

OSPF Working Group
Internet Draft
Intended status: Standard Track
Expires: November 22, 2010

Dimitri Papadimitriou
Alcatel-Lucent
May 23, 2010

Phased OSPF Link-State Database Synchronization
draft-dimitri-ospf-phased-db-sync-00.txt

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>

This Internet-Draft will expire on November 22, 2010.

Internet-Draft Phased OSPF Database Synchronization May 23, 2010

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the [Trust Legal Provisions](#) and are provided without warranty as described in the Simplified BSD License.

Abstract

Opaque Link-State Advertisements (LSA) extend the topological link state of Open Shortest Path First (OSPF). The information contained in Opaque LSA may be used directly by OSPF or indirectly by some application wishing to distribute information throughout the OSPF domain. However the Link-State Database (LSDB) synchronization process is kept unified, i.e., there is no messaging or processing allowing to order the exchanges in the link state database synchronization process. We call this ordering, phasing of logically segmented LSDB into Opaque and non-Opaque. The motivation is to prevent delaying reaching Full state whereas synchronizing over the entire LSDB would delay full adjacency establishment.

Table of Contents

| | | |
|----------------------|---|-------------------|
| 1. | Introduction..... | 3 |
| 2. | Conventions used in this document..... | 4 |
| 3. | Link-State Database (LSDB) Synchronization..... | 4 |
| 3.1. | LSDB Synchronization: General Description..... | 4 |
| 3.2. | Application of LSDB Synchronization to Opaque LSA..... | 5 |
| 4. | Phased Link-State Database (LSDB) Synchronization..... | 6 |
| 4.1. | Phased Link-State Database (LSDB) Synchronization Process..... | 6 |
| 4.2. | Transition from LSDB to Opaque LSDB Synchronization.... | 8 |

| | | |
|------------------------|------------------------------------|--------------------|
| 4.3. | Capability Negotiation..... | 8 |
| 4.3.1. | Option field in Hello Packets..... | 9 |
| 4.3.2. | Link Local Signaling (LLS)..... | 9 |
| 5. | Backward Compatibility..... | 9 |
| 6. | Security Considerations..... | 10 |
| 7. | IANA Considerations..... | 10 |
| 8. | References..... | 10 |

Internet-Draft Phased OSPF Database Synchronization May 23, 2010

| | | |
|----------------------|-----------------------------|--------------------|
| 8.1. | Normative References..... | 10 |
| 8.2. | Informative References..... | 10 |
| 9. | Acknowledgments..... | 10 |

[1.](#) Introduction

Open Shortest Path First (OSPF) link-state routing protocol supports a class of Link-State Advertisements (LSA) called Opaque LSAs that provide a generalized mechanism to allow extensibility of OSPF [[RFC2328](#)]. The information field contained in Opaque LSAs is often indirectly used by some application wishing to distribute information throughout the OSPF domain. Standard OSPF Link-State Database (LSDB) flooding mechanisms are used to distribute Opaque LSAs to all or some limited portion of the OSPF topology [[RFC2370](#)]. Nevertheless, OSPF mandates full synchronization of the LSDB before a routing adjacency is declared in state Full (when LSDB synchronization is completed, the neighbor is in state Full and the two routers are fully adjacent).

The reliable and effective LSDB synchronization but also link-state flooding mechanism provided by OSPF has thus been re-used by many distributed network applications that rely on OSPF to exchange non IP routing information. By non-IP routing information, we mean any information that is not directly or indirectly related to the forwarding of IP datagrams. Another case that often leads to a delayed synchronization process is when the number of entries is not bound by the number of links. This observation also leads us to think that AS-external LSAs in particular are good candidate for the approach proposed here and the mechanism can thus be seen as a complement to [[RFC1765](#)].

