Extending OSPF to Support Demand Circuits

Status of this Memo

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

Abstract

   This memo defines enhancements to the OSPF protocol that allow
   efficient operation over "demand circuits". Demand circuits are
   network segments whose costs vary with usage; charges can be based
   both on connect time and on bytes/packets transmitted. Examples of
   demand circuits include ISDN circuits, X.25 SVCs, and dial-up lines.
   The periodic nature of OSPF routing traffic has until now required a
   demand circuit's underlying data-link connection to be constantly
   open, resulting in unwanted usage charges. With the modifications
   described herein, OSPF Hellos and the refresh of OSPF routing
   information are suppressed on demand circuits, allowing the
   underlying data-link connections to be closed when not carrying
   application traffic.

   Demand circuits and regular network segments (e.g., leased lines) are
   allowed to be combined in any manner. In other words, there are no
   topological restrictions on the demand circuit support. However,
   while any OSPF network segment can be defined as a demand circuit,
   only point-to-point networks receive the full benefit. When broadcast
   and NBMA networks are declared demand circuits, routing update
   traffic is reduced but the periodic sending of Hellos is not, which
   in effect still requires that the data-link connections remain
   constantly open.

   While mainly intended for use with cost-conscious network links such
   as ISDN, X.25 and dial-up, the modifications in this memo may also
   prove useful over bandwidth-limited network links such as slow-speed
   leased lines and packet radio.

   The enhancements defined in this memo are backward-compatible with
   the OSPF specification defined in [1], and with the OSPF extensions
   defined in [3] (OSPF NSSA areas), [4] (MOSPF) and [8] (OSPF Point-

         to-MultiPoint Interface).

         This memo provides functionality similar to that specified for RIP in
         [2], with the main difference being the way the two proposals handle
         oversubscription (see Sections 4.3 and 7 below). However, because
         OSPF employs link-state routing technology as opposed to RIP's
         Distance Vector base, the mechanisms used to achieve the demand
         circuit functionality are quite different.

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

Acknowledgments

         The author would like to acknowledge the helpful comments of Fred
         Baker, Rob Coltun, Dawn Li, Gerry Meyer, Tom Pusateri and Zhaohui
         Zhang. This memo is a product of the OSPF Working Group.

Table of Contents

1. Model for demand circuits

    In this memo, demand circuits refer to those network segments whose
    cost depends on either connect time and/or usage (expressed in terms
    of bytes or packets). Examples include ISDN circuits and X.25 SVCs.
    On these circuits, it is desirable for a routing protocol to send as
    little routing traffic as possible. In fact, when there is no change
    in network topology it is desirable for a routing protocol to send no
    routing traffic at all; this allows the underlying data-link
    connection to be closed when not needed for application data traffic.

    The model used within this memo for the maintenance of demand
    circuits is as follows. If there is no data to send (either routing
    protocol traffic or application data), the data-link connection
    remains closed. As soon as there is data to be sent, an attempt to
    open the data-link connection is made (e.g., an ISDN or X.25 call is
    placed). When/if the data-link connection is established, the data is
    sent, and the connection stays open until some period of time elapses
    without more data to send. At this point the data-link connection is
    again closed, in order to conserve cost and resources (see Section 1
    of [2]).

    The "Presumption of Reachability" described in [2] is also used.
    Even though a circuit's data-link connection may be closed at any
    particular time, it is assumed by the routing layer (i.e., OSPF) that
    the circuit is available unless other information, such as a
    discouraging diagnostic code resulting from an attempted data-link
    connection, is present.

    It may be possible that a data-link connection cannot be established
    due to resource shortages. For example, a router with a single basic
    rate ISDN interface cannot open more than two simultaneous ISDN
    data-link connections (one for each B channel), and limitations in
    interface firmware and/or switch capacity may limit the number of
    X.25 SVCs simultaneously supported. When a router cannot
    simultaneously open all of its circuits' underlying data-link
    connections due to resource limitations, we say that the router is
    oversubscribed. In these cases, datagrams to be forwarded out the
    (temporarily unopenable) data-link connections are discarded, instead
    of being queued. Note also that this temporary inability to open
    data-link connections due to oversubscription is NOT reported by the
    OSPF routing system as unreachability; see Section 4.3 for more
    information.

Either end of a demand circuit may attempt to open the data-link
connection. When both ends attempt to open the connection
simultaneously, there is the possibility of call collision. Not all
data-links specify how call collisions are handled. Also, while OSPF
requires that all periodic timers be randomized to avoid
synchronization (see Section 4.4 of [1]), if call attempts are
strictly data-driven there may still be insufficient spacing of call
attempts to avoid collisions on some data-links. For these reasons,
for those data-links without collision detection/avoidance support,
it is suggested (but not specified herein) that an exponential
backoff scheme for call retries be employed at the data-link layer.
Besides helping with call collisions, such a scheme could minimize
charges (if they exist) for failed call attempts.

As a result of the physical implementation of some demand circuits,
only one end of the circuit may be capable of opening the data-link
connection. For example, some async modems can initiate calls, but
cannot accept incoming calls. In these cases, since connection
initiation in this memo is data-driven, care must be taken to ensure
that the initiating application party is located at the calling end
of the demand circuit.

2. Modifications to all OSPF routers

   While most of the modifications to support demand circuits are
   isolated to the demand circuit endpoints (see Section 3), there are
   changes required of all OSPF routers. These changes are described in
   the subsections below.

   2.1. The OSPF Options field

      A new bit is added to the OSPF Options field to support the demand
      circuit extensions. This bit is called the "DC-bit". The resulting
      format of the Options field is described in Appendix A.

      A router implementing the functionality described in Section 2 of
      this memo sets the DC-bit in the Options field of all LSAs that it
      originates. This is regardless of the LSAs' LS type, and also
      regardless of whether the router implements the more substantial
      modifications required of demand circuit endpoints (see Section
      3). Setting the DC-bit in self-originated LSAs tells the rest of
      the routing domain that the router can correctly process DoNotAge
      LSAs (see Sections 2.2, 2.3 and 2.5).

      There is a single exception to the above rule. A router
      implementing Section 2 of this memo may sometimes originate an
      "indication-LSA"; these LSAs always have the DC-bit clear.
      Indication-LSAs are used to convey across area boundaries the

existence of routers incapable of DoNotAge processing; see Section
2.5.1 for details.

2.2. The LS age field

The semantics of the LSA's LS age field are changed, allowing the
high bit of the LS age field to be set. This bit is called
"DoNotAge"; see Appendix C for its formal definition. LSAs whose
LS age field have the DoNotAge bit set are not aged while they are
held in the link state database, which means that they do not have
to be refreshed every LSRefreshInterval as is done with all other
OSPF LSAs.

By convention, in the rest of this memo we will express LS age
fields having the DoNotAge bit set as "DoNotAge+x", while an LS
age expressed as just "x" is assumed to not have the DoNotAge bit
set. LSAs having DoNotAge set are also sometimes referred to as
"DoNotAge LSAs".

When comparing two LSA instances to see which one is most recent,
the two LSAs' LS age fields are compared whenever the LS sequence
numbers and LS checksums are found identical (see Section 13.1 of
[1]). Before comparing, the LS age fields must have their DoNotAge
bits masked off. For example, in determining which LSA is more
recent, LS ages of 1 and DoNotAge+1 are considered equivalent; an
LSA flooded with LS age of 1 may be acknowledged with a Link State
Acknowledgement listing an LS age of DoNotAge+1, or vice versa. In
particular, DoNotAge+MaxAge is equivalent to MaxAge; however for
backward-compatibility the MaxAge form should always be used when
flushing LSAs from the routing domain (see Section 14.1 of [1]).

