table in the presence of areas.

Router RT4's routing table would improve (i.e., some of the paths in the routing table would become shorter) if an additional virtual link were configured between Router RT4 and Router RT3. The new virtual link would itself be associated with the first entry for area border router RT3 in Table 13 (an intra-area path through Area 1). This would yield a cost of 1 for the virtual link. The routing table entries changes that would be caused by the addition of this virtual link are shown in Table 14.

Type	*Dest*	*Area*	*Path Type*	*Cost*	*Next Hop(s)*	*Advertising Router(s)*
N	Ib	0	intra-area	16	RT3	*
N	Ia	0	intra-area	21	RT3	*
BR	RT3	0	intra-area	1	*	*
BR	RT10	0	intra-area	16	RT3	*
BR	RT11	0	intra-area	19	RT3	*
N	N9-N11,H1	0	inter-area	20	RT3	RT11

Table 14: Changes resulting from an additional virtual link.

12 Link State Advertisements

Each router in the Autonomous System originates one or more link state advertisements. There are five distinct types of link state advertisements, which are described in Section 4.3. The collection of link state advertisements forms the link state or topological database. Each separate type of advertisement has a separate function. Router links and network links advertisements describe how an area's routers and networks are interconnected. Summary link advertisements provide a way of condensing an area's routing information. AS external advertisements provide a way of transparently advertising externally-derived routing information throughout the Autonomous System.

Each link state advertisement begins with a standard 20-byte header. This link state header is discussed below.

12.1 The Link State Advertisement Header

The link state advertisement header contains the LS type, Link State ID and Advertising Router fields. The combination of these three fields uniquely identifies the link state advertisement.

There may be several instances of an advertisement present in the Autonomous System, all at the same time. It must then be determined which instance is more recent. This determination is made by examining the LS sequence, LS checksum and LS age fields. These fields are also contained in the 20-byte link state advertisement header.

Several of the OSPF packet types list link state advertisements. When the instance is not important, an advertisement is referred to by its LS type, Link State ID and Advertising Router (see Link State Request Packets). Otherwise, the LS sequence number, LS age and LS checksum fields must also be referenced.

A detailed explanation of the fields contained in the link state advertisement header follows.

12.1.1 LS age

This field is the age of the link state advertisement in seconds. It should be processed as an unsigned 16-bit integer. It is set to 0 when the link state advertisement is originated. It must be incremented by InfTransDelay on every hop of the flooding procedure. Link state advertisements are also aged as they are held in each router's database.

The age of a link state advertisement is never incremented past MaxAge. Advertisements having age MaxAge are not used in the routing table calculation. When an advertisement's age first reaches MaxAge, it is reflooded. A link state advertisement of age MaxAge is finally flushed from the database when it is no longer needed to ensure database synchronization. For more information on the aging of link state advertisements, consult Section 14.

The LS age field is examined when a router receives two instances of a link state advertisement, both having identical LS sequence numbers and LS checksums. An instance of age MaxAge is then always accepted as most recent; this allows old advertisements to be flushed quickly from the routing domain. Otherwise, if the ages differ by more than MaxAgeDiff, the instance having the smaller age is accepted as most recent. [11] See Section 13.1 for more details.

12.1.2 Options

The Options field in the link state advertisement header indicates which optional capabilities are associated with the advertisement. OSPF's optional capabilities are described in Section 4.5. There are currently two optional capabilities defined; they are represented by the **T-bit** and **E-bit** found in the Options field. The rest of the Options field should be set to zero.

The **E-bit** represents OSPF's ExternalRoutingCapability. This bit should be set in all advertisements associated with the backbone, and all advertisements associated with non-stub areas (see Section 3.6). It should also be set in all AS external link advertisements. It should be reset in all router links, network links and summary link advertisements associated with a stub area. For all link state advertisements, the setting of the **E-bit** is for informational purposes only; it does not affect the routing table calculation.

The **T-bit** represents OSPF's TOS routing capability. This bit should be set in a router links advertisement if and only if the router is capable of calculating separate routes for each IP TOS (see Section 2.4). The **T-bit** should always be set in network links advertisements. It should be set in summary link and AS external link advertisements if and only if the advertisement describes paths for all TOS values, instead of just the TOS 0 path. Note that, with the **T-bit** set, there may still be only a single metric in the advertisement (the TOS 0 metric). This would mean that paths for non-zero TOS exist, but are equivalent to the TOS 0 path. A link state advertisement's **T-bit** is examined when calculating the routing table's non-zero TOS paths (see Section 16.9).

12.1.3 LS type

The LS type field dictates the format and function of the link state advertisement. Advertisements of different types have different names (e.g., router links or network links). All advertisement types, except the AS external link advertisements (LS type = 5), are flooded throughout a single area only. AS external link advertisements are flooded throughout the entire Autonomous System, excepting stub areas (see Section 3.6). Each separate advertisement type is briefly described below in Table 15.

12.1.4 Link State ID

This field identifies the piece of the routing domain that is being described by the advertisement. Depending on the advertisement's LS type, the Link State ID takes on the values listed in Table 16.

Actually, for Type 3 summary link (LS type = 3) advertisements and AS external link (LS type = 5) advertisements, the Link State ID may additionally have one or more of the destination network's "host" bits set. For example, when originating an AS external link for the network `10.0.0.0` with mask of `255.0.0.0`, the Link State ID can be set to anything in the range `10.0.0.0` through `10.255.255.255` inclusive (although `10.0.0.0` should be used whenever possible). The freedom to set certain host bits allows a router to originate separate advertisements for two networks having the same address but different masks. See Appendix F for details.

When the link state advertisement is describing a network (LS type = 2, 3 or 5), the network's IP address is easily derived by masking the Link State ID with the network/subnet mask contained in the body of the link state

---

[11] MaxAgeDiff is an architectural constant. It indicates the maximum dispersion of ages, in seconds, that can occur for a single link state instance as it is flooded throughout the routing domain. If two advertisements differ by more than this, they are assumed to be different instances of the same advertisement. This can occur when a router restarts and loses track of the advertisement's previous LS sequence number. See Section 13.4 for more details.

LS type	*Advertisement description*
1	These are the router links advertisements. They describe the collected states of the router's interfaces. For more information, consult Section 12.4.1.
2	These are the network links advertisements. They describe the set of routers attached to the network. For more information, consult Section 12.4.2.
3 or 4	These are the summary link advertisements. They describe inter-area routes, and enable the condensation of routing information at area borders. Originated by area border routers, the Type 3 advertisements describe routes to networks while the Type 4 advertisements describe routes to AS boundary routers.
5	These are the AS external link advertisements. Originated by AS boundary routers, they describe routes to destinations external to the Autonomous System. A default route for the Autonomous System can also be described by an AS external link advertisement.

Table 15: OSPF link state advertisements.

LS Type	*Link State ID*
1	The originating router's Router ID.
2	The IP interface address of the network's Designated Router.
3	The destination network's IP address.
4	The Router ID of the described AS boundary router.
5	The destination network's IP address.

Table 16: The advertisement's Link State ID.

advertisement. When the link state advertisement is describing a router (LS type = 1 or 4), the Link State ID is always the described router's OSPF Router ID.

When an AS external advertisement (LS Type = 5) is describing a default route, its Link State ID is set to DefaultDestination (`0.0.0.0`).

12.1.5 Advertising Router

This field specifies the OSPF Router ID of the advertisement's originator. For router links advertisements, this field is identical to the Link State ID field. Network link advertisements are originated by the network's Designated Router. Summary link advertisements are originated by area border routers. AS external link advertisements are originated by AS boundary routers.

12.1.6 LS sequence number

The sequence number field is a signed 32-bit integer. It is used to detect old and duplicate link state advertisements. The space of sequence numbers is linearly ordered. The larger the sequence number (when compared as signed 32-bit integers) the more recent the advertisement. To describe to sequence number space more precisely, let N refer in the discussion below to the constant 2^{31}.

The sequence number $-N$ (`0x80000000`) is reserved (and unused). This leaves $-N + 1$ (`0x80000001`) as the smallest (and therefore oldest) sequence number. A router uses this sequence number the first time it originates any link state advertisement. Afterwards, the advertisement's sequence number is incremented each time the router originates a new instance of the advertisement. When an attempt is made to increment the sequence number past the maximum value of $N - 1$ (`0x7fffffff`), the current instance of the advertisement must first be flushed from the routing domain. This is done by prematurely aging the advertisement (see Section 14.1) and reflooding it. As soon as this flood has been acknowledged by all adjacent neighbors, a new instance can be originated with sequence number of $-N + 1$ (`0x80000001`).

The router may be forced to promote the sequence number of one of its advertisements when a more recent instance of the advertisement is unexpectedly received during the flooding process. This should be a rare event. This may indicate that an out-of-date advertisement, originated by the router itself before its last restart/reload, still exists in the Autonomous System. For more information see Section 13.4.

12.1.7 LS checksum

This field is the checksum of the complete contents of the advertisement, excepting the LS age field. The LS age field is excepted so that an advertisement's age can be incremented without updating the checksum. The checksum used is the same that is used for ISO connectionless datagrams; it is commonly referred to as the Fletcher checksum. It is documented in Annex B of [RFC 905]. The link state advertisement header also contains the length of the advertisement in bytes; subtracting the size of the LS age field (two bytes) yields the amount of data to checksum.

The checksum is used to detect data corruption of an advertisement. This corruption can occur while an advertisement is being flooded, or while it is being held in a router's memory. The LS checksum field cannot take on the value of zero; the occurrence of such a value should be considered a checksum failure. In other words, calculation of the checksum is not optional.

The checksum of a link state advertisement is verified in two cases: a) when it is received in a Link State Update Packet and b) at times during the aging of the link state database. The detection of a checksum failure leads to separate actions in each case. See Sections 13 and 14 for more details.

Whenever the LS sequence number field indicates that two instances of an advertisement are the same, the LS checksum field is examined. If there is a difference, the instance with the larger LS checksum is considered to be most recent. [12] See Section 13.1 for more details.

12.2 The link state database

A router has a separate link state database for every area to which it belongs. The link state database has been referred to elsewhere in the text as the topological database. All routers belonging to the same area have identical topological databases for the area.

---

[12] When two advertisements have different LS checksums, they are assumed to be separate instances. This can occur when a router restarts, and loses track of the advertisement's previous LS sequence number. In the case where the two advertisements have the same LS sequence number, it is not possible to determine which link state is actually newer. If the wrong advertisement is accepted as newer, the originating router will originate another instance. See Section 13.4 for further details.

The databases for each individual area are always dealt with separately. The shortest path calculation is performed separately for each area (see Section 16). Components of the area topological database are flooded throughout the area only. Finally, when an adjacency (belonging to Area A) is being brought up, only the database for Area A is synchronized between the two routers.

The area database is composed of router links advertisements, network links advertisements, and summary link advertisements (all listed in the area data structure). In addition, external routes (AS external advertisements) are included in all non-stub area databases (see Section 3.6).

An implementation of OSPF must be able to access individual pieces of an area database. This lookup function is based on an advertisement's LS type, Link State ID and Advertising Router. [13] There will be a single instance (the most up-to-date) of each link state advertisement in the database. The database lookup function is invoked during the link state flooding procedure (Section 13) and the routing table calculation (Section 16). In addition, using this lookup function the router can determine whether it has itself ever originated a particular link state advertisement, and if so, with what LS sequence number.

A link state advertisement is added to a router's database when either a) it is received during the flooding process (Section 13) or b) it is originated by the router itself (Section 12.4). A link state advertisement is deleted from a router's database when either a) it has been overwritten by a newer instance during the flooding process (Section 13) or b) the router originates a newer instance of one of its self-originated advertisements (Section 12.4) or c) the advertisement ages out and is flushed from the routing domain (Section 14). Whenever a link state advertisement is deleted from the database it must also be removed from all neighbors' **Link state retransmission lists** (see Section 10).

12.3 Representation of TOS

All OSPF link state advertisements (with the exception of network links advertisements) specify metrics. In router links advertisements, the metrics indicate the costs of the described interfaces. In summary link and AS external link advertisements, the metric indicates the cost of the described path. In all of these advertisements, a separate metric can be specified for each IP TOS. The encoding of TOS in OSPF link state advertisements is specified in Table 17. That table relates the OSPF encoding to the IP packet header's TOS field (defined in [RFC 1349]). The OSPF encoding is expressed as a decimal integer, and the IP packet header's TOS field is expressed in the binary TOS values used in [RFC 1349].

Each OSPF link state advertisement must specify the TOS 0 metric. Other TOS metrics, if they appear, must appear in order of increasing TOS encoding. For example, the TOS 8 (maximize throughput) metric must always appear before the TOS 16 (minimize delay) metric when both are specified. If a metric for some non-zero TOS is not specified, its cost defaults to the cost for TOS 0, unless the **T-bit** is reset in the advertisement's Options field (see Section 12.1.2 for more details).

12.4 Originating link state advertisements

Into any given OSPF area, a router will originate several link state advertisements. Each router originates a router links advertisement. If the router is also the Designated Router for any of the area's networks, it will originate network links advertisements for those networks.

Area border routers originate a single summary link advertisement for each known inter-area destination. AS boundary routers originate a single AS external link advertisement for each known AS external destination.

---

[13] There is one instance where a lookup must be done based on partial information. This is during the routing table calculation, when a network links advertisement must be found based solely on its Link State ID. The lookup in this case is still well defined, since no two network links advertisements can have the same Link State ID.

OSPF encoding	*RFC 1349 TOS values*
0	0000 normal service
2	0001 minimize monetary cost
4	0010 maximize reliability
6	0011
8	0100 maximize throughput
10	0101
12	0110
14	0111
16	1000 minimize delay
18	1001
20	1010
22	1011
24	1100
26	1101
28	1110
30	1111

Table 17: Representing TOS in OSPF.

Destinations are advertised one at a time so that the change in any single route can be flooded without reflooding the entire collection of routes. During the flooding procedure, many link state advertisements can be carried by a single Link State Update packet.

As an example, consider Router RT4 in Figure 6. It is an area border router, having a connection to Area 1 and the backbone. Router RT4 originates 5 distinct link state advertisements into the backbone (one router links, and one summary link for each of the networks N1-N4). Router RT4 will also originate 8 distinct link state advertisements into Area 1 (one router links and seven summary link advertisements as pictured in Figure 7). If RT4 has been selected as Designated Router for Network N3, it will also originate a network links advertisement for N3 into Area 1.

In this same figure, Router RT5 will be originating 3 distinct AS external link advertisements (one for each of the networks N12-N14). These will be flooded throughout the entire AS, assuming that none of the areas have been configured as stubs. However, if area 3 has been configured as a stub area, the external advertisements for networks N12-N14 will not be flooded into area 3 (see Section 3.6). Instead, Router RT11 would originate a default summary link advertisement that would be flooded throughout area 3 (see Section 12.4.3). This instructs all of area 3's internal routers to send their AS external traffic to RT11.

Whenever a new instance of a link state advertisement is originated, its LS sequence number is incremented, its LS age is set to 0, its LS checksum is calculated, and the advertisement is added to the link state database and flooded out the appropriate interfaces. See Section 13.2 for details concerning the installation of the advertisement into the link state database. See Section 13.3 for details concerning the flooding of newly originated advertisements.

The ten events that can cause a new instance of a link state advertisement to be originated are:

1. *The LS age field of one of the router's self-originated advertisements reaches the value LSRefreshTime.* In this case, a new instance of the link state advertisement is originated, even though the contents of the advertisement (apart from the link state advertisement header) will be the same. This guarantees periodic originations of all link state advertisements. This periodic updating of link state advertisements adds robustness to the link state algorithm. Link state advertisements that solely describe unreachable destinations should not be refreshed, but should instead be flushed from the routing domain (see Section 14.1).

When whatever is being described by a link state advertisement changes, a new advertisement is originated. However, two instances of the same link state advertisement may not be originated within the time period MinLSInterval. This may require that the generation of the next instance be delayed by up to MinLSInterval. The following events may cause the contents of a link state advertisement to change. These events should cause new originations if and only if the contents of the new advertisement would be different:

2. *An interface's state changes* (see Section 9.1). This may mean that it is necessary to produce a new instance of the router links advertisement.
3. *An attached network's Designated Router changes.* A new router links advertisement should be originated. Also, if the router itself is now the Designated Router, a new network links advertisement should be produced. If the router itself is no longer the Designated Router, any network links advertisement that it might have originated for the network should be flushed from the routing domain (see Section 14.1).
4. *One of the neighboring routers changes to/from the FULL state.* This may mean that it is necessary to produce a new instance of the router links advertisement. Also, if the router is itself the Designated Router for the attached network, a new network links advertisement should be produced.

The next four events concern area border routers only:

5. *An intra-area route has been added/deleted/modified* in the routing table. This may cause a new instance of a summary links advertisement (for this route) to be originated in each attached area (possibly including the backbone).
6. *An inter-area route has been added/deleted/modified* in the routing table. This may cause a new instance of a summary links advertisement (for this route) to be originated in each attached area (but NEVER for the backbone).
7. *The router becomes newly attached to an area.* The router must then originate summary link advertisements into the newly attached area for all pertinent intra-area and inter-area routes in the router's routing table. See Section 12.4.3 for more details.
8. *When the state of one of the router's configured virtual links changes,* it may be necessary to originate a new router links advertisement into the virtual link's transit area (see the discussion of the router links advertisement's bit V in Section 12.4.1), as well as originating a new router links advertisement into the backbone.

The last two events concern AS boundary routers (and former AS boundary routers) only:

9. *An external route gained through direct experience* with an external routing protocol (like EGP) changes. This will cause an AS boundary router to originate a new instance of an AS external link advertisement.
10. *A router ceases to be an AS boundary router*, perhaps after restarting. In this situation the router should flush all AS external link advertisements that it had previously originated. These advertisements can be flushed via the premature aging procedure specified in Section 14.1.

The construction of each type of link state advertisement is explained in detail below. In general, these sections describe the contents of the advertisement body (i.e., the part coming after the 20-byte advertisement header). For information concerning the building of the link state advertisement header, see Section 12.1.

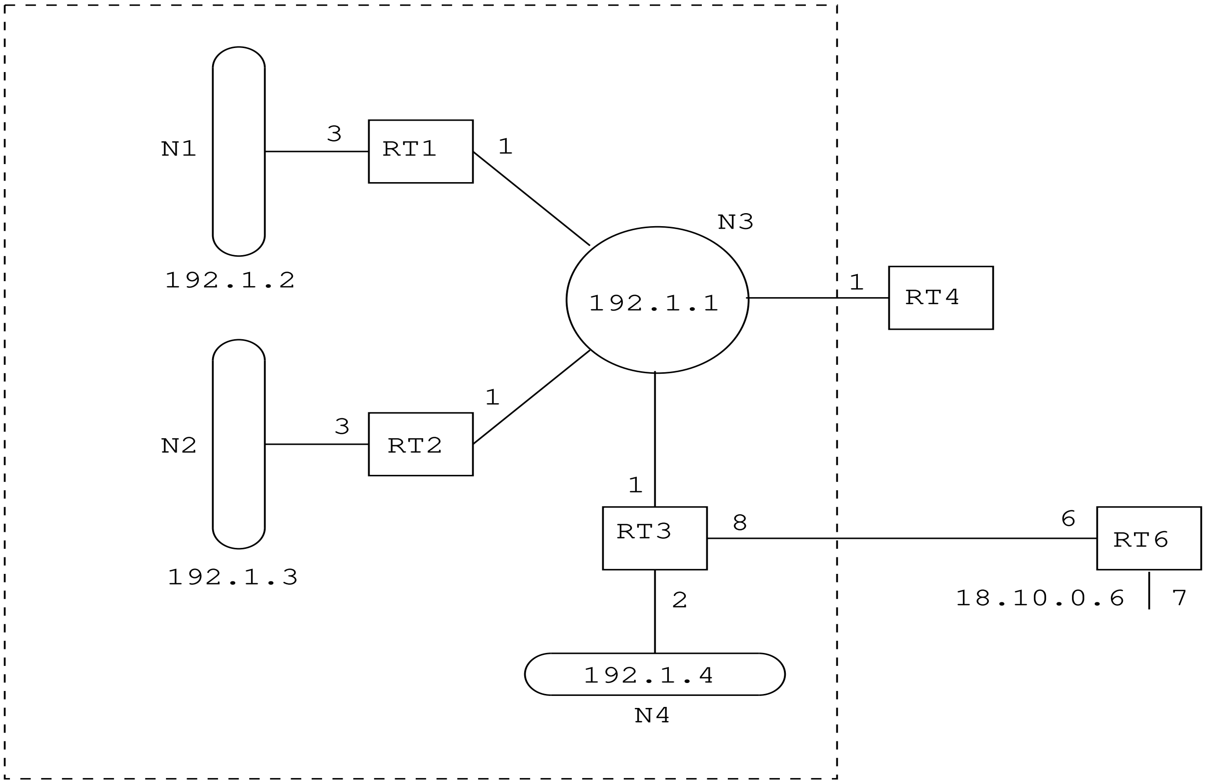


Figure 15: Area 1 with IP addresses shown

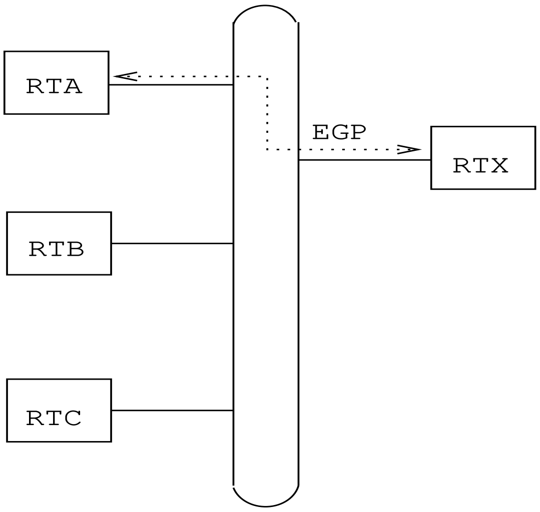


Figure 16: Forwarding address example

12.4.1 Router links

A router originates a router links advertisement for each area that it belongs to. Such an advertisement describes the collected states of the router's links to the area. The advertisement is flooded throughout the particular area, and no further.

