Routing Area Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 13, 2012

A. Atlas, Ed.
R. Kebler
Juniper Networks
G. Enyedi
A. Csaszar
Ericsson
M. Konstantynowicz
R. White
Cisco Systems
M. Shand

March 12, 2012

# An Architecture for IP/LDP Fast-Reroute Using Maximally Redundant Trees

draft-ietf-rtgwg-mrt-frr-architecture-01

## Abstract

As IP and LDP Fast-Reroute are increasingly deployed, the coverage limitations of Loop-Free Alternates are seen as a problem that requires a straightforward and consistent solution for IP and LDP, for unicast and multicast. This draft describes an architecture based on redundant backup trees where a single failure can cut a point-of-local-repair from the destination only on one of the pair of redundant trees.

One innovative algorithm to compute such topologies is maximally disjoint backup trees. Each router can compute its next-hops for each pair of maximally disjoint trees rooted at each node in the IGP area with computational complexity similar to that required by Dijkstra.

The additional state, address and computation requirements are believed to be significantly less than the Not-Via architecture requires.

## Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on September 13, 2012.

## Copyright Notice

Table of Contents

## 1. Introduction

There is still work required to completely provide IP and LDP Fast-Reroute[RFC5714] for unicast and multicast traffic. This draft proposes an architecture to provide 100% coverage for unicast traffic. The associated multicast architecture is described in [I-D.atlas-rtgwg-mrt-mc-arch].

Loop-free alternates (LFAs)[RFC5286] provide a useful mechanism for link and node protection but getting complete coverage is quite hard. [LFARevisited] defines sufficient conditions to determine if a network provides link-protecting LFAs and also proves that augmenting a network to provide better coverage is NP-hard. [I-D.ietf-rtgwg-lfa-applicability] discusses the applicability of LFA to different topologies with a focus on common PoP architectures.

While Not-Via [I-D.ietf-rtgwg-ipfrr-notvia-addresses] is defined as an architecture, in practice, it has proved too complicated and stateful to spark substantial interest in implementation or deployment. Academic implementations [LightweightNotVia] exist and have found the address management complexity high (but no standardization has been done to reduce this).

A different approach is needed and that is what is described here. It is based on the idea of using disjoint backup topologies as realized by Maximally Redundant Trees (described in [LightweightNotVia]); the general architecture can also apply to future improved redundant tree algorithms.

### 1.1. Goals for Extending IP Fast-Reroute coverage beyond LFA

Any scheme proposed for extending IPFRR network topology coverage beyond LFA, apart from attaining basic IPFRR properties, should also aim to achieve the following usability goals:

- ensure maximum physically feasible link and node disjointness regardless of topology,
- automatically compute backup next-hops based on the topology information distributed by link-state IGP,
- do not require any signaling in the case of failure and use pre-programmed backup next-hops for forwarding,
- introduce minimal amount of additional addressing and state on routers,

- o enable gradual introduction of the new scheme and backward compatibility,
- o and do not impose requirements for external computation.

## 2. Terminology

2-connected: A graph that has no cut-vertices. This is a graph that requires two nodes to be removed before the network is partitioned.

2-connected cluster: A maximal set of nodes that are 2-connected.

2-edge-connected: A network graph where at least two links must be removed to partition the network.

ADAG: Almost Directed Acyclic Graph - a graph that, if all links incoming to the root were removed, would be a DAG.

block: Either a 2-connected cluster, a cut-edge, or an isolated vertex.

cut-link: A link whose removal partitions the network. A cut-link by definition must be connected between two cut-vertices. If there are multiple parallel links, then they are referred to as cut-links in this document if removing the set of parallel links would partition the network.

cut-vertex: A vertex whose removal partitions the network.

DAG: Directed Acyclic Graph - a graph where all links are directed and there are no cycles in it.

GADAG: Generalized ADAG - a graph that is the combination of the ADAGs of all blocks.