Thus, the set of allowable values for the LS age field fall into
the two ranges: 0 through MaxAge and DoNotAge through
DoNotAge+MaxAge. (Previously the LS age field could not exceed
the value of MaxAge.) Any LS age field not falling into these two
ranges should be considered to be equal to MaxAge.

When an LSA is flooded out an interface, the constant
InfTransDelay is added to the LSA's LS age field. This happens
even if the DoNotAge bit is set; in this case the LS age field is
not allowed to exceed DoNotAge+MaxAge. If the LS age field reaches
DoNotAge+MaxAge during flooding, the LSA is flushed from the
routing domain. This preserves the protection in [1] afforded
against flooding loops.

The LS age field is not checksum protected. Errors in a router's
memory may mistakenly set an LSA's DoNotAge bit, stopping the
aging of the LSA. However, a router should note that its own

self-originated LSAs should never have the DoNotAge bit set in its
own database. This means that in any case the router's self-
originated LSAs will be refreshed every LSRefreshInterval. As
this refresh is flooded throughout the OSPF routing domain, it
will replace any LSA copies in other routers' databases whose
DoNotAge bits were mistakenly set.

2.3. Removing stale DoNotAge LSAs

Because LSAs with the DoNotAge bit set are never aged, they can
stay in the link state database even when the originator of the
LSA no longer exists. To ensure that these LSAs are eventually
flushed from the routing domain, and that the size of the link
state database doesn't grow without bound, routers are required to
flush a DoNotAge LSA if BOTH of the following conditions are met:

(1) The LSA has been in the router's database for at least
    MaxAge seconds.

(2) The originator of the LSA has been unreachable (according to
    the routing calculations specified by Section 16 of [1]) for
    at least MaxAge seconds.

For an example, see Time T8 in the example of Section 4.1. Note
that the above functionality is an exception to the general OSPF
rule that a router can only flush (i.e., prematurely age; see
Section 14.1 of [1]) its own self-originated LSAs. The above
functionality pertains only to DoNotAge LSAs. An LSA having
DoNotAge clear still can be prematurely aged only by its
originator; otherwise, the LSA must age naturally to MaxAge before
being removed from the routing domain.

An interval as long as MaxAge has been chosen to avoid any
possibility of thrashing (i.e., flushing an LSA only to have it
reoriginated soon afterwards). Note that by the above rules, a
DoNotAge LSA will be removed from the routing domain no faster
than if it were being aged naturally (i.e., if DoNotAge were not
set).

2.4. A change to the flooding algorithm

The following change is made to the OSPF flooding algorithm. When
a Link State Update Packet is received that contains an LSA
instance which is actually less recent than the the router's
current database copy, the router must now process the LSA as
follows (modifying Step 8 of Section 13 in [1] accordingly):

    o   If the database copy has LS age equal to MaxAge and LS
         sequence number equal to MaxSequenceNumber, simply discard
         the received LSA without acknowledging it. (In this case,
         the LSA's sequence number is wrapping, and the
         MaxSequenceNumber LSA must be completely flushed before any
         new LSAs can be introduced). This is identical to the
         behavior specified by Step 8 of Section 13 in [1].

    o   Otherwise, send the database copy back to the sending
         neighbor, encapsulated within a Link State Update Packet. In
         so doing, do not put the database copy of the LSA on the
         neighbor's link state retransmission list, and do not
         acknowledge the received (less recent) LSA instance.

This change is necessary to support flooding over demand circuits.
For example, see Time T4 in the example of Section 4.2.

However, this change is beneficial when flooding over non-demand
interfaces as well. For this reason, the flooding change pertains
to all interfaces, not just interfaces to demand circuits. The
main example involves MaxAge LSAs. There are times when MaxAge
LSAs stay in a router's database for extended intervals: 1) when
they are stuck in a retransmission queue on a slow link or 2) when
a router is not properly flushing them from its database, due to
software bugs. The prolonged existence of these MaxAge LSAs can
inhibit the flooding of new instances of the LSA. New instances
typically start with the initial LS sequence number, and are
treated as less recent (and hence discarded) by routers still
holding MaxAge instances. However, with the above change to
flooding, a router with a MaxAge instance will respond back with
the MaxAge instance. This will get back to the LSA's originator,
which will then pick the next highest LS sequence number and
reflood, overwriting the MaxAge instance.

This change will be included in future revisions of the base OSPF
specification [1].

2.5. Interoperability with unmodified OSPF routers

Unmodified OSPF routers will probably treat DoNotAge LSAs as if
they had LS age of MaxAge. At the very worst, this will cause
continual retransmissions of the DoNotAge LSAs. (An example
scenario follows. Suppose Routers A and B are connected by a
point-to-point link. Router A implements the demand circuit
extensions, Router B does not. Neither one treats their connecting
link as a demand circuit. At some point in time, Router A receives
from another neighbor via flooding a DoNotAge LSA. The DoNotAge
LSA is then flooded by Router A to Router B. Router B, not

understanding DoNotAge LSAs, treats it as a MaxAge LSA and
acknowledges it as such to Router A. Router A receives the
acknowledgment, but notices that the acknowledgment is for a
different instance, and so starts retransmitting the LSA.)

However, to avoid this confusion, DoNotAge LSAs will be allowed in
an OSPF area if and only if, in the area's link state database,
all LSAs have the DC-bit set in their Options field (see Section
2.1). Note that it is not required that the LSAs' Advertising
Router be reachable; if any LSA is found not having its DC-bit set
(regardless of reachability), then the router should flush (i.e.,
prematurely age; see Section 14.1 of [1]) from the area all
DoNotAge LSAs. These LSAs will then be reoriginated at their
sources, this time with DoNotAge clear. Like the change in
Section 2.3, this change is an exception to the general OSPF rule
that a router can only flush its own self-originated LSAs. Both
changes pertain only to DoNotAge LSAs, and in both cases a flushed
LSA's LS age field should be set to MaxAge and not
DoNotAge+MaxAge.

2.5.1. Indicating across area boundaries

AS-external-LSAs are flooded throughout the entire OSPF routing
domain, excepting only OSPF stub areas and NSSAs. For that
reason, if an OSPF router that is incapable of DoNotAge
processing exists in any "regular" area (i.e., an area that is
not a stub nor an NSSA), no AS-external-LSA can have DoNotAge
set. This memo simplifies that requirement by broadening it to
the following rule: LSAs in regular OSPF areas are allowed to
have DoNotAge set if and only if every router in the OSPF
domain (excepting those in stub areas and NSSAs) is capable of
DoNotAge processing. The rest of this section describes how the
rule is implemented.

As described above in Sections 2.1 and 2.5, a router indicates
that it is capable of DoNotAge processing by setting the DC-bit
in the LSAs that it originates. However, there is a problem. It
is possible that, in all areas to which Router X directly
attaches, all the routers are capable of DoNotAge processing,
yet there is some router in a remote "regular" area that cannot
process DoNotAge LSAs. This information must then be conveyed
to Router X, so that it does not mistakenly flood/create
DoNotAge LSAs.

The solution is as follows. Area border routers transmit the
existence of DoNotAge-incapable routers across area boundaries,
using "indication-LSAs". Indication-LSAs are type-4-summary
LSAs (also called ASBR-summary-LSAs), listing the area border

router itself as the described ASBR, with the LSA's cost set to
LSInfinity and the DC-bit clear. Note that indication-LSAs
convey no additional information; in particular, they are used
even if the area border router is not really an AS boundary
router (ASBR).

