Network Working Group
Request for Comments: 1131

J. Moy
Proteon, Inc.
October 1989

The OSPF Specification

Status of this Memo

This RFC suggests a proposed protocol for the Internet community, and requests discussion and suggestions for improvements. Distribution of this memo is unlimited.

Contents

1 Introduction

This document is a specification of the Open Shortest Path First (OSPF) internet routing protocol. OSPF is classified as an Internal Gateway Protocol (IGP). This means that it distributes routing information between routers belonging to a single Autonomous System. The OSPF protocol is based on SPF or link-state technology. This is a departure from the Bellman-Ford base used by traditional 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 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 reponds quickly to topology changes, yet involves small amounts of routing protocol traffic.

The author would like to thank Rob Coltun, Milo Medin, Mike Petry 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 an SPF-based 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 a 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 subnets. 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

Here is a collection of definitions for terms that have a specific meaning to the 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.

Internal Gateway Protocol The routing protocol spoken by the routers belonging to an Autonomous system. Abbreviated as IGP. Each Autonomous System has a single IGP. Different 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 paper, an IP network or subnet. 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. 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 (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 the Hello Protocol.

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

Link state advertisement Refers to 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 PDNs, and the ethernet data link layer for ethernets.

1.3 Brief history of SPF-based routing technology

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

The first SPF-based 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 SPF-based 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 an SPF-based protocol. This was accomplished by going to a lollipop-shaped sequence space and synchronizing the topological database over newly formed adjacencies, enabling the interval between link state advertisements to be increased by an order of magnitude.

An SPF-based 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 internet environment.

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, configuration items and required management statistics 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 paper.

2 The Topological Database

The database of the Autonomous System's topology 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

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 the 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 discovered on these nets using the 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.

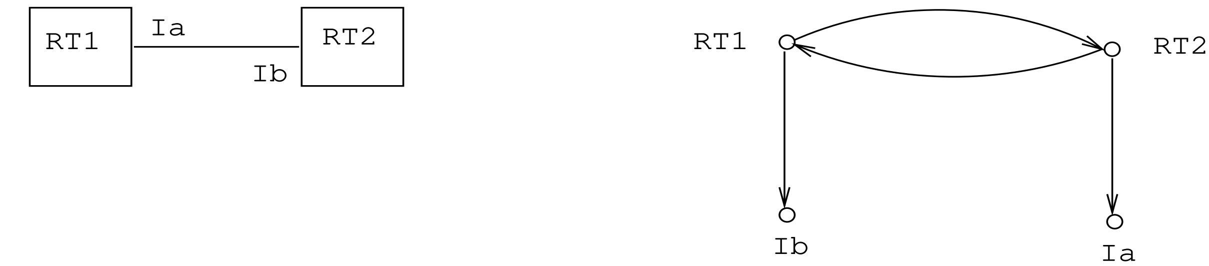
The neighborhood of each network node in the graph depends on whether the network has multi-access capabilities (either broadcast or non-broadcast) and, if so, the number of routers having an interface to the network. The three cases are depicted in Figure 1. Rectangles indicate routers. Circles and oblongs indicate multi-access networks. Router names are prefixed with the letters RT and network names with N. Router interface names are prefixed by 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).

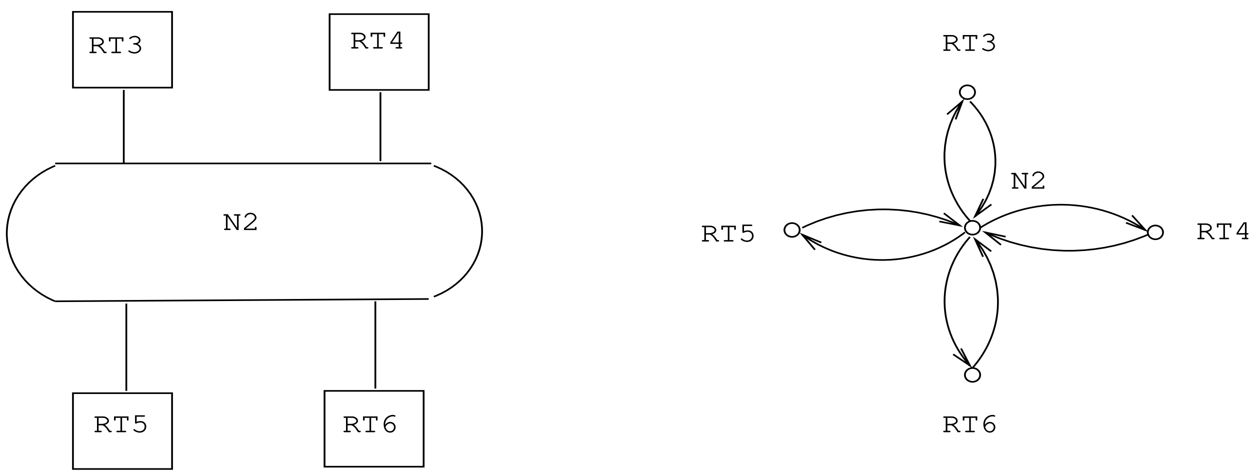
When multiple routers are attached to a multi-access network, the directed graph shows all routers bidirectionally connected to the network vertex (again, see Figure 1). If only a single router is attached to a multi-access network, the network will appear in the directed graph as a stub connection.

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.

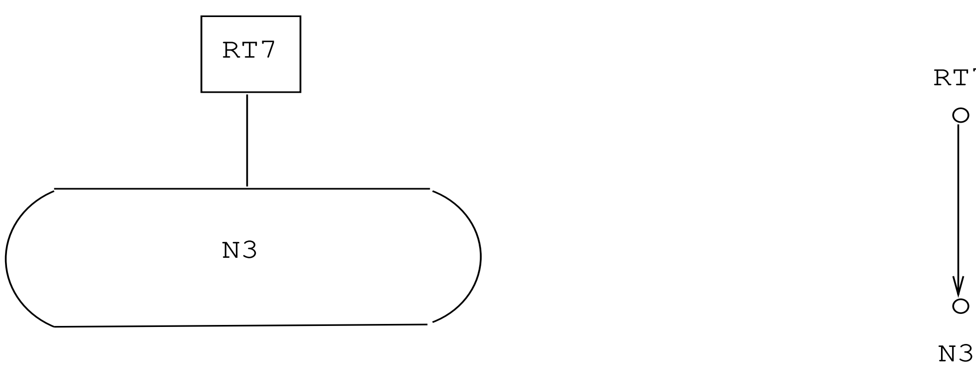
Figure 2 shows a sample map of an Autonomous System. The rectangle labelled H1 indicates a host, which has a SLIP connection to router RT12. Router RT12 is therefore advertising a host route. Lines between routers indicate physical point-to-point networks. The only point-to-point network that has been assigned interface addresses is the



Physical point-to-point networks



Multi-access networks



Stub multi-access networks

Figure 1: Network map components

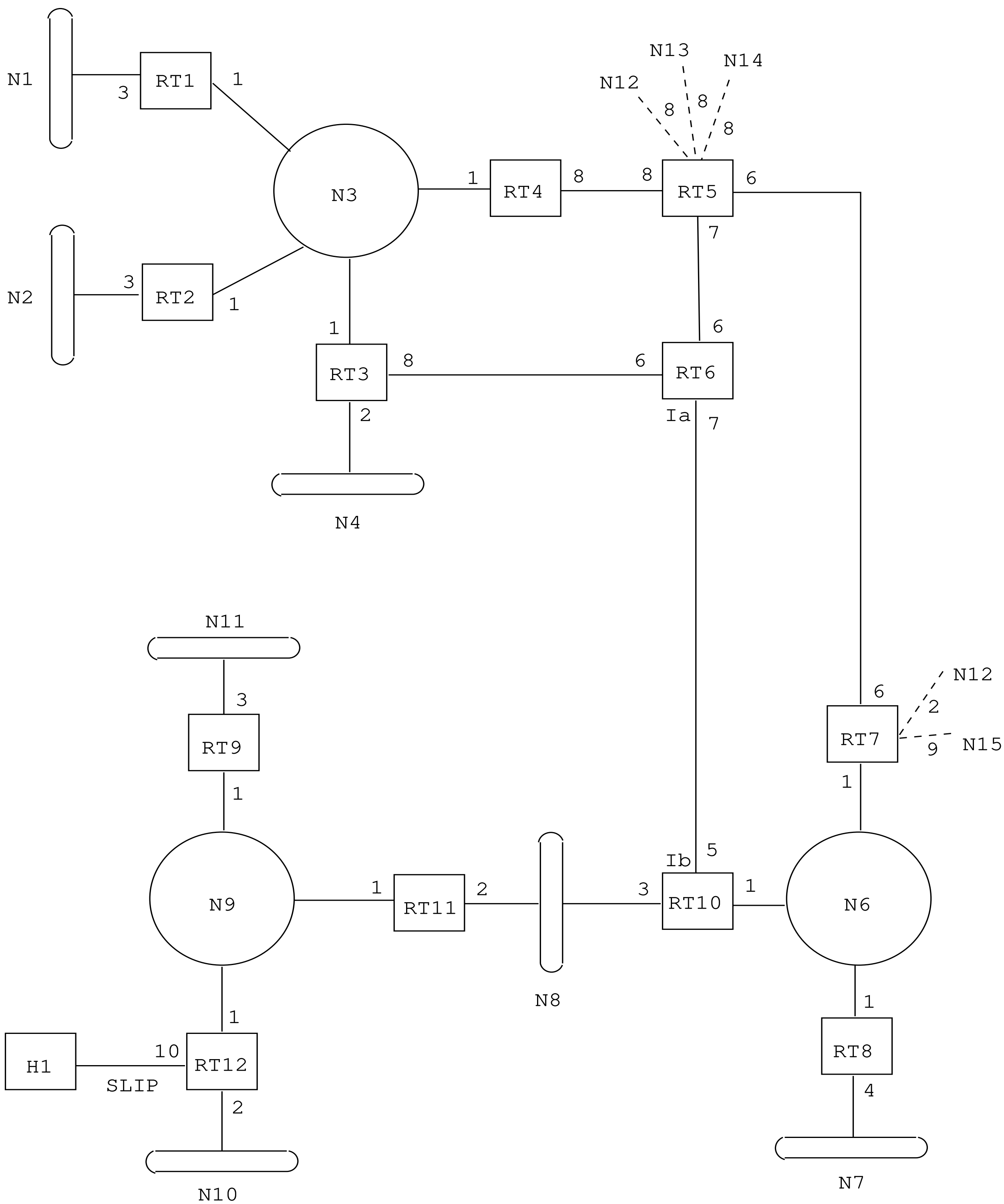


Figure 2: A sample Autonomous System

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 1. 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 are also a form of 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.

Here is an example of Type 1 external metric processing. Suppose that the routers RT7 and RT5 in our example are advertising Type 1 external metrics. For each external route, the distance to Router RT6 is calculated as the sum of the external route's cost and the distance of the advertising router to Router RT6. For every external destination, the

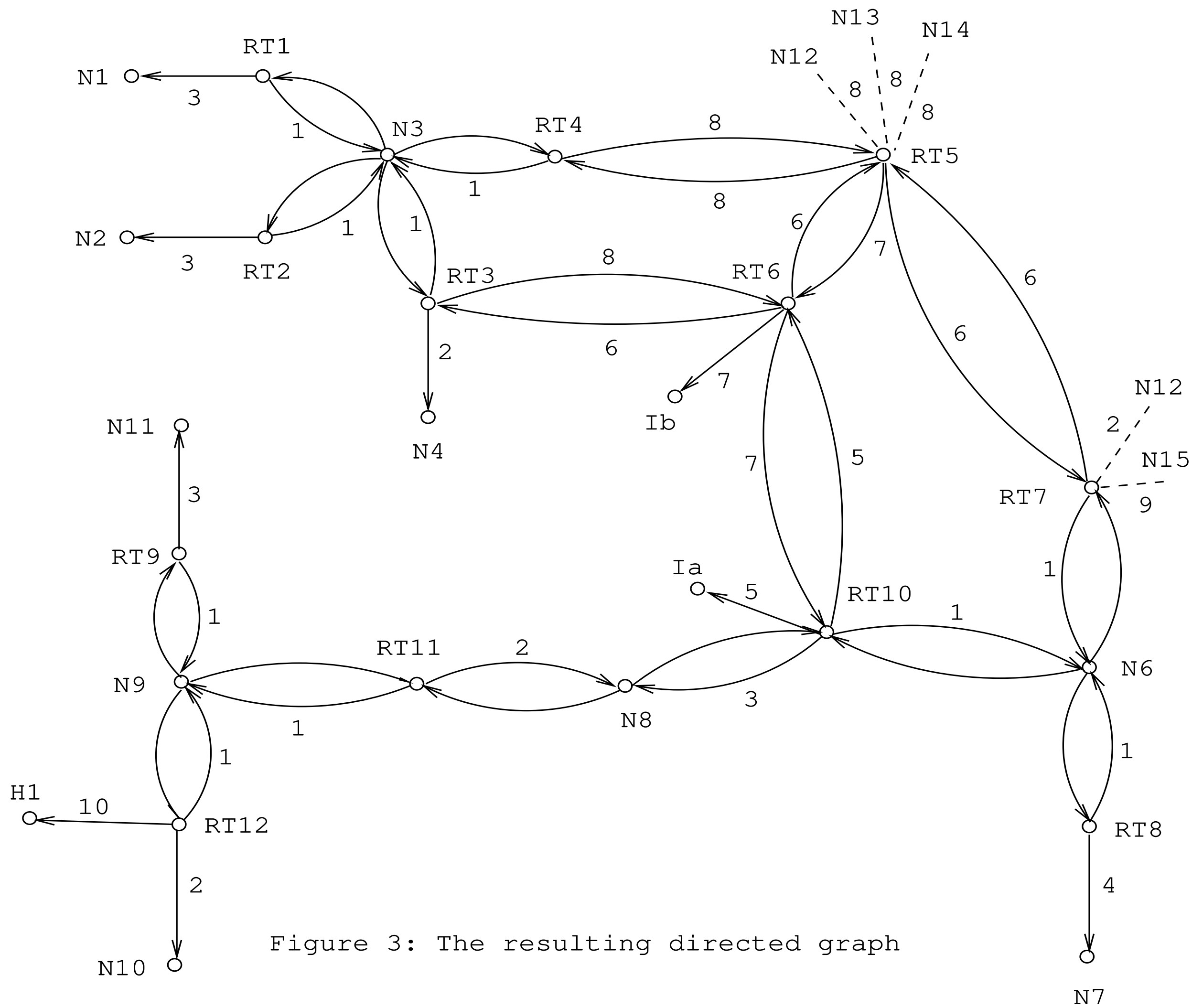
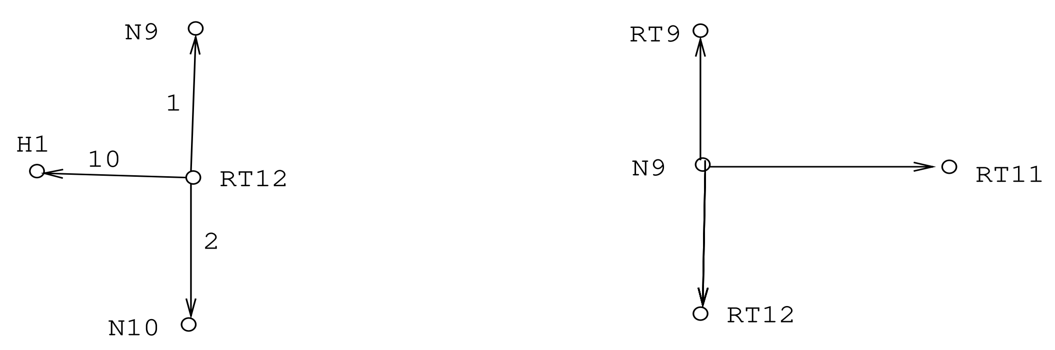


Figure 3: The resulting directed graph



A router links advertisement A network links advertisement

Figure 4: Individual link state components

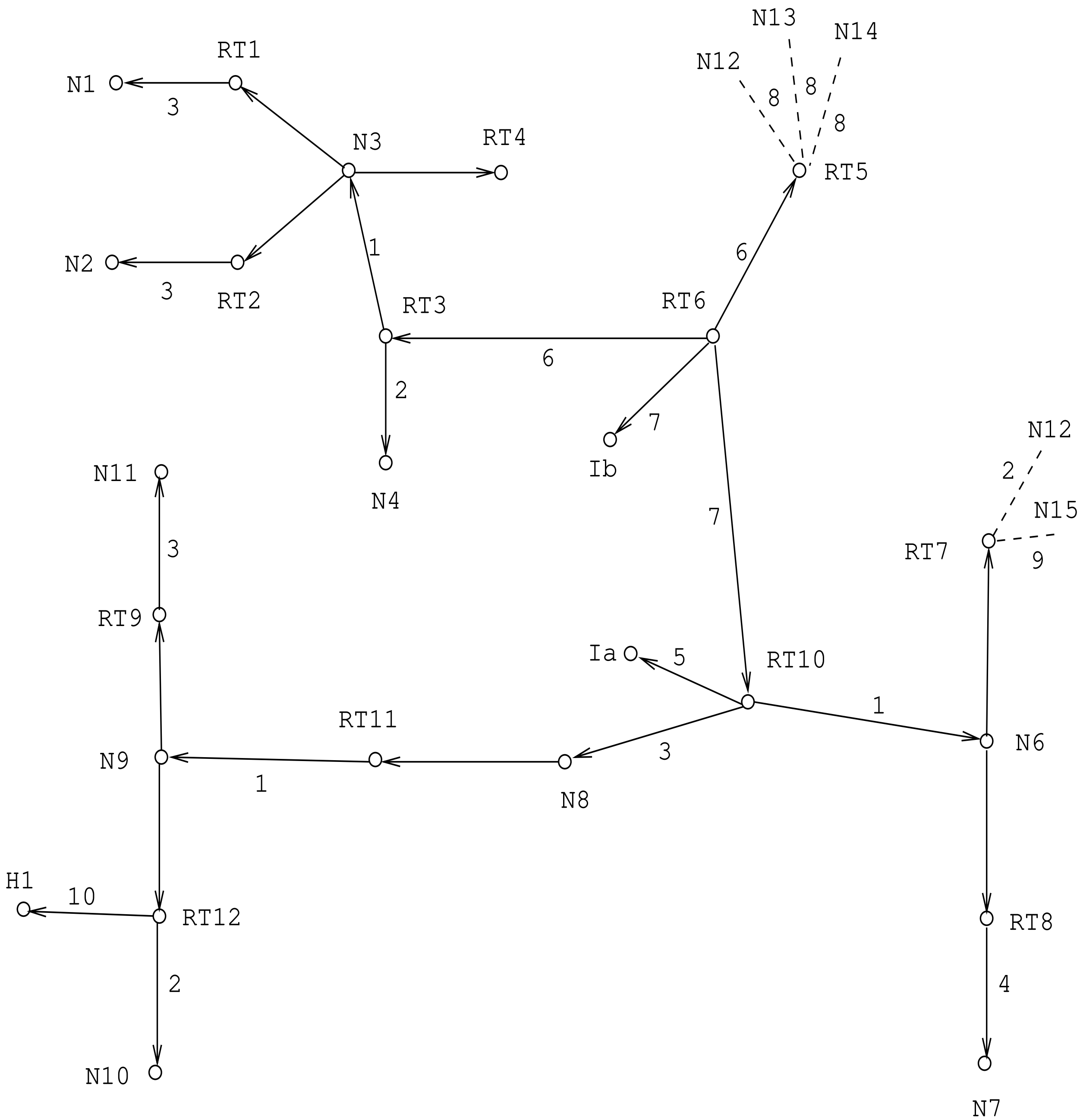


Figure 5: The SPF tree for router RT6

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 1: The portion of router's RT6 routing table listing local destinations.