Maximally Redundant Trees (MRT): A pair of trees where the path from any node X to the root R along the first tree and the path from the same node X to the root along the second tree share the minimum number of nodes and the minimum number of links. Each such shared node is a cut-vertex. Any shared links are cut-links. Any RT is an MRT but many MRTs are not RTs.

network graph: A graph that reflects the network topology where all links connect exactly two nodes and broadcast links have been transformed into the standard pseudo-node representation.

Redundant Trees (RT): A pair of trees where the path from any node X to the root R along the first tree is node-disjoint with the path from the same node X to the root along the second tree. These can be computed in 2-connected graphs.

## 3. Maximally Redundant Trees (MRT)

In the last few years, there's been substantial research on how to compute and use redundant trees. Redundant trees are directed spanning trees that provide disjoint paths towards their common root. These redundant trees only exist and provide link protection if the network is 2-edge-connected and node protection if the network is 2-connected. Such connectiveness may not be the case in real networks, either due to architecture or due to a previous failure. The work on maximally redundant trees has added two useful pieces that make them ready for use in a real network.

- Computable regardless of network topology: The maximally redundant trees are computed so that only the cut-edges or cut-vertices are shared between the multiple trees.

- Computationally practical algorithm is based on a common network topology database. Algorithm variants can compute in O( e) or O(e + n log n), as given in [I-D.enyedi-rtgwg-mrt-frr-algorithm].

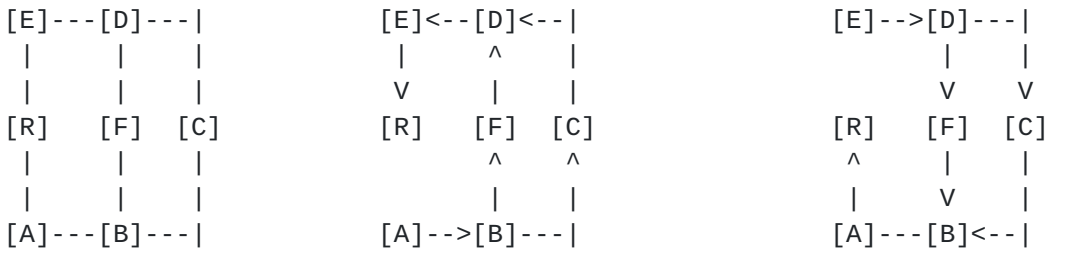
There is, of course, significantly more in the literature related to redundant trees and even fast-reroute, but the formulation of the Maximally Redundant Trees (MRT) algorithm makes it very well suited to use in routers.

A known disadvantage of MRT, and redundant trees in general, is that the trees do not necessarily provide shortest detour paths. The use of the shortest-path-first algorithm in tree-building and including all links in the network as possibilities for one path or another should improve this. Modeling is underway to investigate and compare the MRT alternates to the optimal [I-D.enyedi-rtgwg-mrt-frr-algorithm]. Providing shortest detour paths would require failure-specific detour paths to the destinations, but the state-reduction advantage of MRT lies in the detour being established per destination (root) instead of per destination AND per failure.

The specific algorithms to compute MRTs as well as the logic behind that algorithm and alternative computational approaches are given in detail in [I-D.enyedi-rtgwg-mrt-frr-algorithm]. Those interested are highly recommended to read that document. This document describes how the MRTs can be used and not how to compute them.

The most important thing to understand about MRTs is that for each pair of destination-routed MRTs, there is a path from every node X to the destination D on the Blue MRT that is as disjoint as possible from the path on the Red MRT. The two paths along the two MRTs to a given destination-root of a 2-connected graph are node-disjoint and link-disjoint, while in any non-2-connected graph, only the cut-vertices and cut-edges can be contained by both of the paths.

For example, in Figure 1, there is a network graph that is 2-connected in (a) and associated MRTs in (b) and (c). One can consider the paths from B to R; on the Blue MRT, the paths are B->F->D->E->R or B->C->D->E->R. On the Red MRT, the path is B->A->R. These are clearly link and node-disjoint. These MRTs are redundant trees because the paths are disjoint.