Taking indication-LSAs into account, the rule as to whether
DoNotAge LSAs are allowed in any particular area is EXACTLY the
same as given previously in Section 2.5, namely: DoNotAge LSAs
will be allowed in an OSPF area if and only if, in the area's
link state database, all LSAs have the DC-bit set in their
Options field.

Through origination of indication-LSAs, the existence of
DoNotAge-incapable routers can be viewed as going from non-
backbone regular areas, to the backbone area and from there to
all other regular areas. The following two cases summarize the
requirements for an area border router to originate
indication-LSAs:

   (1) Suppose an area border router (Router X) is connected to
       a regular non-backbone OSPF area (Area A). Furthermore,
       assume that Area A has LSAs with the DC-bit clear, other
       than indication-LSAs. Then Router X should originate
       indication-LSAs into all other directly-connected
       "regular" areas, including the backbone area, keeping
       the guidelines of Section 2.5.1.1 in mind.

   (2) Suppose an area border router (Router X) is connected to
       the backbone OSPF area (Area 0.0.0.0). Furthermore,
       assume that the backbone has LSAs with the DC-bit clear
       that are either a) not indication-LSAs or b)
       indication-LSAs that have been originated by routers
       other than Router X itself. Then Router X should
       originate indication-LSAs into all other directly-
       connected "regular" non-backbone areas, keeping the
       guidelines of Section 2.5.1.1 in mind.

2.5.1.1. Limiting indication-LSA origination

   To limit the number of indication-LSAs originated, the
   following guidelines should be observed by an area border
   router (Router X) when originating indication-LSAs. First,
   indication-LSAs are not originated into an Area A when A
   already has LSAs with DC-bit clear other than indication-
   LSAs. Second, if another area border router has originated a
   indication-LSA into Area A, and that area border router has
   a higher OSPF Router ID than Router X (same tie-breaker as

for forwarding address origination; see Section 12.4.5 of
[1]), then Router X should not originate an indication-LSA
into Area A.

As an example, suppose that three regular OSPF areas (Areas
A, B and C) are connected by routers X, Y and Z
(respectively) to the backbone area. Furthermore, suppose
that all routers are capable of DoNotAge processing, except
for routers in Areas A and B. Finally, suppose that Router
Z has a higher Router ID than Y, which in turn has a higher
Router ID than X. In this case, two indication-LSAs will be
generated (if the rules of Section 2.5.1 and the guidelines
of the preceding paragraph are followed): Router Y will
originate an indication-LSA into the backbone, and Router Z
will originate an indication-LSA into Area C.

3. Modifications to demand circuit endpoints

The following subsections detail the modifications required of the
routers at the endpoints of demand circuits. These consist of
modifications to two main pieces of OSPF: 1) sending and receiving
Hello Packets over demand circuits and 2) flooding LSAs over demand
circuits.

An additional OSPF interface configuration parameter, ospfIfDemand,
is defined to indicate whether an OSPF interface connects to a demand
circuit (see Appendix B). Two routers connecting to a common network
segment need not agree on that segment's demand circuit status.
However, to get full benefit of the demand circuit extensions, the
two ends of a point-to-point link must both agree to treat the link
as a demand circuit (see Section 3.2).

3.1. Interface State machine modifications

An OSPF point-to-point interface connecting to a demand circuit is
considered to be in state "Point-to-point" if and only if its
associated neighbor is in state "1-Way" or greater; otherwise the
interface is considered to be in state "Down". Hellos are sent out
such an interface when it is in "Down" state, at the reduced
interval of PollInterval. If the negotiation in Section 3.2.1
succeeds, Hellos will cease to be sent out the interface whenever
the associated neighbor reaches state "Full".

Note that as a result, an "LLDown" event for the point-to-point
demand circuit's neighbor forces both the neighbor and the
interface into state "Down" (see Section 3.2.2).

For OSPF broadcast and NBMA networks that have been configured as
demand circuits, there are no changes to the Interface State
Machine.

3.2. Sending and Receiving OSPF Hellos

The following sections describe the required modifications to OSPF
Hello Packet processing on point-to-point demand circuits.

For OSPF broadcast and NBMA networks that have been configured as
demand circuits, there is no change to the sending and receiving
of Hellos, nor are there any changes to the Neighbor State
Machine. This is because the proper operation of the Designated
Router election algorithm requires periodic exchange of Hello
Packets.

3.2.1. Negotiating Hello suppression

On point-to-point demand circuits, both endpoints must agree to
suppress the sending of Hello Packets. To ensure this
agreement, a router sets the DC-bit in OSPF Hellos and Database
Description Packets sent out the demand interface. Receiving
an Hello or a Database Description Packet with the DC-bit set
indicates agreement. Receiving an Hello with the DC-bit clear
and also listing the router's Router ID in the body of the
Hello message, or a Database Description Packet with the DC-bit
clear (either one indicating bidirectional connectivity)
indicates that the other end refuses to suppress Hellos. In
these latter cases, the router reverts to the normal periodic
sending of Hello Packets out the interface (see Section 9.5 of
[1]).

A demand point-to-point circuit need be configured in only one
of the two endpoints (see Section 4.1). If a router
implementing Sections 2 and 3 of this memo receives an Hello
Packet with the DC-bit set, it should treat the point-to-point
link as a demand circuit, making the appropriate changes to its
Hello Processing (see Section 3.2.2) and flooding (see Section
3.3).

Even if the above negotiation fails, the router should continue
setting the DC-bit in its Hellos and Database Descriptions (the
neighbor will just ignore the bit). The router will then
automatically attempt to renegotiate Hello suppression whenever
the link goes down and comes back up. For example, if the
neighboring router is rebooted with software that is capable of
operating over demand circuits (i.e., implements Sections 2 and
3 of this memo), a future negotiation will succeed.

Also, even if the negotiation to suppress Hellos fails, the
flooding modifications described in Section 3.3 are still
performed over the link.

3.2.2. Neighbor state machine modifications

When the above negotiation succeeds, Hello Packets are sent
over point-to-point demand circuits only until initial link-
state database synchronization is achieved with the neighbor
(i.e., the state of the neighbor connection reaches "Full", as
defined in Section 10.1 of [1]). After this, Hellos are
suppressed and the data-link connection to the neighbor is
assumed available until evidence is received to the contrary.
When the router finds that the neighbor is no longer available,
presumably from something like a discouraging diagnostic code
contained in a response to a failed call request, the neighbor
connection transitions back to "Down" and Hellos are sent
periodically (at Intervals of PollInterval) in an attempt to
restart synchronization with the neighbor.

This requires changes to the OSPF Neighbor State Machine (see
Section 10.3 of [1]). The receipt of Hellos from demand circuit
neighbors in state "Loading" or "Full" can no longer be
required. In other words, the InactivityTimer event defined in
Section 10.2 of [1] has no effect on demand circuit neighbors
in state "Loading" or "Full". An additional clarification is
needed in the Neighbor State Machine's LLDown event. For demand
circuits, this event should be mapped into the "discouraging
diagnostic code" discussed previously in Section 1, and should
not be generated when the data-link connection has been closed
simply to save resources. Nor should LLDown be generated if a
data-link connection fails due to temporary lack of resources.

3.3. Flooding over demand circuits

Flooding over demand circuits (point-to-point or otherwise) is
modified if and only if all routers have indicated that they can
process LSAs having DoNotAge set. This is determined by examining
the link state database of the OSPF area containing the demand
circuit. All LSAs in the database must have the DC-bit set. If
one or more LSAs have the DC-bit clear, flooding over demand
circuits is unchanged from [1]. Otherwise, flooding is changed as
follows.