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 2 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 2: 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 shortest external route 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.

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.

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 SPF 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 SPF 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. Before the addition of virtual links, the backbone has potentially several components. Routing within any one of these components cannot be influenced by information gained from any of the areas. Routing along virtual links is of course dictated by the topology of the associated 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 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 Any router that is not an internal router. 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, RT11. The third area consists of networks N9-N11 and host H1, along with their attached routers RT9, RT11, 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).

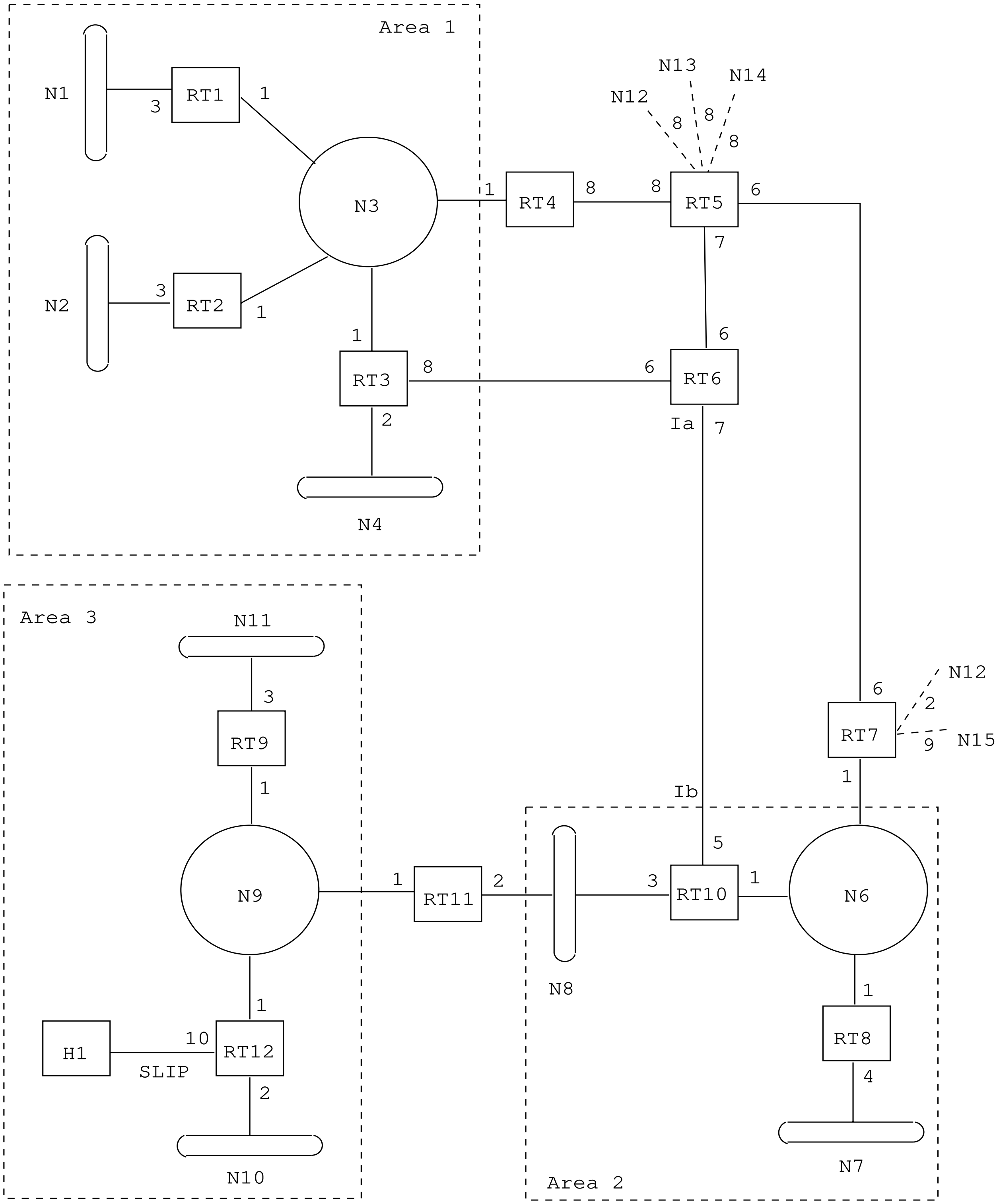


Figure 6: A sample OSPF area configuration

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 3. 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.

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

Table 3: 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. 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 and AS boundary routers that belong to the backbone. These calculation are shown in Table 4.

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 advertisements that router RT3 and RT4 will make into Area 1 are shown in Table 5.

This information enables an internal router in Area 1, 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.

One more iteration of the same logic enables router RT1 to decide between routing towards RT5 or RT7 when sending to a destination in another Autonomous System (one of the networks N12-N15).

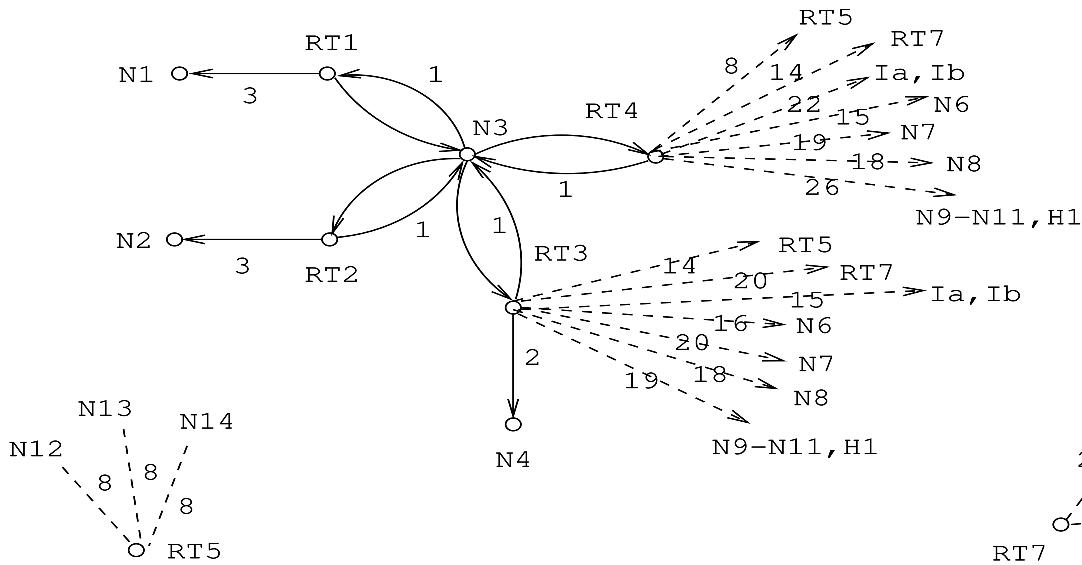


Figure 7: Area 1's Database

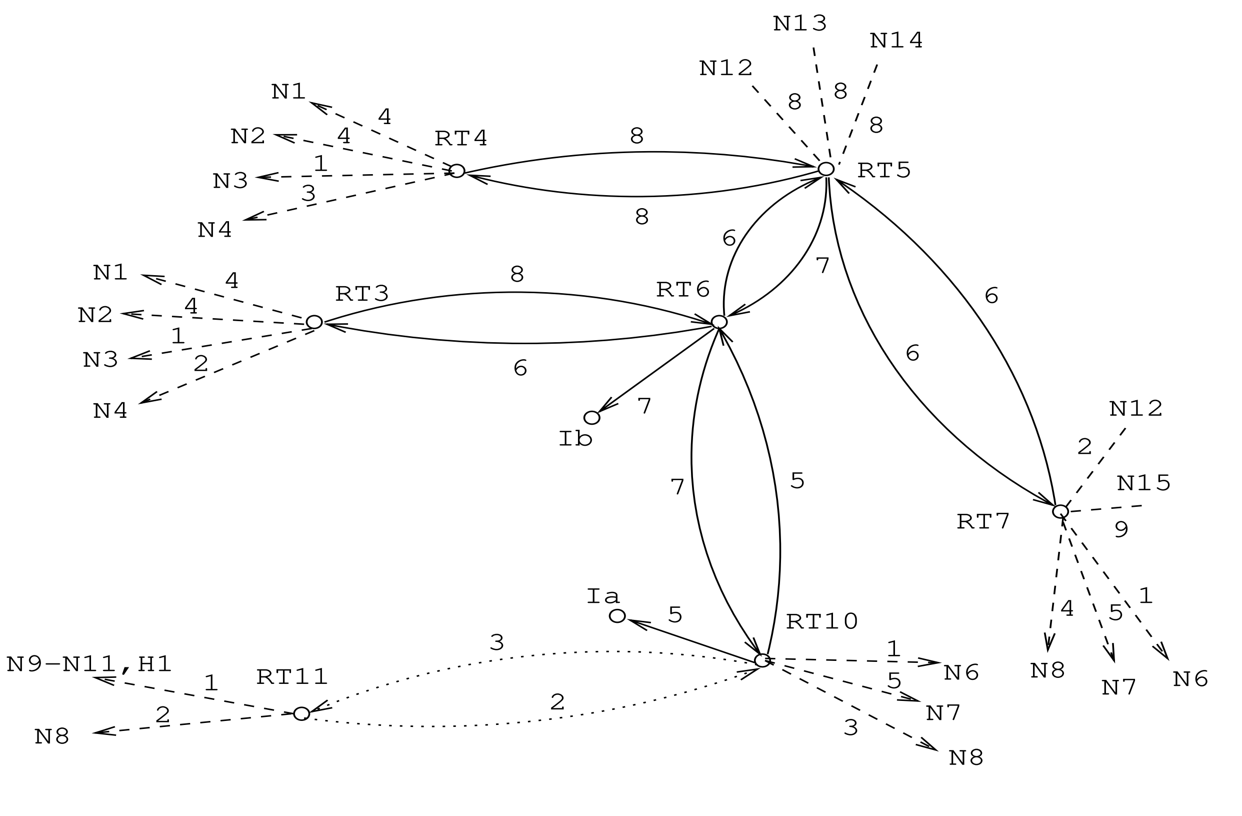


Figure 8: The backbone database

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 4: Backbone distances calculated by routers RT3 and RT4.

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 5: Destinations advertised into Area 1 by routers RT3 and RT4.

A failure of the line between routers RT6 and RT10 will cause the backbone to become disconnected. Configuring another 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 subnet masks. 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 64 variable-sized subnets: 16 subnets of size 4K, 16 subnets of size 256, and 32 subnets of size 8. The following table shows some of the resulting network addresses together with their masks:

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

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. The 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.

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 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.

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 the basic routing algorithm runs in each area. Routers having interfaces to multiple areas run multiple copies of the basic algorithm. A brief summary of the basic 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 Hello Protocol to acquire neighbors. The router sends Hello packets to its neighbors, and in turn receives their Hello packets. On broadcast networks, the router dynamically detects its neighboring routers by sending its Hello packets to the multicast address AllSPFRouters. On multi-access networks, 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 may be 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. A 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 to 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. These routes are distributed verbatim to every participating router.

To utilize this information, the path to all routers advertising such externally derived information must be known throughout the AS. For that reason, the locations of these AS boundary routers are summarized by the 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. OSPF defines the following packet types. 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

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. A link state advertisement can be one of four types, dividing the topological database into four parts:

Router links advertisement Originated by all routers. This advertisement describes the collected states of the router's interfaces to an area. Flooded throughout a single area only.

Network links advertisement Originated by broadcast networks. This advertisement contains the list of routers connected to the network. Flooded throughout a single area only.

Summary link advertisement Originated by area border routers, and flooded throughout their associated areas. Describes a route to a destination outside the area, yet still inside the AS (i.e., an inter-area route).

AS external link advertisement Originated by AS boundary routers, and flooded throughout the AS. Contains a route to a destination in another Autonomous System.

As mentioned above, OSPF routing packets (with the exception of Hellos) are sent only over adjacencies. Note that this means that all 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.

IP multicast Certain OSPF packets use IP multicast. Support for receiving and sending IP multicasts, along with the appropriate lower-level protocol support, is required. These IP multicast packets never travel more than one hop. For information on IP multicast, see [RFC 1112].

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 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 must 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 a PDN a dead 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 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.

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. Areas, OSPF 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.

Pointers to 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.

Pointer to the backbone 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. 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 info. 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 to the entire AS through AS external link advertisements.

List of AS external link advertisements Part of the topological database. These have 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.

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 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. Remember that each area maintains its own topological database. Router interfaces and adjacencies belong to a single area.

The 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 areas.

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

Area ID A 32 bit number identifying the area. 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. 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 adjacencies. A virtual adjacency is identified by the router ID of its other endpoint; its cost is the cost of the intra-area route that exists between the two routers.

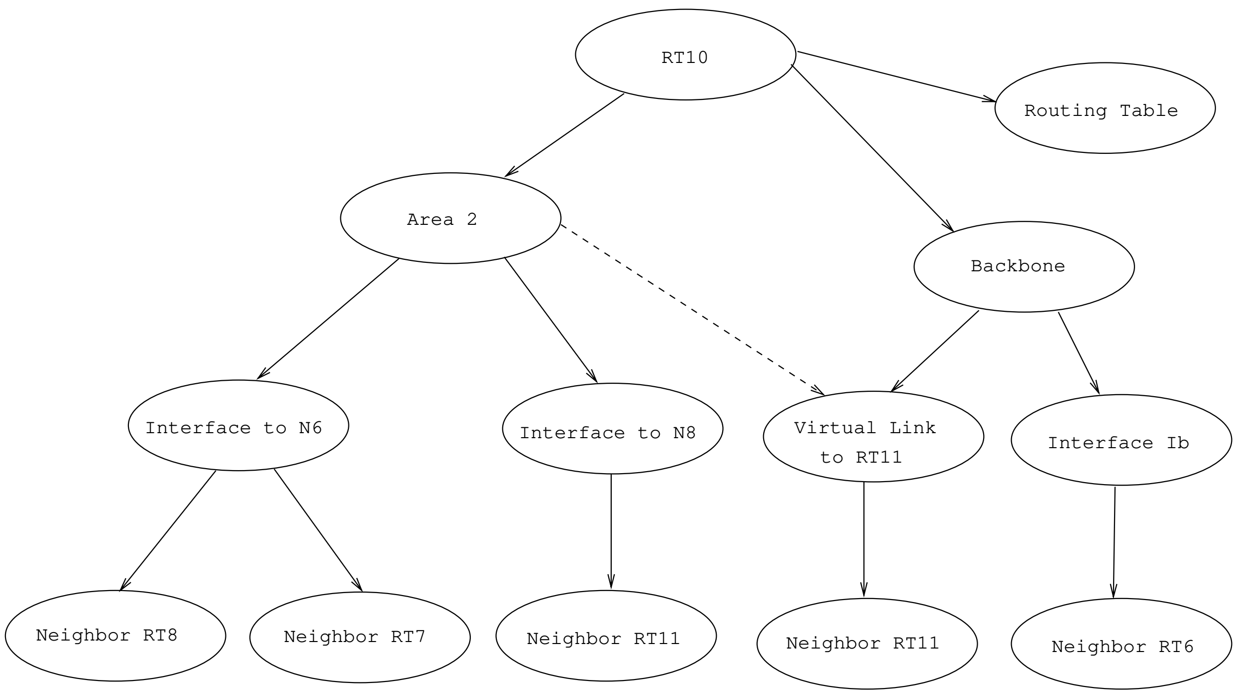


Figure 9: Router RT10's Data Structures

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. It describes the set of routers currently connected to the network.