The format of a router links advertisement is shown in Appendix A (Section A.4.2). The first 20 bytes of the advertisement consist of the generic link state advertisement header that was discussed in Section 12.1. Router links advertisements have LS type = 1. The router indicates whether it is willing to calculate separate routes for each IP TOS by setting (or resetting) the **T-bit** of the link state advertisement's **Options** field.

A router also indicates whether it is an area border router, or an AS boundary router, by setting the appropriate bits (**bit B** and **bit E**, respectively) in its router links advertisements. This enables paths to those types of routers to be saved in the routing table, for later processing of summary link advertisements and AS external link advertisements. Bit B should be set whenever the router is actively attached to two or more areas, even if the router is not currently attached to the OSPF backbone area. Bit E should never be set in a router links advertisement for a stub area (stub areas cannot contain AS boundary routers). In addition, the router sets **bit V** in its router links advertisement for Area A if and only if it is the endpoint of an active virtual link using Area A as its Transit area. This enables the other routers attached to Area A to discover whether the area supports any virtual links (i.e., is a transit area).

The router links advertisement then describes the router's working connections (i.e., interfaces or links) to the area. Each link is typed according to the kind of attached network. Each link is also labelled with its Link ID. This Link ID gives a name to the entity that is on the other end of the link. Table 18 summarizes the values used for the Type and Link ID fields.

Link type	*Description*	*Link ID*
1	Point-to-point link	Neighbor Router ID
2	Link to transit network	Interface address of Designated Router
3	Link to stub network	IP network number
4	Virtual link	Neighbor Router ID

Table 18: Link descriptions in the router links advertisement.

In addition, the Link Data field is specified for each link. This field gives 32 bits of extra information for the link. For links to transit networks, numbered links to routers and virtual links, this field specifies the IP interface address of the associated router interface (this is needed by the routing table calculation, see Section 16.1.1). For links to stub networks, this field specifies the network's IP address mask. For unnumbered point-to-point networks, the Link Data field should be set to the unnumbered interface's MIB-II [RFC 1213] ifIndex value.

Finally, the cost of using the link for output (possibly specifying a different cost for each Type of Service) is specified. The output cost of a link is configurable. It must always be non-zero.

To further describe the process of building the list of link descriptions, suppose a router wishes to build a router links advertisement for Area A. The router examines its collection of interface data structures. For each interface, the following steps are taken:

- If the attached network does not belong to Area A, no links are added to the advertisement, and the next interface should be examined.
- Else, if the state of the interface is **Down**, no links are added.

- Else, if the state of the interface is **Point-to-Point**, then add links according to the following:
  - If the neighboring router is fully adjacent, add a Type 1 link (point-to-point) if this is an interface to a point-to-point network, or add a Type 4 link (virtual link) if this is a virtual link. The Link ID should be set to the Router ID of the neighboring router. For virtual links and numbered point-to-point networks, the Link Data should specify the IP interface address. For unnumbered point-to-point networks, the Link Data field should specify the interface's MIB-II [RFC 1213] ifIndex value.
  - If this is a numbered point-to-point network (i.e, not a virtual link and not an unnumbered point-to-point network) and the neighboring router's IP address is known, add a Type 3 link (stub network) whose Link ID is the neighbor's IP address, whose Link Data is the mask `0xffffffff` indicating a host route, and whose cost is the interface's configured output cost.
- Else if the state of the interface is **Loopback**, add a Type 3 link (stub network) as long as this is not an interface to an unnumbered serial line. The Link ID should be set to the IP interface address, the Link Data set to the mask `0xffffffff` (indicating a host route), and the cost set to 0.
- Else if the state of the interface is **Waiting**, add a Type 3 link (stub network) whose Link ID is the IP network number of the attached network and whose Link Data is the attached network's address mask.
- Else, there has been a Designated Router selected for the attached network. If the router is fully adjacent to the Designated Router, or if the router itself is Designated Router and is fully adjacent to at least one other router, add a single Type 2 link (transit network) whose Link ID is the IP interface address of the attached network's Designated Router (which may be the router itself) and whose Link Data is the router's own IP interface address. Otherwise, add a link as if the interface state were **Waiting** (see above).

Unless otherwise specified, the cost of each link generated by the above procedure is equal to the output cost of the associated interface. Note that in the case of serial lines, multiple links may be generated by a single interface.

After consideration of all the router interfaces, host links are added to the advertisement by examining the list of attached hosts. A host route is represented as a Type 3 link (stub network) whose Link ID is the host's IP address and whose Link Data is the mask of all ones (`0xffffffff`).

As an example, consider the router links advertisements generated by Router RT3, as pictured in Figure 6. The area containing Router RT3 (Area 1) has been redrawn, with actual network addresses, in Figure 15. Assume that the last byte of all of RT3's interface addresses is 3, giving it the interface addresses 192.1.1.3 and 192.1.4.3, and that the other routers have similar addressing schemes. In addition, assume that all links are functional, and that Router IDs are assigned as the smallest IP interface address.

RT3 originates two router links advertisements, one for Area 1 and one for the backbone. Assume that Router RT4 has been selected as the Designated router for network `192.1.1.0`. RT3's router links advertisement for Area 1 is then shown below. It indicates that RT3 has two connections to Area 1, the first a link to the transit network `192.1.1.0` and the second a link to the stub network `192.1.4.0`. Note that the transit network is identified by the IP interface of its Designated Router (i.e., the Link ID = `192.1.1.4` which is the Designated Router RT4's IP interface to `192.1.1.0`). Note also that RT3 has indicated that it is capable of calculating separate routes based on IP TOS, through setting the **T-bit** in the Options field. It has also indicated that it is an area border router.

```
; RT3's router links advertisement for Area 1

LS age = 0                     ;always true on origination
Options = (T-bit|E-bit)        ;TOS-capable
LS type = 1                    ;indicates router links
Link State ID = 192.1.1.3      ;RT3's Router ID
```

```
Advertising Router = 192.1.1.3 ;RT3's Router ID
bit E = 0                      ;not an AS boundary router
bit B = 1                      ;RT3 is an area border router
#links = 2
        Link ID = 192.1.1.4         ;IP address of Designated Router
        Link Data = 192.1.1.3       ;RT3's IP interface to net
        Type = 2                    ;connects to transit network
        # other metrics = 0
        TOS 0 metric = 1

        Link ID = 192.1.4.0         ;IP Network number
        Link Data = 0xffffff00      ;Network mask
        Type = 3                    ;connects to stub network
        # other metrics = 0
        TOS 0 metric = 2
```

Next RT3's router links advertisement for the backbone is shown. It indicates that RT3 has a single attachment to the backbone. This attachment is via an unnumbered point-to-point link to Router RT6. RT3 has again indicated that it is TOS-capable, and that it is an area border router.

```
; RT3's router links advertisement for the backbone

LS age = 0                     ;always true on origination
Options = (T-bit|E-bit)        ;TOS-capable
LS type = 1                    ;indicates router links
Link State ID = 192.1.1.3      ;RT3's router ID
Advertising Router = 192.1.1.3 ;RT3's router ID
bit E = 0                      ;not an AS boundary router
bit B = 1                      ;RT3 is an area border router
#links = 1
        Link ID = 18.10.0.6         ;Neighbor's Router ID
        Link Data = 0.0.0.3         ;MIB-II ifIndex of P-P link
        Type = 1                    ;connects to router
        # other metrics = 0
        TOS 0 metric = 8
```

Even though Router RT3 has indicated that it is TOS-capable in the above examples, only a single metric (the TOS 0 metric) has been specified for each interface. Different metrics can be specified for each TOS. The encoding of TOS in OSPF link state advertisements is described in Section 12.3.

As an example, suppose the point-to-point link between Routers RT3 and RT6 in Figure 15 is a satellite link. The AS administrator may want to encourage the use of the line for high bandwidth traffic. This would be done by setting the metric artificially low for the appropriate TOS value. Router RT3 would then originate the following router links advertisement for the backbone (TOS 8 = maximize throughput):

```
; RT3's router links advertisement for the backbone

LS age = 0                  ;always true on origination
Options = (T-bit|E-bit)     ;TOS-capable
LS type = 1                 ;indicates router links
Link State ID = 192.1.1.3   ;RT3's Router ID
Advertising Router = 192.1.1.3
```

```
bit E = 0                  ;not an AS boundary router
bit B = 1                  ;RT3 is an area border router
#links = 1
        Link ID = 18.10.0.6         ;Neighbor's Router ID
        Link Data = 0.0.0.3         ;MIB-II ifIndex of P-P link
        Type = 1                    ;connects to router
        # other metrics = 1
        TOS 0 metric = 8
                TOS = 8             ;Maximize throughput
                metric = 1          ;traffic preferred
```

12.4.2 Network links

A network links advertisement is generated for every transit multi-access network. (A transit network is a network having two or more attached routers). The network links advertisement describes all the routers that are attached to the network.

The Designated Router for the network originates the advertisement. The Designated Router originates the advertisement only if it is fully adjacent to at least one other router on the network. The network links advertisement is flooded throughout the area that contains the transit network, and no further. The networks links advertisement lists those routers that are fully adjacent to the Designated Router; each fully adjacent router is identified by its OSPF Router ID. The Designated Router includes itself in this list.

The Link State ID for a network links advertisement is the IP interface address of the Designated Router. This value, masked by the network's address mask (which is also contained in the network links advertisement) yields the network's IP address.

A router that has formerly been the Designated Router for a network, but is no longer, should flush the network links advertisement that it had previously originated. This advertisement is no longer used in the routing table calculation. It is flushed by prematurely incrementing the advertisement's age to MaxAge and reflooding (see Section 14.1). In addition, in those rare cases where a router's Router ID has changed, any network links advertisements that were originated with the router's previous Router ID must be flushed. Since the router may have no idea what it's previous Router ID might have been, these network links advertisements are indicated by having their Link State ID equal to one of the router's IP interface addresses and their Advertising Router not equal to the router's current Router ID (see Section 13.4 for more details).

As an example of a network links advertisement, again consider the area configuration in Figure 6. Network links advertisements are originated for Network N3 in Area 1, Networks N6 and N8 in Area 2, and Network N9 in Area 3. Assuming that Router RT4 has been selected as the Designated Router for Network N3, the following network links advertisement is generated by RT4 on behalf of Network N3 (see Figure 15 for the address assignments):

```
; network links advertisement for Network N3

LS age = 0                      ;always true on origination
Options = (T-bit|E-bit)         ;TOS-capable
LS type = 2                     ;indicates network links
Link State ID = 192.1.1.4       ;IP address of Designated Router
Advertising Router = 192.1.1.4 ;RT4's Router ID
Network Mask = 0xffffff00
        Attached Router = 192.1.1.4    ;Router ID
        Attached Router = 192.1.1.1    ;Router ID
        Attached Router = 192.1.1.2    ;Router ID
        Attached Router = 192.1.1.3    ;Router ID
```

12.4.3 Summary links

Each summary link advertisement describes a route to a single destination. Summary link advertisements are flooded throughout a single area only. The destination described is one that is external to the area, yet still belonging to the Autonomous System.

Summary link advertisements are originated by area border routers. The precise summary routes to advertise into an area are determined by examining the routing table structure (see Section 11) in accordance with the algorithm described below. Note that only intra-area routes are advertised into the backbone, while both intra-area and inter-area routes are advertised into the other areas.

To determine which routes to advertise into an attached Area A, each routing table entry is processed as follows. Remember that each routing table entry describes a set of equal-cost best paths to a particular destination:

- Only Destination Types of network and AS boundary router are advertised in summary link advertisements. If the routing table entry's **Destination Type** is area border router, examine the next routing table entry.
- AS external routes are never advertised in summary link advertisements. If the routing table entry has **Path-type** of type 1 external or type 2 external, examine the next routing table entry.
- Else, if the area associated with this set of paths is the Area A itself, do not generate a summary link advertisement for the route. [14]
- Else, if the next hops associated with this set of paths belong to Area A itself, do not generate a summary link advertisement for the route. [15] This is the logical equivalent of a Distance Vector protocol's split horizon logic.
- Else, if the routing table cost equals or exceeds the value LSInfinity, a summary link advertisement cannot be generated for this route.
- Else, if the destination of this route is an AS boundary router, generate a Type 4 link state advertisement for the destination, with **Link State ID** equal to the AS boundary router's Router ID and metric equal to the routing table entry's cost. These advertisements should not be generated if Area A has been configured as a stub area.
- Else, the Destination type is network. If this is an inter-area route, generate a Type 3 advertisement for the destination, with **Link State ID** equal to the network's address (if necessary, the Link State ID can also have one or more of the network's "host" bits set; see Appendix F for details) and metric equal to the routing table cost.
- The one remaining case is an intra-area route to a network. This means that the network is contained in one of the router's directly attached areas. In general, this information must be condensed before appearing in summary link advertisements. Remember that an area has been defined as a list of address ranges, each range consisting of an [address,mask] pair and a status indication of either Advertise or DoNotAdvertise. At most a single Type 3 advertisement is made for each range. When the range's status indicates Advertise, a Type 3 advertisement is generated with Link State ID equal to the range's address (if necessary, the Link State ID can also have one or more of the range's "host" bits set; see Appendix F for details) and cost equal to the smallest cost of any of the component networks. When the range's status indicates DoNotAdvertise, the Type 3 advertisement is suppressed and the component networks remain hidden from other areas.

---

[14] This clause covers the case: Inter-area routes are not summarized to the backbone. This is because inter-area routes are always associated with the backbone area.

[15] This clause is only invoked when Area A is a Transit area supporting one or more virtual links. For example, in the area configuration of Figure 6, Router RT11 need only originate a single summary link having the (collapsed) destination N9-N11,H1 into its connected Transit area Area 2, since all of its other eligible routes have next hops belonging to Area 2 (and as such only need be advertised by other area border routers; in this case, Routers RT10 and RT7).

By default, if a network is not contained in any explicitly configured address range, a Type 3 advertisement is generated with Link State ID equal to the network's address (if necessary, the Link State ID can also have one or more of the network's “host” bits set; see Appendix F for details) and metric equal to the network's routing table cost.

If virtual links are being used to provide/increase connectivity of the backbone, routing information concerning the backbone networks should not be condensed before being summarized into the virtual links' Transit areas. Nor should the advertisement of backbone networks into Transit areas be suppressed. In other words, the backbone's configured ranges should be ignored when originating summary links into Transit areas. The existence of virtual links is determined during the shortest path calculation for the Transit areas (see Section 16.1).

If a router advertises a summary advertisement for a destination which then becomes unreachable, the router must then flush the advertisement from the routing domain by setting its age to MaxAge and reflooding (see Section 14.1). Also, if the destination is still reachable, yet can no longer be advertised according to the above procedure (e.g., it is now an inter-area route, when it used to be an intra-area route associated with some non-backbone area; it would thus no longer be advertisable to the backbone), the advertisement should also be flushed from the routing domain.

For an example of summary link advertisements, consider again the area configuration in Figure 6. Routers RT3, RT4, RT7, RT10 and RT11 are all area border routers, and therefore are originating summary link advertisements. Consider in particular Router RT4. Its routing table was calculated as the example in Section 11.3. RT4 originates summary link advertisements into both the backbone and Area 1. Into the backbone, Router RT4 originates separate advertisements for each of the networks N1-N4. Into Area 1, Router RT4 originates separate advertisements for networks N6-N8 and the AS boundary routers RT5,RT7. It also condenses host routes Ia and Ib into a single summary link advertisement. Finally, the routes to networks N9,N10,N11 and Host H1 are advertised by a single summary link advertisement. This condensation was originally performed by the Router RT11.

These advertisements are illustrated graphically in Figures 7 and 8. Two of the summary link advertisements originated by Router RT4 follow. The actual IP addresses for the networks and routers in question have been assigned in Figure 15.

```
; summary link advertisement for Network N1,
; originated by Router RT4 into the backbone

LS age = 0                  ;always true on origination
Options = (T-bit|E-bit)     ;TOS-capable
LS type = 3                 ;indicates summary link to IP net
Link State ID = 192.1.2.0   ;N1's IP network number
Advertising Router = 192.1.1.4       ;RT4's ID
        TOS = 0
        metric = 4

; summary link advertisement for AS boundary Router RT7
; originated by Router RT4 into Area 1

LS age = 0                  ;always true on origination
Options = (T-bit|E-bit)     ;TOS-capable
LS type = 4                 ;indicates summary link to ASBR
Link State ID = Router RT7's ID
Advertising Router = 192.1.1.4       ;RT4's ID
        TOS = 0
        metric = 14
```

Summary link advertisements pertain to a single destination (IP network or AS boundary router). However, for a single destination there may be separate sets of paths, and therefore separate routing table entries, for each Type of Service. All these entries must be considered when building the summary link advertisement for the destination; a single advertisement must specify the separate costs (if they exist) for each TOS. The encoding of TOS in OSPF link state advertisements is described in Section 12.3.

Clearing the **T-bit** in the Options field of a summary link advertisement indicates that there is a TOS 0 path to the destination, but no paths for non-zero TOS. This can happen when non-TOS-capable routers exist in the routing domain (see Section 2.4).

12.4.4 Originating summary links into stub areas

The algorithm in Section 12.4.3 is optional when Area A is an OSPF stub area. Area border routers connecting to a stub area can originate summary link advertisements into the area according to the above Section's algorithm, or can choose to originate only a subset of the advertisements, possibly under configuration control. The fewer advertisements originated, the smaller the stub area's link state database, further reducing the demands on its routers' resources. However, omitting advertisements may also lead to sub-optimal inter-area routing, although routing will continue to function.

As specified in Section 12.4.3, Type 4 link state advertisements (ASBR summary links) are never originated into stub areas.

In a stub area, instead of importing external routes each area border router originates a "default summary link" into the area. The **Link State ID** for the default summary link is set to DefaultDestination, and the metric set to the (per-area) configurable parameter StubDefaultCost. Note that StubDefaultCost need not be configured identically in all of the stub area's area border routers.

12.4.5 AS external links

AS external link advertisements describe routes to destinations external to the Autonomous System. Most AS external link advertisements describe routes to specific external destinations; in these cases the advertisement's Link State ID is set to the destination network's IP address (if necessary, the Link State ID can also have one or more of the network's "host" bits set; see Appendix F for details). However, a default route for the Autonomous System can be described in an AS external link advertisement by setting the advertisement's **Link State ID** to DefaultDestination (`0.0.0.0`). AS external link advertisements are originated by AS boundary routers. An AS boundary router originates a single AS external link advertisement for each external route that it has learned, either through another routing protocol (such as EGP), or through configuration information.

In general, AS external link advertisements are the only type of link state advertisements that are flooded throughout the entire Autonomous System; all other types of link state advertisements are specific to a single area. However, AS external link advertisements are not flooded into/throughout stub areas (see Section 3.6). This enables a reduction in link state database size for routers internal to stub areas.

The metric that is advertised for an external route can be one of two types. Type 1 metrics are comparable to the link state metric. Type 2 metrics are assumed to be larger than the cost of any intra-AS path. As with summary link advertisements, if separate paths exist based on TOS, separate TOS costs can be included in the AS external link advertisement. The encoding of TOS in OSPF link state advertisements is described in Section 12.3. If the **T-bit** of the advertisement's **Options** field is clear, no non-zero TOS paths to the destination exist.

If a router advertises an AS external link advertisement for a destination which then becomes unreachable, the router must then flush the advertisement from the routing domain by setting its age to MaxAge and reflooding (see Section 14.1).

For an example of AS external link advertisements, consider once again the AS pictured in Figure 6. There are two AS boundary routers: RT5 and RT7. Router RT5 originates three external link advertisements, for networks N12-N14. Router RT7 originates two external link advertisements, for networks N12 and N15. Assume that RT7 has learned its route to N12 via EGP, and that it wishes to advertise a Type 2 metric to the AS. RT7 would then originate the following advertisement for N12:

```
; AS external link advertisement for Network N12,
; originated by Router RT7

LS age = 0                ;always true on origination
Options = (T-bit|E-bit)   ;TOS-capable
LS type = 5               ;indicates AS external link
Link State ID = N12's IP network number
Advertising Router = Router RT7's ID
        bit E = 1         ;Type 2 metric
        TOS = 0
        metric = 2
        Forwarding address = 0.0.0.0
```

In the above example, the **forwarding address** field has been set to `0.0.0.0`, indicating that packets for the external destination should be forwarded to the advertising OSPF router (RT7). This is not always desirable. Consider the example pictured in Figure 16. There are three OSPF routers (RTA, RTB and RTC) connected to a common network. Only one of these routers, RTA, is exchanging EGP information with the non-OSPF router RTX. RTA must then originate AS external link advertisements for those destinations it has learned from RTX. By using the AS external link advertisement's **forwarding address** field, RTA can specify that packets for these destinations be forwarded directly to RTX. Without this feature, Routers RTB and RTC would take an extra hop to get to these destinations.