(a) a 2-connected graph

(b) Blue MRT towards R

(c) Red MRT towards R

Figure 1: A 2-connected Network

By contrast, in Figure 2, the network in (a) is not 2-connected. If F, G or the link F<->G failed, then the network would be partitioned. It is clearly impossible to have two link-disjoint or node-disjoint paths from G, I or J to R. The MRTs given in (b) and (c) offer paths that are as disjoint as possible. For instance, the paths from B to R are the same as in Figure 1 and the path from G to R on the Blue MRT is G->F->D->E->R and on the Red MRT is G->F->B->A->R.

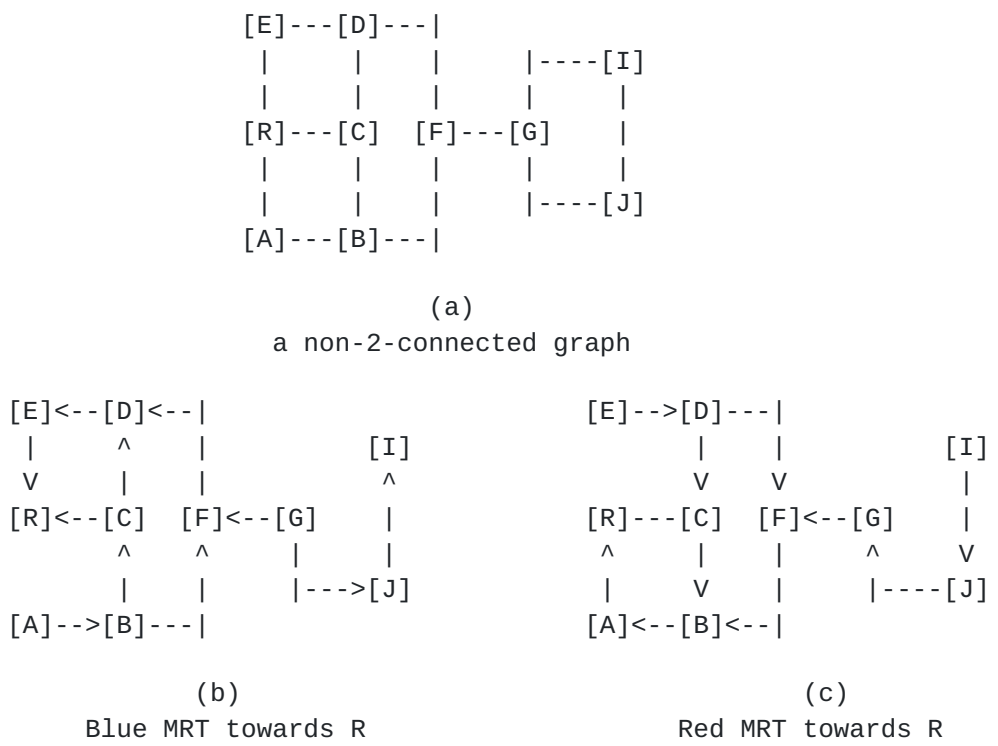



Figure 2: A non-2-connected network

## 4. Maximally Redundant Trees (MRT) and Fast-Reroute

In normal IGP routing, each router has its shortest-path-tree to all destinations. From the perspective of a particular destination, D, this looks like a reverse SPT (rSPT). To use maximally redundant trees, in addition, each destination D has two MRTs associated with it; by convention these will be called the blue and red MRTs.

Any IP/LDP fast-reroute technique beyond LFA requires an additional dataplane procedure, such as an additional forwarding mechanism. The well-known options are tunneling (e.g. [I-D.ietf-rtgwg-ipfrr-notvia-addresses]), per-interface forwarding (e.g. Loop-Free Failure Insensitive Routing in [EnyediThesis]), and multi-topology forwarding. MRT is realized by using multi-topology forwarding. There is a Blue MRT forwarding topology and a Red MRT forwarding topology.