The proposed mechanism phases the LSDB synchronization process by first exchanging IP routing LSAs (Router, Network, Summary, AS-external, and Not-so-stubby area LSAs) and then, the Opaque LSAs

as defined in [\[RFC2370\]](#). The purpose is to prevent delaying the establishment of fully adjacent routers - at this point the adjacency is listed in LSAs - even if the "Opaque part" of the LSDB is not synchronized. We note here that in most cases the application itself makes use of the IP adjacencies for application specific message exchanges and thus the applications would not be slow down by this process. In this sense, the present document reverts back to [\[RFC2328\]](#) the LSDB synchronization process as extended by [\[RFC2370\]](#) (that covers LSDB including both non-Opaque and Opaque LSAs). Phasing is achieved by logically segmenting the LSDB synchronization process: add "on top of" the LSDB synchronization process

Internet-Draft Phased OSPF Database Synchronization May 23, 2010

described in [\[RFC2328\]](#), a synchronization process dedicated to Opaque LSAs. This phasing prevents delaying establishment of full adjacency between two routers (Full state) resulting from the time needed to synchronize Opaque LSAs. This condition occurs in particular when the number of Opaque LSAs >> non-Opaque LSAs. The fundamental aspect of the proposed approach consists thus of considering the state Full as the invariant state for reaching full adjacency.

The purpose of the proposed Opaque LSDB Synchronization process is to devise a less drastic alternative to the current approach developed at OSPF WG that mandates complete separation of OSPF instances when Opaque LSA are decoupled from IP Routing [OSPF-TP]. The latter does not actually solve the so-called "Opaque overload" problem because it separates IP-related from non-IP related routing information instead of Opaque from non-Opaque LSAs. The proposed approach here is to avoid duplicating OSPF instances while keeping Opaque LSA messaging and processing as part of a single OSPF instance.

[2.](#) Conventions used in this document

In examples, "C:" and "S:" indicate lines sent by the client and server respectively.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC-2119](#).

[3.](#) Link-State Database (LSDB) Synchronization

[3.1.](#) LSDB Synchronization: General Description

In link-state routing, it is very important for all routers' Link-State Databases (LSDB) 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 routing adjacency.

Each router describes its database by sending a sequence of Database Description (DD) packets to its neighbor. Each Database Description packet describes a set of LSAs belonging to the router's database. When the neighbor sees an LSA that is more recent than its own database copy, it makes a note that this newer LSA 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 per-interface constant RxmtInterval.

[3.2.](#) Application of LSDB Synchronization to Opaque LSA

Per [\[RFC2370\]](#): an opaque-capable router learns of its neighbor's opaque capability at the beginning of the "Database Exchange Process" (see [Section 10.6 of \[RFC2328\]](#), receiving Database Description packets from a neighbor in state ExStart). A neighbor is opaque-capable if and only if it sets the O-bit in the Options field of its Database Description packets; the O-bit is not set in packets other than Database Description packets. Then, in the next step of the Database Exchange process, Opaque LSAs are included in the Database summary list that is sent to the neighbor if and only if the neighbor is opaque capable. When flooding Opaque LSAs to adjacent neighbors, an opaque-capable router looks at the neighbor's opaque capability. Opaque LSAs are only flooded to opaque-capable neighbors, i.e., Opaque LSAs are

only placed on the link-state retransmission lists of opaque-capable neighbors. In case non-opaque-capable neighbor inadvertently receives Opaque LSAs, the non-opaque-capable router will then simply discard the LSA receiving LSAs having unknown LS types).

Hence, [RFC2370] does not modify the state machine as defined in [Section 10.3 of \[RFC2328\]](#) except for the action associated with State: ExStart, Event: NegotiationDone which is where the Database summary list is built in order to incorporate the Opaque LSA in OSPF (see Figure 1).