(1) Only truly changed LSAs are flooded over demand circuits.
    When a router receives a new LSA instance, it checks first
    to see whether the contents have changed. If not, the new
    LSA is simply a periodic refresh and it is not flooded out

attached demand circuits (it is still flooded out other
interfaces however). This check should be performed in Step
5b of Section 13 in [1]. When comparing an LSA to its
previous instance, the following are all considered to be
changes in contents:

o   The LSA's Options field has changed.

o   One or both of the LSA instances has LS age set to
     MaxAge (or DoNotAge+MaxAge).

o   The length field in the LSA header has changed.

o   The contents of the LSA, excluding the 20-byte link
     state header, have changed. Note that this excludes
     changes in LS Sequence Number and LS Checksum.

(2) When it has been decided to flood an LSA over a demand
    circuit, DoNotAge should be set in the copy of the LSA that
    is flooded out the demand interface. (There is one
    exception: DoNotAge should not be set if the LSA's LS age is
    equal to MaxAge.) Setting DoNotAge will cause the routers on
    the other side of the demand circuit to hold the LSA in
    their databases indefinitely, removing the need for periodic
    refresh. Note that it is perfectly possible that DoNotAge
    will already be set. This simply indicates that the LSA has
    already been flooded over demand circuits. In any case, the
    flooded copy's LS age field must also be incremented by
    InfTransDelay (see Step 5 of Section 13.3 in [1], and
    Section 2.2 of this memo), as protection against flooding
    loops.

    The previous paragraph also pertains to LSAs flooded over
    demand circuits in response to Link State Requests. It also
    pertains to LSAs that are retransmitted over demand
    circuits.

3.4. Virtual link support

OSPF virtual links are essentially unnumbered point-to-point links
(see Section 15 of [1]). Accordingly, demand circuit support for
virtual links resembles that described for point-to-point links in
the previous sections. The main difference is that a router
implementing Sections 2 and 3 of this memo, and supporting virtual
links, always treats virtual links as if they were demand
circuits. Otherwise, when a virtual link's underlying physical
path contains one or more demand circuits, periodic OSPF protocol
exchanges over the virtual link would unnecessarily keep the

underlying demand circuits open.

Demand circuit support on virtual links can be summarized as
follows:

   o   Instead of modifying the Interface state machine for virtual
        links as was done for point-to-point links in Section 3.1,
        the Interface state machine for virtual links remains
        unchanged. A virtual link is considered to be in state
        "Point-to-point" if an intra-area path (through the virtual
        link's transit area) exists to the other endpoint. Otherwise
        it is considered to be in state "Down". See Section 15 of
        [1] for more details.

   o   Virtual links are always treated as demand circuits. In
        particular, over virtual links a router always negotiates to
        suppress the sending of Hellos. See Sections 3.2.1 and 3.2.2
        for details.

   o   In the demand circuit support over virtual links, there is
        no "discouraging diagnostic code" as described in Section 1.
        Instead, the connection is considered to exist if and only
        if an intra-area path (through the virtual link's transit
        area) exists to the other endpoint. See Section 15 of [1]
        for more details.

   o   Since virtual links are always treated as demand circuits,
        flooding over virtual links always proceeds as in Section
        3.3.

3.5. Point-to-MultiPoint Interface support

   The OSPF Point-to-MultiPoint interface has recently been developed
   for use with non-mesh-connected network segments. A common example
   is a Frame Relay subnet where PVCs are provisioned between some
   pairs of routers, but not all pairs. In this case the Point-to-
   Multipoint interface represents the single physical interface to
   the Frame relay network, over which multiple point-to-point OSPF
   conversations (one on each PVC) are taking place. For more
   information on the Point-to-MultiPoint interface, see [8].

   Since an OSPF Point-to-MultiPoint interface essentially consists
   of multiple point-to-point links, demand circuit support on the
   Point-to-Multipoint interface strongly resembles demand circuit
   support for point-to-point links. However, since the Point-to-
   MultiPoint interface requires commonality of its component point-
   to-point links' configurations, there are some differences.

Demand circuit support on Point-to-Multipoint interfaces can be
summarized as follows:

o   Instead of modifying the Interface state machine for Point-
     to-Multipoint interfaces as was done for point-to-point
     links in Section 3.1, the Interface state machine for
     Point-to-Multipoint interfaces remains unchanged.

o   When ospfIfDemand is set on a Point-to-MultiPoint interface,
     the router tries to negotiate Hello suppression separately
     on each of interface's component point-to-point links. This
     negotiation proceeds as in Section 3.2.1. Negotiation may
     fail on some component point-to-point links, and succeed on
     others. This is acceptable. On those component links where
     the negotiation fails, Hellos will always be sent;
     otherwise, Hellos will cease to be sent when the Database
     Description process completes on the component link (see
     Section 3.2.2).

o   Section 3.3 defines the demand circuit flooding behavior for
     all OSPF interface types. This includes Point-to-Multipoint
     interfaces.

4. Examples

   This section gives three examples of the operation over demand
   circuits. The first example is probably the most common and certainly
   the most basic. It shows a single point-to-point demand circuit
   connecting two routers. The second illustrates what happens when
   demand circuits and leased lines are used in parallel. The third
   explains what happens when a router has multiple demand circuits and
   cannot keep them all open (for resource reasons) at the same time.

   4.1. Example 1: Sole connectivity through demand circuits

      Figure 1 shows a sample internetwork with a single demand circuit
      providing connectivity to the LAN containing Host H2. Assume that
      all three routers (RTA, RTB and RTC) have implemented the
      functionality in Section 2 of this memo, and thus will be setting
      the DC-bit in their LSAs. Furthermore assume that Router RTB has
      been configured to treat the link to Router RTC as a demand
      circuit, but Router RTC has not been so configured. Finally assume
      that the LAN interface connecting Router RTA to Host H1 is
      initially down.

      The following sequence of events may then transpire, starting with
      Router RTB booting and bringing up its link to Router RTC:

Time T0: RTB negotiates Hello suppression

Router RTB will start sending Hellos over the demand circuit
with the DC-bit set in the Hello's Options field. Because
RTC is not configured to treat the link as a demand circuit,
the first Hello that RTB receives from RTC may not have the
DC-bit set. However, subsequent Hellos and Database
Description Packets received from RTC will have the DC-bit
set, indicating that the two routers have agreed that the
link will be treated as a demand circuit. The entire
negotiation is pictured in Figure 2. Note that if RTC were
unable or unwilling to suppress Hellos on the link, the
initial Database Description sent from Router RTC to RTB
would have the DC-bit clear, forcing Router RTB to revert to
the periodic sending of Hellos specified in Section 9.5 of
[1].