MRTs are practical to maintain redundancy even after a single link or node failure. If a pair of MRTs is computed rooted at each destination, all the destinations remain reachable along one of the MRTs in the case of a single link or node failure.

When there is a link or node failure affecting the rSPT, each node will still have at least one path via one of the MRTs to reach the destination D. For example, in Figure 2, C would normally forward traffic to R across the C<->R link. If that C<->R link fails, then C could use either the Blue MRT path C->D->E->R or the Red MRT path C->B->A->R.

As is always the case with fast-reroute technologies, forwarding does not change until a local failure is detected. Packets are forwarded along the shortest path. The appropriate alternate to use is pre-computed. [I-D.enyedi-rtgwg-mrt-frr-algorithm] describes exactly how to determine whether the Blue MRT next-hops or the Red MRT next-hops should be the MRT alternate next-hops for a particular primary next-hop N to a particular destination D.

MRT alternates are always available to use, unless the network has been partitioned. It is a local decision whether to use an MRT alternate, a Loop-Free Alternate or some other type of alternate. When a network needs to use a micro-loop prevention mechanism [RFC5715] such as Ordered FIB[I-D.ietf-rtgwg-ordered-fib] or Farside Tunneling[RFC5715], then the whole IGP area needs to have alternates available so that the micro-loop prevention mechanism, which requires slower network convergence, can take the necessary time without impacting traffic badly.

As described in [RFC5286], when a worse failure than is anticipated happens, using LFAs that are not downstream neighbors can cause micro-looping. An example is given of link-protecting alternates causing a loop on node failure. Even if a worse failure than anticipated happened, the use of MRT alternates will not cause looping. Therefore, while node-protecting LFAs may be prefered, an advantage to using MRT alternates when such a node-protecting LFA is not a downstream path is the certainty that no alternate-induced looping will occur.

## 5. Unicast Forwarding with MRT Fast-Reroute

With LFA, there is no need to tunnel unicast traffic, whether IP or LDP. The traffic is simply sent to an alternate. As mentioned earlier in Section 4, MRT needs multi-topology forwarding. Unfortunately, neither IP nor LDP provide extra bits for a packet to indicate its topology.

Once the MRTs are computed, the two sets of MRTs are seen by the forwarding plane as essentially two additional topologies. The same considerations apply for forwarding along the MRTs as for handling multiple topologies.

### 5.1. LDP Unicast Forwarding - Avoid Tunneling

For LDP, it is very desirable to avoid tunneling because, for at least node protection, tunneling requires knowledge of remote LDP label mappings and thus requires targeted LDP sessions and the associated management complexity. There are two different mechanisms that can be used.

1. Option A - Encode MT-ID in Labels: In addition to sending a single label for a FEC, a router would provide two additional labels with the MT-IDs associated with the Blue MRT or Red MRT forwarding topologies. This is very simple for hardware support. It does reduce the label space for other uses. It also increases the memory to store the labels and the communication required by LDP.

2. Option B - Create Topology-Identification Labels: Use the label-stacking ability of MPLS and specify only two additional labels - one for each associated MRT color - by a new FEC type. When sending a packet onto an MRT, first swap the LDP label and then push the topology-identification label for that MRT color. When receiving a packet with a topology-identification label, pop it and use it to guide the next-hop selection in combination with the next label in the stack; then swap the remaining label, if appropriate, and push the topology-identification label for the next-hop. This has minimal usage of additional labels, memory and LDP communication. It does increase the size of packets and the complexity of the required label operations and look-ups. This can use the same mechanisms as are needed for context-aware label spaces.

Note that with LDP unicast forwarding, regardless of whether topology-identification label or encoding topology in label is used, no additional loopbacks per router are required. This is because LDP labels are used on a hop-by-hop basis to identify MRT-blue and MRT-red forwading topologies.