Note that when the **forwarding address** field is non-zero, it should point to a router belonging to another Autonomous System.

A **forwarding address** can also be specified for the default route. For example, in figure 16 RTA may want to specify that all externally-destined packets should by default be forwarded to its EGP peer RTX. The resulting AS external link advertisement is pictured below. Note that the Link State ID is set to DefaultDestination.

```
; Default route, originated by Router RTA
; Packets forwarded through RTX

LS age = 0                ;always true on origination
Options = (T-bit|E-bit)  ;TOS-capable
LS type = 5               ;indicates AS external link
Link State ID = DefaultDestination  ; default route
Advertising Router = Router RTA's ID
        bit E = 1         ;Type 2 metric
        TOS = 0
        metric = 1
        Forwarding address = RTX's IP address
```

In figure 16, suppose instead that both RTA and RTB exchange EGP information with RTX. In this case, RTA and RTB would originate the same set of AS external link advertisements. These advertisements, if they specify the same metric, would be functionally equivalent since they would specify the same destination and forwarding address (RTX). This leads to a clear duplication of effort. If only one of RTA or RTB originated the set of external advertisements, the routing would remain the same, and the size of the link state database would decrease. However, it must be unambiguously defined as to which router originates the advertisements (otherwise neither may, or the identity of the originator may oscillate). The following rule is thereby established: *if two routers, both reachable from one another, originate functionally equivalent AS external advertisements (i.e., same destination, cost and non-zero forwarding address), then the advertisement originated by the router having the highest OSPF Router ID is used.* The router having the lower OSPF Router ID can then flush its advertisement. Flushing a link state advertisement is discussed in Section 14.1.

13 The Flooding Procedure

Link State Update packets provide the mechanism for flooding link state advertisements. A Link State Update packet may contain several distinct advertisements, and floods each advertisement one hop further from its point of origination. To make the flooding procedure reliable, each advertisement must be acknowledged separately. Acknowledgments are transmitted in Link State Acknowledgment packets. Many separate acknowledgments can also be grouped together into a single packet.

The flooding procedure starts when a Link State Update packet has been received. Many consistency checks have been made on the received packet before being handed to the flooding procedure (see Section 8.2). In particular, the Link State Update packet has been associated with a particular neighbor, and a particular area. If the neighbor is in a lesser state than **Exchange**, the packet should be dropped without further processing.

All types of link state advertisements, other than AS external links, are associated with a specific area. However, link state advertisements do not contain an area field. A link state advertisement's area must be deduced from the Link State Update packet header.

For each link state advertisement contained in the packet, the following steps are taken:

1. Validate the advertisement's LS checksum. If the checksum turns out to be invalid, discard the advertisement and get the next one from the Link State Update packet.
2. Examine the link state advertisement's LS type. If the LS type is unknown, discard the advertisement and get the next one from the Link State Update Packet. This specification defines LS types 1-5 (see Section 4.3).
3. Else if this is a AS external link advertisement (LS type = 5), and the area has been configured as a stub area, discard the advertisement and get the next one from the Link State Update Packet. AS external link advertisements are not flooded into/throughout stub areas (see Section 3.6).
4. Else if the advertisement's LS age is equal to MaxAge, and there is currently no instance of the advertisement in the router's link state database, then take the following actions:
    (a) Acknowledge the receipt of the advertisement by sending a Link State Acknowledgment packet back to the sending neighbor (see Section 13.5).
    (b) Purge all outstanding requests for equal or previous instances of the advertisement from the sending neighbor's **Link State Request list** (see Section 10).
    (c) If the sending neighbor is in state **Exchange** or in state **Loading**, then install the MaxAge advertisement in the link state database. Otherwise, simply discard the advertisement. In either case, examine the next advertisement (if any) listed in the Link State Update packet.
5. Otherwise, find the instance of this advertisement that is currently contained in the router's link state database. If there is no database copy, or the received advertisement is more recent than the database copy (see Section 13.1 below for the determination of which advertisement is more recent) the following steps must be performed:

   (a) If there is already a database copy, and if the database copy was installed less than MinLSInterval seconds ago, discard the new advertisement (without acknowledging it) and examine the next advertisement (if any) listed in the Link State Update packet.

   (b) Otherwise immediately flood the new advertisement out some subset of the router's interfaces (see Section 13.3). In some cases (e.g., the state of the receiving interface is **DR** and the advertisement was received from a router other than the Backup DR) the advertisement will be flooded back out the receiving interface. This occurrence should be noted for later use by the acknowledgment process (Section 13.5).

   (c) Remove the current database copy from all neighbors' **Link state retransmission lists**.

   (d) Install the new advertisement in the link state database (replacing the current database copy). This may cause the routing table calculation to be scheduled. In addition, timestamp the new advertisement with the current time (i.e., the time it was received). The flooding procedure cannot overwrite the newly installed advertisement until MinLSInterval seconds have elapsed. The advertisement installation process is discussed further in Section 13.2.

   (e) Possibly acknowledge the receipt of the advertisement by sending a Link State Acknowledgment packet back out the receiving interface. This is explained below in Section 13.5.

   (f) If this new link state advertisement indicates that it was originated by the receiving router itself (i.e., is considered a self-originated advertisement), the router must take special action, either updating the advertisement or in some cases flushing it from the routing domain. For a description of how self-originated advertisements are detected and subsequently handled, see Section 13.4.

6. Else, if there is an instance of the advertisement on the sending neighbor's **Link state request list**, an error has occurred in the Database Exchange process. In this case, restart the Database Exchange process by generating the neighbor event **BadLSReq** for the sending neighbor and stop processing the Link State Update packet.

7. Else, if the received advertisement is the same instance as the database copy (i.e., neither one is more recent) the following two steps should be performed:

   (a) If the advertisement is listed in the **Link state retransmission list** for the receiving adjacency, the router itself is expecting an acknowledgment for this advertisement. The router should treat the received advertisement as an acknowledgment, by removing the advertisement from the **Link state retransmission list**. This is termed an "implied acknowledgment". Its occurrence should be noted for later use by the acknowledgment process (Section 13.5).

   (b) Possibly acknowledge the receipt of the advertisement by sending a Link State Acknowledgment packet back out the receiving interface. This is explained below in Section 13.5.

8. Else, the database copy is more recent. Note an unusual event to network management, discard the advertisement and process the next link state advertisement contained in the Link State Update packet.

13.1 Determining which link state is newer

When a router encounters two instances of a link state advertisement, it must determine which is more recent. This occurred above when comparing a received advertisement to its database copy. This comparison must also be done during the Database Exchange procedure which occurs during adjacency bring-up.

A link state advertisement is identified by its LS type, Link State ID and Advertising Router. For two instances of the same advertisement, the LS sequence number, LS age, and LS checksum fields are used to determine which instance is more recent:

- The advertisement having the newer LS sequence number is more recent. See Section 12.1.6 for an explanation of the LS sequence number space. If both instances have the same LS sequence number, then:
- If the two instances have different LS checksums, then the instance having the larger LS checksum (when considered as a 16-bit unsigned integer) is considered more recent.
- Else, if only one of the instances has its LS age field set to MaxAge, the instance of age MaxAge is considered to be more recent.
- Else, if the LS age fields of the two instances differ by more than MaxAgeDiff, the instance having the smaller (younger) LS age is considered to be more recent.
- Else, the two instances are considered to be identical.

13.2 Installing link state advertisements in the database

Installing a new link state advertisement in the database, either as the result of flooding or a newly self-originated advertisement, may cause the OSPF routing table structure to be recalculated. The contents of the new advertisement should be compared to the old instance, if present. If there is no difference, there is no need to recalculate the routing table. (Note that even if the contents are the same, the LS checksum will probably be different, since the checksum covers the LS sequence number.)

If the contents are different, the following pieces of the routing table must be recalculated, depending on the new advertisement's LS type field:

Router links and network links advertisements The entire routing table must be recalculated, starting with the shortest path calculations for each area (not just the area whose topological database has changed). The reason that the shortest path calculation cannot be restricted to the single changed area has to do with the fact that AS boundary routers may belong to multiple areas. A change in the area currently providing the best route may force the router to use an intra-area route provided by a different area. [16]

Summary link advertisements The best route to the destination described by the summary link advertisement must be recalculated (see Section 16.5). If this destination is an AS boundary router, it may also be necessary to re-examine all the AS external link advertisements.

AS external link advertisements The best route to the destination described by the AS external link advertisement must be recalculated (see Section 16.6).

Also, any old instance of the advertisement must be removed from the database when the new advertisement is installed. This old instance must also be removed from all neighbors' **Link state retransmission lists** (see Section 10).

13.3 Next step in the flooding procedure

When a new (and more recent) advertisement has been received, it must be flooded out some set of the router's interfaces. This section describes the second part of flooding procedure (the first part being the processing that occurred in Section 13), namely, selecting the outgoing interfaces and adding the advertisement to the appropriate

---

[16] By keeping more information in the routing table, it is possible for an implementation to recalculate the shortest path tree only for a single area. In fact, there are incremental algorithms that allow an implementation to recalculate only a portion of a single area's shortest path tree [BBN]. However, these algorithms are beyond the scope of this specification.

neighbors' **Link state retransmission lists**. Also included in this part of the flooding procedure is the maintenance of the neighbors' **Link state request lists**.

This section is equally applicable to the flooding of an advertisement that the router itself has just originated (see Section 12.4). For these advertisements, this section provides the entirety of the flooding procedure (i.e., the processing of Section 13 is not performed, since, for example, the advertisement has not been received from a neighbor and therefore does not need to be acknowledged).

Depending upon the advertisement's LS type, the advertisement can be flooded out only certain interfaces. These interfaces, defined by the following, are called the **eligible interfaces**:

AS external link advertisements (LS Type = 5) AS external link advertisements are flooded throughout the entire AS, with the exception of stub areas (see Section 3.6). The **eligible interfaces** are all the router's interfaces, excluding virtual links and those interfaces attaching to stub areas.

All other LS types All other types are specific to a single area (Area A). The **eligible interfaces** are all those interfaces attaching to the Area A. If Area A is the backbone, this includes all the virtual links.

Link state databases must remain synchronized over all adjacencies associated with the above **eligible interfaces**. This is accomplished by executing the following steps on each eligible interface. It should be noted that this procedure may decide not to flood a link state advertisement out a particular interface, if there is a high probability that the attached neighbors have already received the advertisement. However, in these cases the flooding procedure must be absolutely sure that the neighbors eventually do receive the advertisement, so the advertisement is still added to each adjacency's **Link state retransmission list**. For each eligible interface:

1. Each of the neighbors attached to this interface are examined, to determine whether they must receive the new advertisement. The following steps are executed for each neighbor:
    (a) If the neighbor is in a lesser state than **Exchange**, it does not participate in flooding, and the next neighbor should be examined.
    (b) Else, if the adjacency is not yet full (neighbor state is **Exchange** or **Loading**), examine the **Link state request list** associated with this adjacency. If there is an instance of the new advertisement on the list, it indicates that the neighboring router has an instance of the advertisement already. Compare the new advertisement to the neighbor's copy:
        - If the new advertisement is less recent, then examine the next neighbor.
        - If the two copies are the same instance, then delete the advertisement from the **Link state request list**, and examine the next neighbor. [17]
        - Else, the new advertisement is more recent. Delete the advertisement from the **Link state request** list.
    (c) If the new advertisement was received from this neighbor, examine the next neighbor.
    (d) At this point we are not positive that the neighbor has an up-to-date instance of this new advertisement. Add the new advertisement to the **Link state retransmission list** for the adjacency. This ensures that the flooding procedure is reliable; the advertisement will be retransmitted at intervals until an acknowledgment is seen from the neighbor.
2. The router must now decide whether to flood the new link state advertisement out this interface. If in the previous step, the link state advertisement was NOT added to any of the **Link state retransmission lists**, there is no need to flood the advertisement out the interface and the next interface should be examined.

---

[17] This is how the **Link state request list** is emptied, which eventually causes the neighbor state to transition to **Full**. See Section 10.9 for more details.

3. If the new advertisement was received on this interface, and it was received from either the Designated Router or the Backup Designated Router, chances are that all the neighbors have received the advertisement already. Therefore, examine the next interface.

4. If the new advertisement was received on this interface, and the interface state is **Backup** (i.e., the router itself is the Backup Designated Router), examine the next interface. The Designated Router will do the flooding on this interface. If the Designated Router fails, this router will end up retransmitting the updates.

5. If this step is reached, the advertisement must be flooded out the interface. Send a Link State Update packet (with the new advertisement as contents) out the interface. The advertisement's LS age must be incremented by InfTransDelay (which must be > 0) when copied into the outgoing Link State Update packet (until the LS age field reaches its maximum value of MaxAge).

   On broadcast networks, the Link State Update packets are multicast. The destination IP address specified for the Link State Update Packet depends on the state of the interface. If the interface state is **DR** or **Backup**, the address AllSPFRouters should be used. Otherwise, the address AllDRouters should be used.

   On non-broadcast, multi-access networks, separate Link State Update packets must be sent, as unicasts, to each adjacent neighbor (i.e., those in state **Exchange** or greater). The destination IP addresses for these packets are the neighbors' IP addresses.

13.4 Receiving self-originated link state

It is a common occurrence for a router to receive self-originated link state advertisements via the flooding procedure. A self-originated advertisement is detected when either 1) the advertisement's Advertising Router is equal to the router's own Router ID or 2) the advertisement is a network links advertisement and its Link State ID is equal to one of the router's own IP interface addresses.

However, if the received self-originated advertisement is newer than the last instance that the router actually originated, the router must take special action. The reception of such an advertisement indicates that there are link state advertisements in the routing domain that were originated before the last time the router was restarted. In most cases, the router must then advance the advertisement's LS sequence number one past the received LS sequence number, and originate a new instance of the advertisement.

It may be the case the router no longer wishes to originate the received advertisement. Possible examples include: 1) the advertisement is a summary link or AS external link and the router no longer has an (advertisable) route to the destination, 2) the advertisement is a network links advertisement but the router is no longer Designated Router for the network or 3) the advertisement is a network links advertisement whose Link State ID is one of the router's own IP interface addresses but whose Advertising Router is not equal to the router's own Router ID (this latter case should be rare, and it indicates that the router's Router ID has changed since originating the advertisement). In all these cases, instead of updating the advertisement, the advertisement should be flushed from the routing domain by incrementing the received advertisement's LS age to MaxAge and reflooding (see Section 14.1).

13.5 Sending Link State Acknowledgment packets

Each newly received link state advertisement must be acknowledged. This is usually done by sending Link State Acknowledgment packets. However, acknowledgments can also be accomplished implicitly by sending Link State Update packets (see step 7a of Section 13).

Many acknowledgments may be grouped together into a single Link State Acknowledgment packet. Such a packet is sent back out the interface that has received the advertisements. The packet can be sent in one of two ways: delayed and sent on an interval timer, or sent directly (as a unicast) to a particular neighbor. The particular acknowledgment strategy used depends on the circumstances surrounding the receipt of the advertisement.

Sending delayed acknowledgments accomplishes several things: it facilitates the packaging of multiple acknowledgments in a single Link State Acknowledgment packet; it enables a single Link State Acknowledgment packet to indicate acknowledgments to several neighbors at once (through multicasting); and it randomizes the Link State Acknowledgment packets sent by the various routers attached to a multi-access network. The fixed interval between a router's delayed transmissions must be short (less than RxmtInterval) or needless retransmissions will ensue.

Direct acknowledgments are sent to a particular neighbor in response to the receipt of duplicate link state advertisements. These acknowledgments are sent as unicasts, and are sent immediately when the duplicate is received.

The precise procedure for sending Link State Acknowledgment packets is described in Table 19. The circumstances surrounding the receipt of the advertisement are listed in the left column. The acknowledgment action then taken is listed in one of the two right columns. This action depends on the state of the concerned interface; interfaces in state **Backup** behave differently from interfaces in all other states.

	Action taken in state	
Circumstances	*Backup*	*All other states*
Advertisement has been flooded back out receiving interface (see Section 13, step 5b).	No acknowledgment sent.	No acknowledgment sent.
Advertisement is more recent than database copy, but was not flooded back out receiving interface	Delayed acknowledgment sent if advertisement received from Designated Router, otherwise do nothing.	Delayed acknowledgment sent.
Advertisement is a duplicate, and was treated as an implied acknowledgment (see Section 13, step 7a).	Delayed acknowledgment sent if advertisement received from Designated Router, otherwise do nothing.	No acknowledgment sent.
Advertisement is a duplicate, and was not treated as an implied acknowledgment.	Direct acknowledgment sent.	Direct acknowledgment sent.
Advertisement's LS age is equal to MaxAge, and there is no current instance of the advertisement in the link state database (see Section 13, step 4).	Direct acknowledgment sent.	Direct acknowledgment sent.

Table 19: Sending link state acknowledgements.

Delayed acknowledgments must be delivered to all adjacent routers associated with the interface. On broadcast networks, this is accomplished by sending the delayed Link State Acknowledgment packets as multicasts. The Destination IP address used depends on the state of the interface. If the state is **DR** or **Backup**, the destination

AllSPFRouters is used. In other states, the destination AllDRouters is used. On non-broadcast networks, delayed Link State Acknowledgment packets must be unicast separately over each adjacency (i.e., neighbor whose state is $>=$ **Exchange**).

The reasoning behind sending the above packets as multicasts is best explained by an example. Consider the network configuration depicted in Figure 15. Suppose RT4 has been elected as Designated Router, and RT3 as Backup Designated Router for the network N3. When Router RT4 floods a new advertisement to Network N3, it is received by routers RT1, RT2, and RT3. These routers will not flood the advertisement back onto net N3, but they still must ensure that their topological databases remain synchronized with their adjacent neighbors. So RT1, RT2, and RT4 are waiting to see an acknowledgment from RT3. Likewise, RT4 and RT3 are both waiting to see acknowledgments from RT1 and RT2. This is best achieved by sending the acknowledgments as multicasts.

The reason that the acknowledgment logic for Backup DRs is slightly different is because they perform differently during the flooding of link state advertisements (see Section 13.3, step 4).

13.6 Retransmitting link state advertisements

Advertisements flooded out an adjacency are placed on the adjacency's **Link state retransmission list**. In order to ensure that flooding is reliable, these advertisements are retransmitted until they are acknowledged. The length of time between retransmissions is a configurable per-interface value, RxmtInterval. If this is set too low for an interface, needless retransmissions will ensue. If the value is set too high, the speed of the flooding, in the face of lost packets, may be affected.

Several retransmitted advertisements may fit into a single Link State Update packet. When advertisements are to be retransmitted, only the number fitting in a single Link State Update packet should be transmitted. Another packet of retransmissions can be sent when some of the advertisements are acknowledged, or on the next firing of the retransmission timer.

Link State Update Packets carrying retransmissions are always sent as unicasts (directly to the physical address of the neighbor). They are never sent as multicasts. Each advertisement's LS age must be incremented by InfTransDelay (which must be > 0) when copied into the outgoing Link State Update packet (until the LS age field reaches its maximum value of MaxAge).

If the adjacent router goes down, retransmissions may occur until the adjacency is destroyed by OSPF's Hello Protocol. When the adjacency is destroyed, the **Link state retransmission list** is cleared.

13.7 Receiving link state acknowledgments

Many consistency checks have been made on a received Link State Acknowledgment packet before it is handed to the flooding procedure. In particular, it has been associated with a particular neighbor. If this neighbor is in a lesser state than **Exchange**, the Link State Acknowledgment packet is discarded.

Otherwise, for each acknowledgment in the Link State Acknowledgment packet, the following steps are performed:

- Does the advertisement acknowledged have an instance on the **Link state retransmission list** for the neighbor? If not, examine the next acknowledgment. Otherwise:
- If the acknowledgment is for the same instance that is contained on the list, remove the item from the list and examine the next acknowledgment. Otherwise:
- Log the questionable acknowledgment, and examine the next one.

14 Aging The Link State Database

Each link state advertisement has an LS age field. The LS age is expressed in seconds. An advertisement's LS age field is incremented while it is contained in a router's database. Also, when copied into a Link State Update Packet for flooding out a particular interface, the advertisement's LS age is incremented by InfTransDelay.

An advertisement's LS age is never incremented past the value MaxAge. Advertisements having age MaxAge are not used in the routing table calculation. As a router ages its link state database, an advertisement's LS age may reach MaxAge. [18] At this time, the router must attempt to flush the advertisement from the routing domain. This is done simply by reflooding the MaxAge advertisement just as if it was a newly originated advertisement (see Section 13.3).

When creating a **Database summary list** for a newly forming adjacency, any MaxAge advertisements present in the link state database are added to the neighbor's **Link state retransmission list** instead of the neighbor's **Database summary list**. See Section 10.3 for more details.

A MaxAge advertisement must be removed immediately from the router's link state database as soon as both a) it is no longer contained on any neighbor **Link state retransmission lists** and b) none of the router's neighbors are in states **Exchange** or **Loading**.

When, in the process of aging the link state database, an advertisement's LS age hits a multiple of CheckAge, its LS checksum should be verified. If the LS checksum is incorrect, a program or memory error has been detected, and at the very least the router itself should be restarted.