Time T1: Database exchange over demand circuit

The initial synchronization of link state databases (the
Database Exchange Process) over the demand circuit then
occurs as over any point-to-point link, with one exception.
LSAs included in Link State Updates Packets sent over the

```
                +              +                            +
                | +---+        |                            |
        +--+    |---|RTA|---   |                            |     +--+
        |H1|--- |   +---+      |                            |  ---|H2|
        +--+    |              |  +---+    ODL     +---+    |     +--+
                | LAN Y        |---|RTB|-----------|RTC|--- |
                +              |   +---+           +---+    |
                                +                           +
```
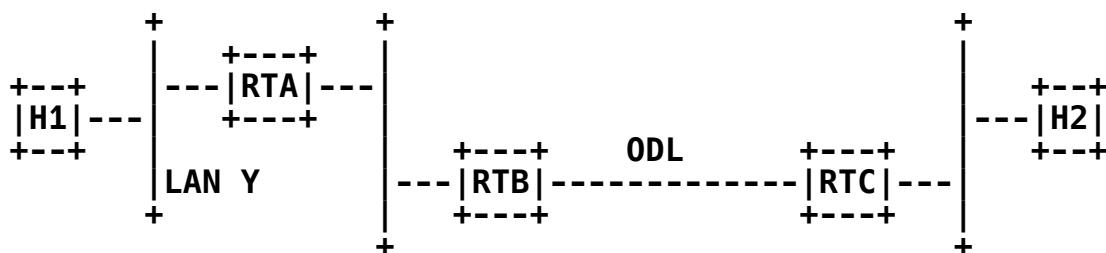
Figure 1: In the example of Section 4.1,
a single demand circuit (labeled
ODL) bisects an internetwork.

```
        +---+                                    +---+
        |RTB|                                    |RTC|
        +---+                                    +---+
                    Hello (DC-bit set)
        ------------------------------------------>
                    Hello (DC-bit clear)
        <------------------------------------------
                 Hello (DC-bit set, RTC seen)
        ------------------------------------------>
              Database Description (DC-bit set)
        <------------------------------------------
```
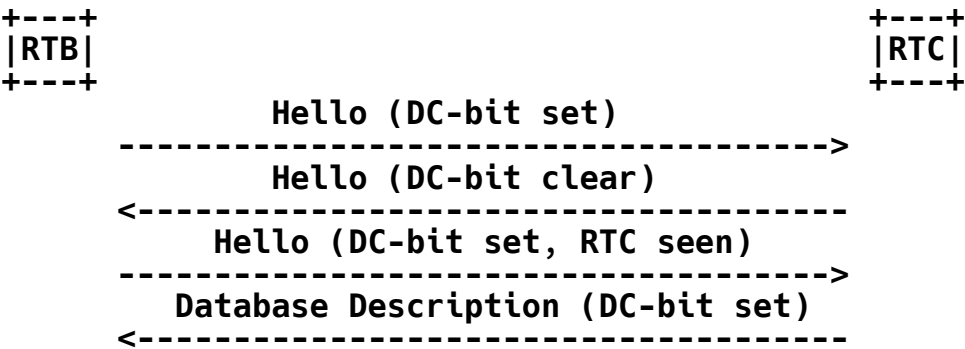
Figure 2: Successful negotiation of Hello
suppression.

demand circuit (in response to Link State Request Packets),
will have the DoNotAge bit set in their LS age field. So,
after the Database Exchange Process is finished, all routers
will have 3 LSAs in their link state databases (router-LSAs
for Routers RTA, RTB and RTC), but the LS age fields
belonging to the LSAs will vary depending on which side of
the demand circuit they were originated from (see Table 1).
For example, all routers other than Router RTC have the
DoNotAge bit set in Router RTC's router-LSA; this removes
the need for Router RTC to refresh its router-LSA over the
demand circuit.

| | LS age | |
LSA	in RTB	in RTC
RTA's Router-LSA	1000	DoNotAge+1001
RTB's Router-LSA	10	DoNotAge+11
RTC's Router-LSA	DoNotAge+11	10

Table 1: After Time T1 in Section 4.1,
possible LS age fields on either
side of the demand circuit

Time T2: Hello traffic ceases

After the Database Exchange Process has completed, no Hellos
are sent over the demand circuit. If there is no application
data to be sent over the demand circuit, the circuit will be
idle.

Time T3: Underlying data-link connection torn down

    After some period of inactivity, the underlying data-link
    connection will be torn down (e.g., an ISDN call would be
    cleared) in order to save connect charges. This will be
    transparent to the OSPF routing; no LSAs or routing table
    entries will change as a result.

Time T4: Router RTA's LSA is refreshed

    At some point Router RTA will refresh its own router-LSA
    (i.e., when the LSA's LS age hits LSRefreshInterval). This
    refresh will be flooded to Router RTB, who will look at it
    and decide NOT to flood it over the demand circuit to Router
    RTC, because the LSA's contents have not really changed
    (only the LS Sequence Number). At this point, the LS
    sequence numbers that the routers have for RTA's router-LSA
    differ depending on which side of the demand circuit the
    routers lie. Because there is still no application traffic,
    the underlying data-link connection remains disconnected.

Time T5: Router RTA's LAN interface comes up

    When Router RTA's LAN interface (connecting to Host H1)
    comes up, RTA will originate a new router-LSA. This router-
    LSA WILL be flooded over the demand circuit because its
    contents have now changed. The underlying data-link
    connection will have to be brought up to flood the LSA.
    After flooding, routers on both sides of the demand circuit
    will again agree on the LS Sequence Number for RTA's
    router-LSA.

Time T6: Underlying data-link connection is torn down again

    Assuming that there is still no application traffic
    transiting the demand circuit, the underlying data-link
    connection will again be torn down after some period of
    inactivity.

Time T7: File transfer started between Hosts H1 and H2

    As soon as application data needs to be sent across the
    demand circuit the underlying data-link connection is
    brought back up.

Time T8: Physical link becomes inoperative

> If an indication is received from the data-link or physical
> layers indicating that the demand circuit can no longer be
> established, Routers RTB and RTC declare their point-to-
> point interfaces down, and originate new router-LSAs. Both
> routers will attempt to bring the connection back up by
> sending Hellos at the reduced rate of PollInterval. Note
> that while the connection is inoperative, Routers RTA and
> RTB will continue to have an old router-LSA for RTC in their
> link state database, and this LSA will not age out because
> it has the DoNotAge bit set. However, according to Section
> 2.3 they will flush Router RTC's router-LSA if the demand
> circuit remains inoperative for longer than MaxAge.

4.2. Example 2: Demand and non-demand circuits in parallel

This example demonstrates the demand circuit functionality when
both demand circuits and non-demand circuits (e.g., leased lines)
are used to interconnect regions of an internetwork. Such an
internetwork is shown in Figure 3. Host H1 can communicate with
Host H2 either over the demand link between Routers RTB and RTC,
or over the leased line between Routers RTB and RTD.

Because the basic properties of the demand circuit functionality
were presented in the previous example, this example will only
address the unique issues involved when using both demand and
non-demand circuits in parallel.

Assume that Routers RTB and RTY are initially powered off, but
that all other routers and their attached links are both
operational and implement the demand circuit modifications to
OSPF. Throughout the example, a TCP connection between Hosts H1
and H2 is transmitting data. Furthermore, assume that the cost of
the demand circuit from RTB to RTC has been set considerably
higher than the cost of the leased line between RTB and RTD; for
this reason traffic between Hosts H1 and H2 will always be sent
over the leased line when it is operational.

The following events may then transpire:

```
                                       +
                              +---+    |
                              |RTC|--  |
                              +---+    |    +---+    |
                                /      |  --|RTE|--  | +--+
                               /ODL    |    +---+    |--|H2|
    +--+    +                 /        |        +    | +  +--+
    |H1|----|    +---+    +---+/       |
    +--+    |  --|RTA|-------|RTB|     |
            |    +---+       +---+\    |    +---+    |
            +                  \  |  --|RTY|--  |
                                \      |    +---+    |
                                 \     |        +
                              +---+    |
                              |RTD|--  |
                              +---+    |
                                       +
```
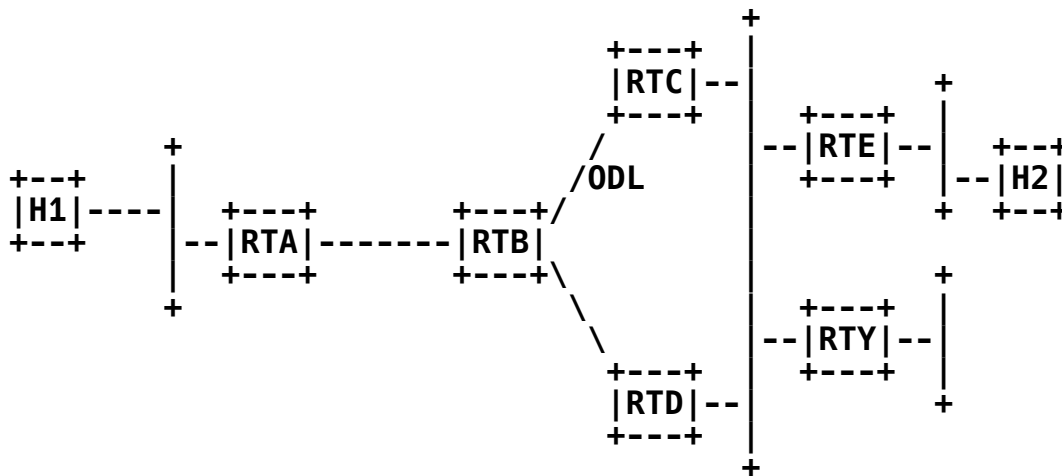
Figure 3: Example 2's internetwork.