For greatest hardware compatibility, routers implementing MRT LDP fast-reroute MUST support Option A of encoding the MT-ID in the labels. The extensions to indicate an MT-ID for a FEC are described in Section 3.2.1 of [I-D.ietf-mpls-ldp-multi-topology]

### 5.2. IP Unicast Traffic

For IP, there is no currently practical alternative except tunneling. The tunnel egress could be the original destination in the area, the next-next-hop, etc.. If the tunnel egress is the original destination router, then the traffic remains on the redundant tree

with sub-optimal routing. If the tunnel egress is the next-next-hop, then protection of multi-homed prefixes and node-failure for ABRs is not available. Selection of the tunnel egress is a router-local decision.

There are three options available for marking IP packets with which MRT it should be forwarded in.

1. Tunnel IP packets via an LDP LSP. This has the advantage that more installed routers can do line-rate encapsulation and decapsulation. Also, no additional IP addresses would need to be allocated or signaled.

    A. Option A - LDP Destination-Topology Label: Use a label that indicates both destination and MRT. This method allows easy tunneling to the next-next-hop as well as to the IGP-area destination. For multi-homed prefixes, this requires that additional labels be advertised for each proxy-node.

    B. Option B - LDP Topology Label: Use a Topology-Identifier label on top of the IP packet. This is very simple and doesn't require additional labels for proxy-nodes. If tunneling to a next-next-hop is desired, then a two-deep label stack can be used with [ Topology-ID label, Next-Next-Hop Label ].

2. Tunnel IP packets in IP. Each router supporting this option would announce two additional loopback addresses and their associated MRT color. Those addresses are used as destination addresses for MRT-blue and MRT-red IP tunnels respectively. They allow the transit nodes to identify the traffic as being forwarded along either MRT-blue or MRT-red tree topology to reach the tunnel destination. Announcements of these two additional loopback addresses per router with their MRT color requires IGP extensions.

For greatest hardware compatibility and ease in removing the MRT-topology marking at area/level boundaries, routers that support MPLS and implement IP MRT fast-reroute SHOULD support Option A - using an LDP label that indicates the destination and MT-ID.

For proxy-nodes associated with one or more multi-homed prefixes, there is no router associated with the proxy-node, so its loopbacks can't be known or used. Instead, the loopback addresses of the two routers that are attached to the proxy-node can be used. One of those routers will be on the Red MRT and the other on the Blue MRT. The MRT-red loopback of the first router would be used to reach the router on the Red MRT and similarly the MRT-blue loopback of the

second router would be used. The routers connected to the proxy-node are the end of the area/level and can decapsulate the traffic and properly forward it into the next area.

## 6. Protocol Extensions and Considerations: OSPF and ISIS

This captures an initial understanding of what may need to be specified. In cases of partial deployment, it is necessary for a router to determine a consistent set of routers to include in the island of MRT support. To facilitate this, each router can announce both what its capabilities are and what it requires from other routers to add them to the MRT island. Generally, there will be a set of information advertised about the MRT support. This information has only area/level-wide scope.

MRT Island Creation ID: This identifies the process that the router uses to form an MRT Island. By advertising an ID for the process, it is possible to have different processes in the future. It may be desirable to advertise a list ordered by preference to allow transitions.

MRT Algorithm ID: This identifies the particular MRT algorithm used by the router. By having an Algorithm ID, it is possible to change the algorithm used or use different ones in different networks. It may be desirable to advertise a list ordered by preference to allow transitions.

Red MRT MT-ID: This specifies the MT-ID to be associated with the Red MRT forwarding topology. It is needed for use in signaling. All routers in the MRT Island MUST agree on a value.

Blue MRT MT-ID: This specifies the MT-ID to be associated with the Blue MRT forwarding topology. It is needed for use in signaling. All routers in the MRT Island MUST agree on a value.