14.1 Premature aging of advertisements

A link state advertisement can be flushed from the routing domain by setting its LS age to MaxAge and reflooding the advertisement. This procedure follows the same course as flushing an advertisement whose LS age has naturally reached the value MaxAge (see Section 14). In particular, the MaxAge advertisement is removed from the router's link state database as soon as a) it is no longer contained on any neighbor **Link state retransmission lists** and b) none of the router's neighbors are in states **Exchange** or **Loading**. We call the setting of an advertisement's LS age to MaxAge **premature aging**.

Premature aging is used when it is time for a self-originated advertisement's sequence number field to wrap. At this point, the current advertisement instance (having LS sequence number of `0x7fffffff`) must be prematurely aged and flushed from the routing domain before a new instance with sequence number `0x80000001` can be originated. See Section 12.1.6 for more information.

Premature aging can also be used when, for example, one of the router's previously advertised external routes is no longer reachable. In this circumstance, the router can flush its external advertisement from the routing domain via premature aging. This procedure is preferable to the alternative, which is to originate a new advertisement for the destination specifying a metric of LSInfinity. Premature aging is also be used when unexpectedly receiving self-originated advertisements during the flooding procedure (see Section 13.4.

A router may only prematurely age its own self-originated link state advertisements. The router may not prematurely age advertisements that have been originated by other routers. An advertisement is considered self-originated when either 1) the advertisement's Advertising Router is equal to the router's own Router ID or 2) the advertisement is a network links advertisement and its Link State ID is equal to one of the router's own IP interface addresses.

---

[18] It should be a relatively rare occurrence for an advertisement's LS age to reach MaxAge in this fashion. Usually, the advertisement will be replaced by a more recent instance before it ages out.

15 Virtual Links

The single backbone area (Area ID = `0.0.0.0`) cannot be disconnected, or some areas of the Autonomous System will become unreachable. To establish/maintain connectivity of the backbone, virtual links can be configured through non-backbone areas. Virtual links serve to connect physically separate components of the backbone. The two endpoints of a virtual link are area border routers. The virtual link must be configured in both routers. The configuration information in each router consists of the other virtual endpoint (the other area border router), and the non-backbone area the two routers have in common (called the transit area). Virtual links cannot be configured through stub areas (see Section 3.6).

The virtual link is treated as if it were an unnumbered point-to-point network (belonging to the backbone) joining the two area border routers. An attempt is made to establish an adjacency over the virtual link. When this adjacency is established, the virtual link will be included in backbone router links advertisements, and OSPF packets pertaining to the backbone area will flow over the adjacency. Such an adjacency has been referred to in this document as a “virtual adjacency”.

In each endpoint router, the cost and viability of the virtual link is discovered by examining the routing table entry for the other endpoint router. (The entry’s associated area must be the configured transit area). Actually, there may be a separate routing table entry for each Type of Service. These are called the virtual link’s corresponding routing table entries. The **InterfaceUp** event occurs for a virtual link when its corresponding TOS 0 routing table entry becomes reachable. Conversely, the **InterfaceDown** event occurs when its TOS 0 routing table entry becomes unreachable. [19] In other words, the virtual link’s viability is determined by the existence of an intra-area path, through the transit area, between the two endpoints. Note that a virtual link whose underlying path has cost greater than hexadecimal `0xffff` (the maximum size of an interface cost in a router links advertisement) should be considered inoperational (i.e., treated the same as if the path did not exist).

The other details concerning virtual links are as follows:

- AS external links are NEVER flooded over virtual adjacencies. This would be duplication of effort, since the same AS external links are already flooded throughout the virtual link’s transit area. For this same reason, AS external link advertisements are not summarized over virtual adjacencies during the Database Exchange process.
- The cost of a virtual link is NOT configured. It is defined to be the cost of the intra-area path between the two defining area border routers. This cost appears in the virtual link’s corresponding routing table entry. When the cost of a virtual link changes, a new router links advertisement should be originated for the backbone area.
- Just as the virtual link’s cost and viability are determined by the routing table build process (through construction of the routing table entry for the other endpoint), so are the IP interface address for the virtual interface and the virtual neighbor’s IP address. These are used when sending OSPF protocol packets over the virtual link. Note that when one (or both) of the virtual link endpoints connect to the transit area via an unnumbered point-to-point link, it may be impossible to calculate either the virtual interface’s IP address and/or the virtual neighbor’s IP address, thereby causing the virtual link to fail.
- In each endpoint’s router links advertisement for the backbone, the virtual link is represented as a **Type 4** link whose **Link ID** is set to the virtual neighbor’s OSPF Router ID and whose **Link Data** is set to the virtual interface’s IP address. See Section 12.4.1 for more information. Note that it may be the case that there is a TOS 0 path, but no non-zero TOS paths, between the two endpoint routers. In this case, both routers must revert to being non-TOS-capable, clearing the **T-bit** in the **Options** field of their backbone router links advertisements.

---

[19] Only the TOS 0 routes are important here because all OSPF protocol packets are sent with $\mathrm{TOS}=0$. See Appendix A.

- When virtual links are configured for the backbone, information concerning backbone networks should not be condensed before being summarized into the transit areas. In other words, each backbone network should be advertised into the transit areas in a separate summary link advertisement, regardless of the backbone's configured **area address ranges**. See Section 12.4.3 for more information.
- The time between link state retransmissions, RxmtInterval, is configured for a virtual link. This should be well over the expected round-trip delay between the two routers. This may be hard to estimate for a virtual link; it is better to err on the side of making it too large.

16 Calculation Of The Routing Table

This section details the OSPF routing table calculation. Using its attached areas' link state databases as input, a router runs the following algorithm, building its routing table step by step. At each step, the router must access individual pieces of the link state databases (e.g., a router links advertisement originated by a certain router). This access is performed by the lookup function discussed in Section 12.2. The lookup process may return a link state advertisement whose LS age is equal to MaxAge. Such an advertisement should not be used in the routing table calculation, and is treated just as if the lookup process had failed.

The OSPF routing table's organization is explained in Section 11. Two examples of the routing table build process are presented in Sections 11.2 and 11.3. This process can be broken into the following steps:

1. *The present routing table is invalidated.* The routing table is built again from scratch. The old routing table is saved so that changes in routing table entries can be identified.
2. *The intra-area routes are calculated by building the shortest-path tree for each attached area.* In particular, all routing table entries whose Destination Type is "area border router" are calculated in this step. This step is described in two parts. At first the tree is constructed by only considering those links between routers and transit networks. Then the stub networks are incorporated into the tree. During the area's shortest-path tree calculation, the area's **TransitCapability** is also calculated for later use in Step 4.
3. *The inter-area routes are calculated, through examination of summary link advertisements.* If the router is attached to multiple areas (i.e., it is an area border router), only backbone summary link advertisements are examined.
4. *In area border routers connecting to one or more transit areas* (i.e, non-backbone areas whose **TransitCapability** is found to be TRUE), the transit areas' summary link advertisements are examined to see whether better paths exist using the transit areas than were found in Steps 2-3 above.
5. *Routes to external destinations are calculated, through examination of AS external link advertisements.* The locations of the AS boundary routers (which originate the AS external link advertisements) have been determined in steps 2-4.

Steps 2-5 are explained in further detail below. The explanations describe the calculations for TOS 0 only. It may also be necessary to perform each step (separately) for each of the non-zero TOS values. [20] For more information concerning the building of non-zero TOS routes see Section 16.9.

Changes made to routing table entries as a result of these calculations can cause the OSPF protocol to take further actions. For example, a change to an intra-area route will cause an area border router to originate new summary link advertisements (see Section 12.4). See Section 16.7 for a complete list of the OSPF protocol actions resulting from routing table changes.

---

[20] It may be the case that paths to certain destinations do not vary based on TOS. For these destinations, the routing calculation need not be repeated for each TOS value. In addition, there need only be a single routing table entry for these destinations (instead of a separate entry for each TOS value).

16.1 Calculating the shortest-path tree for an area

This calculation yields the set of intra-area routes associated with an area (called hereafter Area A). A router calculates the shortest-path tree using itself as the root. [21] The formation of the shortest path tree is done here in two stages. In the first stage, only links between routers and transit networks are considered. Using the Dijkstra algorithm, a tree is formed from this subset of the link state database. In the second stage, leaves are added to the tree by considering the links to stub networks.

The procedure will be explained using the graph terminology that was introduced in Section 2. The area's link state database is represented as a directed graph. The graph's vertices are routers, transit networks and stub networks. The first stage of the procedure concerns only the transit vertices (routers and transit networks) and their connecting links. Throughout the shortest path calculation, the following data is also associated with each transit vertex:

Vertex (node) ID A 32-bit number uniquely identifying the vertex. For router vertices this is the router's OSPF Router ID. For network vertices, this is the IP address of the network's Designated Router.

A link state advertisement Each transit vertex has an associated link state advertisement. For router vertices, this is a router links advertisement. For transit networks, this is a network links advertisement (which is actually originated by the network's Designated Router). In any case, the advertisement's Link State ID is always equal to the above Vertex ID.

List of next hops The list of next hops for the current set of shortest paths from the root to this vertex. There can be multiple shortest paths due to the equal-cost multipath capability. Each next hop indicates the outgoing router interface to use when forwarding traffic to the destination. On multi-access networks, the next hop also includes the IP address of the next router (if any) in the path towards the destination.

Distance from root The link state cost of the current set of shortest paths from the root to the vertex. The link state cost of a path is calculated as the sum of the costs of the path's constituent links (as advertised in router links and network links advertisements). One path is said to be "shorter" than another if it has a smaller link state cost.

The first stage of the procedure (i.e., the Dijkstra algorithm) can now be summarized as follows. At each iteration of the algorithm, there is a list of candidate vertices. Paths from the root to these vertices have been found, but not necessarily the shortest ones. However, the paths to the candidate vertex that is closest to the root are guaranteed to be shortest; this vertex is added to the shortest-path tree, removed from the candidate list, and its adjacent vertices are examined for possible addition to/modification of the candidate list. The algorithm then iterates again. It terminates when the candidate list becomes empty.

The following steps describe the algorithm in detail. Remember that we are computing the shortest path tree for Area A. All references to link state database lookup below are from Area A's database.

1. Initialize the algorithm's data structures. Clear the list of candidate vertices. Initialize the shortest-path tree to only the root (which is the router doing the calculation). Set Area A's **TransitCapability** to FALSE.

2. Call the vertex just added to the tree vertex V. Examine the link state advertisement associated with vertex V. This is a lookup in the Area A's link state database based on the **Vertex ID**. If this is a router links advertisement, and **bit V** of the router links advertisement (see Section A.4.2) is set, set Area A's **TransitCapability** to TRUE. In any case, each link described by the advertisement gives the cost to an adjacent vertex. For each described link, (say it joins vertex V to vertex W):

---

[21] Strictly speaking, because of equal-cost multipath, the algorithm does not create a tree. We continue to use the "tree" terminology because that is what occurs most often in the existing literature.

(a) If this is a link to a stub network, examine the next link in V's advertisement. Links to stub networks will be considered in the second stage of the shortest path calculation.

(b) Otherwise, W is a transit vertex (router or transit network). Look up the vertex W's link state advertisement (router links or network links) in Area A's link state database. If the advertisement does not exist, or its LS age is equal to MaxAge, or it does not have a link back to vertex V, examine the next link in V's advertisement. [22]

(c) If vertex W is already on the shortest-path tree, examine the next link in the advertisement.

(d) Calculate the link state cost D of the resulting path from the root to vertex W. D is equal to the sum of the link state cost of the (already calculated) shortest path to vertex V and the advertised cost of the link between vertices V and W. If D is:

- Greater than the value that already appears for vertex W on the candidate list, then examine the next link.
- Equal to the value that appears for vertex W on the candidate list, calculate the set of next hops that result from using the advertised link. Input to this calculation is the destination (W), and its parent (V). This calculation is shown in Section 16.1.1. This set of hops should be added to the next hop values that appear for W on the candidate list.
- Less than the value that appears for vertex W on the candidate list, or if W does not yet appear on the candidate list, then set the entry for W on the candidate list to indicate a distance of D from the root. Also calculate the list of next hops that result from using the advertised link, setting the next hop values for W accordingly. The next hop calculation is described in Section 16.1.1; it takes as input the destination (W) and its parent (V).

3. If at this step the candidate list is empty, the shortest-path tree (of transit vertices) has been completely built and this stage of the procedure terminates. Otherwise, choose the vertex belonging to the candidate list that is closest to the root, and add it to the shortest-path tree (removing it from the candidate list in the process). Note that when there is a choice of vertices closest to the root, network vertices must be chosen before router vertices in order to necessarily find all equal-cost paths. This is consistent with the tie-breakers that were introduced in the modified Dijkstra algorithm used by OSPF's Multicast routing extensions (MOSPF).

4. Possibly modify the routing table. For those routing table entries modified, the associated area will be set to Area A, the path type will be set to intra-area, and the cost will be set to the newly discovered shortest path's calculated distance.

   If the newly added vertex is an area border router (call it ABR), a routing table entry is added whose destination type is "area border router". The **Options** field found in the associated router links advertisement is copied into the routing table entry's **Optional capabilities** field. If in addition ABR is the endpoint of one of the calculating router's configured virtual links that uses Area A as its Transit area: the virtual link is declared up, the IP address of the virtual interface is set to the IP address of the outgoing interface calculated above for ABR, and the virtual neighbor's IP address is set to the ABR interface address (contained in ABR's router links advertisement) that points back to the root of the shortest-path tree; equivalently, this is the interface that points back to ABR's parent vertex on the shortest-path tree (similar to the calculation in Section 16.1.1).

   If the newly added vertex is an AS boundary router, the routing table entry of type "AS boundary router" for the destination is located. Since routers can belong to more than one area, it is possible that several sets of intra-area paths exist to the AS boundary router, each set using a different area. However, the AS boundary router's routing table entry must indicate a set of paths which utilize a single area. The area leading to the routing table entry is selected as follows: The area providing the shortest path is always chosen; if more than

---

[22] Note that the presence of any link back to V is sufficient; it need not be the matching half of the link under consideration from V to W. This is enough to ensure that, before data traffic flows between a pair of neighboring routers, their link state databases will be synchronized.

one area provides paths with the same minimum cost, the area with the largest OSPF Area ID (when considered as an unsigned 32-bit integer) is chosen. Note that whenever an AS boundary router's routing table entry is added/modified, the **Options** found in the associated router links advertisement is copied into the routing table entry's **Optional capabilities** field.

If the newly added vertex is a transit network, the routing table entry for the network is located. The entry's Destination ID is the IP network number, which can be obtained by masking the **Vertex ID** (Link State ID) with its associated subnet mask (found in the body of the associated network links advertisement). If the routing table entry already exists (i.e., there is already an intra-area route to the destination installed in the routing table), multiple vertices have mapped to the same IP network. For example, this can occur when a new Designated Router is being established. In this case, the current routing table entry should be overwritten if and only if the newly found path is just as short and the current routing table entry's **Link State Origin** has a smaller Link State ID than the newly added vertex' **link state advertisement**.

If there is no routing table entry for the network (the usual case), a routing table entry for the IP network should be added. The routing table entry's **Link State Origin** should be set to the newly added vertex' **link state advertisement**.

5. Iterate the algorithm by returning to Step 2.

The stub networks are added to the tree in the procedure's second stage. In this stage, all router vertices are again examined. Those that have been determined to be unreachable in the above first phase are discarded. For each reachable router vertex (call it V), the associated router links advertisement is found in the link state database. Each stub network link appearing in the advertisement is then examined, and the following steps are executed:

1. Calculate the distance D of stub network from the root. D is equal to the distance from the root to the router vertex (calculated in stage 1), plus the stub network link's advertised cost. Compare this distance to the current best cost to the stub network. This is done by looking up the stub network's current routing table entry. If the calculated distance D is larger, go on to examine the next stub network link in the advertisement.

2. If this step is reached, the stub network's routing table entry must be updated. Calculate the set of next hops that would result from using the stub network link. This calculation is shown in Section 16.1.1; input to this calculation is the destination (the stub network) and the parent vertex (the router vertex). If the distance D is the same as the current routing table cost, simply add this set of next hops to the routing table entry's list of next hops. In this case, the routing table already has a **Link State Origin**. If this **Link State Origin** is a router links advertisement whose Link State ID is smaller than V's Router ID, reset the **Link State Origin** to V's router links advertisement.

   Otherwise D is smaller than the routing table cost. Overwrite the current routing table entry by setting the routing table entry's cost to D, and by setting the entry's list of next hops to the newly calculated set. Set the routing table entry's **Link State Origin** to V's router links advertisement. Then go on to examine the next stub network link.

For all routing table entries added/modified in the second stage, the associated area will be set to Area A and the path type will be set to intra-area. When the list of reachable router links is exhausted, the second stage is completed. At this time, all intra-area routes associated with Area A have been determined.

The specification does not require that the above two stage method be used to calculate the shortest path tree. However, if another algorithm is used, an identical tree must be produced. For this reason, it is important to note that links between transit vertices must be bidirectional in ordered to be included in the above tree. It should also be mentioned that more efficient algorithms exist for calculating the tree; for example, the incremental SPF algorithm described in [BBN].

16.1.1 The next hop calculation

This section explains how to calculate the current set of next hops to use for a destination. Each next hop consists of the outgoing interface to use in forwarding packets to the destination together with the next hop router (if any). The next hop calculation is invoked each time a shorter path to the destination is discovered. This can happen in either stage of the shortest-path tree calculation (see Section 16.1). In stage 1 of the shortest-path tree calculation a shorter path is found as the destination is added to the candidate list, or when the destination's entry on the candidate list is modified (Step 2d of Stage 1). In stage 2 a shorter path is discovered each time the destination's routing table entry is modified (Step 2 of Stage 2).

The set of next hops to use for the destination may be recalculated several times during the shortest-path tree calculation, as shorter and shorter paths are discovered. In the end, the destination's routing table entry will always reflect the next hops resulting from the absolute shortest path(s).

Input to the next hop calculation is a) the destination and b) its parent in the current shortest path between the root (the calculating router) and the destination. The parent is always a transit vertex (i.e., always a router or a transit network).

If there is at least one intervening router in the current shortest path between the destination and the root, the destination simply inherits the set of next hops from the parent. Otherwise, there are two cases. In the first case, the parent vertex is the root (the calculating router itself). This means that the destination is either a directly connected network or directly connected router. The next hop in this case is simply the OSPF interface connecting to the network/router; no next hop router is required. If the connecting OSPF interface in this case is a virtual link, the setting of the next hop should be deferred until the calculation in Section 16.3.

In the second case, the parent vertex is a network that directly connects the calculating router to the destination router. The list of next hops is then determined by examining the destination's router links advertisement. For each link in the advertisement that points back to the parent network, the link's Link Data field provides the IP address of a next hop router. The outgoing interface to use can then be derived from the next hop IP address (or it can be inherited from the parent network).

16.2 Calculating the inter-area routes

The inter-area routes are calculated by examining summary link advertisements. If the router has active attachments to multiple areas, only backbone summary link advertisements are examined. Routers attached to a single area examine that area's summary links. In either case, the summary links examined below are all part of a single area's link state database (call it Area A).

Summary link advertisements are originated by the area border routers. Each summary link advertisement in Area A is considered in turn. Remember that the destination described by a summary link advertisement is either a network (Type 3 summary link advertisements) or an AS boundary router (Type 4 summary link advertisements). For each summary link advertisement:

1. If the cost specified by the advertisement is LSInfinity, or if the advertisement's LS age is equal to MaxAge, then examine the next advertisement.

2. If the advertisement was originated by the calculating router itself, examine the next advertisement.

3. If the collection of destinations described by the summary link advertisement falls into one of the router's configured **area address ranges** (see Section 3.5) and the particular **area address range** is active, the summary link advertisement should be ignored. Active means that there are one or more reachable (by intra-area paths) networks contained in the area range. In this case, all addresses in the area range are assumed to be either reachable via intra-area paths, or else to be unreachable by any other means.

4. Else, call the destination described by the advertisement N (for Type 3 summary links, N's address is obtained by masking the advertisement's Link State ID with the network/subnet mask contained in the body of the advertisement), and the area border originating the advertisement BR. Look up the routing table entry for BR having Area A as its associated area. If no such entry exists for router BR (i.e., BR is unreachable in Area A), do nothing with this advertisement and consider the next in the list. Else, this advertisement describes an inter-area path to destination N, whose cost is the distance to BR plus the cost specified in the advertisement. Call the cost of this inter-area path IAC.
5. Next, look up the routing table entry for the destination N. (The entry's Destination Type is either Network or AS boundary router.) If no entry exists for N or if the entry's path type is "type 1 external" or "type 2 external", then install the inter-area path to N, with associated area Area A, cost IAC, next hop equal to the list of next hops to router BR, and Advertising router equal to BR.
6. Else, if the paths present in the table are intra-area paths, do nothing with the advertisement (intra-area paths are always preferred).
7. Else, the paths present in the routing table are also inter-area paths. Install the new path through BR if it is cheaper, overriding the paths in the routing table. Otherwise, if the new path is the same cost, add it to the list of paths that appear in the routing table entry.

16.3 Examining transit areas' summary links

This step is only performed by area border routers attached to one or more transit areas. Transit areas are those areas supporting one or more virtual links; their **TransitCapability** parameter has been set to TRUE in Step 2 of the Dijkstra algorithm (see Section 16.1). They are the only non-backbone areas that can carry data traffic that neither originates nor terminates in the area itself.