Vertical lines are LAN segments. Six routers
are pictured, Routers RTA-RTE and RTY.
RTB has three serial line interfaces, two of
which are leased lines and the third (connecting to
RTC) a demand circuit. Two hosts, H1 and
H2, are pictured to illustrate the effect of
application traffic.

Time T0: Router RTB comes up.

Assume RTB supports the demand circuit OSPF modifications.
When Router RTB comes up and establishes links to Routers
RTC and RTD, it will flood the same information over both.
However, LSAs sent over the demand circuit (to Router RTC)
will have the DoNotAge bit set, while those sent over the
leased line to Router RTD will not. Because the DoNotAge bit
is not taken into account when comparing LSA instances, the
routers on the right side of RTB (RTC, RTE and RTD) may or
may not have the DoNotAge bit set in their database copies
of RTA's and RTB's router-LSAs. This depends on whether the
LSAs sent over the demand link reach the routers before
those sent over the leased line. One possibility is pictured
in Table 2.

| | | LS age | |
LSA	in RTC	in RTD	in RTE
RTA's Router-LSA	DoNotAge+20	21	21
RTB's Router-LSA	DoNotAge+5	6	6

   Table 2: After Time T0 in Example 2, LS age
    fields on the right side of Router RTB.

| | | LS age | |
LSA	in RTC	in RTD	in RTE
RTA's Router-LSA	5	6	6
RTB's Router-LSA	DoNotAge+5	1785	1785

   Table 3: After Time T2 in Example 2, LS age
    fields on the right side of Router RTB.

| | | LS age | |
LSA	in RTC	in RTD	in RTE
RTA's Router-LSA	325	326	326
RTB's Router-LSA	DoNotAge+5	DoNotAge+6	DoNotAge+6

   Table 4: After Time T3 in Example 2, LS age
    fields on the right side of Router RTB.

| | | LS age | |
LSA	in RTC	in RTD	in RTE
RTA's Router-LSA	DoNotAge+7	DoNotAge+8	DoNotAge+8
RTB's Router-LSA	DoNotAge+5	DoNotAge+6	DoNotAge+6

   Table 5: After Time T4 in Example 2, LS age
    fields on the right side of Router RTB.

Time T1: Underlying data-link connection is torn down.

All application traffic is flowing over the leased line
connecting Routers RTB and RTD instead of the demand
circuit, due to the leased line's lesser OSPF cost. After
some period of inactivity, the data-link connection
underlying the demand circuit will be torn down. This does
not affect the OSPF database or the routers' routing tables.

Time T2: Router RTA refreshes its router-LSA.

When Router RTA refreshes its router-LSA (as all routers do
every LSRefreshInterval), Router RTB floods the refreshed
LSA over the leased line but not over the demand circuit,
because the contents of the LSA have not changed. This new
LSA will not have the DoNotAge bit set, and will replace the
old instances (whether or not they have the DoNotAge bit
set) by virtue of its higher LS Sequence number. This is
pictured in Table 3.

Time T3: Leased line becomes inoperational.

When the leased line becomes inoperational, the data-link
connection underlying the demand circuit will be reopened,
in order to flood a new (and changed) router-LSA for RTB and
also to carry the application traffic between Hosts H1 and
H2. After flooding the new LSA, all routers on the right
side of the demand circuit will have DoNotAge set in their
copy of RTB's router-LSA and DoNotAge clear in their copy of
RTA's router-LSA (see Table 4).

Time T4: In Router RTE, Router RTA's router-LSA times out.

Refreshes of Router RTA's router-LSA are not being flooded
over the demand circuit. However, RTA's router-LSA is aging
in all of the routers to the right of the demand circuit.
For this reason, the router-LSA will eventually be aged out
and reflooded (by router RTE in our example). Because this
aged out LSA constitutes a real change (see Section 3.3), it
is flooded over the demand circuit from Router RTC to RTB.
There are then two possible scenarios. First, the LS
Sequence number for RTA's router-LSA may be larger on RTB's
side of the demand link. In this case, when router RTB
receives the flushed LSA it will respond by flooding back
the more recent instance (see Section 2.4). If instead the
LS sequence numbers are the same, the flushed LSA will be
flooded all the way back to Router RTA, which will then be
forced to reoriginate the LSA.

In any case, after a small period all the routers on the
right side of the demand link will have the DoNotAge bit set
in their copy of RTA's router-LSA (see Table 5). In the
small interval between the flushing and waiting for a new
instance of the LSA, there will be a temporary loss of
connectivity between Hosts H1 and H2.

Time T5: A non-supporting router joins.

Suppose Router RTY now becomes operational, and does not
support the demand circuit OSPF extensions. Router RTY's
router-LSA then will not have the DC-bit set in its Options
field, and as the router-LSA is flooded throughout the
internetwork it flushes all LSAs having the DoNotAge bit set
and causes the flooding behavior over the demand circuit to
revert back to the normal flooding behavior defined in [1].
However, although all LSAs will now be flooded over the
demand circuit, regardless of whether their contents have
really changed, Hellos will still continue to be suppressed
on the demand circuit (see Section 3.2.2).

4.3. Example 3: Operation when oversubscribed

The following example shows the behavior of the demand circuit
extensions in the presence of oversubscribed interfaces. Note that
the example's topology excludes the possibility of alternative
paths. The combination of oversubscription and redundant topology
(i.e., alternative paths) poses special problems for the demand
circuit extensions. These problems are discussed later in Section
7.

Figure 4 shows a single Router (RT1) connected via demand circuits
to three other routers (RT2-RT4). Assume that RT1 can only have
two out of three underlying data-link connections open at once.
This may be due to one of the following reasons: Router RT1 may be
using a single Basic Rate ISDN interface (2 B channels) to support
all three demand circuits, or, RT1 may be connected to a data-link
switch (e.g., an X.25 or Frame relay switch) that is only capable
of so many simultaneous data-link connections.

The following events may transpire, starting with Router RT1
coming up.

Time T0: Router RT1 comes up.