4. Phased Link-State Database (LSDB) Synchronization

Compared to the [RFC2370] processing, the Phase Link-State Database (LSDB) synchronization modifies the LSDB exchange process as follows: Opaque LSAs are included in the LSDB summary list that is sent to the neighbor, if and only if

- i) The neighbor is Opaque capable (see [Section 4](#) and [Appendix A of \[RFC2370\]](#))

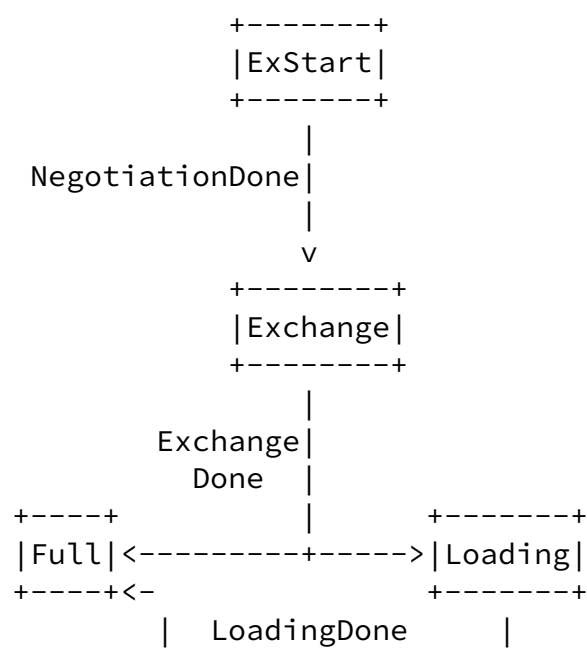


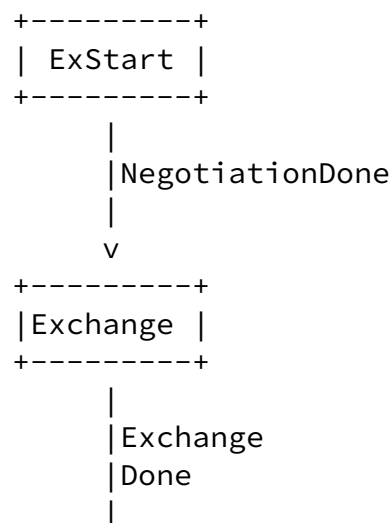
Fig.1 Neighbor state changes (Database Exchange)

ii) The neighbor has fully exchanged the area LSDB that consists of the router-LSAs (Type 1), network-LSAs (Type 2), and summary-LSAs (Type 3, 4) contained in the area structure, along with the AS-external-LSAs (Type 5) contained in the global structure, and Not-So-Stubby Area (NSSA) LSAs (Type 7) [[RFC3101](#)], i.e., the "Full" state has been reached.

iii) Both local and neighbor router supports the phased LSDB synchronization (see [Section 4.3](#)).

[4.1](#). Phased Link-State Database (LSDB) Synchronization Process

The process is depicted in Fig.2, the ExStart, Exchange, Loading and Full states are defined per [[RFC2328](#)]. Note that in Full State, the router can perform all subsequent operations per [[RFC 2328](#)] including, the computation of the shortest-path tree for an area, and the computation of the AS external routes, as described in [Section 16 of \[RFC2328\]](#). Events NegotiationDone, ExchangeDone and LoadingDone are used as defined per [[RFC2328](#)].