List of summary links 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.

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

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 Hellos 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 hellos 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 hellos 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 an SPF-based 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 database. When the neighbor sees a link state advertisement that is more recent than its own instantiation of the same advertisement, 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.

After the Database Exchange Process is over, each router has a list of those link state advertisements for which the neighbor has more up to date instantiations. These are then requested in Link State Request Packets. Link State Request packets that are not satisfied are retransmitted at fixed intervals of time RxmtInterval. When all Link State Requests have been satisfied, the databases are synchronized and the routers are 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 12.1.2) 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. Broadcast 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 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 take 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 Updates, 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. 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.

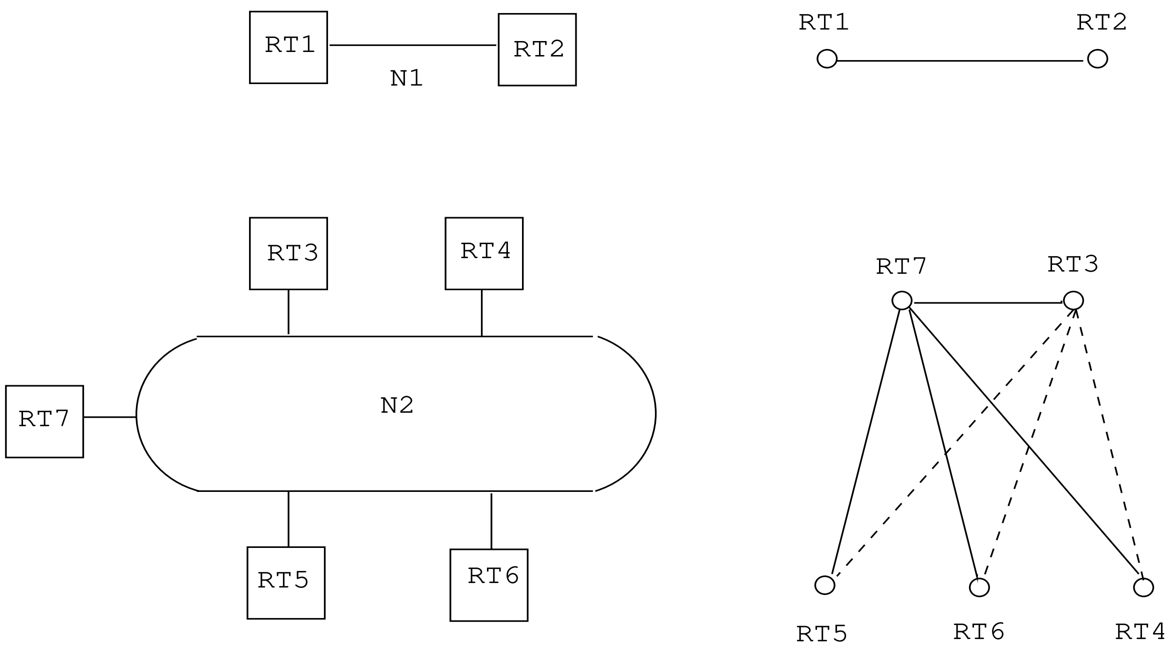
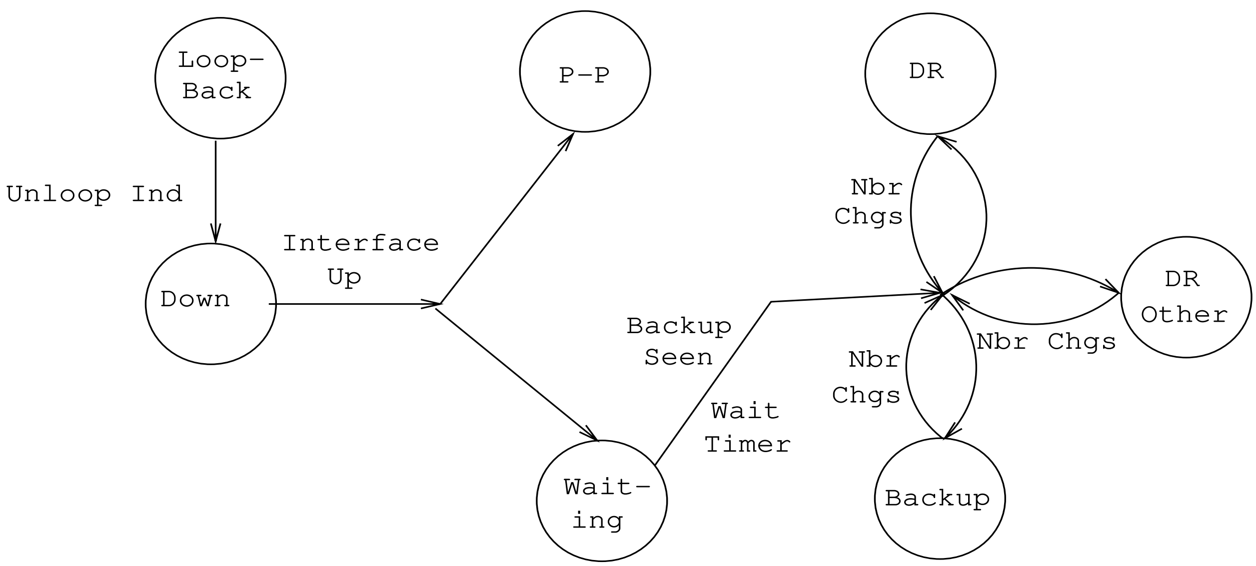


Figure 10: The graph of adjacencies



* Interface Down forces Down state
* Loop Ind forces Loopback state

Figure 11: Interface state changes

8 Protocol Packet Processing

This section discusses the general processing of 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 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.

8.1 Sending protocol packets

When a router sends a routing protocol packet, it fills in the fields of that standard header as follows. For more details on the header format consult Section A.2:

Version # Set to 1, 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 header.

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

Area ID The 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 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 **interface IP 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 packet types, consult the sections:

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

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 RT19 (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 1.
- The 16-bit 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:

[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.

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

  - *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.
  - *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 Authentication type specified must match the authentication type specified for the associated area.

Next, the packet must be authenticated. This depends on the authentication type 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 only on adjacencies. This means the packet must have been sent by one of the router's active neighbors. The sender is identified by the Router ID (source router) found in the 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:

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

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 number. 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 link state advertisement.

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 This indicates the portion of the IP interface address that identifies the attached network. This is often referred to as the subnet mask. Masking the IP interface address with this value yields the IP network number of the attached network.

Area ID The area ID 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 hellos. 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 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 interface IP address on the network. The Designated Router advertises link state for the network. The network link state advertisement is labelled with the Designated Router's IP address. This item is initialized to 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. Initialized to 0 indicating the lack of a Backup Designated Router.

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 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 authentication type 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 interface 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 intermediates 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 table 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 Hellos it receives. The router is not allowed to elect a Backup Designated Router nor 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. Hellos are sent to the neighbor every HelloInterval seconds.

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.

[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 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 os these events on OSPF protocol operation, consult Section 9.3.

Interface Up 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).

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

Backup seen 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, a Hello Packet may be received from a neighbor claiming to be itself the Backup Designated Router. Alternatively, a Hello Packet may be received from a neighbor claiming to be itself the Designated Router, and indicating that there is no Backup. 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.

Neighbor Change 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 **Neighbor Change** 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 (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 (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.

Loop Ind 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.

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

Interface Down 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 12.3 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: **Interface Up**

New state: Depends on 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: **Backup Seen**

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: **Wait Timer**

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: **Neighbor Change**

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: **Interface Down**
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: **Loop Ind**
New state: **Loopback**
Action: Since this interface is no longer connected to the attached network the actions associated with the above **Interface Down** event are executed.

State(s): **Loopback**
Event: **Unloop Ind**
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 **Interface Up** event is necessary before the interface again becomes fully functional.

9.4 Electing the Designated Router

The Designated Router calculation 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. If one or more of the routers have declared themselves Backup Designated Router (they have listed themselves as Backup 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, 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 (they have listed 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, promote the new Backup Designated Router to Designated Router.

4. If Router X is now newly the (Backup) Designated Router, or is now no longer the (Backup) Designated Router, repeat steps 2 and 3. 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 (Backup) Designated Router, it must start sending hellos 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 the (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 router can be chosen until the old Backup accepts its new Designated Router responsibilities.

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, hellos 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 a Hello packet is detailed in Section A.4. The Hello Packet contains the router's Router Priority (used in choosing the Designated Router), and the interval between Hello broadcasts (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.

In order to ensure two-way communication between adjacent routers, the Hello packet contains the list of all routers from which hellos 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 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

[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 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.

[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.

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 hellos to be sent. Hellos are then sent directly (as unicasts) to some subset of a router's neighbors. Sometimes an hello is sent periodically on a timer; at other times it is sent as a response to a received hello. A router's hello-sending behavior varies depending on whether the router itself is eligible to become Designated Router.

If the router is eligible to become Designated Router, it must periodically send hellos 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 hellos to all other neighbors. This means that any two eligible routers are always exchanging hellos, which is necessary for the correct operation of the Designated Router election algorithm. To minimize the number of hellos 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 hellos to both the Designated Router and the Backup Designated Router (if they exist). It must also send an hello in reply to an hello 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 hellos is determined by the neighbor's state. If the neighbor is in state **Down**, hellos are sent every PollInterval seconds. Otherwise, hellos 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 the next section.

Inactivity Timer A single shot timer whose firing indicates that no Hello Packet has been seen from this neighbor recently.

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**.

Sequence Number A 32-bit number identifying individual Database Description packets. When the neighbor state ExStart is entered, the 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 sequence number is then incremented by the master with each new Database Description packet sent. The slave's sequence number indicates the last packet received from the master. Only one packet is allowed outstanding at a time.

Neighbor ID The router ID of the neighboring router.

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 to construct the link state Originating ID for the attached network if the neighboring router is selected to be Designated Router. The neighbor IP address is learned when Hello packets are received from the neighbor, or is configured if this is a virtual adjacency (see Section C.4).

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 are four distinct types of link state advertisements in an area topological database: router links, network links, and 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 but not acknowledged on this adjacency. These will be retransmitted at intervals until they are acknowledged, or until the adjacency is destroyed.

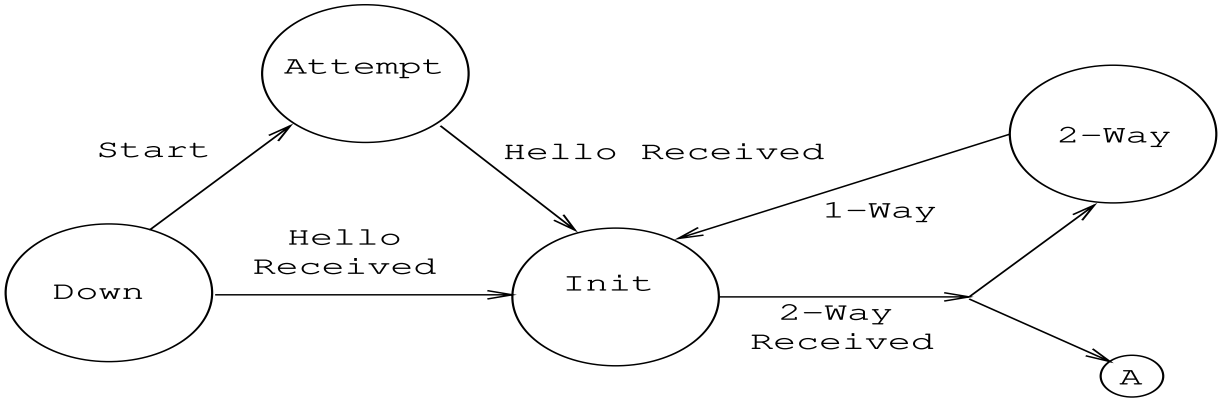
Database summary list The complete list of link state advertisements that make up the area topological database, at the moment the neighbor goes into Database Exchange state. This list is sent to the neighbor in Database Description packets.

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. The neighbor is later sent Link State Request packets until the initialization list becomes empty.

10.1 Neighbor states

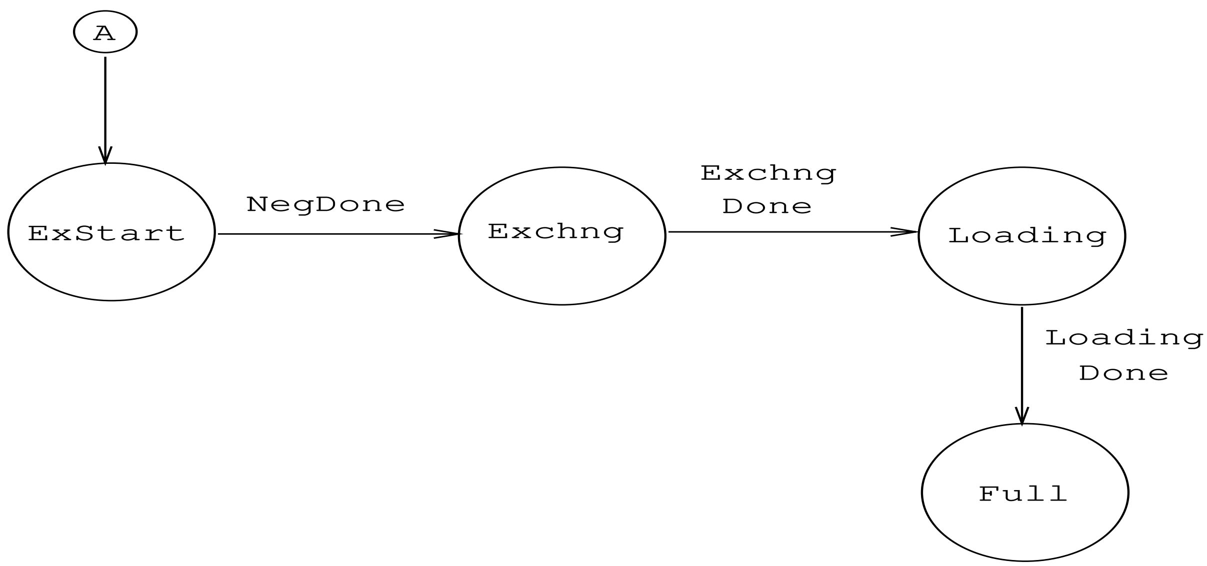
The state of a neighbor (really, the state of a conversation being held with a neighboring router) is documented in the following sections. The states are listed in order of progressing functionality. For example, the inoperative state is listed first, followed by a list of intermediates states before 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 the next section.

The graph in Figure 12 show 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.



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

Figure 12: Neighbor state changes (Hello Protocol)



* Event Seq Number Mismatch forces Exstart state
  Event 1-Way forces Init state
  Event KillNbr forces Down state
  Event Inactivity Timer forces Down State
  Event LLDown forces Down state
  Event AdjOK? leads to adjacency forming/breaking

Figure 13: Neighbor state changes (Database Exchange)

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 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 non-broadcast networks (or neighbors associated with virtual links). 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 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 sequence number, and is explicitly acknowledged. Only one Database Description Packet is allowed outstanding at any one time. All adjacencies in this 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 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:

Hello Received 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-Way Received 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 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.

Exchange Done 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.

Seq Number Mismatch A Database Description packet has been received that either a) has an unexpected sequence number or b) unexpectedly has the Init bit set. This indicates that some error has occurred during adjacency establishment.

BadLSReq A Link State Request has been received for a link state advertisement not contained in the database. This indicates an error in the synchronization 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 processing a Link State Update.

AdjOK? A decision must be made (again) as to whether an adjacency should be established 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.

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

Inactivity Timer 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 an neighbor's state changes, it may be necessary to rerun the Designated Router election algorithm. This is determined by whether the interface **Neighbor Change** 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 12.3).

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 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: **Hello Received**
New state: **Init**
Action: Restart the inactivity timer for the neighbor, since the neighbor has now been heard from.

State(s): **Down**
Event: **Hello Received**
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: **Hello Received**
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-Way Received**
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 sequence number for this neighbor. If this is the first time that an adjacency has been attempted, the 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: **NegDone**
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**. 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 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: **Exchange Done**
New state: **Loading**
Action: Start 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 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 **Exchange** and perform the actions associated with the above state machine entry for state **Init** and event **2-Way Received**.

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: **Seq Number Mismatch**
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 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): 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: **Inactivity Timer**
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-Way Received**
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-Way received**
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 if 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.4 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**, **HelloInt**, and **DeadInt** 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.

The source of the Hello Packet is then matched to one of the receiving interface's neighbors. The source is identified either by the Router ID found in the OSPF packet header or by IP source address found in the Hello's IP 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. When it is created, the neighbor structure's Neighbor ID and Neighbor IP address are set. The initial state of the neighbor is set to **Down**.

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 **Hello Received**.
- 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-Way Received**. Otherwise, the neighbor state machine should be executed with the event **1-Way Received**, 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 set accordingly.
- Next the Designated Router field in the Hello Packet is examined. If the neighbor is declaring itself to be Designated Router (Designated Router field = neighbor ID) 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 Designated Router item in the neighbor structure is set 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 ID) 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 Backup Designated Router item in the neighbor structure is set accordingly.