The purpose of the calculation below is to examine the transit areas to see whether they provide any better (shorter) paths than the paths previously calculated in Sections 16.1 and 16.2. Any paths found that are better than or equal to previously discovered paths are installed in the routing table.

The calculation proceeds as follows. All the transit areas' summary link advertisements are examined in turn. Each such summary link advertisement describes a route through a transit area Area A to a Network N (N's address is obtained by masking the advertisement's Link State ID with the network/subnet mask contained in the body of the advertisement) or in the case of a Type 4 summary link advertisement, to an AS boundary router N. Suppose also that the summary link advertisement was originated by an area border router BR.

1. If the cost advertised by the summary link advertisement is LSInfinity, or if the advertisement's LS age is equal to MaxAge, then examine the next advertisement.
2. If the summary link advertisement was originated by the calculating router itself, examine the next advertisement.
3. Look up the routing table entry for N. If it does not exist, or if the route type is other than intra-area or inter-area, or if the area associated with the routing table entry is not the backbone area, then examine the next advertisement. In other words, this calculation only updates backbone intra-area routes found in Section 16.1 and inter-area routes found in Section 16.2.
4. Look up the routing table entry for the advertising router BR associated with the Area A. If it is unreachable, examine the next advertisement. Otherwise, the cost to destination N is the sum of the cost in BR's Area A routing table entry and the cost advertised in the advertisement. Call this cost IAC.

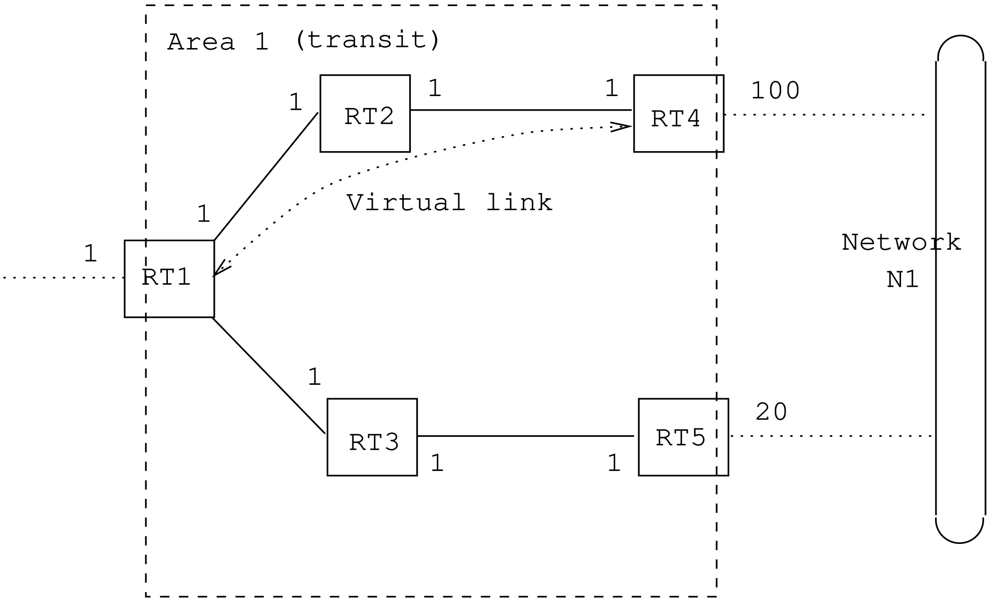


Figure 17: Routing through transit areas

5. If this cost is less than the cost occurring in N's routing table entry, overwrite N's list of next hops with those used for BR, and set N's routing table cost to IAC. Else, if IAC is the same as N's current cost, add BR's list of next hops to N's list of next hops. In any case, the area associated with N's routing table entry must remain the backbone area, and the path type (either intra-area or inter-area) must also remain the same.

It is important to note that the above calculation never makes unreachable destinations reachable, but instead just potentially finds better paths to already reachable destinations. Also, unlike Section 16.3 of [RFC 1247], the above calculation installs any better cost found into the routing table entry, from which it may be readvertised in summary link advertisements to other areas.

As an example of the calculation, consider the Autonomous System pictured in Figure 17. There is a single non-backbone area (Area 1) that physically divides the backbone into two separate pieces. To maintain connectivity of the backbone, a virtual link has been configured between Routers RT1 and RT4. On the right side of the figure, Network N1 belongs to the backbone. The dotted lines indicate that there is a much shorter intra-area backbone path between router RT5 and Network N1 (cost 20) than there is between Router RT4 and Network N1 (cost 100). Both Router RT4 and Router RT5 will inject summary link advertisements for Network N1 into Area 1.

After the shortest-path tree has been calculated for the backbone in Section 16.1, Router RT1 (left end of the virtual link) will have calculated a path through Router RT4 for all data traffic destined for Network N1. However, since Router RT5 is so much closer to Network N1, all routers internal to Area 1 (e.g., Routers RT2 and RT3) will forward their Network N1 traffic towards Router RT5, instead of RT4. And indeed, after examining Area 1's summary link advertisements by the above calculation, Router RT1 will also forward Network N1 traffic towards RT5. Note that in this example the virtual link enables Network N1 traffic to be forwarded through the transit area Area 1, but the actual path the data traffic takes does not follow the virtual link. In other words, virtual links allow transit traffic to be forwarded through an area, but do not dictate the precise path that the traffic will take.

16.4 Calculating AS external routes

AS external routes are calculated by examining AS external link advertisements. Each of the AS external link advertisements is considered in turn. Most AS external link advertisements describe routes to specific IP destinations. An AS external link advertisement can also describe a default route for the Autonomous System (Destination ID = DefaultDestination, network/subnet mask = `0x00000000`). For each AS external link advertisement:

1. If the cost specified by the advertisement is LSInfinity, or if the advertisement's LS age is equal to MaxAge, then examine the next advertisement.

2. If the advertisement was originated by the calculating router itself, examine the next advertisement.

3. Call the destination described by the advertisement N. N's address is obtained by masking the advertisement's Link State ID with the network/subnet mask contained in the body of the advertisement. Look up the routing table entry for the AS boundary router (ASBR) that originated the advertisement. If no entry exists for router ASBR (i.e., ASBR is unreachable), do nothing with this advertisement and consider the next in the list.

   Else, this advertisement describes an AS external path to destination N. Examine the **forwarding address** specified in the external advertisement. This indicates the IP address to which packets for the destination should be forwarded. If **forwarding address** is set to `0.0.0.0`, packets should be sent to the ASBR itself. Otherwise, look up the **forwarding address** in the routing table.[23] An intra-area or inter-area path must exist to the forwarding address. If no such path exists, do nothing with the advertisement and consider the next in the list.

   Call the routing table distance to the **forwarding address** X (when the forwarding address is set to `0.0.0.0`, this is the distance to the ASBR itself), and the cost specified in the advertisement Y. X is in terms of the link state metric, and Y is a type 1 or 2 external metric.

4. Next, look up the routing table entry for the destination N. If no entry exists for N, install the AS external path to N, with next hop equal to the list of next hops to the **forwarding address**, and advertising router equal to ASBR. If the external metric type is 1, then the path-type is set to type 1 external and the cost is equal to X+Y. If the external metric type is 2, the path-type is set to type 2 external, the link state component of the route's cost is X, and the type 2 cost is Y.

5. Else, if the paths present in the table are not type 1 or type 2 external paths, do nothing (AS external paths have the lowest priority).

6. Otherwise, compare the cost of this new AS external path to the ones present in the table. Type 1 external paths are always shorter than type 2 external paths. Type 1 external paths are compared by looking at the sum of the distance to the **forwarding address** and the advertised type 1 metric (X+Y). Type 2 external paths are compared by looking at the advertised type 2 metrics, and then if necessary, the distance to the **forwarding addresses**.

   If the new path is shorter, it replaces the present paths in the routing table entry. If the new path is the same cost, it is added to the routing table entry's list of paths.

---

[23] When the **forwarding address** is non-zero, it should point to a router belonging to another Autonomous System. See Section 12.4.5 for more details.

16.5 Incremental updates — summary link advertisements

When a new summary link advertisement is received, it is not necessary to recalculate the entire routing table. Call the destination described by the summary link advertisement N (N's address is obtained by masking the advertisement's Link State ID with the network/subnet mask contained in the body of the advertisement), and let Area A be the area to which the advertisement belongs. There are then two separate cases:

Case 1: Area A is the backbone and/or the router is not an area border router. In this case, the following calculations must be performed. First, if there is presently an inter-area route to the destination N, N's routing table entry is invalidated, saving the entry's values for later comparisons. Then the calculation in Section 16.2 is run again for the single destination N. In this calculation, all of Area A's summary link advertisements that describe a route to N are examined. In addition, if the router is an area border router attached to one or more transit areas, the calculation in Section 16.3 must be run again for the single destination. If the results of these calculations have changed the cost/path to an AS boundary router (as would be the case for a Type 4 summary link advertisement) or to any forwarding addresses, all AS external link advertisements will have to be reexamined by rerunning the calculation in Section 16.4. Otherwise, if N is now newly unreachable, the calculation in Section 16.4 must be rerun for the single destination N, in case an alternate external route to N exists.

Case 2: Area A is a transit area and the router is an area border router. In this case, the following calculations must be performed. First, if N's routing table entry presently contains one or more inter-area paths that utilize the transit area Area A, these paths should be removed. If this removes all paths from the routing table entry, the entry should be invalidated. The entry's old values should be saved for later comparisons. Next the calculation in Section 16.3 must be run again for the single destination N. If the results of this calculation have caused the cost to N to increase, the complete routing table calculation must be rerun starting with the Dijkstra algorithm specified in Section 16.1. Otherwise, if the cost/path to an AS boundary router (as would be the case for a Type 4 summary link advertisement) or to any forwarding addresses has changed, all AS external link advertisements will have to be reexamined by rerunning the calculation in Section 16.4. Otherwise, if N is now newly unreachable, the calculation in Section 16.4 must be rerun for the single destination N, in case an alternate external route to N exists.

16.6 Incremental updates — AS external link advertisements

When a new AS external link advertisement is received, it is not necessary to recalculate the entire routing table. Call the destination described by the AS external link advertisement N. N's address is obtained by masking the advertisement's Link State ID with the network/subnet mask contained in the body of the advertisement. If there is already an intra-area or inter-area route to the destination, no recalculation is necessary (internal routes take precedence).

Otherwise, the procedure in Section 16.4 will have to be performed, but only for those AS external link advertisements whose destination is N. Before this procedure is performed, the present routing table entry for N should be invalidated.

16.7 Events generated as a result of routing table changes

Changes to routing table entries sometimes cause the OSPF area border routers to take additional actions. These routers need to act on the following routing table changes:

- The cost or path type of a routing table entry has changed. If the destination described by this entry is a Network or AS boundary router, and this is not simply a change of AS external routes, new summary link advertisements may have to be generated (potentially one for each attached area, including the backbone). See Section 12.4.3 for more information. If a previously advertised entry has been deleted, or is no longer advertisable to a particular area, the advertisement must be flushed from the routing domain by setting its LS age to MaxAge and reflooding (see Section 14.1).

- A routing table entry associated with a configured virtual link has changed. The destination of such a routing table entry is an area border router. The change indicates a modification to the virtual link's cost or viability.

  If the entry indicates that the area border router is newly reachable (via TOS 0), the corresponding virtual link is now operational. An **InterfaceUp** event should be generated for the virtual link, which will cause a virtual adjacency to begin to form (see Section 10.3). At this time the virtual link's IP interface address and the virtual neighbor's Neighbor IP address are also calculated.

  If the entry indicates that the area border router is no longer reachable (via TOS 0), the virtual link and its associated adjacency should be destroyed. This means an **InterfaceDown** event should be generated for the associated virtual link.

  If the cost of the entry has changed, and there is a fully established virtual adjacency, a new router links advertisement for the backbone must be originated. This in turn may cause further routing table changes.

16.8 Equal-cost multipath

The OSPF protocol maintains multiple equal-cost routes to all destinations. This can be seen in the steps used above to calculate the routing table, and in the definition of the routing table structure.

Each one of the multiple routes will be of the same type (intra-area, inter-area, type 1 external or type 2 external), cost, and will have the same associated area. However, each route specifies a separate next hop and Advertising router.

There is no requirement that a router running OSPF keep track of all possible equal-cost routes to a destination. An implementation may choose to keep only a fixed number of routes to any given destination. This does not affect any of the algorithms presented in this specification.

16.9 Building the non-zero-TOS portion of the routing table

The OSPF protocol can calculate a different set of routes for each IP TOS (see Section 2.4). Support for TOS-based routing is optional. TOS-capable and non-TOS-capable routers can be mixed in an OSPF routing domain. Routers not supporting TOS calculate only the TOS 0 route to each destination. These routes are then used to forward all data traffic, regardless of the TOS indications in the data packet's IP header. A router that does not support TOS indicates this fact to the other OSPF routers by clearing the **T-bit** in the **Options** field of its router links advertisement.

The above sections detailing the routing table calculations handle the TOS 0 case only. In general, for routers supporting TOS-based routing, each piece of the routing table calculation must be rerun separately for the non-zero TOS values. When calculating routes for TOS X, only TOS X metrics can be used. Any link state advertisement may specify a separate cost for each TOS (a cost for TOS 0 must always be specified). The encoding of TOS in OSPF link state advertisements is described in Section 12.3.

An advertisement can specify that it is restricted to TOS 0 (i.e., non-zero TOS is not handled) by clearing the **T-bit** in the link state advertisement's **Option** field. Such advertisements are not used when calculating routes for non-zero TOS. For this reason, it is possible that a destination is unreachable for some non-zero TOS. In this case, the TOS 0 path is used when forwarding packets (see Section 11.1).

The following lists the modifications needed when running the routing table calculation for a non-zero TOS value (called TOS X). In general, routers and advertisements that do not support TOS are omitted from the calculation.

Calculating the shortest-path tree (Section 16.1). Routers that do not support TOS-based routing should be omitted from the shortest-path tree calculation. These routers are identified as those having the **T-bit** reset in the **Options** field of their router links advertisements. Such routers should never be added to the Dijktra algorithm's candidate list, nor should their router links advertisements be examined when adding the stub networks to the tree. In particular, if the **T-bit** is reset in the calculating router's own router links advertisement, it does not run the shortest-path tree calculation for non-zero TOS values.

Calculating the inter-area routes (Section 16.2). Inter-area paths are the concatenation of a path to an area border router with a summary link. When calculating TOS X routes, both path components must also specify TOS X. In other words, only TOS X paths to the area border router are examined, and the area border router must be advertising a TOS X route to the destination. Note that this means that summary link advertisements having the **T-bit** reset in their **Options** field are not considered.

Examining transit areas' summary links (Section 16.3). This calculation again considers the concatenation of a path to an area border router with a summary link. As with inter-area routes, only TOS X paths to the area border router are examined, and the area border router must be advertising a TOS X route to the destination.

Calculating AS external routes (Section 16.4). This calculation considers the concatenation of a path to a **forwarding address** with an AS external link. Only TOS X paths to the **forwarding address** are examined, and the AS boundary router must be advertising a TOS X route to the destination. Note that this means that AS external link advertisements having the **T-bit** reset in their **Options** field are not considered.

In addition, the advertising AS boundary router must also be reachable for its advertisements to be considered (see Section 16.4). However, if the advertising router and the **forwarding address** are not one in the same, the advertising router need only be reachable via TOS 0.

References

[BBN]	McQuillan, J, I. Richer and E. Rosen, *ARPANET Routing Algorithm Improvements*, BBN Technical Report 3803, April 1978.
[DEC]	Digital Equipment Corporation, *Information processing systems – Data communications – Intermediate System to Intermediate System Intra-Domain Routing Protocol*, October 1987.
[McQuillan]	McQuillan, J. et.al., *The New Routing Algorithm for the Arpanet*, IEEE Transactions on Communications, May 1980.
[Perlman]	Perlman, R., *Fault-Tolerant Broadcast of Routing Information*, Computer Networks, December 1983.
[RFC 791]	Postel, J., *Internet Protocol*, STD 5, RFC 791, USC/Information Sciences Institute, September 1981.
[RFC 905]	McKenzie, A., *ISO Transport Protocol specification ISO DP 8073*, RFC 905, ISO, April 1984.
[RFC 1112]	Deering, S., *Host extensions for IP multicasting*, STD 5, RFC 1112, Stanford University, May 1988.
[RFC 1213]	McCloghrie, K., and M. Rose, *Management Information Base for network management of TCP/IP-based internets: MIB-II*, STD 17, RFC 1213, Hughes LAN Systems, Performance Systems International, March 1991.
[RFC 1247]	Moy, J., *OSPF Version 2*, RFC 1247, Proteon, Inc., July 1991.
[RFC 1519]	Fuller, V., T. Li, J. Yu, and K. Varadhan, *Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy*, RFC1519, BARRNet, cisco, MERIT, OARnet, September 1993.
[RFC 1340]	Reynolds, J. and J. Postel, *Assigned Numbers*, STD 2, RFC 1340, USC/Information Sciences Institute, July 1992.
[RFC 1349]	Almquist, P., *Type of Service in the Internet Protocol Suite*, RFC 1349, July 1992.
[RS-85-153]	Leiner, B., et.al., *The DARPA Internet Protocol Suite*, DDN Protocol Handbook, April 1985.

A OSPF data formats

This appendix describes the format of OSPF protocol packets and OSPF link state advertisements. The OSPF protocol runs directly over the IP network layer. Before any data formats are described, the details of the OSPF encapsulation are explained.

Next the OSPF Options field is described. This field describes various capabilities that may or may not be supported by pieces of the OSPF routing domain. The OSPF Options field is contained in OSPF Hello packets, Database Description packets and in OSPF link state advertisements.

OSPF packet formats are detailed in Section A.3. A description of OSPF link state advertisements appears in Section A.4.

A.1 Encapsulation of OSPF packets

OSPF runs directly over the Internet Protocol's network layer. OSPF packets are therefore encapsulated solely by IP and local data-link headers.

OSPF does not define a way to fragment its protocol packets, and depends on IP fragmentation when transmitting packets larger than the network MTU. The OSPF packet types that are likely to be large (Database Description Packets, Link State Request, Link State Update, and Link State Acknowledgment packets) can usually be split into several separate protocol packets, without loss of functionality. This is recommended; IP fragmentation should be avoided whenever possible. Using this reasoning, an attempt should be made to limit the sizes of packets sent over virtual links to 576 bytes. However, if necessary, the length of OSPF packets can be up to 65,535 bytes (including the IP header).

The other important features of OSPF's IP encapsulation are:

- *Use of IP multicast.* Some OSPF messages are multicast, when sent over multi-access networks. Two distinct IP multicast addresses are used. Packets sent to these multicast addresses should never be forwarded; they are meant to travel a single hop only. To ensure that these packets will not travel multiple hops, their IP TTL must be set to 1.

  AllSPFRouters This multicast address has been assigned the value `224.0.0.5`. All routers running OSPF should be prepared to receive packets sent to this address. Hello packets are always sent to this destination. Also, certain OSPF protocol packets are sent to this address during the flooding procedure.

  AllDRouters This multicast address has been assigned the value `224.0.0.6`. Both the Designated Router and Backup Designated Router must be prepared to receive packets destined to this address. Certain OSPF protocol packets are sent to this address during the flooding procedure.

- *OSPF is IP protocol number 89.* This number has been registered with the Network Information Center. IP protocol number assignments are documented in [RFC 1340].

- *Routing protocol packets are sent with IP TOS of 0.* The OSPF protocol supports TOS-based routing. Routes to any particular destination may vary based on TOS. However, all OSPF routing protocol packets are sent using the normal service TOS value of binary 0000 defined in [RFC 1349].

- *Routing protocol packets are sent with IP precedence set to Internetwork Control.* OSPF protocol packets should be given precedence over regular IP data traffic, in both sending and receiving. Setting the IP precedence field in the IP header to Internetwork Control [RFC 791] may help implement this objective.

A.2 The Options field

The OSPF Options field is present in OSPF Hello packets, Database Description packets and all link state advertisements. The Options field enables OSPF routers to support (or not support) optional capabilities, and to communicate their capability level to other OSPF routers. Through this mechanism routers of differing capabilities can be mixed within an OSPF routing domain.

When used in Hello packets, the Options field allows a router to reject a neighbor because of a capability mismatch. Alternatively, when capabilities are exchanged in Database Description packets a router can choose not to forward certain link state advertisements to a neighbor because of its reduced functionality. Lastly, listing capabilities in link state advertisements allows routers to route traffic around reduced functionality routers, by excluding them from parts of the routing table calculation.

Two capabilities are currently defined. For each capability, the effect of the capability's appearance (or lack of appearance) in Hello packets, Database Description packets and link state advertisements is specified below. For example, the ExternalRoutingCapability (below called the E-bit) has meaning only in OSPF Hello Packets. Routers should reset (i.e. clear) the unassigned part of the capability field when sending Hello packets or Database Description packets and when originating link state advertisements.

Additional capabilities may be assigned in the future. Routers encountering unrecognized capabilities in received Hello Packets, Database Description packets or link state advertisements should ignore the capability and process the packet/advertisement normally.

*	*	*	*	*	*	E	T

The options field

T-bit This describes the router's TOS capability. If the T-bit is reset, then the router supports only a single TOS (TOS 0). Such a router is also said to be incapable of TOS-routing, and elsewhere in this document referred to as a TOS-0-only router. The absence of the T-bit in a router links advertisement causes the router to be skipped when building a non-zero TOS shortest-path tree (see Section 16.9). In other words, routers incapable of TOS routing will be avoided as much as possible when forwarding data traffic requesting a non-zero TOS. The absence of the T-bit in a summary link advertisement or an AS external link advertisement indicates that the advertisement is describing a TOS 0 route only (and not routes for non-zero TOS).