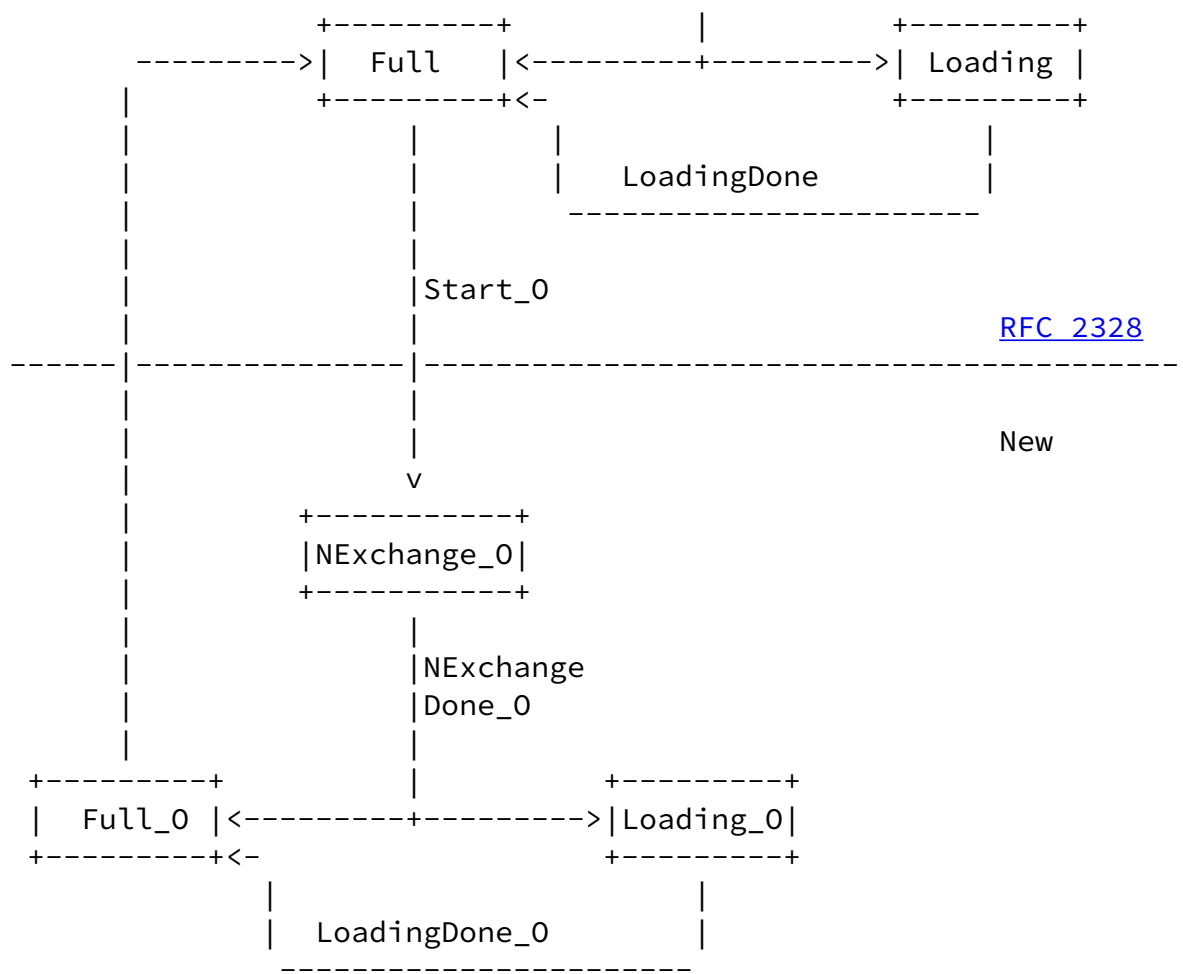


Fig.2 Modified Neighbor state changes (Database Exchange)

[4.2](#). Transition from LSDB to Opaque LSDB Synchronization

In case Full state is not reached due to e.g. corruption or Fletcher checksum error, the exchange process restarts (go back to ExStart).

In case Full state is reached, the process continues as follows (note that the master remains the master as negotiated during the ExStart step):

- Start_0 (Event): LSDB contain Opaque_LSA's AND capability as described in [Section 4.3](#) successfully negotiated. This

ensures backward compatibility with [\[RFC2370\]](#).