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-Way Received**. This causes an immediate state change to either state **2-Way** or state **Exstart**. The processing of the current packet should then continue in this new state.

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**) 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 sequence number to that specified by the master.
- The initialize(I) and master(MS) bits are off, the packet's sequence number equals the router's own 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 **Seq Number Mismatch** and stop processing the packet. Otherwise:

- If the initialize(I) bit is set, generate the neighbor event **Seq Number Mismatch** and stop processing the packet.
- If the router is master, and the packet's sequence number equals the router's own 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 sequence number is one less than the router's sequence number, the packet is a duplicate. Duplicates should be discarded by the master.
- If the router is slave, and the packet's sequence number is one more than the router's own 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 sequence number is equal to the router's sequence number, the packet is a duplicate. The slave must respond to duplicates by repeating the last Database Description packet that it sent.

[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.

- Else, generate the neighbor event **Seq Number Mismatch** and stop processing the packet.

Loading or Full In this state, the router has sent and received an entire sequence of Database Descriptions. The only packets received should be duplicates (see above). Any other packets received, including the reception of a packet with the Initialize(I) bit set, should generate the neighbor event **Seq Number Mismatch**. [8] Duplicates should be discarded by the master. The slave must respond to duplicates by repeating the last Database Description packet that it 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 router looks in its database to see whether it also has an instantiation 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 when the neighbor's state transitions to **Loading**.

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

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

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 by the master when the neighbor is in states **Exchange**, **Loading**, or **Full**. Link state Request Packets should be accepted by the slave when the neighbor is in states **Loading** or **Full**. In all other states Link State requests 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 synchronization procedure, 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 sending of these 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

[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 **Seq Number Mismatch** event, and therefore to also go back to **ExStart** state.

transitions into **Exchange** state) is listed in the neighbor **Database summary list**. When a new packet is to be sent, the 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 packet depends on whether the router is master or slave:

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

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

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

10.9 Sending Link State Request Packets

When the neighbor enters the state **Loading**, the link state request list contains a list of those link state advertisements that should be obtained from the neighbor. The beginning of this list is packaged into a Link State Request packet for transmission to the neighbor.

When the neighbor satisfies this request by sending the proper Link State Update packet(s), the link state request list is truncated accordingly and a new Link State Request packet is sent. Unsatisfied Link State Requests 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, 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 that 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 hellos, 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 indicates that it is itself the Designated Router and that it has heard hellos from RT1. This in turn causes RT1 to go to state **ExStart**, as it starts to bring up the adjacency.

RT1 begins by asserting itself as the master. When it sees that RT2 is indeed the master, RT1 transitions to slave state and adopts its neighbor's 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.

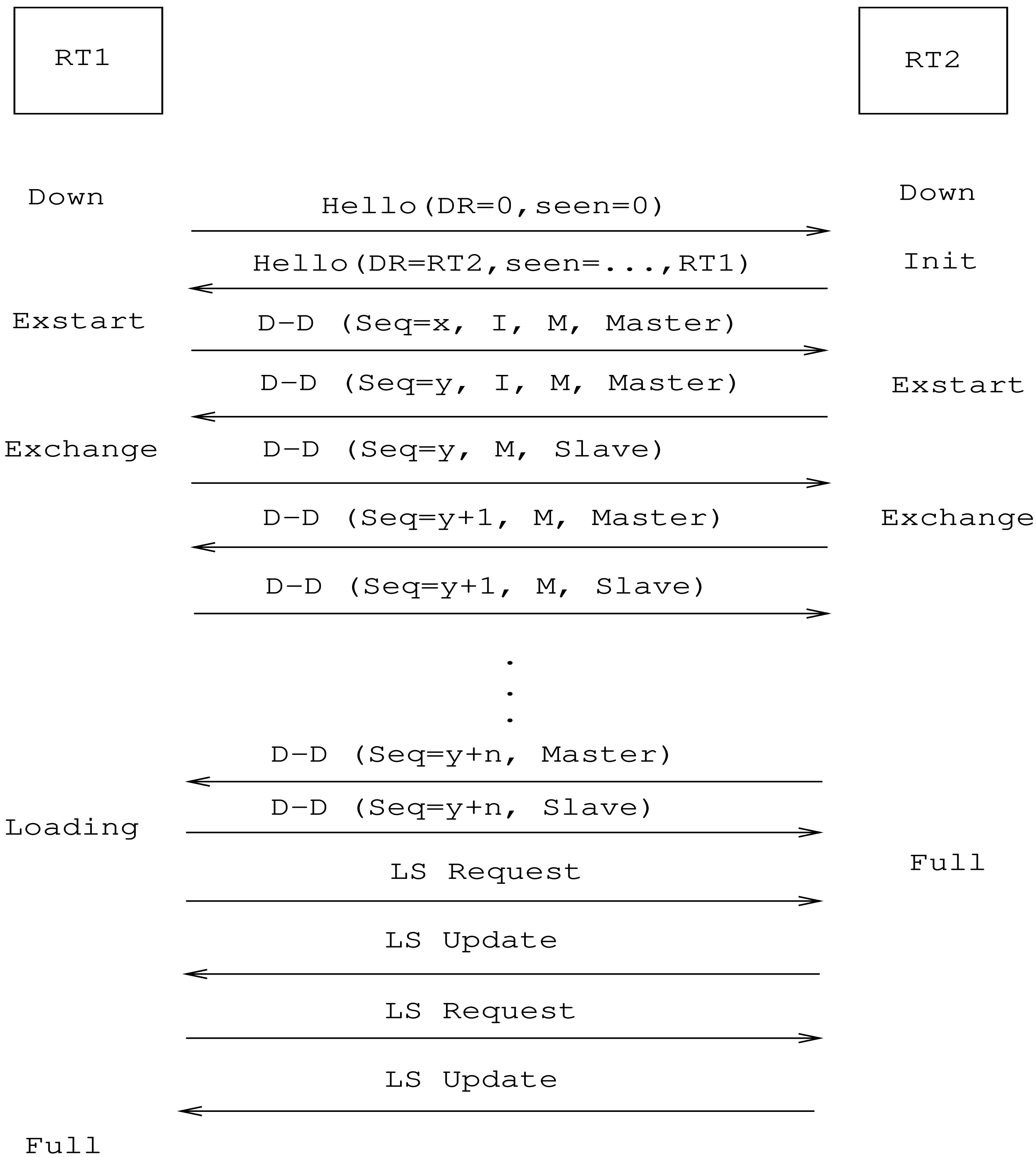


Figure 14: An adjacency bring-up example

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 routing table entry indicates the next hop towards the packet's destination. OSPF also provides for the existence of a default route (Destination ID = DefaultDestination). When the default route exists, it matches all IP destinations (although any other matching entry is a better match).

There is a single routing table in each router, built out of the link state information obtained from each attached area. A router begins building its routing table by examining the topological database (router links and network links) for each attached area. From these databases, the router constructs a shortest-path tree for each area, with itself as each tree's root. This yields the set of known intra-area routes. A byproduct of this calculation is the distance to all routers in the attached areas.

In particular, the distance to all area border routers is discovered. By examining the summary link advertisements originated by these area border routers, all inter-area routes are discovered, as well as the routes to all AS boundary routers. Finally, by examining the AS external link advertisements originated by these AS boundary routers, the complete set of routes is discovered. The building of the routing table is discussed in greater detail in Section 16.

Some of the routing table entries describe intermediate destinations used in the construction of the routing table, i.e., the area border routers and the AS boundary routers. A routing table entry can be identified by a combination of the following fields:

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 by 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, 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. 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.

Destination ID The destination's identifier or name. This depends on the destination's 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`).

Type of Service There can be a separate set of routes for each IP Type of Service.

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.

[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.

The rest of the routing table entry describes the set of paths to the described destination. The following fields pertain to the set of paths as a whole:

Path type There are three possible types of path(s) used to route traffic to the destination, listed here in order of preference: intra-area, inter-area, or AS 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 cost of the path(s) to the destination. For intra-area and inter-area paths, this cost is in terms of the link state metric. For AS external paths, this field indicates the link state metric component of the path's cost. For Type 1 external metrics, this describes the path's entire cost. For Type 2 external metrics, this describes the distance to the advertising AS boundary router.

For AS external destinations, the following fields are also specified, again for the set of paths as a whole:

External metric type The type of external metric type advertised by the AS boundary router(s). Either Type 1 (comparable to link state) or Type 2 (incomparable to link state). Type 2 metrics are always larger than the cost of any intra-area or inter-area path.

Type 2 cost For path(s) described by Type 2 external metrics, the cost advertised by the advertising AS boundary router(s).

Multiple equal-cost paths to a destination are stored when they exist. All these paths must be associated with the same area. Each one of the paths is further described by the fields:

Next hop The 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 external link that led to this path.

11.1 Two examples

The following two examples are based on the network map presented in Figure 2. The corresponding directed graph is shown in Figure 3. Note that both of these figures show a single metric per outbound interface, indicating that routes will not vary based on TOS.

First, assume that no areas are configured. All networks and routers belong to the backbone. The calculation of the routing table for router RT6 proceeds as described in Section 2.1. The resulting routing table is shown in Table 6. Destination types are abbreviated: Network as "N", area border router as "BR" and AS boundary router as "ASBR". Intra-area paths are indicated by the value 1 in the path type field, inter-area paths by the value 2, and AS external paths by the value 3.

Since there are no areas in this first example, there are no inter-area paths. Routers RT5 and RT7 are AS boundary routers. They are both assumed to be advertising Type 1 external metrics. In this example, there are no instances of multiple equal-cost shortest paths.

Next, assume that areas have been configured as in Figure 6. The following example describes Router RT4's routing table for this area configuration. Router RT4 has a connection to Area 1 and a backbone connection. This causes

Type	*Dest*	*Area*	*Path Type*	*Cost*	*Next Hop(s)*	*Advertising Router(s)*
N	N1	0	1	10	RT3	*
N	N2	0	1	10	RT3	*
N	N3	0	1	7	RT3	*
N	N4	0	1	8	RT3	*
N	Ib	0	1	7	*	*
N	Ia	0	1	12	RT10	*
N	N6	0	1	8	RT10	*
N	N7	0	1	12	RT10	*
N	N8	0	1	10	RT10	*
N	N9	0	1	11	RT10	*
N	N10	0	1	13	RT10	*
N	N11	0	1	14	RT10	*
N	H1	0	1	21	RT10	*
ASBR	RT5	0	1	6	RT5	*
ASBR	RT7	0	1	8	RT10	*
N	N12	*	3	10	RT10	RT7
N	N13	*	3	14	RT5	RT5
N	N14	*	3	14	RT5	RT5
N	N15	*	3	17	RT10	RT7

Table 6: The routing table for Router RT6 (no configured areas).

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 7.

Again, routers RT5 and RT7 are AS boundary routers. Routers RT3, RT4, RT7, RT10 and RT11 are area boundary routers. Note that there are two routing 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 router is itself an area border router. Note also that the backbone path to router RT4 is quite long. This can be fixed (below) by configuring a virtual link between RT4 and RT3. Such a virtual link would also lead to a much better path to the networks contained in Area 3.

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 N1 are all condensed to a single route when advertised to 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 are two equal-cost paths to network N12. However, they both use the same next hop (Router RT5).

As mentioned above, a few routes would improve if a virtual link to RT3 were configured. The routing table entries that would be affected by this change are displayed in Table 8. In this case, the virtual link would be associated with the first routing table entry for router RT3 appearing in Table 7. Note that this entry's associated area is equal to the virtual link's transit area (Area 1).

Type	*Dest*	*Area*	*Path Type*	*Cost*	*Next Hop(s)*	*Advertising Router(s)*
N	N1	1	1	4	RT1	*
N	N2	1	1	4	RT2	*
N	N3	1	1	1	*	*
N	N4	1	1	3	RT3	*
BR	RT3	1	1	1	*	*
N	Ib	0	1	22	RT5	*
N	Ia	0	1	27	RT5	*
BR	RT3	0	1	21	RT5	*
BR	RT7	0	1	14	RT5	*
BR	RT10	0	1	22	RT5	*
BR	RT11	0	1	25	RT5	*
ASBR	RT5	0	1	8	*	*
ASBR	RT7	0	1	14	RT5	*
N	N6	0	2	15	RT5	RT7
N	N7	0	2	19	RT5	RT7
N	N8	0	2	18	RT5	RT7
N	N9-N11,H1	0	2	26	RT5	RT11
N	N12	*	3	16	RT5	RT5,RT7
N	N13	*	3	16	RT5	RT5
N	N14	*	3	16	RT5	RT5
N	N15	*	3	23	RT5	RT7

Table 7: Router RT4's routing table in the presence of areas.

Type	*Dest*	*Area*	*Path Type*	*Cost*	*Next Hop(s)*	*Advertising Router(s)*
N	Ib	0	1	16	RT3	*
N	Ia	0	1	21	RT3	*
BR	RT3	0	1	1	*	*
BR	RT10	0	1	16	RT3	*
BR	RT11	0	1	19	RT3	*
N	N9-N11,H1	0	2	20	RT3	RT11

Table 8: Changes resulting from a configured virtual link.

12 Link State Advertisements

Each router in the Autonomous System originates one or more link state advertisements. There are four 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 through the Autonomous System.

Each link state advertisement begins with a standard 20-byte header. This link state header is discussed in the next section.

12.1 The Link State Header

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

Several of the OSPF packet types list link state advertisements. When the instantiation 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 header follows.

12.1.1 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. Each separate advertisement type is briefly described below:

LS type = 1 These are the router links advertisements. They describe the collected states of the router's interfaces. For more information, consult Section 12.3.1.

LS type = 2 These are the network links advertisements. They describe the set of routers attached to the network. For more information, consult Section 12.3.2

LS type = 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.

LS type = 5 These are the AS external link advertisements. Originated by AS boundary routers, they describe routes to destinations external to the Autonomous System.

12.1.2 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 following values:

Router links The originating router's Router ID.

Network links The IP interface address of the Designated Router on the network. Note that masking this ID with the network's subnet mask yields the network's IP address.

Summary link For Type 3 advertisements, the destination network's IP address. For Type 4 advertisements, the Router ID of the described AS boundary router.

AS external link The destination network's IP address.

12.1.3 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. Finally, AS external link advertisements are originated by AS boundary routers.

12.1.4 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 has a defined ordering. Given two different sequence numbers, this ordering determines which sequence number is newer. In the text, this is indicated as *newer*.

The shape of the sequence number space is shown in Figure 15. This sequence number space has been described in [Perlman] and [DEC]. It is often referred to as lollipop-shaped. The tail of the sequence space is ordered linearly, and the entire circular portion of the sequence space is *newer* than the tail.

The precise definition of *newer* is as follows. Twos complement arithmetic is used in the definitions. $<$ refers the comparison of twos complement integers, and $-$ refers to the subtraction of twos complement integers. N refers to the constant 2^{31}. Sequence number a is *newer* than sequence number b if one of the following holds:

$$b < 0, \quad a > b \quad \text{or}$$

$$a > 0, \quad b > 0, \quad \frac{N-1}{2} > (a-b) > 0 \quad \text{or}$$

$$a > 0, \quad b > 0, \quad (a-b) < \frac{N-1}{2}$$

The sequence number of a link state advertisement is incremented just before a new instantiation of the advertisement is originated. When the sequence number increments past $N-2$, it becomes 0. This logic yields the circular part of the sequence space. The sequence number value $N-1$ is illegal.

The sequence number $-N$ is reserved (and unused). Therefore the oldest sequence number is $(-N+1)$. A router uses this sequence number the first time it originates any link state advertisement. The router may then promote the sequence number if its link state advertisements from a previous instantiation still exist in the Autonomous System. This is covered in Section 13.4.