Router RT1 attempts to establish neighbor connections and
synchronize OSPF databases with routers RT2-RT4. But,

```
                                        +  +--+
                              +---+    |--|H2|
                 +---------|RT2|--|  +--+
                /          +---+    |
               / ODL                +
              /
+--+ +      /                         +
|H1|--|    /           ODL    +---+  |  +--+
+--+  |  +---+-----------|RT3|--|--|H3|
      |--|RT1|-----------+---+  |  +--+
      |  +---+                   +
      |     \
      +      \ODL
              \
               \           +---+    +  +--+
                +--------|RT4|--|--|H4|
                           +---+    |  +--+
                                     +
```
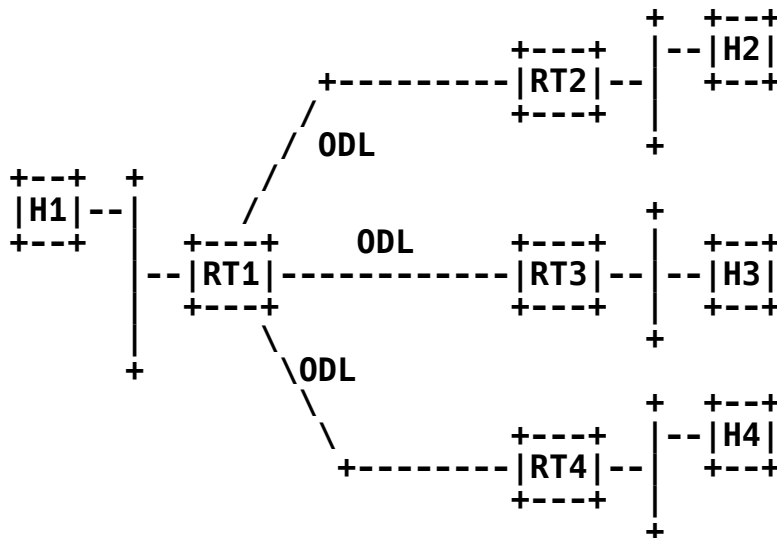
Figure 4: Example 3's internetwork.

because it cannot have data-link connections open to all
three at once, it will synchronize with RT2 and RT3, while
Hellos sent to RT4 will be discarded (see Section 1).

Time T1: Data-link connection to RT2 closed due to inactivity.

Assuming that no application traffic is being sent to/from
Host H2, the underlying data-link connection to RT2 will
eventually close due to inactivity. This will allow RT1 to
finally synchronize with RT4; the next Hello that RT1
attempts to send to RT4 will cause that data-link connection
to open and synchronization with RT4 will ensue. Note that,
until this time, H4 will have been considered unreachable by
OSPF routing. However, data traffic would not have been
deliverable to H4 until now in any case.

Time T2: RT2's LAN interface becomes inoperational

This causes RT2 to reissue its router-LSA. However, it may
be unable to flood it to RT1 if RT1 already has data-link
connections open to RT3 and RT4. While the data-link
connection from RT2 to RT1 cannot be opened due to resource
shortages, the new router-LSA will be continually
retransmitted (and dropped by RT2's ISDN interface; see
Section 1). This means that the routers RT1, RT3 and RT4
will not detect the unreachability of Host H2 until a data-
link connection on RT1 becomes available.

5. Topology recommendations

Because LSAs indicating topology changes are still flooded over
demand circuits, it is still advantageous to design networks so that
the demand circuits are isolated from as many topology changes as
possible. In OSPF, this is done by encasing the demand circuits
within OSPF stub areas or within NSSAs (see [3]). In both cases, this
isolates the demand circuits from AS external routing changes, which
in many networks are the most frequent (see [6]). Stub areas can even
isolate the demand circuits from changes in other OSPF areas.

Also, considering the interoperation of OSPF routers supporting
demand circuits and those that do not (see Section 2.5), isolated
stub areas or NSSAs can be converted independently to support demand
circuits. In contrast, regular OSPF areas must all be converted
before the functionality can take effect in any particular regular
OSPF area.

6. Lost functionality

The enhancements defined in this memo to support demand circuits come
at some cost. Although we gain an efficient use of demand circuits,
holding them open only when there is actual application data to send,
we lose the following:

Robustness
In regular OSPF [1], all LSAs are refreshed every
LSRefreshInterval. This provides protection against routers
losing LSAs from (or LSAs getting corrupted in) their link state
databases due to software errors, etc. Over demand circuits
this periodic refresh is removed, and we depend on routers
correctly holding LSAs marked with DoNotAge in their databases
indefinitely.

Database Checksum
OSPF supplies network management variables, namely
ospfExternLSACksumSum and ospfAreaLSACksumSum in [7], allowing a
network management station to verify OSPF database
synchronization among routers. However, these variables are sums
of the individual LSAs' LS checksum fields, which are no longer
guaranteed to be identical across demand circuits (because the
LS checksum covers the LS Sequence Number, which will in general
differ across demand circuits). This means that these variables
can no longer be used to verify database synchronization in OSPF
networks containing demand circuits.

7. Future work: Oversubscription

An internetwork is oversubscribed when not all of its demand
circuits' underlying connections can be open at once, due to resource
limitations. These internetworks were addressed in Section 4.3.
However, when all possible sources in the internetwork are active at
once, problems can occur which are not addressed in this memo:

(1) There is a network design problem. Does a subset of demand
    circuits exist such that a) their data-link connections can be
    open simultaneously and b) they can provide connectivity for all
    possible sources? This requires that (at least) a spanning tree
    be formed out of established connections. Figure 4 shows an
    example where this is not possible; Hosts H1 through H4 cannot
    simultaneously talk, since Router RT1 is limited to two
    simultaneously open demand circuits.

(2) Even if it is possible that a spanning tree can form, will one?
    Given the model in Section 1, demand circuits are brought up
    when needed for data traffic, and stay established as long as
    data traffic is present. One example is shown in Figure 5. Four
    routers are interconnected via demand circuits, with each router
    being able to establish a circuit to any other. However, we
    assume that each router can only have two circuits open at once
    (e.g., the routers could be using Basic Rate ISDN). In this
    case, one would hope that the data-link connections in Figure 5a
    would form. But the connections in Figure 5b are equally
    likely, which leave Host H2 unable to communicate.

    One possible approach to this problem would be for a) the OSPF
    database to indicate which demand circuits have actually been
    established and b) implement a distributed spanning tree
    construction (see for example Chapter 5.2.2 of [9]) when
    necessary.