- NExchange_0 (State): the router lists the contents of its Opaque area LSDB in the neighbor Database summary list. The Opaque area LSDB consists of Type 9 and Type 10 Opaque LSAs along with Type 11 Opaque LSAs contained in the global structure. The router sends the Database Description (DD) packets for these Opaque LSAs to the neighbor. Each DD Packet has a DD sequence number, and is explicitly acknowledged. Only one DD Packet is allowed outstanding at any one time. In this state, LS Request Packets may also be sent asking for the neighbor's more recent Opaque LSAs.
- NExchangeDone_0 (Event): both routers have successfully transmitted a full sequence of DD packets. Each router now knows what parts of its Opaque LSDB are out of date.
- Loading_0 (State): LS Request packets are sent to the neighbor asking for the more recent Opaque LSAs that have been discovered (but not yet received) in the NExchange_0 state.
- LoadingDone_0 (Event): LS Updates have been received for all out-of-date portions of the Opaque LSDB. This is indicated by the Link state request list becoming empty after the Database Exchange process has completed.
- Full_0 (State): neighboring nodes have completed Opaque LSA exchange.

[4.3.](#) Capability Negotiation

Negotiating Phased LSDB synchronization can be performed by inserting a Phased LSDB Flag in:

- i) Option field of Hello Packets and DD Packets
- ii) Data block of Link Local Signaling (LLS) and DD Packets

[4.3.1.](#) Option field in Hello Packets

The Options field (8-bits) enables OSPF routers to support (or

not support) optional capabilities, and to communicate their capability level to other OSPF routers. Through this mechanism routers of differing capabilities can be mixed within an OSPF routing domain. When used in Hello packets, the Options field allows a router to accept a neighbor at the condition of adaptation to neighbors's capability (due to the initial mismatch): if either of the neighbors does not support or does not recognize the capability the synchronization is not phased. In this case, routers encountering the unrecognized Option bits in received Hello Packets ignore the capability and process the packet normally.

Then, by exchanging this capability in Database Description (DD) packets a router can sequence its exchange of LSAs (starting by non-Opaque LSAs and then Opaque LSAs): if both neighbors are capable of phased synchronization, they may still decide to use or not.

This alternative is perfectly valid but requires usage of one bit of the Option field that is a very sparse resource.

[4.3.2](#). Link Local Signaling (LLS)

To Link-local signaling (LLS), OSPF routers add a special data block to the end of OSPF packets (or right after the authentication data block when cryptographic authentication is used). The LLS block is attached to OSPF Hello packets. The drawback of this alternative is that the delivery of LLS data in Hello packets is not guaranteed.

To circumvent this problem, the solution consists in piggy bagging the Phased DB Flag in the Database Description packets.

[5](#). Backward Compatibility

The proposed synchronization process is backward compatible since the mechanism extends the current process if and only if the mechanism is locally (see [Section 4.2](#)) and remotely supported (see [Section 4.3](#)). If either of these conditions is not met LSDB synchronization falls back to the linear process currently

However, it does not prevent that routers may be required to support both methods. This could be the case typically for ABR's.

6. Security Considerations

TBD

7. IANA Considerations

TBD

8. References

8.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2234] Crocker, D. and Overell, P.(Editors), "Augmented BNF for Syntax Specifications: ABNF", [RFC 2234](#), Internet Mail Consortium and Demon Internet Ltd., November 1997.
- [RFC2328] J. Moy, "OSPF version 2", [RFC 2328](#), Internet Engineering Task Force, April 1998.
- [RFC2370] R. Coltun, "The OSPF Opaque LSA Option", [RFC 2370](#), July 1998.

8.2. Informative References

- [OSPF-TP] A.Lindem, et al. "OSPF Transport Instance Extensions", IETF Draft, Work in Progress, [draft-ietf-ospf-transport-instance-04.txt](#), April 2010.
- [RFC3101] P. Murphy, "The OSPF Not-So-Stubby Area (NSSA) Option", [RFC 3101](#), Internet Engineering Task Force, January 2003.

9. Acknowledgments

Authors would like to thank Marc Lasserre, Devendra Raut, Andrew Lange, and Manav Bhatia for their comments.

This document was prepared using 2-Word-v2.0.template.dot.

Authors' Addresses

Dimitri Papadimitriou
Alcatel-Lucent Bell
Copernicuslaan 50
2018 Antwerpen
Belgium
Phone: +32 3 2408491
Email: dimitri.papadimitriou@alcatel-lucent.com