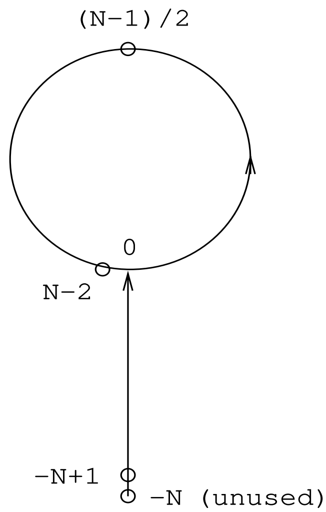


Figure 15: The lollipop-shaped sequence space

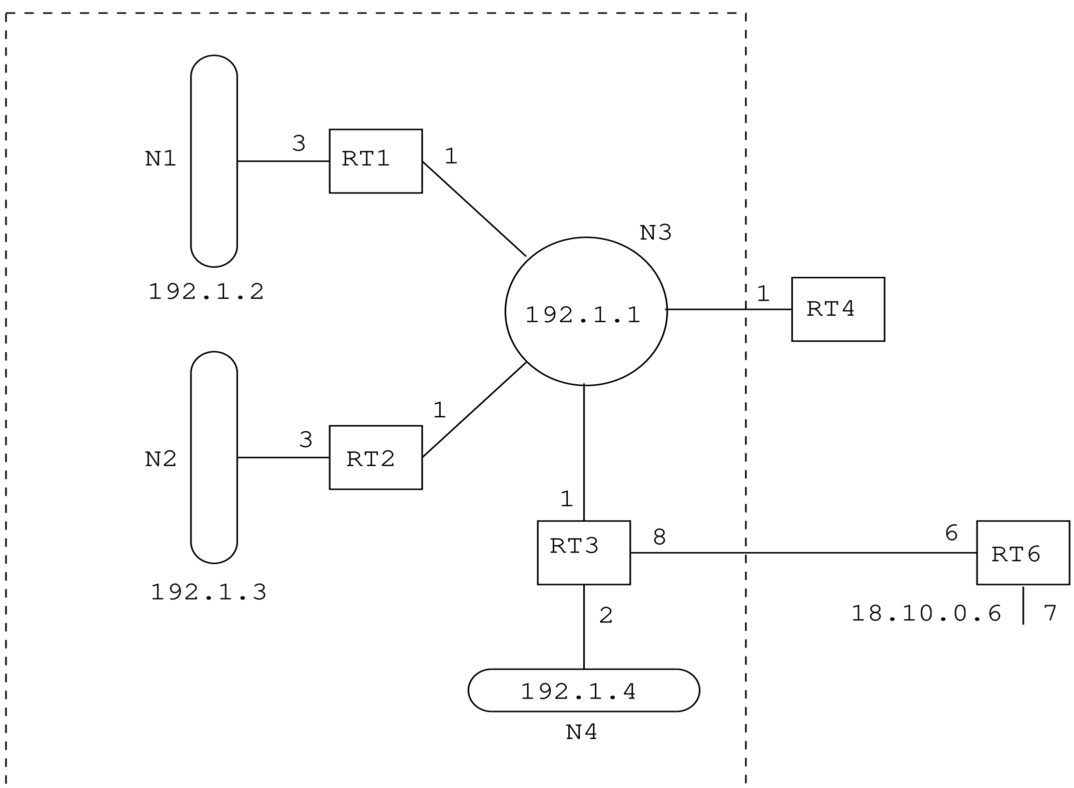


Figure 16: Area 1 with IP addresses shown

12.1.5 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 contained on any neighbor **Link state retransmission lists**. This indicates that it has been acknowledged by all adjacent neighbors.

Ages are examined when a router receives two instantiations of a link state advertisement, both having identical sequence numbers and checksums. An instantiation 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 instantiation having the smaller age is accepted as most recent. [10]

12.1.6 LS checksum

This field is the checksum of the complete contents of the advertisement, excepting the age field. The age field is excepted so that it can be updated easily. The length of the advertisement is also indicated in the link state header. 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 C of [RFC 994].

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 instantiations of an advertisement are the same, the LS checksum field is examined. If there is a difference, the instantiation with the larger checksum is considered to be most recent. [11]

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.

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.

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

[11] When two advertisements have different checksums, they are assumed to be separate instantiations. This can occur when a router restarts, and loses track of its previous sequence number. In this case, 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 instantiation. See Section 13.4 for further details.

The pieces of an area database are: the router links advertisements, network links advertisements, and summary links advertisements for the area (all listed in the area data structure) and the AS external link advertisements for the whole Autonomous System. Note that the AS external link advertisements are common to all area databases.

An implementation of OSPF must be able to access individual pieces of an area database. This lookup function is based on LS type, Link State ID and Advertising Router. [12] There will be a single instantiation (the most up-to-date) of each link state advertisement in the database. 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.

12.3 Originating link state advertisements

A router may originate many types of link state advertisements. A router originates a router links advertisement for each area to which it belongs. If the router is also the Designated Router for one of its attached networks, it will originate a link state packet for that network.

Area border routers originate a single summary links advertisement for each known inter-area destination. AS boundary routers originate a single AS external links advertisement for each known AS external destination. 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. Remember that many link state advertisements can be contained in 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 links 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 link state advertisement for N3 into Area 1.

In this same figure, router RT5 will be originating 3 distinct AS external links advertisements (one for each of the networks N12-N14). These will be flooded throughout the entire AS.

Whenever a new instantiation 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 13.3 for details.

The events that cause a new instantiation of a link state advertisement to be originated are:

- *The LS refresh timer firing*. There is a LS refresh timer for each link state advertisement that the router has originated. The LS refresh timer is an interval timer, with length LSRefreshTimer. This periodic updating of link state advertisements is necessary for the maintenance of the LS sequence space. The LS refresh timer guarantees periodic originations regardless of any other events that cause new instantiations. There is one exception; summary link and AS external link advertisements that are solely indicating unreachability should not be refreshed.

When whatever is being described by a link state advertisement changes, a new advertisement is originated. Two instantiations of the same link state advertisement may not be originated within the time period MinLSInterval. This may require that the generation of the next instantiation to be delayed by up to MinLSInterval. The following events may cause a router to originate a new instantiation of an advertisement. These events should cause new originations only if the contents of the new advertisement would be different.

- *An interface's state changes* (see Section 9.1). This may mean that it is necessary to produce a new instantiation of the router links advertisement.

[12] 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 advertisements can have the same Link State ID.

- *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.
- *One of the neighboring routers changes to/from the FULL state.* This may mean that it is necessary to produce a new instantiation 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 two events concern area border routers only.

- *An intra-area (Type 1) route has been added/deleted/modified* in the routing table. This may cause a new instantiation of a summary links advertisement (for this route) to be originated in each attached area (this includes the backbone).
- *An inter-area (Type 2) route has been added/deleted/modified* in the routing table. This may cause a new instantiation of a summary links advertisement (for this route) to be originated in each attached area (but NEVER for the backbone).

The last event concerns AS boundary routers only.

- *An external route gained through direct experience* with an external routing protocol (like EGP) changes. This will cause the AS boundary router to originate a new instantiation of an external links advertisement.

The construction of each of the link state types is explained below. Each section assumes that the paths do not vary based on Type of Service. For the implications of separate costs for separate TOS values, consult Section 12.4.

12.3.1 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.7.1). The first 20 bytes of the advertisement consist of the generic link state header that was discussed in Section 12.1. Router links advertisements have LS type = 1.

A router indicates whether it is an area border router, or an AS boundary router, by setting the appropriate bits 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.

The router links advertisement then describes the router's working links to the area. Each link is typed. These link types indicate the kind of entity that is on the other end of the link. Each link is also labelled with its link ID. This ID gives a name to the entity that is on the other end of the link. The following table summarizes the values used for the type and Link ID fields:

Link type	*Description*	*Link ID*
1	Link to router	Neighbor Router ID
2	Link to transit network	Interface address of Designated Router
3	Link to stub network	IP network number

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 routers and transit networks, this field specifies the IP interface address of the associated router interface (this is needed by the routing table calculation). For links to stub networks, this field specifies the network's IP address mask.

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 describe the process of building the list of link records, suppose a router wishes to build router links advertisement for an 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 (router) whose link ID is the Router ID of the neighboring router and whose Link Data specifies the interface IP address.
  - If the neighboring router's IP address is known, add a Type 3 link (stub network) whose 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. In the case of an unnumbered serial line, the neighbor will not have an IP address and no stub link should be added.
- Else, if the state of the interface is **Loopback**, add a Type 3 link (stub network) whose link ID is the IP interface address, whose Link Data is the mask `0xffffffff` indicating a host route, and whose cost is 0. Unnumbered serial line interfaces do not generate link state information in **Loopback** state.
- 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 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 interface IP address. Otherwise, add a link as if the interface state were **Waiting** (see above).

Unless specified above, the cost of each link generated 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 Date 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 16. 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 N3. RT3's two router advertisements then have the following values (refer to Section A.7.1 for the field definitions).

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

LS age = 0                      ;always true on origination
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

; RT3's router links advertisement for the backbone

LS age = 0                      ;always true on origination
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 = 0.0.0.0         ;Interface to unnumbered SL
        Type = 1                    ;connects to router
        # other metrics = 0
        TOS 0 metric = 8
```

12.3.2 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 an 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 routers listed in the advertisement are those that are fully adjacent to the Designated Router They are identified in the advertisement by their Router IDs. Each link from the transit network to an attached router has cost 0.

The Link State ID for a network links advertisement is the IP interface address of the Designated Router.

As an example, 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 16 for the address assignments):

```
; network links advertisement for network N3

LS age = 0                      ;always true on origination
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
```

12.3.3 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 links advertisements are generated by area border routers. The precise summary routes to advertise into an area are determined by examining the routing table structure (see Section 11). Only intra-area routes are advertised into the backbone. 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:

- 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 AS 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 links advertisement for the route. [13]
- 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 ID and metric equal to the routing table entry's cost.
- 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 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. A single Type 3 advertisement must be made for each range, with Link State ID equal to the range's address and cost equal to the smallest cost of any of the component networks.

If a router advertises a summary advertisement for a destination which then becomes unreachable, the router must then originate a new summary advertisement, having metric LSInfinity. Also, if the destination is still reachable, yet cannot 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

[13] 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.

associated with some non-backbone area; it would thus no longer be advertisable to the backbone), a new summary advertisement must be advertised with metric LSInfinity.

As an example, 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 links advertisements. Consider in particular router RT4. Its routing table was calculated as the second example in Section 11.1. 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 advertisement. Finally, the routes to networks N9,N10,N11 and host H9 are advertised by a single summary link. 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 16.

```
; summary link advertisement for network N1,
; originated by router RT4 into the backbone

LS age = 0                  ;always true on origination
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
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
```

12.3.4 AS external links

Each AS external link advertisement describes a route to a destination that is external to the AS. AS external link advertisements are the only type of link state advertisements that are flooded throughout the entire AS; all other types of link state advertisements are specific to a single area. 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. A default route may also be advertised. The destination for the default route is defined to be DefaultDestination.

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 an example, 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
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
```

12.4 TOS metrics

In each type of link state advertisement, different metrics can be advertised for each IP Type of Service (TOS). The TOS fields specified in the link state advertisements map directly to the TOS field in the IP header.

A metric for TOS 0 must always be specified. Metrics for other TOS values can be specified; if they are not, these metrics are assumed equal to the metric specified for TOS 0.

As an example, suppose the point-to-point link between routers RT3 and RT6 in Figure 16 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 that TOS. Router RT3 would then originate the following router links advertisement for the backbone:

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

LS age = 0                  ;always true on origination
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 = 0.0.0.0         ;Interface to unnumbered SL
        Type = 1                    ;connects to router
        # other metrics = 1
        TOS 0 metric = 8
                TOS = 2             ;High bandwidth
                metric = 1          ;traffic preferred
```

Conversely, suppose that the administrator does not want any high bandwidth traffic to go over a certain link. The cost of the link for TOS 2 would then be set to LSInfinity. [14]

Summary link advertisements and AS external 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.

[14] A similar technique can be done for routers that are unable to route based on Type of Service, yet wish to run the OSPF protocol. Such routers must avoid forwarding IP data traffic with non-zero TOS, since they cannot determine the best route (and so cannot be sure to avoid looping) for the packets with non-zero TOS. These routers should originate router links advertisements that indicate their interfaces are unavailable for non-zero TOS traffic. This is again accomplished using the metric LSInfinity.

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 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. The advertisement does not contain an area field, however, and the 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 link state checksum. If the checksum turns out to be invalid, discard the advertisement and get the next one from the Link State Update packet.
2. Find the instantiation 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) 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).
   (b) Remove the current database copy from all lists of link state advertisements (e.g., from all neighbors' **Link state retransmission lists**).
   (c) Install the new advertisement in the link state database (replacing the current database copy). This may cause the routing table calculation to be scheduled. The advertisement installation process is discussed further in Section 13.2.
   (d) 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.
   (e) If this new link state advertisement indicates that it was originated by this router itself, the router must advance the advertisement's link state sequence number, and issue a new instantiation of the advertisement (see Section 13.4).
3. Else, if the received advertisement is the same instantiation as the database copy (i.e., neither one is more recent) the following two steps should be preformed:
   (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.

4. 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 packet.

13.1 Determining which link state is newer

When a router encounters two instantiations of a link state advertisement, it must determine which is more recent. This occurred above when comparing a received advertisement to the 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 instantiations of the same advertisement, the LS sequence number, LS age, and LS checksum fields are used to determine which instantiation is more recent:

- The advertisement having the *newer* LS sequence number is more recent. See Section 12.1.4 for an explanation of the LS sequence number space. If both instantiations have the same LS sequence number, then:
- If the two instantiations have different LS checksums, then the instantiation having the larger LS checksum (when considered as a 16-bit unsigned integer) is considered more recent.
- Else, if only one of the instantiations is of age MaxAge, the instantiation of age MaxAge is considered to be more recent.
- Else, if the ages of the two instantiations differ by more than MaxAgeDiff, the instantiation having the smaller (younger) age is considered to be more recent.
- Else, the two instantiations 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 routing table structure to be recalculated. The contents of the new advertisement should be compared to the old instantiation, 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 LS type field:

Router links, network links 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. [15]

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

[15] 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 the shortest path tree [BBN]. These algorithms are beyond the scope of this specification.

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

Also, any old version of the advertisement must be removed from the database when the new advertisement is installed. This old version must also be removed from all lists of link state advertisements (e.g., the **Link state retransmission lists** for all neighbors; 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. The set of interfaces to flood the advertisement out of depends on the type of advertisement:

AS external links AS external links are flooded throughout the entire AS. The eligible interfaces are all interfaces, regardless of associated area, yet excluding the virtual links.

All other types All other types are specific to a single area (Area A). The eligible interfaces are all those interfaces associated with 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 instantiation of the new advertisement on the list, it indicates that the neighboring router has an instantiation of the advertisement already. Compare the new advertisement to the neighbor's copy:

   - If the new advertisement is less recent, then try the next neighbor.
   - If the two copies are the same instantiation, then delete the advertisement from the **Link state request list**, and try the next neighbor. [16]
   - 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, try the next neighbor.

   (d) At this point we are not positive that the new neighbor has an up-to-date instantiation 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. Any old instantiations of the advertisement should be removed from the **Link state retransmission list** at this time.

[16] This is how the **Link state request list** is emptied, which eventually causes the neighbor state to transition to **Full**. See Section 10.2.

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 and the next interface should be examined.

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 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 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 to receive a self-originated link state advertisement via the flooding procedure. If the advertisement received is a newer instantiation than the last instantiation 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 this case, the router must advance the sequence number for the advertisement one past the received sequence number, and originate a new instantiation of the advertisement.

Note also that if the type of the advertisement is Summary link or AS external link, the router may no longer have an (advertisable) route to the destination. In this case, a new advertisement must still be originated, with metric equal to LSInfinity.

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 3a 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 packet; it enables a single packet to indicate acknowledgments to several neighbors at once (through multicasting); and it randomizes the 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 the following table. 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 2a).	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 DR, otherwise do nothing.	Delayed acknowledgment sent.
Advertisement is a duplicate, and was treated as an implied acknowledgment (see Section 13, step 3a).	Delayed acknowledgment sent if advertisement received from DR, otherwise do nothing.	No acknowledgment sent.
Advertisement is a duplicate, and was not treated as an implied acknowledgment.	Direct acknowledgment sent.	Direct acknowledgment sent.

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 acks must be unicast separately over each adjacency (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 16. Suppose RT4 has been elected as DR, and RT3 as Backup 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 RT3, 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 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 the 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 packet is discarded.

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

- Does the advertisement acknowledged have an instantiation on the **Link state retransmission list** for the neighbor? If not, examine the next acknowledgment. Otherwise:
- If the acknowledgment is for the same instantiation 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 age field. The age is expressed in seconds. An advertisement's 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 age is incremented by InfTransDelay.

An advertisement's age is never incremented past the value MaxAge. As a router ages its link state database, an advertisement's age may reach MaxAge. At this time, the advertisement is reflooded just as if it was a newly originated advertisement. This flooding process is described in Section 13.3. In addition, when adding advertisements to a neighbor's **Database summary list**, those advertisements having age MaxAge are instead added to the neighbor's **Link state retransmission list**.

It will be a relatively rare occurrence for an advertisement's age to reach MaxAge. Usually, the advertisement will be replaced by a more recent instantiation before it ages out.

Advertisements having age MaxAge are not used in the routing table calculation. When such an advertisement is no longer contained on any neighbor **Link state retransmission lists** it is removed entirely from the link state database.

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

15 Virtual Links

The single backbone area (Area ID = 0) cannot be disconnected, or some areas of the Autonomous System will become unreachable. This is because all inter-area traffic traverses the backbone. The backbone is also responsible for distributing the inter-area routing information.

To establish/maintain connectivity of the backbone, virtual links can be configured through non-backbone areas. Virtual links serve to connect 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).

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 as a "virtual adjacency".

The presence of a virtual link can be detected only by the two endpoint routers. These two routers must determine the viability and cost of the virtual link. The mechanisms behind this determination are as follows:

- 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 **Interface Up** event occurs for a virtual link when its corresponding TOS 0 routing table entry becomes reachable. Conversely, the **Interface Down** event occurs when its TOS 0 routing table entry becomes unreachable. [17] 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.
- Virtual links belong to the backbone. Only routing traffic for the backbone area should be traversing the associated virtual adjacency.
- Virtual links are represented as UNNUMBERED point-to-point networks in backbone router links advertisements.
- 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.
- 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 protocol packets over the virtual link.
- 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.

[17] Only the TOS 0 routes are important here. This is because all routing protocol packets are sent with TOS= 0. See Appendix A.

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 = 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. The first step of the routing table calculation is to invalidate the present routing table. The contents of the old table should be remembered however, so that routing table changes can be identified.

Changes made to the routing table 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 12.3).

The routing table calculation consists of the following steps.

1. *The present routing table is invalidated.* The routing table is built again from scratch. The old routing table is saved for comparison purposes.

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.

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. *For those routing entries whose next hop is over a virtual link, a real (physical) next hop is calculated.* The real next hop will be on one of the router's directly attached networks. This step only concerns routers having configured virtual links.

5. *AS external routes are calculated, through examination of AS external link advertisements.* The location of the AS boundary routers (which originate the AS external link advertisements) has been determined in steps 2-4.

Steps 2-5 are explained in further detail below. The explanations describe the calculations for a single TOS only. In general, each calculation must be performed for as many TOS values as there are differing routes. Any link state advertisement may specify a separate cost for each TOS (see Section 12.4). A cost for TOS 0 must always be specified. The cost of any other TOS, when not specified, defaults to the cost of TOS 0.

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. [18] The formation of the shortest path tree is done here in two stages. In the first step, 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 step, leaves are added to the tree by considering the links to stub networks.

[18] 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. Equal-cost multipath causes the algorithm to change only slightly.

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. Each transit vertex has an associated link state advertisement. Throughout the shortest path calculation, the following data is also associated with each transit vertex:

Vertex (node) ID The vertex's identifier. For router vertices this is an OSPF Router ID. For network vertices, this is the IP address of the network's Designated Router (the actual originator of the advertisement). In any case, the Vertex ID is the same as the associated link state advertisement's Link State ID.

Distance to root The current best distance from the root to the vertex, expressed in the link state metric.

List of next hops the list of next hops for the current best paths from the root to this vertex. There can be multiple best paths due to the equal-cost multipath capability.

The first stage of the procedure can now be summarized as follows: At any step, there is a list of candidate vertices. The best paths from the root to these vertices have not been found. The candidate vertex closest to the root is added to the shortest-path tree (and the routing table), removed from the candidate list, and its adjacent vertices are examined for possible addition to the candidate list. The algorithm then iterates. It terminates when the candidate list becomes empty.

This is described in detail below. 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.

Step 0 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 itself).

Step 1 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 link state database based on the vertex identifier. Each link described by the advertisement gives the cost to an adjacent vertex. For each advertised link, (say it joins vertex V to vertex W):

- 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.
- W is then a transit vertex (router ot transit network). Look up the vertex W's advertisement (router links or network links) in the area link state database. If the advertisement does not exist, or its age is $= MaxAge$, or it does not have a link back to vertex V, examine the next link in V's advertisement. Both ends of a link must advertise it before it will be used for data traffic. [19]
- If vertex W is already on the shortest-path tree, examine the next link in the advertisement.
- If the cost of the link (from V to W) is LSInfinity, the link should not be used for data traffic. In this case, examine the next link in the advertisement.
- Calculate the distance D of vertex W from the root, when the advertised link is used. D is equal to the distance from the root to vertex V, plus the advertised link's cost. If D is:
  - Greater than the value that already appears for vertex W on the candidate list, then examine the next link.

[19] This means that before data traffic will flow between a pair of neighboring routers, their link state databases must be synchronized. Before synchronization (neighbor state $<$ Full), neither router will advertise the other in its link state advertisements.

- Equal to the value that appears for vertex W on the 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 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 (see Section 16.1.1), setting the next hop values for W accordingly.

Step 2 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 algorithm 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).

Step 3 Possibly modify the routing table. For those routing table entries modified, the associated area will be Area A, the path type will be intra-area, and its cost will be equal to the distance from the root to the vertex.

If the newly added vertex is a router, multiple routing table entries may be added/modified, or none at all. If the new added router is an area border router, a routing table entry is added whose destination type is “area border router”. In addition, if the newly added router is an AS boundary router, the routing table entry of type “AS boundary router” for the router is located. Since routers can belong to more than one area, it is possible that an intra-area route of equal or better cost already exists using another area. In this case, do not modify the already existing entry. Otherwise, install the new values for the AS boundary router.

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 identifier (Link State ID) with its associated subnet mask (found in the associated link state 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 not be overwritten (because the previously found route will be shorter). Otherwise, a routing table entry for the IP network should be added.

Step 4 Iterate the algorithm by returning to Step 1.

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, the associated router links advertisement is found in the link state database. Each stub network link appearing in the advertisement is then examined.

If the cost of the stub network link is LSInfinity, the link should not be used for data traffic. In this case, go on to examine the next stub network link in the advertisement. Otherwise, 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 cost to the current best distance to the stub network. This is done by looking up the 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.

Otherwise, the stub link is added to the tree. As a result, the corresponding 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 distance, simply add this set of next hops to the routing table entry’s list of next hops. Otherwise, set the routing table entry’s distance to D, and set the entry’s list of next hops to the newly calculated set. Then go on to examine the next stub network link.

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 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 algorithms exist for incrementally updating the shortest-path tree (see [BBN]).

16.1.1 The next hop calculation

This section explains how to calculate the set of next hops as each vertex is added to the shortest path tree. Input to this calculation is the newly added vertex (also called the destination) and its parent in the tree.

If there is at least one intervening router between the newly added vertex and the root, the newly added vertex 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 the interface indicated by the newly added link. No IP address is required for these next hops.

In the second case, the newly added vertex is a router, and its parent vertex is a network. The list of next hops is then determined by examining the newly added router's link state 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.

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 or an AS boundary router. For each summary link advertisement:

- If the cost specified by the advertisement is LSInfinity, then examine the next advertisement.
- If the advertisement was originated by the router itself, examine the next advertisement.
- Else, call the destination described by the advertisement N, and the area border originating the advertisement BR. Look up the routing table entry for BR having 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.
- 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 "AS external", install the inter-area path to N, with associated area A, cost IAC, next hop equal to the list of next hops to router BR, and advertising router equal to BR.
- Else, if the paths present in the table are intra-area paths, do nothing with the advertisement (intra-area paths are always preferred).
- 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 Resolving virtual next hops

This step is only necessary in area-border routers having configured virtual links. In these routers, some of the routing table entries may have virtual next hops. That is, one or more of the next hops installed in Sections 16.1 and 16.2 may be over a virtual link. However, when forwarding data traffic to a destination, the next hops must always be on a directly attached network.

In this section, each virtual next hop is replaced by a real next hop. In the process a new routing table distance is calculated that may be smaller than the previously calculated distance. In this case, the list of next hops is pruned so that only those giving rise to the new shortest distance are included, and the routing table entry's distance is updated accordingly.

This resolution of virtual next hops is done only for Destination types Network or AS Boundary router. Suppose that one of a routing table entry's next hops is a virtual link. This is determined by the following combination: the routing table entry's path type is either intra-area or inter-area, the area associated with the routing table entry must be the backbone, yet the next hop belongs to a different area (the virtual link's transit area).

Let N be the above entry's destination, and A the virtual link's transit area. The real next hop (and new distance) is calculated as follows. Let D be a distance counter, and set the real next hop NH to null. Then, look up all the summary link advertisements for N in area A's database, performing the following steps for each advertisement: [20]

1. Call the border router that originated the advertisement BR. If there is no routing table entry for BR having A as associated area (i.e., BR is unreachable through Area A), examine the next advertisement.
2. Else, let X be the distance to BR via Area A. If the cost advertised by BR (call it Y) to the destination is LSInfinity, examine the next summary link advertisement. Else, the cost to destination N through area border router BR is X+Y.
3. If next hop NH is null or X+Y is smaller is smaller than D, set D to X+Y and set the next hop NH to the next hop specified in router BR's entry.

At this point, the real next hop NH should be set, and the distance D calculated should be less than or equal to the cost originally specified in destination N's routing table entry. This same calculation should be done for all of N's virtual next hops, and then N's new cost set to the minimum calculated distance, with the its new set of next hops that combination of non-virtual and recalculated next hops that correspond to this (possibly same as original) distance.

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. Remember that the destination described by an AS external link advertisement is always a network. For each AS external link advertisement:

- If the cost specified by the advertisement is LSInfinity, then examine the next advertisement.
- Call the destination described by the advertisement N. 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. Call the distance to router ASBR X, 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.

[20] Note the similarity between this procedure and the calculation of inter-area routes by a router internal to Area A.

- 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 router ASBR, and advertising router equal to ASBR. If the external metric type is 1, then the cost is equal to X+Y. Else, the link state component of the route's cost is X, and the Type 2 cost is Y.
- Else, if the paths present in the table are not AS external paths, do nothing (AS external paths have the lowest priority).
- Otherwise, compare the cost of this new AS external path to the ones present in the table. Type 1 paths are always shorter than Type 2 paths. Type 1 paths are compared by looking at the sum of the distance to the advertising router and the advertised Type 1 metric (X+Y). Type 2 paths are compared by looking at the distance to the advertising routers, and then if necessary, the advertised Type 2 metrics.

  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.

16.5 Incremental updates — summary links

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, and let A be the area to which the advertisement belongs.

Look up the routing table entry for N. If the next hop to N is a virtual link through Area A (this means that the entry's associated area is the backbone, and the listed next hop does not belong to the backbone, but instead belongs to Area A), the real next hop must again be resolved. This means running the algorithm in Section 16.3 for destination N only.

Else, if there is an intra-area route to destination N nothing need be done (intra-area routes always take precedence). Otherwise, if Area A is the router's sole attached area, or Area A is the backbone, the procedure in Section 16.2 will have to be performed, but only for those summary link advertisements whose destination is N. Before this procedure is performed, the present routing table entry for N should be invalidated (but kept for comparison purposes). If this procedure leads to a virtual next hop, the algorithm in Section 16.3 will again have to be performed in order to calculate the real next hop.

If N's routing table entry changes, and N is an AS boundary router, the AS external links will have to be reexamined (Section 16.4).

16.6 Incremental updates — AS external links

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. If there is already an intra-area or inter-area route to the destination, no recalculation is necessary (these 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 12.3.3 for more information. If a previously advertised entry has been deleted, or is no longer advertisable to a particular area, it must be advertised with cost LSInfinity.

- 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 **Interface Up** 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 interface's IP address and the virtual neighbor's 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 **Interface Down** 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, or AS 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.

References

[BBN] McQuillan, J.M., Richer, I. and Rosen, E.C. *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, Radia. *Fault-Tolerant Broadcast of Routing Information.* Computer Networks, Dec. 1983.

[RFC 791] Postel, Jon. *Internet Protocol.* September 1981

[RFC 944] ANSI X3S3.3 86-60. *Final Text of DIS 8473, Protocol for Providing the Connectionless-mode Network Service.* March 1986.

[RFC 1010] Reynolds, J. and Postel, J. *Assigned Numbers.* May 1987.

[RFC 1112] Deering, S.E. *Host extensions for IP multicasting.* May 1988.

[RS-85-153] Leiner, Dr. Barry M., et.al. *The DARPA Internet Protocol Suite.* DDN Protocol Handbook, April 1985.

A Packet Formats

This appendix gives formats for the various OSPF packets. For each packet type, the division into fields is displayed, and then the field definitions are enumerated. Each OSPF packet begins with a standard 24 byte header. This header is described before the details of the type-specific portions of the packets 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. Each link state advertisement begins with a standard 20-byte link state header. Since all packets (except Hello) reference fields from the link state header, the link state header is described early in this Appendix.

There are four distinct types of link state advertisements (router links, network links, summary link and AS external link advertisements). All advertisements are transported by Link State Update packets; for this reason the individual advertisement formats are given in the section explaining the Link State Update packet.

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 network headers.

OSPF does not define a way to fragment its protocol packets, and depends on IP fragmentation when packets larger than the link layer segment sizes need to be transmitted. 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. 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 the 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 destined to these multicast addresses should never be forwarded. Such packets are meant to travel a single hop only; they must have their IP TTL 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 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 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 1010].

- *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 with the IP TOS field set to 0.

- *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 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.

```
0            8            16           24           32
             |            |            |
+------------+------------+-------------------------+
| Version #  |    Type    |     Packet length       |
+------------+------------+-------------------------+
|                     Router ID                     |
+---------------------------------------------------+
|                      Area ID                      |
+-------------------------+-------------------------+
|        Checksum         |         Autype          |
+-------------------------+-------------------------+
|                                                   |
+------------      Authentication      ------------+
|                                                   |
+---------------------------------------------------+
```

Version # The OSPF version number. This specification documents version 1 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.

1 – Hello
2 – Database Description
3 – Link State Request
4 – Link State Update
5 – Link State Ack

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 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.

Checksum The standard IP checksum of the entire contents of the packet, 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 save 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 E of the specification. Authentication types of greater than 255 may be assigned on a per Autonomous System basis. Authentication types of 0-255 are reserved for definition by this specification. The authentication types currently defined are:

0 – None
1 – Simple password

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

A.3 The Link State Advertisement header

The topological database is composed of individual link state advertisements. Each advertisement describes some local piece of the routing domain, e.g., a router links advertisement (LS type = 1) indicates those networks/neighbors to which a particular router is connected.

All link state advertisements begin with a common 20 byte header. This link state advertisement header contains enough information to uniquely identify the advertisement (LS type, Link State ID, and Advertising Router).

Multiple instantiations of a link state advertisement may exist in the routing domain at the same time. It is then necessary to determine which instantiation 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 header.