E-bit This bit reflects the associated area's ExternalRoutingCapability. AS external link advertisements are not flooded into/through OSPF stub areas (see Section 3.6). The E-bit ensures that all members of a stub area agree on that area's configuration. The E-bit is meaningful only in OSPF Hello packets. When the E-bit is reset in the Hello packet sent out a particular interface, it means that the router will neither send nor receive AS external link state advertisements on that interface (in other words, the interface connects to a stub area). Two routers will not become neighbors unless they agree on the state of the E-bit.

A.3 OSPF Packet Formats

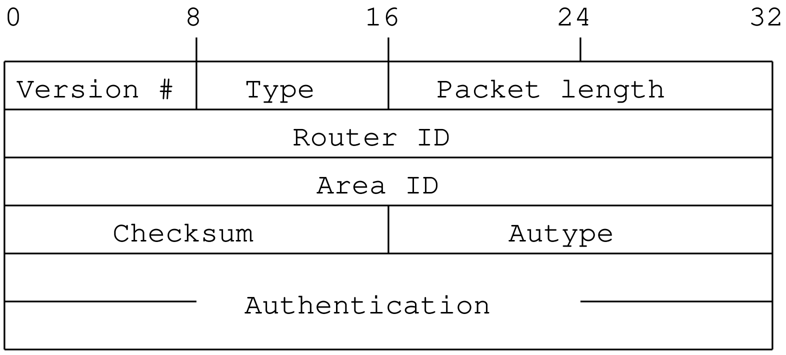
There are five distinct OSPF packet types. All OSPF packet types begin with a standard 24 byte header. This header is described first. Each packet type is then described in a succeeding section. In these sections each packet's division into fields is displayed, and then the field definitions are enumerated.

All OSPF packet types (other than the OSPF Hello packets) deal with lists of link state advertisements. For example, Link State Update packets implement the flooding of advertisements throughout the OSPF routing domain. Because of this, OSPF protocol packets cannot be parsed unless the format of link state advertisements is also understood. The format of Link state advertisements is described in Section A.4.

The receive processing of OSPF packets is detailed in Section 8.2. The sending of OSPF packets is explained in Section 8.1.

A.3.1 The OSPF packet header

Every OSPF packet starts with a common 24 byte header. This header contains all the necessary information to determine whether the packet should be accepted for further processing. This determination is described in Section 8.2 of the specification.



Version # The OSPF version number. This specification documents version 2 of the protocol.

Type The OSPF packet types are as follows. The format of each of these packet types is described in a succeeding section.

Type	*Description*
1	Hello
2	Database Description
3	Link State Request
4	Link State Update
5	Link State Acknowledgment

Packet length The length of the protocol packet in bytes. This length includes the standard OSPF header.

Router ID The Router ID of the packet's source. In OSPF, the source and destination of a routing protocol packet are the two ends of an (potential) adjacency.

Area ID A 32 bit number identifying the area that this packet belongs to. All OSPF packets are associated with a single area. Most travel a single hop only. Packets travelling over a virtual link are labelled with the backbone Area ID of `0.0.0.0`.

Checksum The standard IP checksum of the entire contents of the packet, starting with the OSPF packet header but excluding the 64-bit authentication field. This checksum is calculated as the 16-bit one's complement of the one's complement sum of all the 16-bit words in the packet, excepting the authentication field. If the packet's length is not an integral number of 16-bit words, the packet is padded with a byte of zero before checksumming.

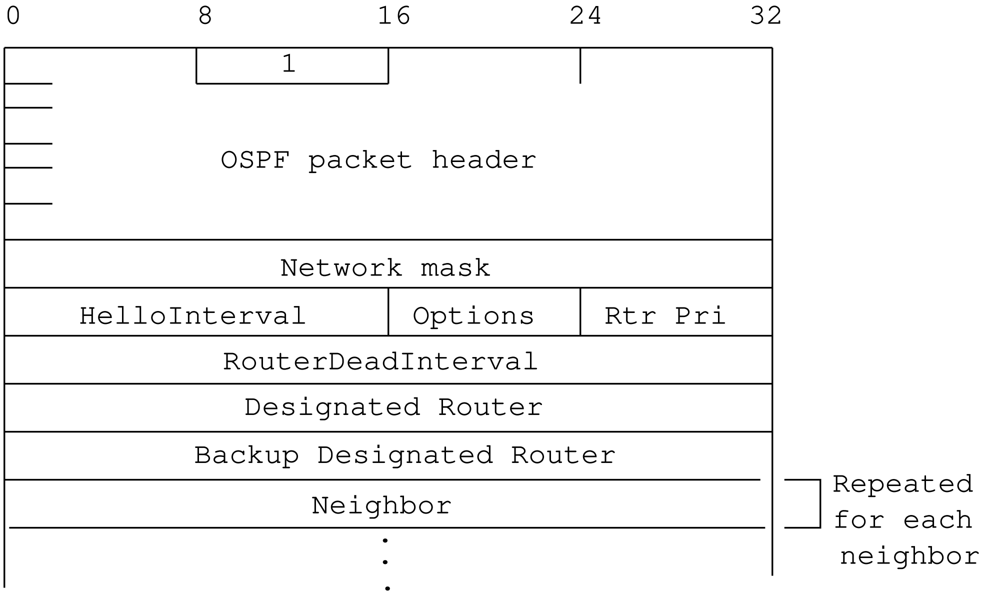
AuType Identifies the authentication scheme to be used for the packet. Authentication is discussed in Appendix D of the specification. Consult Appendix D for a list of the currently defined authentication types.

Authentication A 64-bit field for use by the authentication scheme.

A.3.2 The Hello packet

Hello packets are OSPF packet type 1. These packets are sent periodically on all interfaces (including virtual links) in order to establish and maintain neighbor relationships. In addition, Hello Packets are multicast on those physical networks having a multicast or broadcast capability, enabling dynamic discovery of neighboring routers.

All routers connected to a common network must agree on certain parameters (Network mask, HelloInterval and RouterDeadInterval). These parameters are included in Hello packets, so that differences can inhibit the forming of neighbor relationships. A detailed explanation of the receive processing for Hello packets is presented in Section 10.5. The sending of Hello packets is covered in Section 9.5.



Network mask The network mask associated with this interface. For example, if the interface is to a class B network whose third byte is used for subnetting, the network mask is `0xffffff00`.

Options The optional capabilities supported by the router, as documented in Section A.2.

HelloInterval The number of seconds between this router's Hello packets.

Rtr Pri This router's Router Priority. Used in (Backup) Designated Router election. If set to 0, the router will be ineligible to become (Backup) Designated Router.

RouterDeadInterval The number of seconds before declaring a silent router down.

Designated Router The identity of the Designated Router for this network, in the view of the advertising router. The Designated Router is identified here by its IP interface address on the network. Set to `0.0.0.0` if there is no Designated Router.

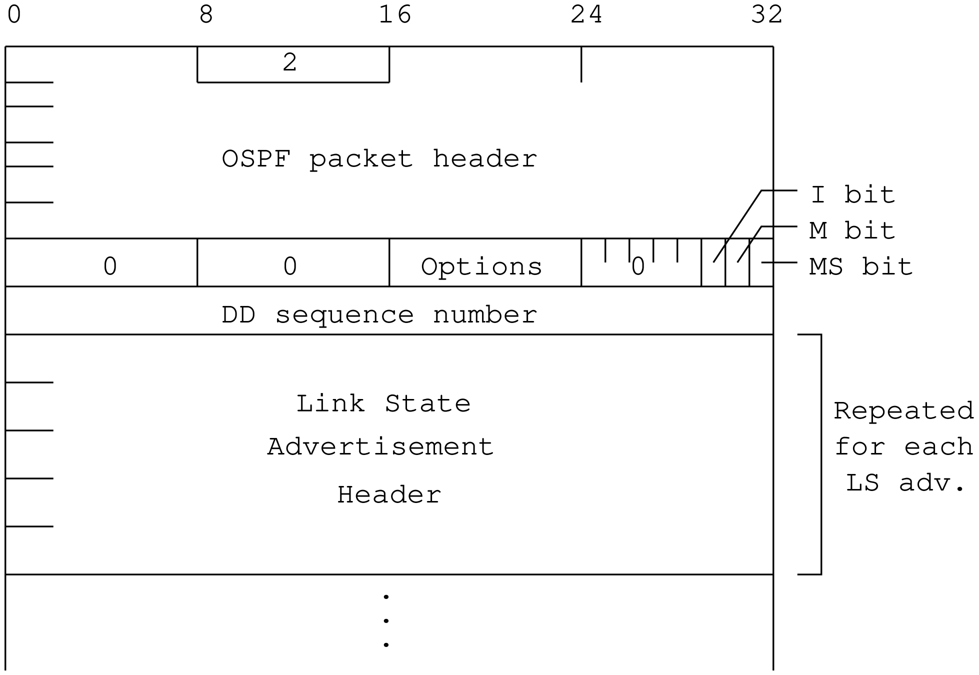
Backup Designated Router The identity of the Backup Designated Router for this network, in the view of the advertising router. The Backup Designated Router is identified here by its IP interface address on the network. Set to `0.0.0.0` if there is no Backup Designated Router.

Neighbor The Router IDs of each router from whom valid Hello packets have been seen recently on the network. Recently means in the last RouterDeadInterval seconds.

A.3.3 The Database Description packet

Database Description packets are OSPF packet type 2. These packets are exchanged when an adjacency is being initialized. They describe the contents of the topological database. Multiple packets may be used to describe the database. For this purpose a poll-response procedure is used. One of the routers is designated to be master, the other a slave. The master sends Database Description packets (polls) which are acknowledged by Database Description packets sent by the slave (responses). The responses are linked to the polls via the packets' DD sequence numbers.

The format of the Database Description packet is very similar to both the Link State Request and Link State Acknowledgment packets. The main part of all three is a list of items, each item describing a piece of the topological database. The sending of Database Description Packets is documented in Section 10.8. The reception of Database Description packets is documented in Section 10.6.



0 These fields are reserved. They must be 0.

Options The optional capabilities supported by the router, as documented in Section A.2.

I-bit The Init bit. When set to 1, this packet is the first in the sequence of Database Description Packets.

M-bit The More bit. When set to 1, it indicates that more Database Description Packets are to follow.

MS-bit The Master/Slave bit. When set to 1, it indicates that the router is the master during the Database Exchange process. Otherwise, the router is the slave.

DD sequence number Used to sequence the collection of Database Description Packets. The initial value (indicated by the Init bit being set) should be unique. The DD sequence number then increments until the complete database description has been sent.

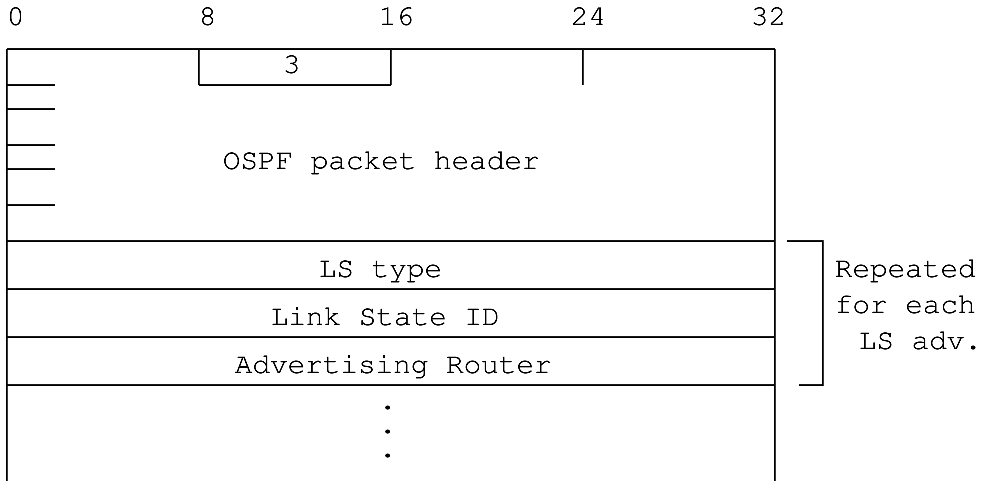
The rest of the packet consists of a (possibly partial) list of the topological database's pieces. Each link state advertisement in the database is described by its link state advertisement header. The link state advertisement header is documented in Section A.4.1. It contains all the information required to uniquely identify both the advertisement and the advertisement's current instance.

A.3.4 The Link State Request packet

Link State Request packets are OSPF packet type 3. After exchanging Database Description packets with a neighboring router, a router may find that parts of its topological database are out of date. The Link State Request packet is used to request the pieces of the neighbor's database that are more up to date. Multiple Link State Request packets may need to be used. The sending of Link State Request packets is the last step in bringing up an adjacency.

A router that sends a Link State Request packet has in mind the precise instance of the database pieces it is requesting, defined by LS sequence number, LS checksum, and LS age, although these fields are not specified in the Link State Request Packet itself. The router may receive even more recent instances in response.

The sending of Link State Request packets is documented in Section 10.9. The reception of Link State Request packets is documented in Section 10.7.

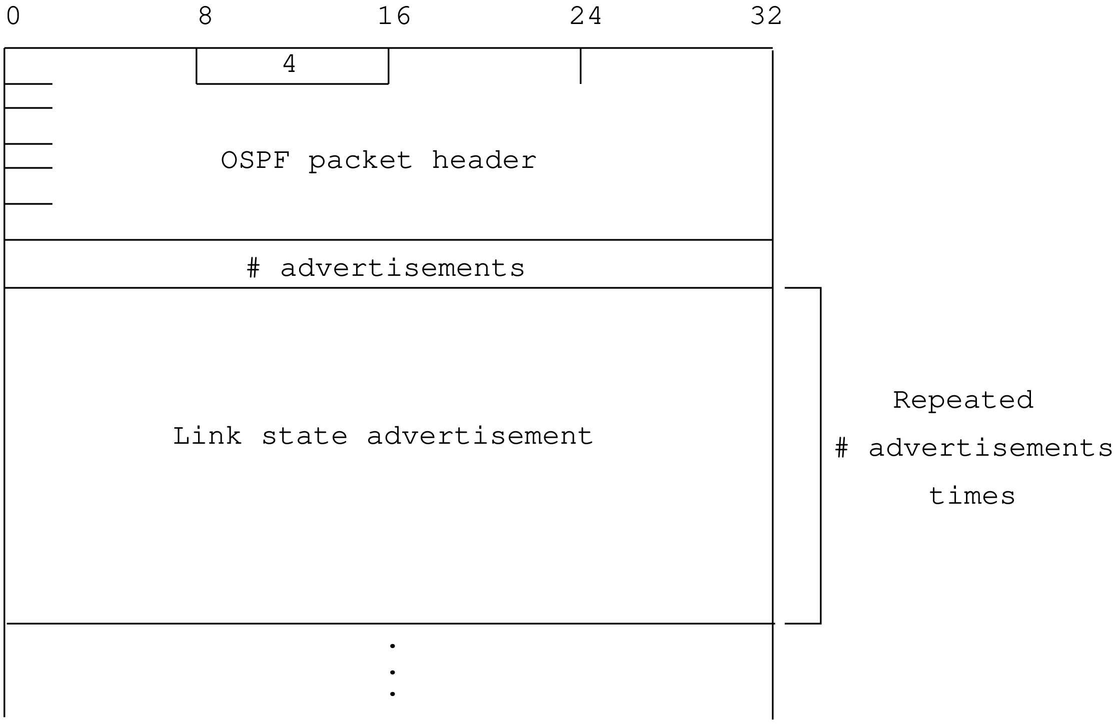


Each advertisement requested is specified by its LS type, Link State ID, and Advertising Router. This uniquely identifies the advertisement, but not its instance. Link State Request packets are understood to be requests for the most recent instance (whatever that might be).

A.3.5 The Link State Update packet

Link State Update packets are OSPF packet type 4. These packets implement the flooding of link state advertisements. Each Link State Update packet carries a collection of link state advertisements one hop further from its origin. Several link state advertisements may be included in a single packet.

Link State Update packets are multicast on those physical networks that support multicast/broadcast. In order to make the flooding procedure reliable, flooded advertisements are acknowledged in Link State Acknowledgment packets. If retransmission of certain advertisements is necessary, the retransmitted advertisements are always carried by unicast Link State Update packets. For more information on the reliable flooding of link state advertisements, consult Section 13.



 # advertisements The number of link state advertisements included in this update.

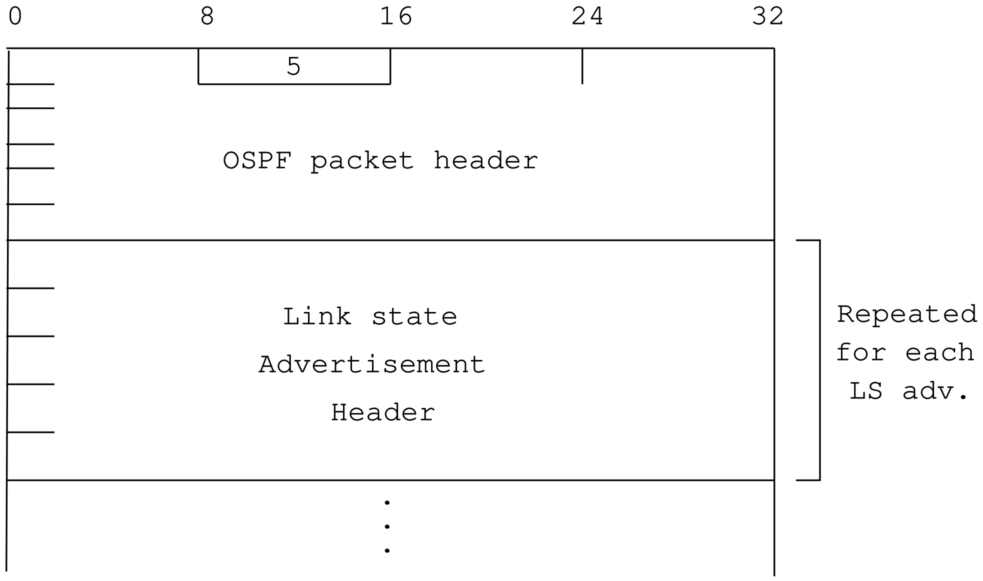
The body of the Link State Update packet consists of a list of link state advertisements. Each advertisement begins with a common 20 byte header, the link state advertisement header. This header is described in Section A.4.1. Otherwise, the format of each of the five types of link state advertisements is different. Their formats are described in Section A.4.

A.3.6 The Link State Acknowledgment packet

Link State Acknowledgment Packets are OSPF packet type 5. To make the flooding of link state advertisements reliable, flooded advertisements are explicitly acknowledged. This acknowledgment is accomplished through the sending and receiving of Link State Acknowledgment packets. Multiple link state advertisements can be acknowledged in a single Link State Acknowledgment packet.

Depending on the state of the sending interface and the source of the advertisements being acknowledged, a Link State Acknowledgment packet is sent either to the multicast address AllSPFRouters, to the multicast address AllDRouters, or as a unicast. The sending of Link State Acknowledgement packets is documented in Section 13.5. The reception of Link State Acknowledgement packets is documented in Section 13.7.

The format of this packet is similar to that of the Data Description packet. The body of both packets is simply a list of link state advertisement headers.



Each acknowledged link state advertisement is described by its link state advertisement header. The link state advertisement header is documented in Section A.4.1. It contains all the information required to uniquely identify both the advertisement and the advertisement's current instance.

A.4 Link state advertisement formats

There are five distinct types of link state advertisements. Each link state advertisement begins with a standard 20-byte link state advertisement header. This header is explained in Section A.4.1. Succeeding sections then diagram the separate link state advertisement types.

Each link state advertisement describes a piece of the OSPF routing domain. Every router originates a router links advertisement. In addition, whenever the router is elected Designated Router, it originates a network links advertisement. Other types of link state advertisements may also be originated (see Section 12.4). All link state advertisements are then flooded throughout the OSPF routing domain. The flooding algorithm is reliable, ensuring that all routers have the same collection of link state advertisements. (See Section 13 for more information concerning the flooding algorithm). This collection of advertisements is called the link state (or topological) database.

From the link state database, each router constructs a shortest path tree with itself as root. This yields a routing table (see Section 11). For the details of the routing table build process, see Section 16.

A.4.1 The Link State Advertisement header

All link state advertisements begin with a common 20 byte header. This header contains enough information to uniquely identify the advertisement (LS type, Link State ID, and Advertising Router). Multiple instances of the link state advertisement may exist in the routing domain at the same time. It is then necessary to determine which instance is more recent. This is accomplished by examining the LS age, LS sequence number and LS checksum fields that are also contained in the link state advertisement header.

<table>
<tr><th>0</th><th>8</th><th>16</th><th>24 32</th></tr>
<tr><td colspan="2">LS age</td><td>Options</td><td>LS type</td></tr>
<tr><td colspan="4">Link State ID</td></tr>
<tr><td colspan="4">Advertising Router</td></tr>
<tr><td colspan="4">LS sequence number</td></tr>
<tr><td colspan="2">LS checksum</td><td colspan="2">length</td></tr>
</table>

LS age The time in seconds since the link state advertisement was originated.