GADAG Root Election Priority: This specifies the priority of the router for being used as the GADAG root of its island. A GADAG root is elected from the set of routers with the highest priority; ties are broken based upon highest Router ID. The sensitivity of the MRT Algorithms to GADAG root selection is still being evaluated. This provides the network operator with a knob to force particular GADAG root selection.

Forwarding Mechanism for IP: This specifies which forwarding mechanisms the router supports for IP traffic. An MRT island must support a common set of forwarding mechanisms, which may be less than the full set advertised. Multiple forwarding mechanisms may

be specified, such as IP-in-IPv4, IP-in-IPv6 or IP-in-LDP-Destination-Topology Label. None is also an option.

Forwarding Mechanism for LDP: This specifies which forwarding mechanisms the router supports for LDP traffic. An MRT island must support a common set of forwarding mechanisms, which may be less than the full set advertised. The expected mechanisms are "Encode MT-ID in Labels" or None.

Red MRT Loopback Address: This provides the router's loopback address to reach the router via the Red MRT forwarding topology. It can, of course, be specified for both IPv4 and IPv6.

Blue MRT Loopback Address: This provides the router's loopback address to reach the router via the Blue MRT forwarding topology. It can, of course, be specified for both IPv4 and IPv6.

MRT Capabilities Available: This is the set of capabilities that the router is configured to support.

MRT Capabilities Required: This is the set of capabilities that other routers must have available to be added into the MRT island.

MRT Capability: Computes MRTs: The router can compute MRTs.

MRT Capability: IP Fast-Reroute: The router can use the computed MRTs for IP fast-reroute.

MRT Capability: LDP Fast-Reroute: The router can use the computed MRTs for LDP fast-reroute.

MRT Capability: PIM Fast-Reroute: The router can use the computed MRTs for PIM fast-reroute.

MRT Capability: mLDP Fast-Reroute: The router can use the computed MRTs for mLDP fast-reroute.

MRT Capability: PIM Global Protection: The router can use the computed MRTs for PIM Global Protection 1+1.

MRT Capability: mLDP Global Protection: The router can use the computed MRTs for mLDP Global Protection 1+1.

The assumption is that a router will form 1 MRT island, compute MRTs within that island, and then use those MRTs for the different purposes. Including a router that, for instance, doesn't support mLDP Global Protection would mean that the whole MRT island could not support it. In a fully deployed case, of course, the whole area/

level would support MRT and the complexities of MRT island formation would be minimal.

If a router wanted to form multiple MRT islands for different application purposes, that could be done by specifying different Red MRT MT-ID and Blue MRT MT-IDs.

As with LFA, it is expected that OSPF Virtual Links will not be supported.

## 7. Multi-homed Prefixes

One advantage of LFAs that is necessary to preserve is the ability to protect multi-homed prefixes against ABR failure. For instance, if a prefix from the backbone is available via both ABR A and ABR B, if A fails, then the traffic should be redirected to B. This can also be done for backups via MRT.

This generalizes to any multi-homed prefix. A multi-homed prefix could be:

- An out-of-area prefix announced by more than one ABR,
- An AS-External route announced by 2 or more ASBRs,
- A prefix with iBGP multipath to different ASBRs,
- etc.

For each prefix, the two lowest total cost ABRs are selected and a proxy-node is created connected to those two ABRs. If there exist multiple multi-homed prefixes that share the same two best connectivity, then a single proxy-node can be used to represent the set. An example of this is shown in Figure 3.