```
0          8          16          24         32
           |          |           |
+---------------------+----------------------+
|       LS age        |       LS type        |
+---------------------+----------------------+
|               Link State ID                |
+--------------------------------------------+
|            Advertising Router              |
+--------------------------------------------+
|            LS sequence number              |
+---------------------+----------------------+
|     LS checksum     |        length        |
+---------------------+----------------------+
```

LS age The time in seconds since the link state advertisement was originated.

LS type The type of the link state advertisement. Each link state type has a separate advertisement format; these formats are described in Section A.7. The link state types are as follows (see Section 12.1.1 for further explanation):

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 that piece of the internet environment that is being described by the advertisement. This is further discussed in Section 12.1.2. The contents of this field depend on the advertisement's LS type. For link state advertisement Types 1 and 4 it is a Router ID, for Types 3 and 5 it is an IP network number, and for Type 2 it is the IP interface address of the Designated Router (from which the IP network number can be derived).

Advertising Router The Router ID of the router that originated the link state advertisement. For Type 1 advertisements, this field is identical to the Link State ID. For Type 2 advertisements, it is the Router ID of the network's Designated Router. For advertisement Types 3 and 4, it is the Router ID of an area border router. Finally, for Type 5 advertisements it is the Router ID of an AS boundary router.

LS sequence number Detects old or duplicate link state advertisements. Successive instantiations of a link state advertisement are given successive LS sequence numbers.

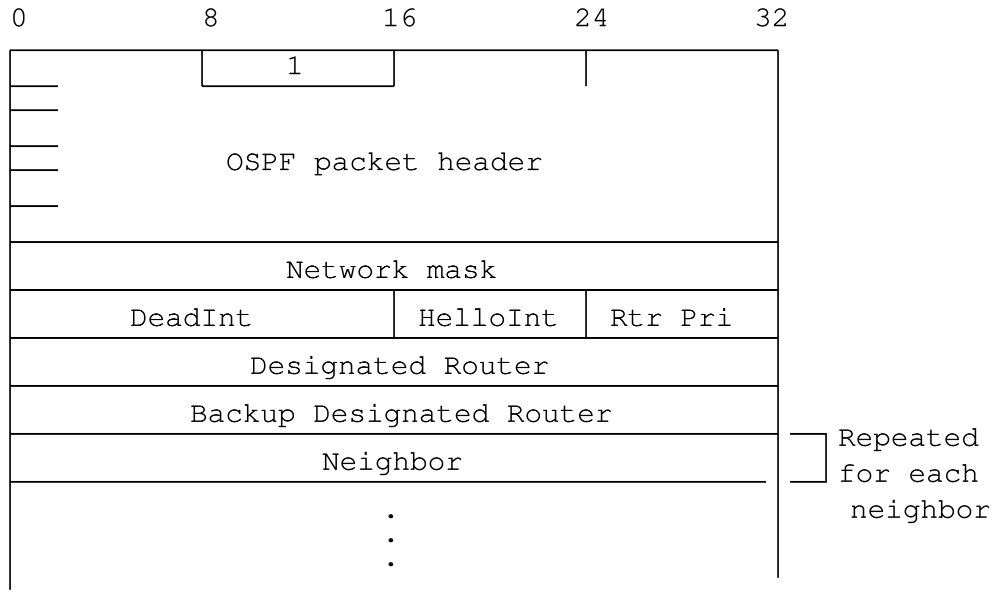
LS checksum The Fletcher checksum of the complete contents of the link state advertisement. See Section 12.1.6 for more details.

length The length in bytes of the link state advertisement. This includes the 20 byte link state header.

A.4 The Hello packet

Hello packets are OSPF packet type 1. These packets are sent periodically on all interfaces, including virtual links. Neighbor relationships are established and maintained through the exchanges of Hello packets. These packets are multicast on those physical networks having a multicast or broadcast capability. This enables dynamic discovery of neighboring routers.

All routers connected to a common network must agree on certain parameters (network mask, hello and dead intervals). These parameters are included in Hello packets, so that differences can inhibit the forming of neighbor relationships. Bidirectionality of communication is determined by including the list of all routers whose Hello packets have been seen recently. The packet fields Router Priority, Backup Designated Router, and Designated Router enable the (Backup) Designated Router to be selected through the exchange of Hello packets.



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`. If the interface is to a class A network, the network mask is `0xff000000`.

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

HelloInt 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.

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 if there is no Designated Router.

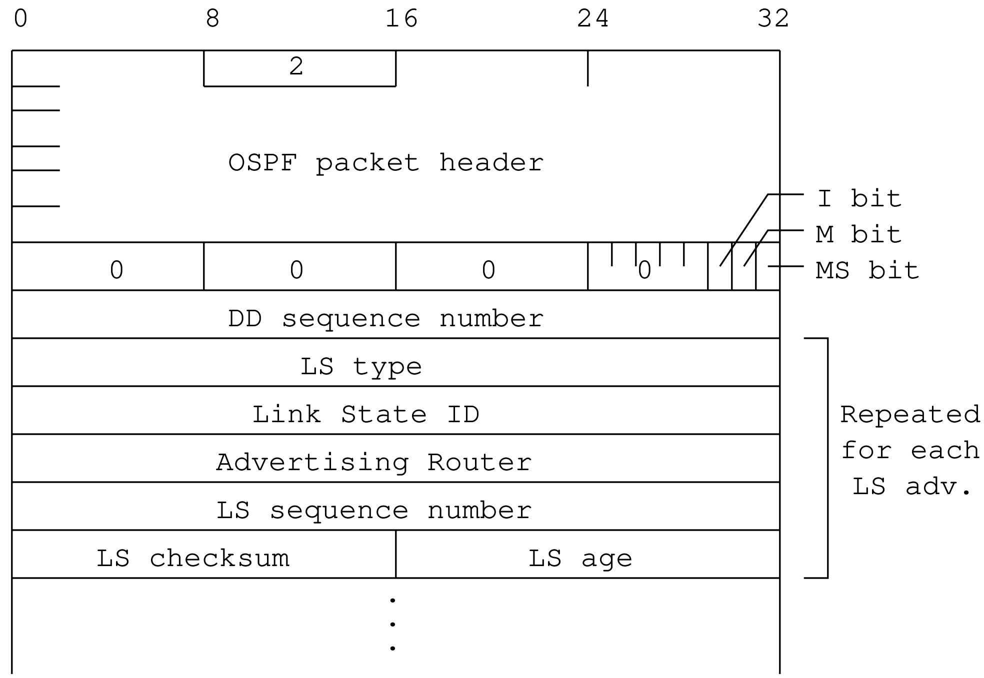
Backup Designated Router The identity of the Backup Designated Router for this network, in the view of the advertising router. The Backup Designated Router is identified here by its IP interface address on the network. Set to 0 if there is no backup Designated Router.

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

A.5 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' 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.



0 These fields are reserved. They must be 0.

I bit The Init bit. When set to 1, this packet is the first in the sequence of database descriptions.

M bit The more bit. When set to 1, it indicates that more database descriptions 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.

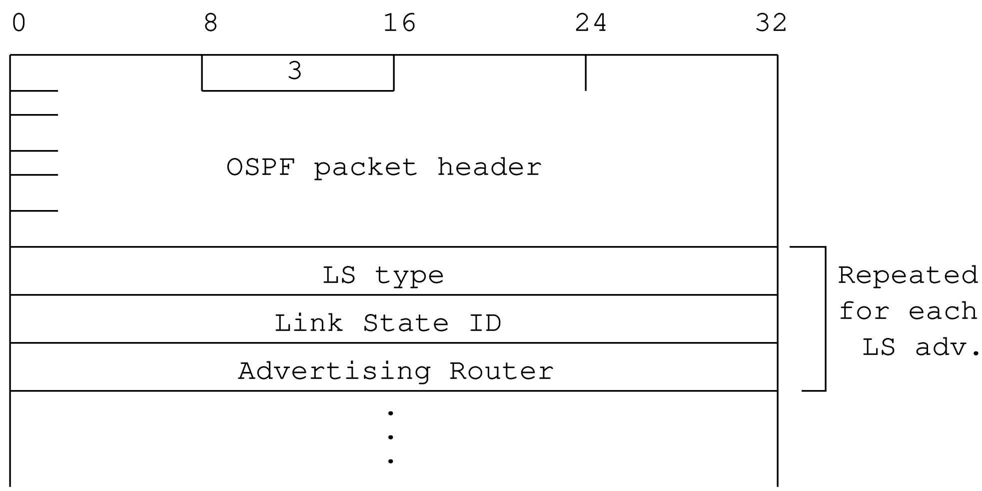
DD sequence number Used to sequence the collection of database description packets. The initial value (indicated by the init bit being set) should be unique. The sequence number then increments until the complete database description has been sent.

The rest of the packet consists of a (possibly partial) list of the topological database's pieces. Each piece of the database is described by the following fields: LS type, Link State ID, Advertising Router, LS sequence number, LS checksum and LS age. Taken together, these fields uniquely identify an advertisement and its instantiation. These fields are all contained in the advertisement's link state header. For further information, see Section A.3.

A.6 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 instantiation of the database pieces it is requesting (defined by LS sequence number, LS checksum, and LS age). It may receive even more recent instantiations in response.

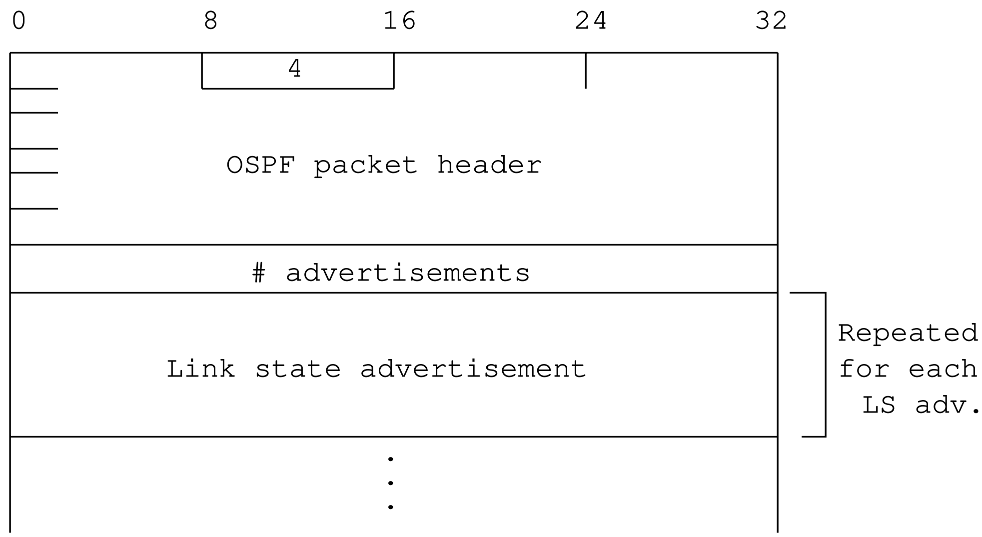


Each advertisement requested is specified by its LS type, Link State ID, and Advertising Router. This uniquely identifies the advertisement, but not its instantiation. Link State Request packets are understood to be requests for the most recent instantiation (whatever that might be).

A.7 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, Link State Update packets are acknowledged by Link State Acknowledgment packets. If retransmission is necessary, the retransmitted Link State Update is always done as a unicast packet.



 # advertisements The number of link state advertisements included in this update.

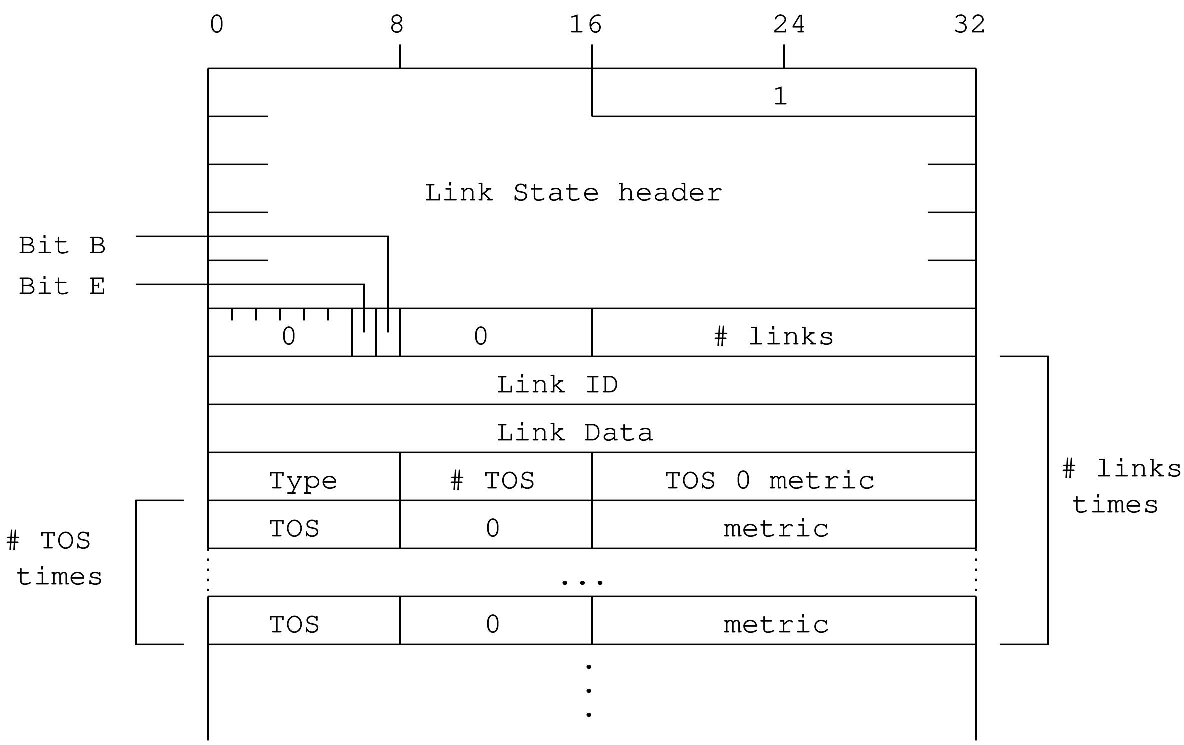
Each link state advertisement contains a type field. The format of each of the four types of link state advertisements is different. Their formats are described in the following sections. All advertisements begin with a common 20 byte header, the link state advertisement header. This is described in Section A.3.

A.7.1 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 collected states of the router's links to the area. (The router links to an area can be derived from the list of router interface data structures and configured host routes, see Section 12.3.1). The advertisement's Link State ID field specifies the router's OSPF Router ID. Such an advertisement is flooded throughout the single area only.

The router links connect to transit networks, stub networks (including attached hosts), or to other routers. Each link also has an associated 32-bit data field; for links to stub networks this specifies the network mask and for the other link types this specifies the appropriate IP interface address. Host routes are classified as links to stub networks whose network mask is `0xffffffff`.

All of the router's links to the area must be described in a single router links advertisement. Multiple advertisements may be included in a single Link State Update packet. For each link, there may be separate metrics for each Type of Service (TOS). The metric for TOS 0 must always be included, and listed first.



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 to the area.

The following fields are used to describe each router link. Each router link is typed (see the below Type field), indicating the kind of object to which the router connects. 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.

Link ID Identifies the object that this router link connects to. This depends on the link's Type field. 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 other 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. Note also that the network number of an attached transit network can be obtained by masking the Link ID with the appropriate network mask.

Type	*Link ID*
1	Neighboring router's ID
2	IP address of Designated Router
3	IP network/subnet number

Link Data Contents again depend on the link's Type field. For connections to stub network, it specifies the network mask. 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.

Type What the router link connects to. One of:

1 – Connects to another router
2 – Connects to a transit network
3 – Connects to a stub network

#metrics 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.

metric (TOS 0) The cost of using this router link for TOS 0.

Each metric is described as follows. There is potentially one metric for each type of service. The TOS 0 metric is used for all types of service unless others are explicitly specified.

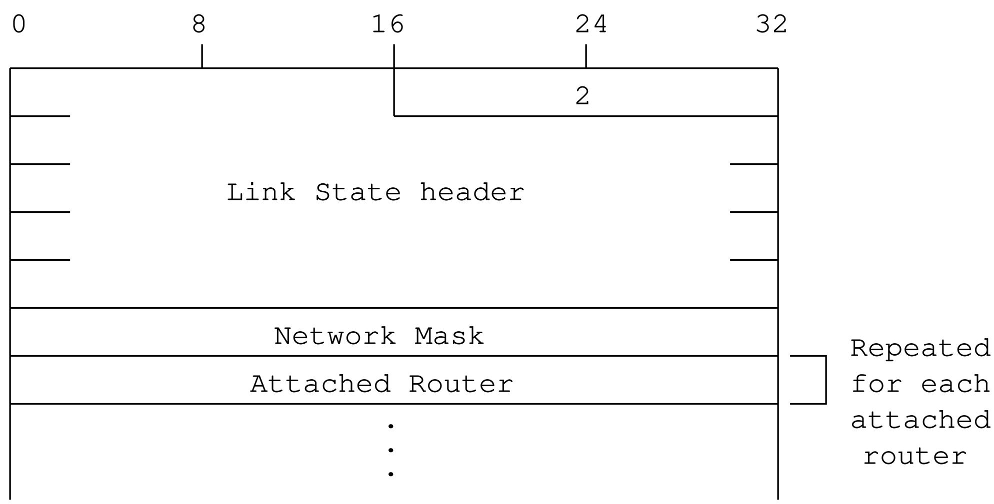
TOS IP type of service that this metric refers to. Represented exactly as it would appear in the IP header's TOS field. This implies that the only valid values are 0–7.

metric The cost of using this outbound router link, for traffic of the specified TOS.

A.7.2 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 router links advertisement.



Network Mask The IP network 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 length field.

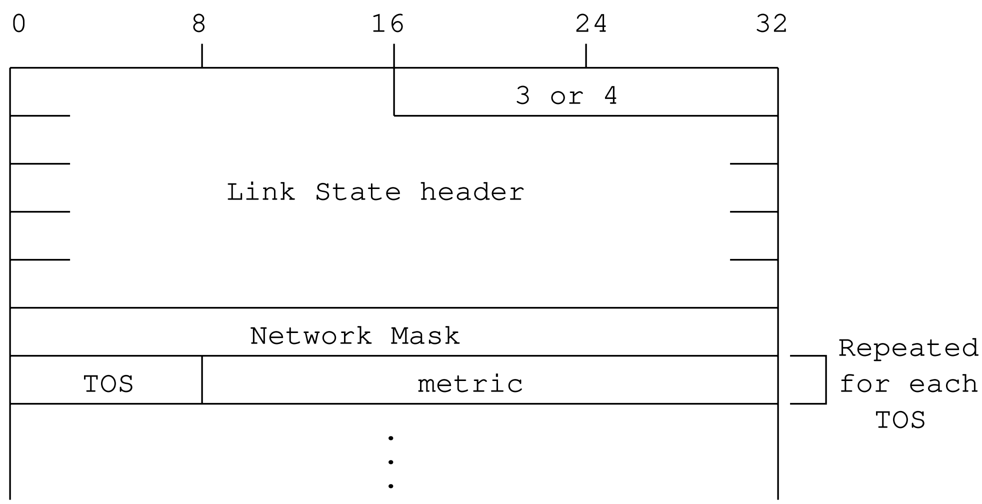
A.7.3 Summary links advertisements

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. 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.

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.

Separate costs may be advertised for each IP Type of Service. The cost for TOS 0 must be included, and is always listed first. If a cost for a certain type of service is not included, its cost defaults to that specified for TOS 0.



Network Mask For Type 3 link state advertisements, this indicates the destination's IP network 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 type of service, the following fields are defined. The number of TOS routes included can be calculated from the link state advertisement's length field. Values for TOS 0 must be specified; they are listed first.

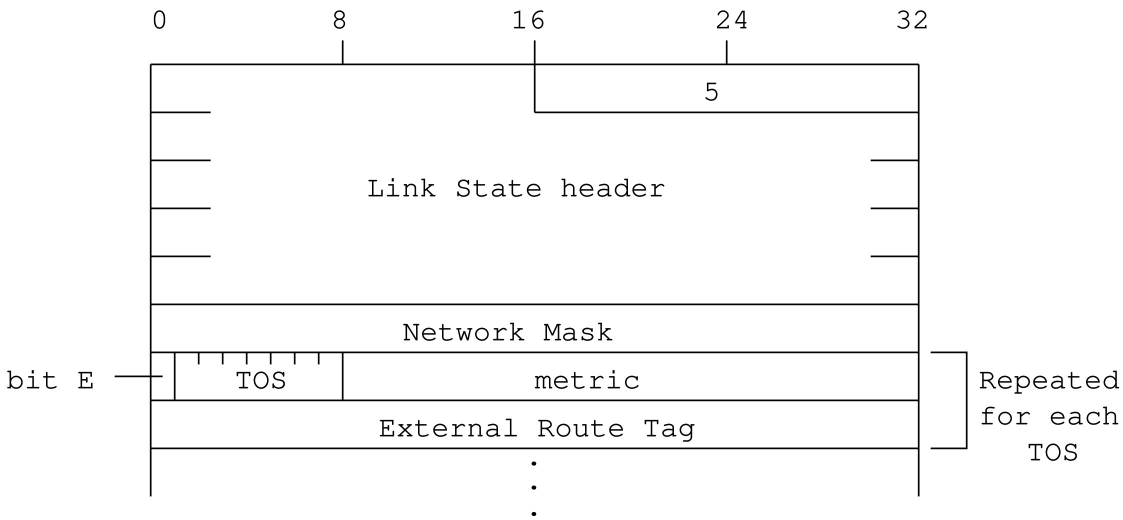
TOS The Type of Service that the following cost concerns.

metric The cost of this route. Expressed in the same units as the interface costs in the router links advertisements.

A.7.4 AS external links 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. The destination is always an IP network; the advertisement's Link State ID field specifies an IP network number.

If a route for a certain type of service is not included, that TOS is assumed to have the same cost as TOS 0. The cost for TOS 0 must be included, and is always listed first.



Network Mask The IP network mask for the advertised destination. For example, when advertising a class A network the mask `0xff000000` would be used.

For each type of service, the following fields are defined. The number of TOS routes included can be calculated from the link state advertisement's length field. Values for TOS 0 must be specified; they are listed first.

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.

TOS The Type of Service that the following cost concerns.

metric The cost of this route. Interpretation depends on the external type indication (bit E above).

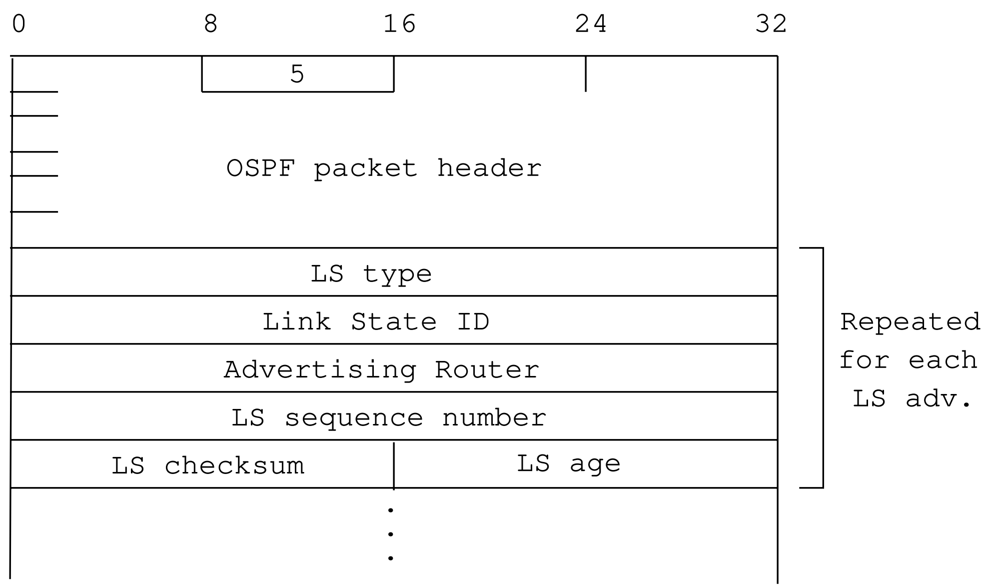
External Route Tag A 32-bit field attached to each external route. This is not used by the OSPF protocol itself. It may be used to communicate information between AS boundary routers; the precise nature of such information is outside the scope of this specification.

A.8 The Link State Acknowledgment packet

Link State Acknowledgment Packets are OSPF packet type 5. To make the flooding of link state advertisements reliable, the 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 packet. Unacknowledged advertisements will be retransmitted.

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. See Section 13.5 for more details.

The format of this packet is similar to that of the Data Description packet. The body of the packet is simply a list of link state advertisement descriptions.



The database pieces are described precisely as in the Database Description Packets. The LS type, Link State ID and Advertising Router fields identify a specific advertisement, while the LS sequence number, LS checksum and LS age fields isolate a particular instantiation of the advertisement. See Section A.3 for more details.

B Architectural Constants

Several OSPF protocol parameters have fixed architectural values. These parameters have been referred to in the text by names such as LSRefreshTimer. The same naming convention is used for the configurable protocol parameters. They are defined in another appendix.

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. For each link state advertisement that a router originates, an interval timer should be set to this value. Firing of this timer causes a new instantiation of the link state advertisement to be originated. 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 age reaches MaxAge, it is reflooded. It is then removed from the database as soon as this flood is acknowledged, i.e., as soon as it has been removed from all neighbor **Link state retransmission lists**. Advertisements having 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 link state metric value indicating that the destination is unreachable. It is defined to be the binary value of all ones. It depends on the size of the metric field, which is 16 bits in router links advertisements, and 24 bits in both summary and AS external links advertisements.

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. 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

The only global configurable parameter is the router's Router ID. This 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.

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 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 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 [IP address, mask] pairs. Each pair describes a range of IP addresses. 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. Routing information is condensed at area boundaries. External to the area, a single route is advertised for each address range.

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 internet mask. A single route would be advertised external to the area, describing the entire subnetted network.

Authentication type Each area can be configured for a separate type of authentication. See Appendix E for a discussion of the defined authentication types.

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 This denotes the portion of the IP interface address that identifies the attached network. This is often referred to as the subnet mask.

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 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 for 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 hello interval, the faster topological changes will be detected, but more routing traffic will ensue. Sample value for a X.25 PDN network: 30 seconds. Sample value for a local area network: 10 seconds.

RouterDeadInterval The number of seconds that a router's Hellos have not been seen before its neighbors declare the router down. This is also advertised in the router's Hello Packets in the DeadInt 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. For example, if the authentication type 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 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 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).

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 many 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 DR 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, Hello packets will be sent only to those routers eligible to become Designated Router, until the identity of the Designated Router is discovered.