(3) Even when a spanning tree has been built, will it be used?
    Routers implementing the functionality described in this memo do
    not necessarily know which data-link connections are established
    at any one time. In fact, they view all demand circuits as being
    equally available, whether or not they are currently
    established. So for example, even when the established
    connections form the pattern in Figure 5a, Router RT1 may still
    believe that the best path to Router RT3 is through the direct
    demand circuit. However, this circuit cannot be established due
    to resource shortages.

```
       +--+ +               +               + +--+
       |H1|--|    +---+  ODL   +---+  |--|H2|
       +--+  |--|RT1|-------|RT2|--|  +--+
             |    +---+               +---+  |
              +    |   \           /   |    +
                   |    \         /    |
                   |     \       /     |
                   |ODL   \     /   |ODL
                   |       \   /\ODL  |
                   |      / \ODL \    |
              +    |    /ODL    \  |    +
       +--+ |    +---+               +---+ |    +--+
       |H4|--|--|RT4|-------|RT3|--|--|H3|
       +--+  |    +---+  ODL   +---+  |    +--+
              +                               +
```
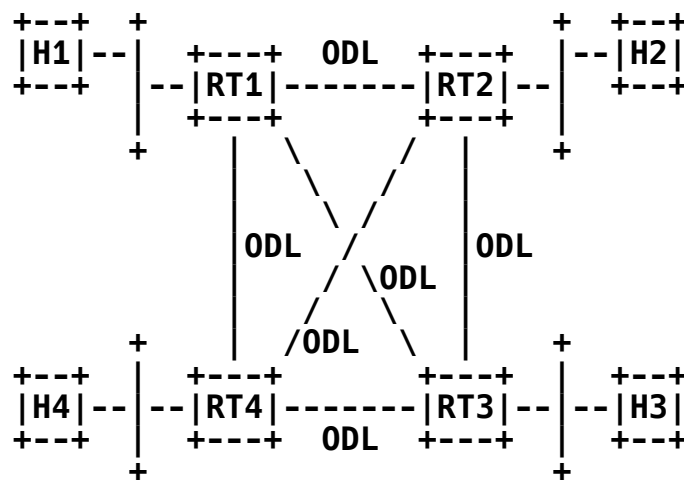
                Figure 5: Example of an oversubscribed
                           internetwork
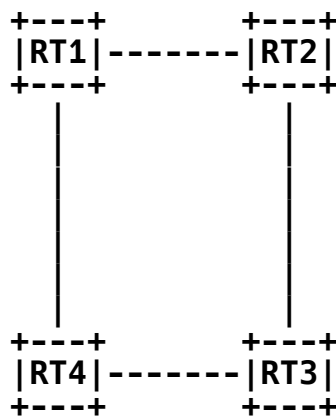
```
     +---+        +---+              +---+        +---+
     |RT1|--------|RT2|              |RT1|        |RT2|
     +---+        +---+              +---+        +---+
       |            |                  |   \
       |            |                  |    \
       |            |                  |     \
       |            |                  |      \
       |            |                  |       \
       |            |                  |        \
     +---+        +---+              +---+        +---+
     |RT4|--------|RT3|              |RT4|--------|RT3|
     +---+        +---+              +---+        +---+

     Figure 5a: One possible         Figure 5b: Another possible
       pattern of data-link            pattern of data-link
           connections                     connections
```
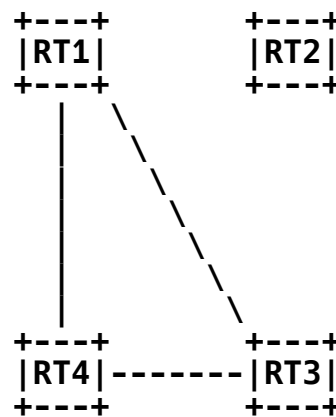
On possible approach to this problem is to increase the OSPF cost of
demand circuits that are currently discarding application packets
(i.e., can't be established) due to resource shortages. This may help
the routing find paths that can actually deliver the packets. On the
downside, it would create more routing traffic. Also, unwanted
routing oscillations may result when you start varying routing
metrics to reflect dynamic network conditions; see [10].

8. Unsupported capabilities

    The following possible capabilities associated with demand circuit
    routing have explicitly not been supported by this memo:

    o   When the topology of an OSPF area changes, the changes are
         flooded over the area's demand circuits, even if this requires
         (re)establishing the demand circuits' data-link connections. One
         might imagine a routing system where the flooding of topology
         changes over demand circuits were delayed until the demand
         circuits were (re)opened for application traffic. However, this
         capability is unsupported because delaying the flooding in this
         manner would sometimes impair the ability to discover new
         network destinations.

    o   Refining the previous capability, one might imagine that the
         network administrator would be able to configure for each demand
         interface whether flooding should be immediate, or whether it
         should be delayed until the data-link connection is established
         for application traffic. This would allow certain "application-
         specific" routing behaviors. For example, a demand circuit may
         connect a collection of client-based subnets to a collection of

server-based subnets. If the client end was configured to delay
flooding, while the server end was configured to flood changes
immediately, then new servers would be discovered promptly while
clients might not be discovered until they initiate
conversations. However, this capability is unsupported because
of the increased complexity of (and possibility for error in)
the network configuration.

A. Format of the OSPF Options field

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

   The memo defines one of the Option bits: the DC-bit (for Demand
   Circuit capability). The DC-bit is set in a router's self-originated
   LSAs if and only if it supports the functionality defined in Section
   2 of this memo. Note that this does not necessarily mean that the
   router can be the endpoint of a demand circuit, but only that it can
   properly process LSAs having the DoNotAge bit set. In contrast, the
   DC-bit is set in Hello Packets and Database Description Packets sent
   out an interface if and only if the router wants to treat the
   attached point-to-point network as a demand circuit (see Section
   3.2.1).

   The addition of the DC-bit makes the current assignment of the OSPF
   Options field as follows:

```
                    +--------------------------------------+
                    | * | * | DC | EA | N/P | MC | E | T |
                    +--------------------------------------+
```

                        Figure 5: The OSPF Options field

     T-bit
         This bit describes TOS-based routing capability, as specified in
         [1].

     E-bit
         This bit describes the way AS-external-LSAs are flooded, as
         described in [1].

     MC-bit
         This bit describes whether IP multicast datagrams are forwarded
         according to the specifications in [4].

     N/P-bit
         This bit describes the handling of Type-7 LSAs, as specified in
         [3].

EA-bit
This bit describes the router's willingness to receive and
forward External-Attributes-LSAs, as specified in [5].

DC-bit
This bit describes the handling of demand circuits, as specified
in this memo. Its setting in Hellos and Database Description
Packets is described in Sections 3.2.1 and 3.2.2. Its setting in
LSAs is described in Sections 2.1 and 2.5.

B. Configurable Parameters

This memo defines a single additional configuration parameter for
OSPF interfaces. In addition, the OSPF Interface configuration
parameter PollInterval, previously used only on NBMA networks, is now
also used on point-to-point networks (see Sections 3.1 and 3.2.2).

ospfIfDemand
Indicates whether the interface connects to a demand circuit.
When set to TRUE, the procedures described in Section 3 of this
memo are followed, in order to send a minimum of routing traffic
over the demand circuit. On point-to-point networks, this allows
the circuit to be closed when not carrying application traffic.
When a broadcast or NBMA interface is configured to connect to a
demand circuit (see Section 1.2 of [1]), the data-link
connections will be kept open constantly due to OSPF Hello
traffic, but the amount of flooding traffic will still be
greatly reduced.

C. Architectural Constants

This memo defines a single additional OSPF architectural constant.

DoNotAge
Equal to the hexadecimal value 0x8000, which is the high bit of
the 16-bit LS age field in OSPF LSAs. When this bit is set in
the LS age field, the LSA is not aged as it is held in the
router's link state database. This allows the elimination of the
periodic LSA refresh over demand circuits. See Section 2.2 for
more information on processing the DoNotAge bit.

References

    [1] Moy, J., "OSPF Version 2", RFC 1583, Proteon, Inc., March 1994.

    [2] Meyer, G., "Extensions to RIP to Support Demand Circuits", RFC
        1582, Spider Systems, February 1994.

    [3] Coltun, R. and V. Fuller, "The OSPF NSSA Option", RFC 1587,
        RainbowBridge Communications, Stanford University, March 1994.

    [4] Moy, J., "Multicast Extensions to OSPF", RFC 1584, Proteon, Inc.,
        March 1994.

    [5] Ferguson, D., "The OSPF External Attributes LSA", Work in
        Progress.

    [6] Moy, J., Editor, "OSPF Protocol Analysis", RFC 1245, Proteon,
        Inc., July 1991.

    [7] Baker F. and R. Coltun, "OSPF Version 2 Management Information
        Base", RFC 1253, ACC, University of Maryland, August 1991.

    [8] Baker F., "OSPF Point-to-MultiPoint Interface", Work in Progress.

    [9] Bertsekas, D., and R. Gallager, "Data Networks", Prentice Hall,
        Inc., 1992.

   [10] Khanna, A., "Short-Term Modifications to Routing and Congestion
        Control", BBN Report 6714, BBN, February 1988.

Security Considerations

    Security issues are not discussed in this memo.

Author's Address

    John Moy
    Cascade Communications Corp.
    5 Carlisle Road
    Westford, MA 01886

    Phone: 508-692-2600 Ext. 394
    Fax: 508-692-9214
    EMail: jmoy@casc.com