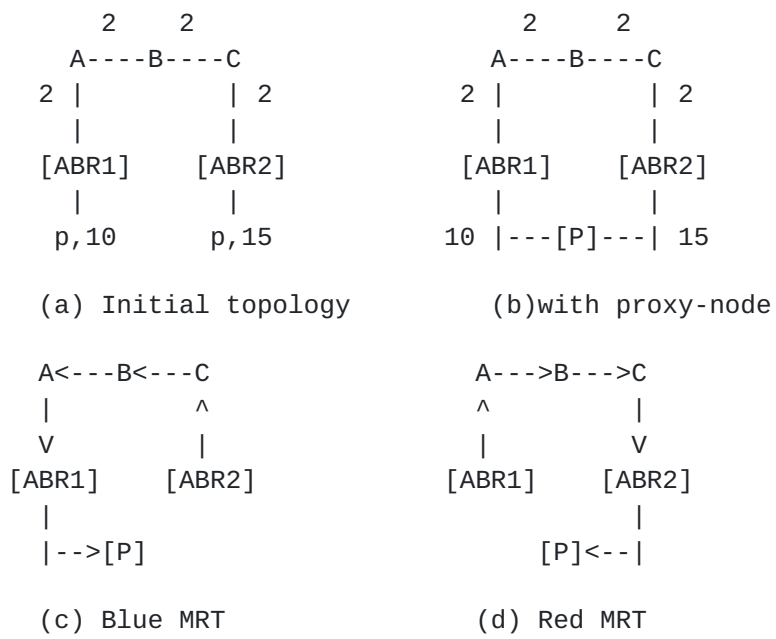



Figure 3: Prefixes Advertised by Multiple ABRs

The proxy-nodes and associated links are added to the network topology after all real links have been assigned to a direction and before the actual MRTs are computed. Proxy-nodes cannot be transited when computing the MRTs. In addition to computing the pair of MRTs associated with each router destination D in the area, a pair of MRTs can be computed for each such proxy-node to fully protect against ABR failure.

Each ABR or attaching router must remove the MRT marking[see Section 5] and then forward the traffic outside of the area (or island of MRT-fast-reroute-supporting routers).

If ASBR protection is desired, this has additonal complexities if the ASBRs are in different areas. Similarly, protecting labeled BGP traffic in the event of an ASBR failure has additional complexities due to the per-ASBR label spaces involved.

## 8. Inter-Area and ABR Forwarding Behavior

In regular forwarding, packets destined outside the area arrive at the ABR and the ABR forwards them into the other area because the next-hops from the area with the best route (according to tie-breaking rules) are used by the ABR. The question is then what to do with packets marked with an MRT that are received by the ABR.

For unicast fast-reroute, the need to stay on an MRT forwarding topology terminates at the ABR/LBR whose best route is via a different area/level. It is highly desirable to go back to the default forwarding topology when leaving an area/level. There are three basic reasons for this. First, the default topology uses shortest paths; the packet will thus take the shortest possible route to the destination. Second, this allows failures that might appear in multiple areas (e.g. ABR/LBR failures) to be separately identified and repaired around. Third, the packet can be fast-rerouted again, if necessary, due to a failure in a different area.

An ABR/LBR that receives a packet marked with an MRT towards a destination in another area should forward the MRT marked packet in the area with the best route along its associated MRT. If the packet came from that area, this correctly avoids the failure.

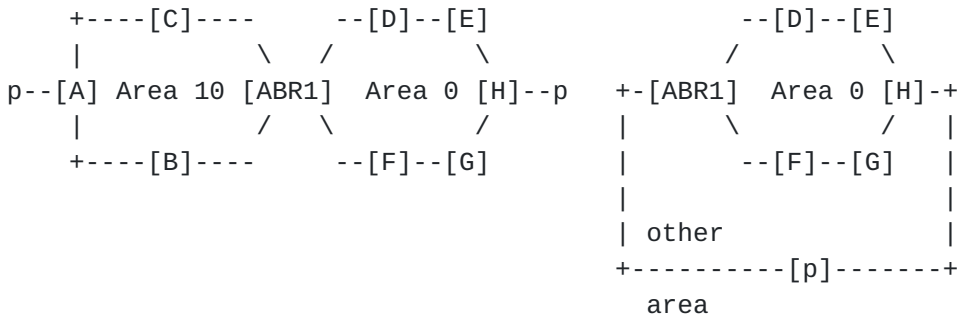
How does an ABR/LBR ensure that MRT-marked packets do not arrive at the ABR/LBR? There are two different mechanisms depending upon the forwarding mechanism being used.