PollInterval If a neighboring router has become inactive (hellos have not been seen for RouterDeadInterval seconds), it may still be necessary to send Hellos to the dead neighbor. These Hellos 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 network 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 directly connected host, 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. Note that this doesn’t really matter unless the host is multiply homed.

D Required Statistics

An OSPF implementation must provide a minimum set of statistics indicating the operational state of the protocol. These statistics must be accessible to the user; this will probably be accomplished through some sort of network management interface.

It is hoped that these statistics will aid in the debugging of the implementation, and in the analysis of the protocol's performance.

The statistics can be broken into two broad categories. The first consists of what we will call logging messages. These are messages produced in real time, with generally a single message produced as the result of a single protocol event. Such messages are also commonly referred to as traps.

The second category will be referred to as cumulative statistics. These are counters whose value have collected over time, such as the count of link state retransmissions over the last hour. Also falling into this category are dumps of the various routing data structures.

D.1 Logging messages

A logging message should be produced on every significant protocol event. The major events are listed below. Most of these events indicate a topological change in the routing domain. However, some number of logging messages can be expected even when the routing domain remains intact for long periods of time. For example, link state originations will still happen due to the link state refresh timer firing.

Any of the messages that refer to link state advertisements should print the area associated with the advertisement. There is no area associated with AS external link advertisements.

The following list of logging messages indicate topological changes in the routing domain:

- T1 The state of a router interface changes. Interface state changes are documented in Section 9.3. In general, they will cause new link state advertisements to be originated. The logging message produced should include the interface's IP address (or other name), interface type (virtual link, etc.) and old and new state values (as documented in Section 9.1).

- T2 The state of a neighbor changes. Neighbor state changes are documented in Section 10.3. The logging message produced should include the neighbor IP address, and old and new state values.

- T3 The (Backup) Designated Router has changed on one of the attached networks. See Section 9.4. The logging message produced should include the network IP address, and the old and new (Backup) Designated Routers.

- T4 The router is originating a new instantiation of a link state advertisement. The logging message produced should indicate the LS type, Link State ID and Advertising Router associated with the advertisement (see Section 12.3).

- T5 The router has received a new instantiation of a link state advertisement. The router receives these in Link State Update packets. This will cause recalculation of the routing table. The logging message produced should indicate the advertisement's LS type, Link State ID and Advertising Router. The message should also include the neighbor from whom the advertisement was received.

- T6 An entry in the routing table has changed (see Section 11). The logging message produced should indicate the Destination type, Destination ID, and the old and new paths to the destination.

The following logging messages may indicate that there is a network configuration error:

C1 A received OSPF packet is rejected due to errors in its IP/OSPF header. The reasons for rejection are documented in Section 8.2. They include OSPF checksum failure, authentication failure, and inability to match the source with an active OSPF neighbor. The logging message produced should include the IP source and destination addresses, the router ID in the OSPF header, and the reason for the rejection.

C2 An incoming Hello packet is rejected due to mismatches between the Hello's parameters and those configured for the receiving interface (see Section 10.5). This indicates a configuration problem on the attached network. The logging message should include the Hello's source, the receiving interface, and the non-matching parameters.

C3 An incoming Database Description packet, Link State Request Packet, Link State Acknowledgment Packet or Link State Update packet is rejected due to the source neighbor being in the wrong state (see Sections 10.6, 10.7, 13.7 , and 13 respectively). This can be normal when the identity of the network's Designated Router changes, causing momentary disagreements over the validity of adjacencies. The logging message should include the source neighbor, its state, and the packet's type.

C4 A Database Description packet has been retransmitted. This may mean that the value of RxmtInterval that has been configured for the associated interface is too small. The logging message should include the neighbor to whom the packet is being sent.

The following messages can be caused by packet transmission errors, or software errors in an OSPF implementation:

E1 The checksum in a received link state advertisement is incorrect. The advertisement is discarded (see Section 13). The logging message should include the advertisement's LS type, Link State ID and Advertising Router (which may be incorrect). The message should also include the neighbor from whom the advertisement was received.

E2 During the aging process, it is discovered that one of the link state advertisements in the database has an incorrect checksum. This indicates memory corruption or a software error in the router itself. The router should be dumped and restarted.

The following messages are an indication that a router has restarted, losing track of its previous LS sequence number. Should these messages continue, it may indicate the presence of duplicate Router IDs:

R1 Two link state advertisements have been seen, whose LS type, Link State ID, Advertising Router and LS sequence number are the same, yet with differing LS checksums. These are considered to be different instantiations of the same advertisement. The instantiation with the larger checksum is accepted as more recent (see Section 12.1.6, 13.1). The logging message should include the LS type, Link State ID, Advertising Router, LS sequence number and the two differing checksums.

R2 Two link state advertisements have been seen, whose LS type, Link State ID, Advertising Router, LS sequence number and LS checksum are the same, yet can be distinguished by their LS age fields. This means that one of the advertisement's LS age is MaxAge, or the two LS age fields differ by more than MaxAgeDiff. The logging message should include the LS type, Link State ID, Advertising Router, LS sequence number and the two differing ages.

R3 The router has received an instantiation of one of its self-originated advertisements, that is considered to be more recent. This forces the router to originate a new advertisement (see Section 13.4). The logging message should include the advertisement's LS type, Link State ID, and Advertising Router along with the neighbor from whom the advertisement was received.

R4 An acknowledgment has been received for an instantiation of an advertisement that is not currently contained in the router's database (see Section 13.7). The logging message should detail the instantiation being acknowledged and the database copy (if any), along with the neighbor from whom the acknowledgment was received.

R5 An advertisement has been received through the flooding procedure that is LESS recent the the router's current database copy (see Section 13). The logging message should include the received advertisement's LS type, Link State ID, Advertising Router, LS sequence number, LS age and LS checksum. Also, the message should display the neighbor from whom the advertisement was received.

The following messages are indication of normal, yet infrequent protocol events. These messages will help in the interpretation of some of the above messages:

N1 The Link state refresh timer has fired for one of the router's self-originated advertisements (see Section 12.3). A new instantiation of the advertisement must be originated. The message should include the advertisement's LS type, Link State ID and Advertising Router.

N2 One of the advertisements in the router's link state database has aged to MaxAge (see Section 14). At this point, the advertisement is no longer included in the routing table calculation, and is reflooded. The message should list the advertisement's LS type, Link State ID and Advertising Router.

N3 An advertisement of age MaxAge has been flushed from the router's database. This occurs after the advertisement has been acknowledged by all adjacent neighbors. The message should list the advertisement's LS type, Link State ID and Advertising Router.

D.2 Cumulative statistics

These statistics display collections of the routing data structures. They should be able to be obtained interactively, through some kind of network management facility.

All the following statistics displays, with the exception of the area list, routing table and the AS external links, are specific to a single area. As noted in Section 4, most OSPF protocol mechanisms work on each area separately.

The following statistics displays should be available:

1. A list of all the areas attached to the router, along with the authentication type to use for the area, the number of router interfaces attaching to the area, and the total number of nets and routers belonging to the area.

   For example, consider the router RT3 pictured in Figure 16. It has interfaces to two separate areas, Area 1 and the backbone (Area 0). The following display then indicates that the backbone is using a simple password for authentication, and that Area 1 is not using any authentication. The number of nets includes IP networks, subnets, and hosts (this is the reason for 2 backbone nets – they are the host routes corresponding to the serial line between backbone routers RT6 and RT10).

Area ID	*# ifcs*	*AuType*	*# nets*	*# routers*
0	1	1	2	7
1	2	0	4	4

2. A list of all the router's interfaces to an area, along with their addresses, output cost, current state, the (Backup) Designated Router for the attached network, and the number of neighbors currently associated with the interface. Some number of these neighbors will have become adjacent, the number of these is noted in the display also.

   Again consider router RT3 in Figure 16. The display below indicates that RT4 has been selected as Designated Router for network N3, and router RT1 has been selected as Backup. Adjacencies have been established to both of these routers. There are no routers besides RT3 attached to network N4, so it becomes DR, yet still advertises the network as a stub in its router links advertisements.

Interface IP address	*state*	*cost*	*DR*	*Backup*	*# nbrs*	*# adjs*
192.1.1.3	DR other	1	192.1.1.4	192.1.1.1	3	2
192.1.4.3	DR	2	192.1.4.3	none	0	0

3. The list of neighbors associated with a particular interface. Each neighbor's IP address, router ID, state, and the length of the three link state advertisement queues (see Section 10) to the neighbor is displayed.

   Suppose router RT4 is the Designated Router for network N3, and router RT1 is the Backup Designated router. Suppose also that the adjacency between router RT3 and RT1 has not yet fully formed. The display of router RT3's neighbors (associated with its interface to network N3) may then appear as follows. The display indicates that RT3 and RT1 are still in the database exchange procedure, Router RT3 has more Database Description packets to send to RT1, and RT1 has at least one link state advertisement that RT3 doesn't. Also, there is a single link state advertisement that has been flooded, but not acknowledged, to each neighbor that participates in the flooding procedure (state >= Exchng). (In the following examples we assume that a router's Router ID is assigned to be its smallest IP interface address).

Neighbor IP address	*Router ID*	*state*	*LS rxmt len*	*DB summ len*	*LS req len*
192.1.1.1	192.1.1.1	Exchng	1	10	1
192.1.1.2	192.1.1.2	2-Way	0	0	0
192.1.1.4	192.1.1.4	Full	1	0	0

4. A list of the area's link state database. This is the same in all of the routers attached to the area. It is composed of that area's router links, network links, and summary links advertisements. Also, the AS external link advertisements are a part of all the areas' databases.

   The link state database for Area 1 in Figure 16 might look as follows (compare this with Figure 7). Assume the the Designated Router for network N3 is router RT4, as above. Also, assume that router RT3 was formerly the Designated Router for network N3. Its network links advertisement is still part of the database (it has not aged to FlushAge).

   Both routers RT3 and RT4 are originating summary link advertisements into Area 1, since they are area border routers. Routers RT5 and RT7 are AS external routers. Their location must be described in summary links advertisements. Also, their AS external link advertisements are flooded throughout the entire AS.

   Router RT3 can locate its self-originated advertisements by looking for its own router ID (192.1.1.3) in advertisements' Advertising Router fields.

   The LS sequence number, LS age, and LS checksum fields indicate the advertisement's instantiation. Their values are stored in the advertisement's link state header; we have not bothered to make up values for the example.

LS type	*Link State ID*	*Advertising Router*	*LS seq no*	*LS age*	*LS checksum*
1	192.1.1.1	192.1.1.1	*	*	*
1	192.1.1.2	192.1.1.2	*	*	*
1	192.1.1.3	192.1.1.3	*	*	*
1	192.1.1.4	192.1.1.4	*	*	*
2	192.1.1.3	192.1.1.3	*	older	*
2	192.1.1.4	192.1.1.4	*	newer	*
3	Ia,Ib	192.1.1.3	*	*	*
3	N6	192.1.1.3	*	*	*
3	N7	192.1.1.3	*	*	*
3	N8	192.1.1.3	*	*	*
3	N9-N11,H1	192.1.1.3	*	*	*
3	Ia,Ib	192.1.1.4	*	*	*
3	N6	192.1.1.4	*	*	*
3	N7	192.1.1.4	*	*	*
3	N8	192.1.1.4	*	*	*
3	N9-N11,H1	192.1.1.4	*	*	*
4	RT5	192.1.1.3	*	*	*
4	NT7	192.1.1.3	*	*	*
4	RT5	192.1.1.4	*	*	*
4	NT7	192.1.1.4	*	*	*
4	N12	RT5's ID	*	*	*
4	N13	RT5's ID	*	*	*
4	N14	RT5's ID	*	*	*
4	N12	RT7's ID	*	*	*
4	N15	RT7's ID	*	*	*

5. The contents of any particular link state advertisement. For example, a listing of the router links advertisement for Area 1, with LS type = 1 and Link State ID = 192.1.1.3 is shown in Section 12.3.1.

6. A listing of the entire routing table. Such listings are shown in Section 11. The routing table is calculated from the combined databases of each attached area (see Section 16). It may be desirable to sort the routing table by Type of Service, or by destination, or a combination of the two.

E Authentication

All OSPF protocol exchanges are authenticated. The OSPF packet header (see Section A.2) includes an authentication type field, and 64-bits of data used 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 of 0-255 are specified below. Other authentication types may be assigned locally on a per Autonomous System basis.

E.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.

E.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 coming up in the area. They must first be configured with their attached networks' passwords before they can join the routing domain.

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 E.

Author's Address

John Moy
Proteon, Inc.
2 Technology Drive
Westborough, MA 01581-5008

Phone: 508-898-2800

EMail: jmoy@proteon.com