Options The optional capabilities supported by the described portion of the routing domain. OSPF's optional capabilities are documented in Section A.2.

LS type The type of the link state advertisement. Each link state type has a separate advertisement format. The link state types are as follows (see Section 12.1.3 for further explanation):

LS Type	*Description*
1	Router links
2	Network links
3	Summary link (IP network)
4	Summary link (ASBR)
5	AS external link

Link State ID This field identifies the portion of the internet environment that is being described by the advertisement. The contents of this field depend on the advertisement's LS type. For example, in network links advertisements the Link State ID is set to the IP interface address of the network's Designated Router (from which the network's IP address can be derived). The Link State ID is further discussed in Section 12.1.4.

Advertising Router The Router ID of the router that originated the link state advertisement. For example, in network links advertisements this field is set to the Router ID of the network's Designated Router.

LS sequence number Detects old or duplicate link state advertisements. Successive instances of a link state advertisement are given successive LS sequence numbers. See Section 12.1.6 for more details.

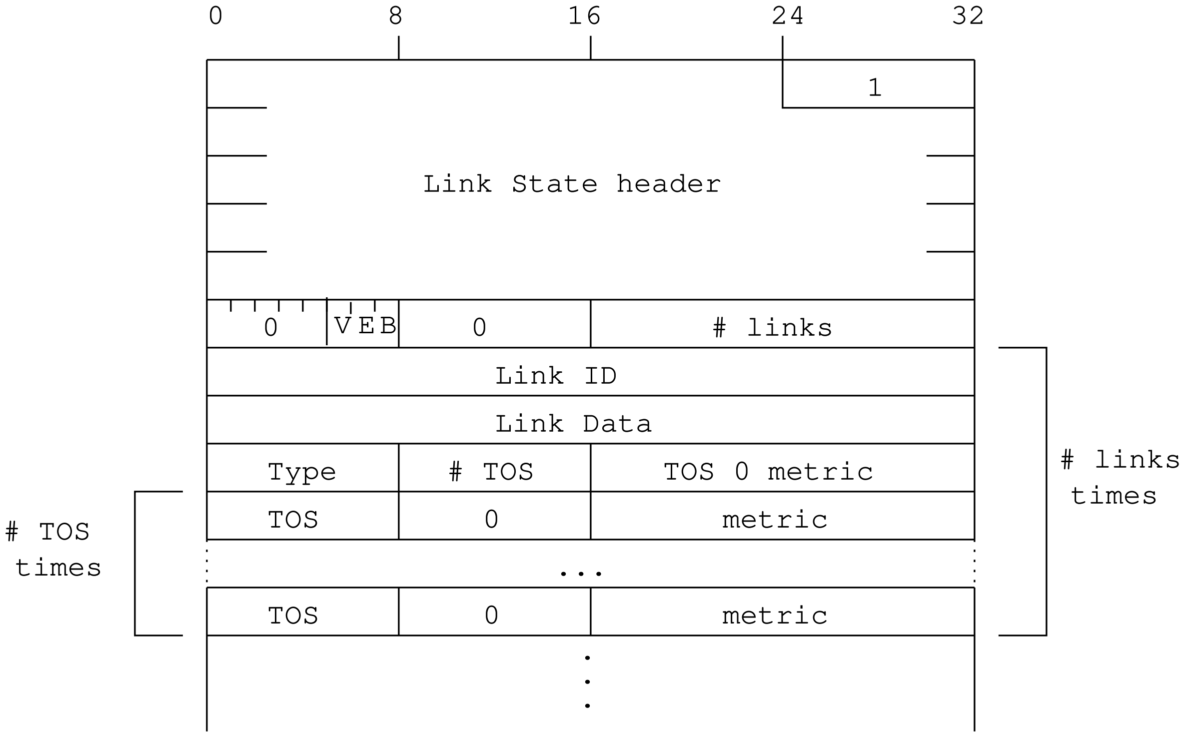
LS checksum The Fletcher checksum of the complete contents of the link state advertisement, including the link state advertisement header but excepting the LS age field. See Section 12.1.7 for more details.

length The length in bytes of the link state advertisement. This includes the 20 byte link state advertisement header.

A.4.2 Router links advertisements

Router links advertisements are the Type 1 link state advertisements. Each router in an area originates a router links advertisement. The advertisement describes the state and cost of the router's links (i.e., interfaces) to the area. All of the router's links to the area must be described in a single router links advertisement. For details concerning the construction of router links advertisements, see Section 12.4.1.

In router links advertisements, the Link State ID field is set to the router's OSPF Router ID. The **T-bit** is set in the advertisement's **Option** field if and only if the router is able to calculate a separate set of routes for each IP TOS. Router links advertisements are flooded throughout a single area only.



bit V	When set, the router is an endpoint of an active virtual link that is using the described area as a Transit area (V is for virtual link endpoint).
bit E	When set, the router is an AS boundary router (E is for external)
bit B	When set, the router is an area border router (B is for border)
# links	The number of router links described by this advertisement. This must be the total collection of router links (i.e., interfaces) to the area.

The following fields are used to describe each router link (i.e., interface). Each router link is typed (see the below Type field). The Type field indicates the kind of link being described. It may be a link to a transit network, to another router or to a stub network. The values of all the other fields describing a router link depend on the link's Type. For example, each link has an associated 32-bit data field. For links to stub networks this field specifies the network's IP address mask. For other link types the Link Data specifies the router's associated IP interface address.

Type A quick description of the router link. One of the following. Note that host routes are classified as links to stub networks whose network mask is `0xffffffff`.

Type	*Description*
1	Point-to-point connection to another router
2	Connection to a transit network
3	Connection to a stub network
4	Virtual link

Link ID Identifies the object that this router link connects to. Value depends on the link's Type. When connecting to an object that also originates a link state advertisement (i.e., another router or a transit network) the Link ID is equal to the neighboring advertisement's Link State ID. This provides the key for looking up said advertisement in the link state database. See Section 12.2 for more details.

Type	*Link ID*
1	Neighboring router's Router ID
2	IP address of Designated Router
3	IP network/subnet number
4	Neighboring router's Router ID

Link Data Contents again depend on the link's Type field. For connections to stub networks, it specifies the network's IP address mask. For unnumbered point-to-point connections, it specifies the interface's MIB-II [RFC 1213] ifIndex value. For the other link types it specifies the router's associated IP interface address. This latter piece of information is needed during the routing table build process, when calculating the IP address of the next hop. See Section 16.1.1 for more details.

 # TOS The number of different TOS metrics given for this link, not counting the required metric for TOS 0. For example, if no additional TOS metrics are given, this field should be set to 0.

TOS 0 metric The cost of using this router link for TOS 0.

For each link, separate metrics may be specified for each Type of Service (TOS). The metric for TOS 0 must always be included, and was discussed above. Metrics for non-zero TOS are described below. The encoding of TOS in OSPF link state advertisements is described in Section 12.3. Note that the cost for non-zero TOS values that are not specified defaults to the TOS 0 cost. Metrics must be listed in order of increasing TOS encoding. For example, the metric for TOS 16 must always follow the metric for TOS 8 when both are specified.

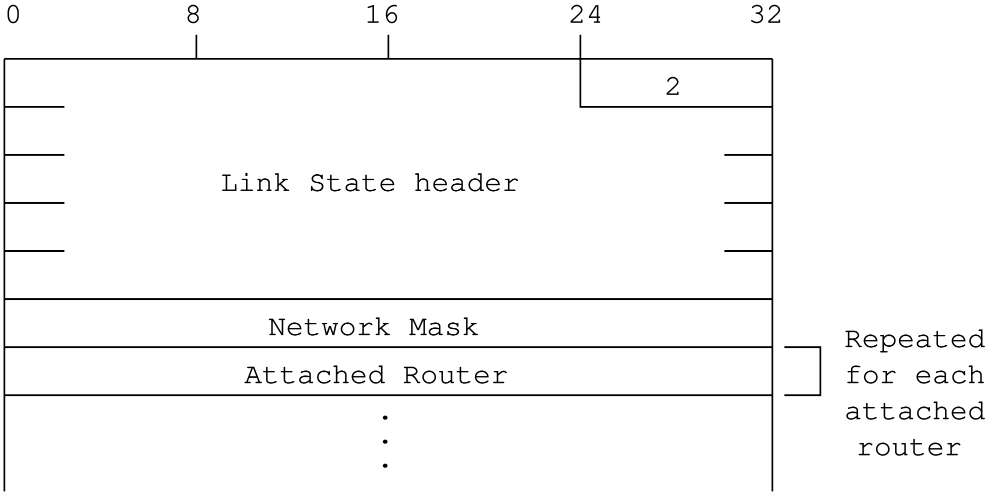
TOS IP Type of Service that this metric refers to. The encoding of TOS in OSPF link state advertisements is described in Section 12.3.

metric The cost of using this outbound router link, for traffic of the specified TOS.

A.4.3 Network links advertisements

Network links advertisements are the Type 2 link state advertisements. A network links advertisement is originated for each transit network in the area. A transit network is a multi-access network that has more than one attached router. The network links advertisement is originated by the network's Designated Router. The advertisement describes all routers attached to the network, including the Designated Router itself. The advertisement's Link State ID field lists the IP interface address of the Designated Router.

The distance from the network to all attached routers is zero, for all Types of Service. This is why the TOS and metric fields need not be specified in the network links advertisement. For details concerning the construction of network links advertisements, see Section 12.4.2.



Network Mask	The IP address mask for the network. For example, a class A network would have the mask `0xff000000`.
Attached Router	The Router IDs of each of the routers attached to the network. Actually, only those routers that are fully adjacent to the Designated Router are listed. The Designated Router includes itself in this list. The number of routers included can be deduced from the link state advertisement header's length field.

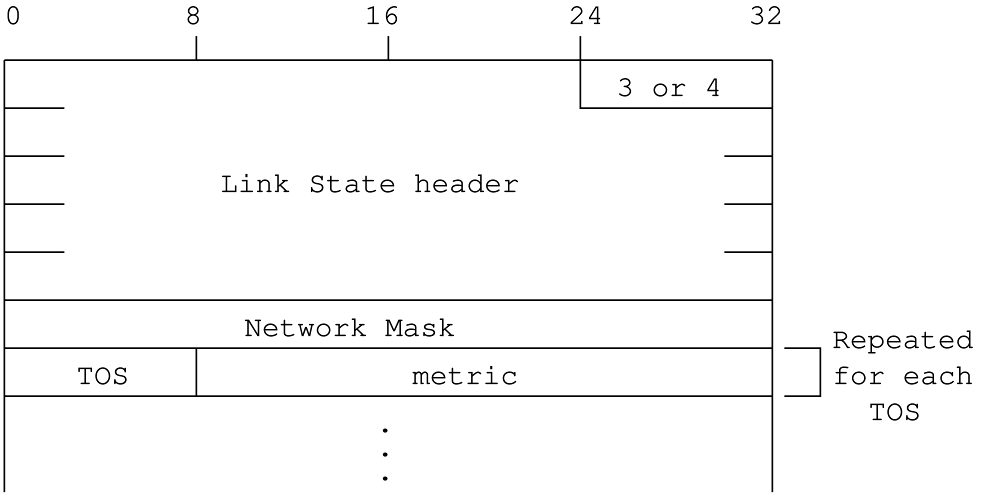
A.4.4 Summary link advertisements

Summary link advertisements are the Type 3 and 4 link state advertisements. These advertisements are originated by area border routers. A separate summary link advertisement is made for each destination (known to the router) which belongs to the AS, yet is outside the area. For details concerning the construction of summary link advertisements, see Section 12.4.3.

Type 3 link state advertisements are used when the destination is an IP network. In this case the advertisement's Link State ID field is an IP network number (if necessary, the Link State ID can also have one or more of the network's "host" bits set; see Appendix F for details). When the destination is an AS boundary router, a Type 4 advertisement is used, and the Link State ID field is the AS boundary router's OSPF Router ID. (To see why it is necessary to advertise the location of each ASBR, consult Section 16.4.) Other than the difference in the Link State ID field, the format of Type 3 and 4 link state advertisements is identical.

For stub areas, Type 3 summary link advertisements can also be used to describe a (per-area) default route. Default summary routes are used in stub areas instead of flooding a complete set of external routes. When describing a default summary route, the advertisement's Link State ID is always set to DefaultDestination (`0.0.0.0`) and the Network Mask is set to `0.0.0.0`.

Separate costs may be advertised for each IP Type of Service. The encoding of TOS in OSPF link state advertisements is described in Section 12.3. Note that the cost for TOS 0 must be included, and is always listed first. If the **T-bit** is reset in the advertisement's **Option field**, only a route for TOS 0 is described by the advertisement. Otherwise, routes for the other TOS values are also described; if a cost for a certain TOS is not included, its cost defaults to that specified for TOS 0.



Network Mask For Type 3 link state advertisements, this indicates the destination network's IP address mask. For example, when advertising the location of a class A network the value `0xff000000` would be used. This field is not meaningful and must be zero for Type 4 link state advertisements.

For each specified Type of Service, the following fields are defined. The number of TOS routes included can be calculated from the link state advertisement header's length field. Values for TOS 0 must be specified; they are listed first. Other values must be listed in order of increasing TOS encoding. For example, the cost for TOS 16 must always follow the cost for TOS 8 when both are specified.

TOS The Type of Service that the following cost concerns. The encoding of TOS in OSPF link state advertisements is described in Section 12.3.

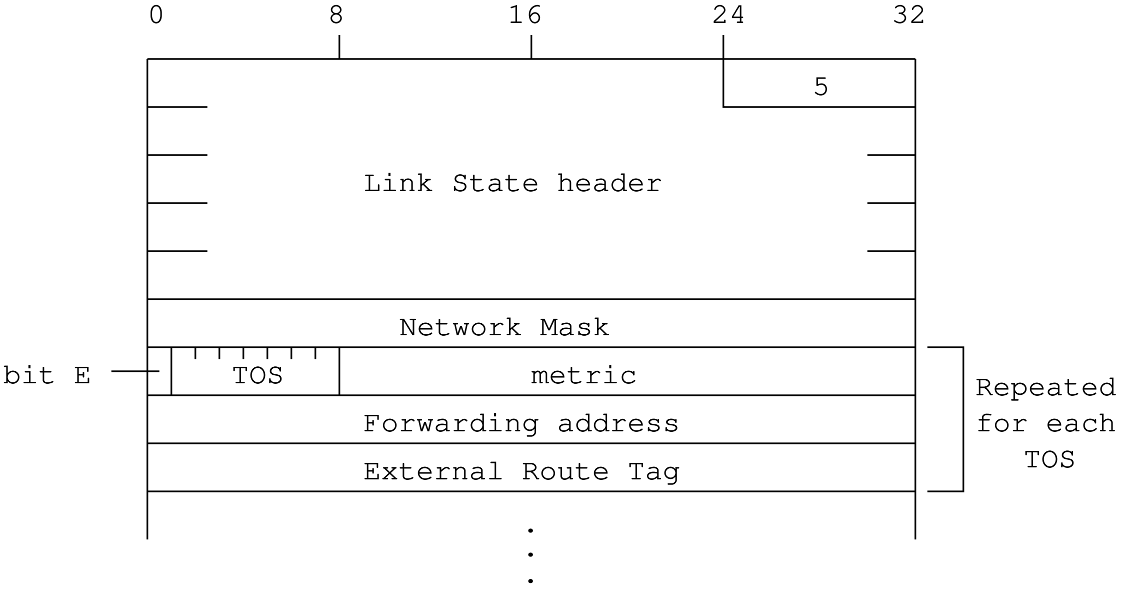
metric The cost of this route. Expressed in the same units as the interface costs in the router links advertisements.

A.4.5 AS external link advertisements

AS external link advertisements are the Type 5 link state advertisements. These advertisements are originated by AS boundary routers. A separate advertisement is made for each destination (known to the router) which is external to the AS. For details concerning the construction of AS external link advertisements, see Section 12.4.3.

AS external link advertisements usually describe a particular external destination. For these advertisements the Link State ID field specifies an IP network number (if necessary, the Link State ID can also have one or more of the network's "host" bits set; see Appendix F for details). AS external link advertisements are also used to describe a default route. Default routes are used when no specific route exists to the destination. When describing a default route, the Link State ID is always set to DefaultDestination (`0.0.0.0`) and the Network Mask is set to `0.0.0.0`.

Separate costs may be advertised for each IP Type of Service. The encoding of TOS in OSPF link state advertisements is described in Section 12.3. Note that the cost for TOS 0 must be included, and is always listed first. If the **T-bit** is reset in the advertisement's **Option field**, only a route for TOS 0 is described by the advertisement. Otherwise, routes for the other TOS values are also described; if a cost for a certain TOS is not included, its cost defaults to that specified for TOS 0.



Network Mask The IP address mask for the advertised destination. For example, when advertising a class A network the mask `0xff000000` would be used.

For each specified Type of Service, the following fields are defined. The number of TOS routes included can be calculated from the link state advertisement header's length field. Values for TOS 0 must be specified; they are listed first. Other values must be listed in order of increasing TOS encoding. For example, the cost for TOS 16 must always follow the cost for TOS 8 when both are specified.

bit E The type of external metric. If bit E is set, the metric specified is a Type 2 external metric. This means the metric is considered larger than any link state path. If bit E is zero, the specified metric is a Type 1 external metric. This means that is is comparable directly (without translation) to the link state metric.

Forwarding address Data traffic for the advertised destination will be forwarded to this address. If the Forwarding address is set to `0.0.0.0`, data traffic will be forwarded instead to the advertisement's originator (i.e., the responsible AS boundary router).

TOS The Type of Service that the following cost concerns. The encoding of TOS in OSPF link state advertisements is described in Section 12.3.

metric The cost of this route. Interpretation depends on the external type indication (bit E above).

External Route Tag A 32-bit field attached to each external route. This is not used by the OSPF protocol itself. It may be used to communicate information between AS boundary routers; the precise nature of such information is outside the scope of this specification.

B Architectural Constants

Several OSPF protocol parameters have fixed architectural values. These parameters have been referred to in the text by names such as LSRefreshTime. The same naming convention is used for the configurable protocol parameters. They are defined in Appendix C.

The name of each architectural constant follows, together with its value and a short description of its function.

LSRefreshTime The maximum time between distinct originations of any particular link state advertisement. When the LS age field of one of the router's self-originated advertisements reaches the value LSRefreshTime, a new instance of the link state advertisement is originated, even though the contents of the advertisement (apart from the link state header) will be the same. The value of LSRefreshTime is set to 30 minutes.

MinLSInterval The minimum time between distinct originations of any particular link state advertisement. The value of MinLSInterval is set to 5 seconds.

MaxAge The maximum age that a link state advertisement can attain. When an advertisement's LS age field reaches MaxAge, it is reflooded in an attempt to flush the advertisement from the routing domain (See Section 14). Advertisements of age MaxAge are not used in the routing table calculation. The value of MaxAge must be greater than LSRefreshTime. The value of MaxAge is set to 1 hour.

CheckAge When the age of a link state advertisement (that is contained in the link state database) hits a multiple of CheckAge, the advertisement's checksum is verified. An incorrect checksum at this time indicates a serious error. The value of CheckAge is set to 5 minutes.

MaxAgeDiff The maximum time dispersion that can occur, as a link state advertisement is flooded throughout the AS. Most of this time is accounted for by the link state advertisements sitting on router output queues (and therefore not aging) during the flooding process. The value of MaxAgeDiff is set to 15 minutes.

LSInfinity The metric value indicating that the destination described by a link state advertisement is unreachable. Used in summary link advertisements and AS external link advertisements as an alternative to premature aging (see Section 14.1). It is defined to be the 24-bit binary value of all ones: `0xffffff`.

DefaultDestination The Destination ID that indicates the default route. This route is used when no other matching routing table entry can be found. The default destination can only be advertised in AS external link advertisements and in stub areas' type 3 summary link advertisements. Its value is the IP address `0.0.0.0`.

C Configurable Constants

The OSPF protocol has quite a few configurable parameters. These parameters are listed below. They are grouped into general functional categories (area parameters, interface parameters, etc.). Sample values are given for some of the parameters.

Some parameter settings need to be consistent among groups of routers. For example, all routers in an area must agree on that area's parameters, and all routers attached to a network must agree on that network's IP network number and mask.

Some parameters may be determined by router algorithms outside of this specification (e.g., the address of a host connected to the router via a SLIP line). From OSPF's point of view, these items are still configurable.

C.1 Global parameters

In general, a separate copy of the OSPF protocol is run for each area. Because of this, most configuration parameters are defined on a per-area basis. The few global configuration parameters are listed below.

Router ID This is a 32-bit number that uniquely identifies the router in the Autonomous System. One algorithm for Router ID assignment is to choose the largest or smallest IP address assigned to the router. If a router's OSPF Router ID is changed, the router's OSPF software should be restarted before the new Router ID takes effect. Before restarting in order to change its Router ID, the router should flush its self-originated link state advertisements from the routing domain (see Section 14.1), or they will persist for up to MaxAge minutes.

TOS capability This item indicates whether the router will calculate separate routes based on TOS. For more information, see Sections 4.5 and 16.9.

C.2 Area parameters

All routers belonging to an area must agree on that area's configuration. Disagreements between two routers will lead to an inability for adjacencies to form between them, with a resulting hindrance to the flow of routing protocol and data traffic. The following items must be configured for an area:

Area ID This is a 32-bit number that identifies the area. The Area ID of `0.0.0.0` is reserved for the backbone. If the area represents a subnetted network, the IP network number of the subnetted network may be used for the Area ID.

List of address ranges An OSPF area is defined as a list of address ranges. Each address range consists of the following items:

> **[IP address, mask]** Describes the collection of IP addresses contained in the address range. Networks and hosts are assigned to an area depending on whether their addresses fall into one of the area's defining address ranges. Routers are viewed as belonging to multiple areas, depending on their attached networks' area membership.
>
> **Status** Set to either Advertise or DoNotAdvertise. Routing information is condensed at area boundaries. External to the area, at most a single route is advertised (via a summary link advertisement) for each address range. The route is advertised if and only if the address range's Status is set to Advertise. Unadvertised ranges allow the existence of certain networks to be intentionally hidden from other areas. Status is set to Advertise by default.

As an example, suppose an IP subnetted network is to be its own OSPF area. The area would be configured as a single address range, whose IP address is the address of the subnetted network, and whose mask is the natural class A, B, or C address mask. A single route would be advertised external to the area, describing the entire subnetted network.

AuType Each area can be configured for a separate type of authentication. See Appendix D for a discussion of the defined authentication types.

ExternalRoutingCapability Whether AS external advertisements will be flooded into/throughout the area. If AS external advertisements are excluded from the area, the area is called a "stub". Internal to stub areas, routing to external destinations will be based solely on a default summary route. The backbone cannot be configured as a stub area. Also, virtual links cannot be configured through stub areas. For more information, see Section 3.6.

StubDefaultCost If the area has been configured as a stub area, and the router itself is an area border router, then the StubDefaultCost indicates the cost of the default summary link that the router should advertise into the area. There can be a separate cost configured for each IP TOS. See Section 12.4.3 for more information.

C.3 Router interface parameters

Some of the configurable router interface parameters (such as IP interface address and subnet mask) actually imply properties of the attached networks, and therefore must be consistent across all the routers attached to that network. The parameters that must be configured for a router interface are:

IP interface address The IP protocol address for this interface. This uniquely identifies the router over the entire internet. An IP address is not required on serial lines. Such a serial line is called "unnumbered".

IP interface mask Also referred to as the subnet mask, this indicates the portion of the IP interface address that identifies the attached network. Masking the IP interface address with the IP interface mask yields the IP network number of the attached network. On point-to-point networks and virtual links, the IP interface mask is not defined. On these networks, the link itself is not assigned an IP network number, and so the addresses of each side of the link are assigned independently, if they are assigned at all.

Interface output cost(s) The cost of sending a packet on the interface, expressed in the link state metric. This is advertised as the link cost for this interface in the router's router links advertisement. There may be a separate cost for each IP Type of Service. The interface output cost(s) must always be greater than 0.

RxmtInterval The number of seconds between link state advertisement retransmissions, for adjacencies belonging to this interface. Also used when retransmitting Database Description and Link State Request Packets. This should be well over the expected round-trip delay between any two routers on the attached network. The setting of this value should be conservative or needless retransmissions will result. It will need to be larger on low speed serial lines and virtual links. Sample value for a local area network: 5 seconds.

InfTransDelay The estimated number of seconds it takes to transmit a Link State Update Packet over this interface. Link state advertisements contained in the update packet must have their age incremented by this amount before transmission. This value should take into account the transmission and propagation delays of the interface. It must be greater than 0. Sample value for a local area network: 1 second.

Router Priority An 8-bit unsigned integer. When two routers attached to a network both attempt to become Designated Router, the one with the highest Router Priority takes precedence. If there is still a tie, the router with the highest Router ID takes precedence. A router whose Router Priority is set to 0 is ineligible to become Designated Router on the attached network. Router Priority is only configured for interfaces to multi-access networks.

HelloInterval The length of time, in seconds, between the Hello Packets that the router sends on the interface. This value is advertised in the router's Hello Packets. It must be the same for all routers attached to a common network. The smaller the HelloInterval, the faster topological changes will be detected, but more OSPF routing protocol traffic will ensue. Sample value for a X.25 PDN network: 30 seconds. Sample value for a local area network: 10 seconds.

RouterDeadInterval After ceasing to hear a router's Hello Packets, the number of seconds before its neighbors declare the router down. This is also advertised in the router's Hello Packets in their RouterDeadInterval field. This should be some multiple of the HelloInterval (say 4). This value again must be the same for all routers attached to a common network.

Authentication key This configured data allows the authentication procedure to generate and/or verify the authentication field in the OSPF header. This value again must be the same for all routers attached to a common network. For example, if the AuType indicates simple password, the Authentication key would be a 64-bit password. This key would be inserted directly into the OSPF header when originating routing protocol packets. There could be a separate password for each network.

C.4 Virtual link parameters

Virtual links are used to restore/increase connectivity of the backbone. Virtual links may be configured between any pair of area border routers having interfaces to a common (non-backbone) area. The virtual link appears as an unnumbered point-to-point link in the graph for the backbone. The virtual link must be configured in both of the area border routers.

A virtual link appears in router links advertisements (for the backbone) as if it were a separate router interface to the backbone. As such, it has all of the parameters associated with a router interface (see Section C.3). Although a virtual link acts like an unnumbered point-to-point link, it does have an associated **IP interface address**. This address is used as the IP source in OSPF protocol packets it sends along the virtual link, and is set dynamically during the routing table build process. **Interface output cost** is also set dynamically on virtual links to be the cost of the intra-area path between the two routers. The parameter **RxmtInterval** must be configured, and should be well over the expected round-trip delay between the two routers. This may be hard to estimate for a virtual link; it is better to err on the side of making it too large. **Router Priority** is not used on virtual links.

A virtual link is defined by the following two configurable parameters: the Router ID of the virtual link's other endpoint, and the (non-backbone) area through which the virtual link runs (referred to as the virtual link's **Transit area**). Virtual links cannot be configured through stub areas.

C.5 Non-broadcast, multi-access network parameters

OSPF treats a non-broadcast, multi-access network much like it treats a broadcast network. Since there may be many routers attached to the network, a Designated Router is selected for the network. This Designated Router then originates a networks links advertisement, which lists all routers attached to the non-broadcast network.

However, due to the lack of broadcast capabilities, it is necessary to use configuration parameters in the Designated Router selection. These parameters need only be configured in those routers that are themselves eligible to become Designated Router (i.e., those router's whose Router Priority for the network is non-zero):

List of all other attached routers The list of all other routers attached to the non-broadcast network. Each router is listed by its IP interface address on the network. Also, for each router listed, that router's eligibility to become Designated Router must be defined. When an interface to a non-broadcast network comes up, the router sends Hello Packets only to those neighbors eligible to become Designated Router, until the identity of the Designated Router is discovered.

PollInterval If a neighboring router has become inactive (Hello Packets have not been seen for RouterDeadInterval seconds), it may still be necessary to send Hello Packets to the dead neighbor. These Hello Packets will be sent at the reduced rate PollInterval, which should be much larger than HelloInterval. Sample value for a PDN X.25 network: 2 minutes.

C.6 Host route parameters

Host routes are advertised in router links advertisements as stub networks with mask `0xffffffff`. They indicate either router interfaces to point-to-point networks, looped router interfaces, or IP hosts that are directly connected to the router (e.g., via a SLIP line). For each host directly connected to the router, the following items must be configured:

Host IP address The IP address of the host.

Cost of link to host The cost of sending a packet to the host, in terms of the link state metric. There may be multiple costs configured, one for each IP TOS. However, since the host probably has only a single connection to the internet, the actual configured cost(s) in many cases is unimportant (i.e., will have no effect on routing).

D Authentication

All OSPF protocol exchanges are authenticated. The OSPF packet header (see Section A.3.1) includes an authentication type field, and 64-bits of data for use by the appropriate authentication scheme (determined by the type field).

The authentication type is configurable on a per-area basis. Additional authentication data is configurable on a per-interface basis. For example, if an area uses a simple password scheme for authentication, a separate password may be configured for each network contained in the area.

Authentication types 0 and 1 are defined by this specification. All other authentication types are reserved for definition by the IANA (iana@ISI.EDU). The current list of authentication types is described below in Table 20.

AuType	*Description*
0	No authentication
1	Simple password
All others	Reserved for assignment by the IANA (iana@ISI.EDU)

Table 20: OSPF authentication types.

D.1 AuType 0 – No authentication

Use of this authentication type means that routing exchanges in the area are not authenticated. The 64-bit field in the OSPF header can contain anything; it is not examined on packet reception.

D.2 AuType 1 – Simple password

Using this authentication type, a 64-bit field is configured on a per-network basis. All packets sent on a particular network must have this configured value in their OSPF header 64-bit authentication field. This essentially serves as a "clear" 64-bit password.

This guards against routers inadvertently joining the area. They must first be configured with their attached networks' passwords before they can participate in the routing domain.

E Differences from RFC 1247

This section documents the differences between this memo and RFC 1247. These differences include a fix for a problem involving OSPF virtual links, together with minor enhancements and clarifications to the protocol. All differences are backward-compatible. Implementations of this memo and of RFC 1247 will interoperate.

E.1 A fix for a problem with OSPF Virtual links

In RFC 1247, certain configurations of OSPF virtual links can cause routing loops. The root of the problem is that while there is an information mismatch at the boundary of any virtual link's Transit area, a backbone path can still cross the boundary. RFC 1247 attempted to compensate for this information mismatch by adjusting any backbone path as it enters the transit area (see Section 16.3 in RFC 1247). However, this proved not to be enough. This memo fixes the problem by having all area border routers determine, by looking at summary links, whether better backbone paths can be found through the transit areas.

This fix simplifies the OSPF virtual link logic, and consists of the following components:

- A new bit has been defined in the router links advertisement, called **bit V**. Bit V is set in a router's router links advertisement for Area A if and only if the router is an endpoint of an active virtual link that uses Area A as its Transit area (see Sections 12.4.1 and A.4.2). This enables the other routers attached to Area A to discover whether the area supports any virtual links (i.e., is a transit area). This discovery is done during the calculation of Area A's shortest-path tree (see Section 16.1).
- To aid in the description of the algorithm, a new parameter has been added to the OSPF area structure: **TransitCapability**. This parameter indicates whether the area supports any active virtual links. Equivalently, it indicates whether the area can carry traffic that neither originates nor terminates in the area itself.
- The calculation in Section 16.3 of RFC 1247 has been replaced. The new calculation, performed by area border routers only, examines the summary links belonging to all attached transit areas to see whether the transit areas can provide better paths than those already found in Sections 16.1 and 16.2.
- The incremental calculations in Section 16.5 have been updated as a result of the new calculations in Section 16.3.

E.2 Supporting supernetting and subnet 0

In RFC 1247, an OSPF router cannot originate separate AS external link advertisements (or separate summary link advertisements) for two networks that have the same address but different masks. This situation can arise when subnet 0 of a network has been assigned (a practice that is generally discouraged), or when using supernetting as described in [RFC 1519] (a practice that is generally encouraged to reduce the size of routing tables), or even when in transition from one mask to another on a subnet. Using supernetting as an example, you might want to aggregate the four class C networks `192.9.4.0-192.9.7.0`, advertising one route for the aggregation and another for the single class C network `192.9.4.0`.

The reason behind this limitation is that in RFC 1247, the Link State ID of AS external link advertisements and summary link advertisements is set equal to the described network's IP address. In the above example, RFC 1247 would assign both advertisements the Link State ID of `192.9.4.0`, making them in essence the same advertisement. This memo fixes the problem by relaxing the setting of the Link State ID so that any of the "host" bits of the network address can also be set. This allows you to disambiguate advertisements for networks having the same address but different masks. Given an AS external link advertisement (or a summary link advertisement), the described network's address can now be obtained by masking the Link State ID with the network mask carried in the body of the advertisement. Again using the above example, the aggregate can now be advertised using a Link State ID of `192.9.4.0` and the single class C network advertised simultaneously using the Link State ID of 192.9.4.255.

Appendix F gives one possible algorithm for setting one or more "host" bits in the Link State ID in order to disambiguate advertisements. It should be noted that this is a local decision. Each router in an OSPF system is free to use its own algorithm, since only those advertisements originated by the router itself are affected.

It is believed that this change will be more or less compatible with implementations of RFC 1247. Implementations of RFC 1247 will probably either a) install routing table entries that won't be used or b) do the correct processing as outlined in this memo or c) mark the advertisement as unusable when presented with a Link State ID that has one or more of the host bits set. However, in the interest of interoperability, implementations of this memo should only set the host bits in Link State IDs when absolutely necessary.

The change affects Sections 12.1.4, 12.4.3, 12.4.5, 16.2, 16.3, 16.4, 16.5, 16.6, A.4.4 and A.4.5.

E.3 Obsoleting LSInfinity in router links advertisements

The metric of LSInfinity can no longer be used in router links advertisements to indicate unusable links. This is being done for several reasons:

- First, it removes any possible confusion in an OSPF area as to just which routers/networks are reachable in the area. For example, the above virtual link fix relies on detecting the existence of virtual links when running the Dijkstra. However, when one-directional links (i.e., cost of LSInfinity in one direction, but not the other) are possible, some routers may detect the existence of virtual links while others may not. This may defeat the fix for the virtual link problem.
- Second, it also helps OSPF's Multicast routing extensions (MOSPF), because one-way reachability can lead to places that are reachable via unicast but not multicast, or vice versa.

The two prior justifications for using LSInfinity in router links advertisements were 1) it was a way to not support TOS before TOS was optional and 2) it went along with strong TOS interpretations. These justifications are no longer valid. However, LSInfinity will continue to mean "unreachable" in summary link advertisements and AS external link advertisements, as some implementations use this as an alternative to the premature aging procedure specified in Section 14.1.

This change has one other side effect. When two routers are connected via a virtual link whose underlying path is non-TOS-capable, they must now revert to being non-TOS-capable routers themselves, instead of the previous behavior of advertising the non-zero TOS costs of the virtual link as LSInfinity. See Section 15 for details.

E.4 TOS encoding updated

The encoding of TOS in OSPF link state advertisements has been updated to reflect the new TOS value (minimize monetary cost) defined by [RFC 1349]. The OSPF encoding is defined in Section 12.3, which is identical in content to Section A.5 of [RFC 1349].

E.5 Summarizing routes into transit areas

RFC 1247 mandated that routes associated with Area A are never summarized back into Area A. However, this memo further reduces the number of summary links originated by refusing to summarize into Area A those routes having next hops belonging to Area A. This is an optimization over RFC 1247 behavior when virtual links are present. For example, in the area configuration of Figure 6, Router RT11 need only originate a single summary link having the (collapsed) destination N9-N11,H1 into its connected transit area Area 2, since all of its other eligible routes have next hops belonging to Area 2 (and as such only need be advertised by other area border routers; in this case, Routers RT10 and RT7). This is the logical equivalent of a Distance Vector protocol's split horizon logic.

This change appears in Section 12.4.3.

E.6 Summarizing routes into stub areas

RFC 1247 mandated that area border routers attached to stub areas must summarize all inter-area routes into the stub areas. However, while area border routers connected to OSPF stub areas must originate default summary links into the stub area, they need not summarize other routes into the stub area. The amount of summarization done into stub areas can instead be put under configuration control. The network administrator can then make the trade-off between optimal routing and database size.

This change appears in Sections 12.4.3 and 12.4.4.

E.7 Flushing anomalous network links advertisements

Text was added indicating that a network links advertisement whose Link State ID is equal to one of the router's own IP interface addresses should be considered to be self-originated, regardless of the setting of the advertisement's Advertising Router. If the Advertising Router of such an advertisement is not equal to the router's own Router ID, the advertisement should be flushed from the routing domain using the premature aging procedure specified in Section 14.1. This case should be rare, and it indicates that the router's Router ID has changed since originating the advertisement.

Failure to flush these anomalous advertisements could lead to multiple network links advertisements having the same Link State ID. This in turn could cause the Dijkstra calculation in Section 16.1 to fail, since it would be impossible to tell which network links advertisement is valid (i.e., more recent).

This change appears in Sections 13.4 and 14.1.

E.8 Required Statistics appendix deleted

Appendix D of RFC 1247, which specified a list of required statistics for an OSPF implementation, has been deleted. That appendix has been superseded by the two documents: the OSPF Version 2 Management Information Base and the OSPF Version 2 Traps.

E.9 Other changes

The following small changes were also made to RFC 1247:

- When representing unnumbered point-to-point networks in router links advertisements, the corresponding **Link Data** field should be set to the unnumbered interface's MIB-II [RFC 1213] ifIndex value.
- A comment was added to Step 3 of the Dijkstra algorithm in Section 16.1. When removing vertices from the candidate list, and when there is a choice of vertices closest to the root, network vertices must be chosen before router vertices in order to necessarily find all equal-cost paths.

- A comment was added to Section 12.4.3 noting that a summary link advertisement cannot express a reachable destination whose path cost equals or exceeds LSInfinity.
- A comment was added to Section 15 noting that a virtual link whose underlying path has cost greater than hexadecimal `0xffff` (the maximum size of an interface cost in a router links advertisement) should be considered inoperational.
- An option was added to the definition of area address ranges, allowing the network administrator to specify that a particular range should not be advertised to other OSPF areas. This enables the existence of certain networks to be hidden from other areas. This change appears in Sections 12.4.3 and C.2.
- A note was added reminding implementors that bit E (the AS boundary router indication) should never be set in a router links advertisement for a stub area, since stub areas cannot contain AS boundary routers. This change appears in Section 12.4.1.

F An algorithm for assigning Link State IDs

In RFC 1247, the Link State ID in AS external link advertisements and summary link advertisements is set to the described network's IP address. This memo relaxes that requirement, allowing one or more of the network's host bits to be set in the Link State ID. This allows the router to originate separate advertisements for networks having the same addresses, yet different masks. Such networks can occur in the presence of supernetting and subnet 0s (see Section E.2 for more information).

This appendix gives one possible algorithm for setting the host bits in Link State IDs. The choice of such an algorithm is a local decision. Separate routers are free to use different algorithms, since the only advertisements affected are the ones that the router itself originates. The only requirement on the algorithms used is that the network's IP address should be used as the Link State ID (the RFC 1247 behavior) whenever possible.

The algorithm below is stated for AS external link advertisements. This is only for clarity; the exact same algorithm can be used for summary link advertisements. Suppose that the router wishes to originate an AS external link advertisement for a network having address NA and mask NM1. The following steps are then used to determine the advertisement's Link State ID:

1. Determine whether the router is already originating an AS external link advertisement with Link State ID equal to NA (in such an advertisement the router itself will be listed as the advertisement's Advertising Router). If not, set the Link State ID equal to NA (the RFC 1247 behavior) and the algorithm terminates. Otherwise,
2. Obtain the network mask from the body of the already existing AS external link advertisement. Call this mask NM2. There are then two cases:
   - NM1 is longer (i.e., more specific) than NM2. In this case, set the Link State ID in the new advertisement to be the network [NA,NM1] with all the host bits set (i.e., equal to NA or'ed together with all the bits that are not set in NM1, which is network [NA,NM1]'s broadcast address).
   - NM2 is longer than NM1. In this case, change the existing advertisement (having Link State ID of NA) to reference the new network [NA,NM1] by incrementing the sequence number, changing the mask in the body to NM1 and using the cost for the new network. Then originate a new advertisement for the old network [NA,NM2], with Link State ID equal to NA or'ed together with the bits that are not set in NM2 (i.e., network [NA,NM2]'s broadcast address).

The above algorithm assumes that all masks are contiguous; this ensures that when two networks have the same address, one mask is more specific than the other. The algorithm also assumes that no network exists having an address equal to another network's broadcast address. Given these two assumptions, the above algorithm always produces unique Link State IDs. The above algorithm can also be reworded as follows: When originating an AS external link state advertisement, try to use the network number as the Link State ID. If that produces a conflict, examine the two networks in conflict. One will be a subset of the other. For the less specific network, use the network number as the Link State ID and for the more specific use the network's broadcast address instead (i.e., flip all the "host" bits to 1). If the most specific network was originated first, this will cause you to originate two link state advertisements at once.

As an example of the algorithm, consider its operation when the following sequence of events occurs in a single router (Router A).

1. Router A wants to originate an AS external link advertisement for [`10.0.0.0,255.255.255.0`]:
   (a) A Link State ID of `10.0.0.0` is used.
2. Router A then wants to originate an AS external link advertisement for [`10.0.0.0,255.255.0.0`]:

    (a) The advertisement for [`10.0.0,0,255.255.255.0`] is reoriginated using a new Link State ID of `10.0.0.255`.
    (b) A Link State ID of `10.0.0.0` is used for [`10.0.0.0,255.255.0.0`].

3. Router A then wants to originate an AS external link advertisement for [`10.0.0.0,255.0.0.0`]:
    (a) The advertisement for [`10.0.0.0,255.255.0.0`] is reoriginated using a new Link State ID of `10.0.255.255`.
    (b) A Link State ID of `10.0.0.0` is used for [`10.0.0.0,255.0.0.0`].
    (c) The network [`10.0.0.0,255.255.255.0`] keeps its Link State ID of `10.0.0.255`.

Security Considerations

All OSPF protocol exchanges are authenticated. This is accomplished through authentication fields contained in the OSPF packet header. For more information, see Sections 8.1, 8.2, and Appendix D.

Author's Address

John Moy
Proteon, Inc.
9 Technology Drive
Westborough, MA 01581

Phone: 508-898-2800
Fax: 508-898-3176
EMail: jmoy@proteon.com