If the LDP label encodes the MT-ID as well as the destination, then the ABR/LBR is responsible for advertising a particular label to each neighbor. Additionally, an LDP label is associated with an MT-ID due to the MT FEC that was used and not due to any intrisic particular value for the label. Assume that an ABR/LBR has allocated three labels for a particular destination; those labels are L_primary, L_blue, and L_red. When the ABR/LBR advertises label bindings to routers in the area with the best route to the destination, the ABR/LBR provides L_primary for the default topology, L_blue for the Blue MRT MT-ID and L_red for the Red MRT MT-ID, exactly as expected. However, when the ABR/LBR advertises label bindings to routers in other areas, the ABR/LBR advertises L_primary for the default topology, for the Blue MRT MT-ID, and for the Red MRT MT-ID. The ABR/LBR installs next-hops from the best area for L_primary based on the default topology, for L_blue based on the Blue MRT forwarding topology, and for L_red based on the Red MRT forwarding topology. Therefore, packets from the non-best area will arrive at the ABR/LBR with a label L_primary and will be forwarded into the best area along the default topology. By controlling what labels are advertised, the ABR/LBR can thus enforce that packets exiting the area do so on the shortest-path default topology.

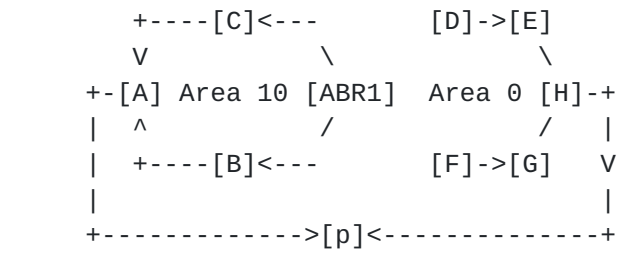
If IP-in-IP forwarding is used, then the ABR/LBR behavior is dependent upon the outermost IP address. If the outermost IP address is an MRT loopback address of the ABR/LBR, then the packet is decapsulated and forwarded based upon the inner IP address, which should go on the default SPT topology. If the outermost IP address is not an MRT loopback address of the ABR/LBR, then the packet is

simply forwarded along the associated forwarding topology. A PLR sending traffic to a destination outside its local area/level will pick the MRT and use the associated MRT loopback address of the ABR/LBR immediately before the proxy-node on that MRT.

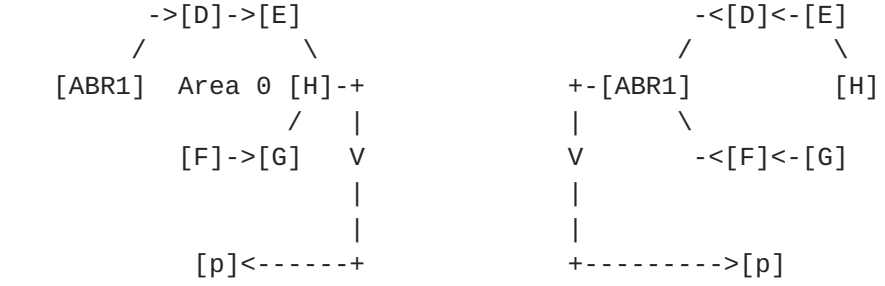
Thus, regardless of which of these two forwarding mechanisms are used, there is no need for additional computation or per-area forwarding state.




(a) Example topology (b) Proxy node view in Area 0 nodes




(c) rSPT towards destination p




(d) Blue MRT in Area 0 (e) Red MRT in Area 0

Figure 4: ABR Forwarding Behavior and MRTs

The other potential forwarding mechanisms require additional computation by the penultimate router along the in-local-area MRT immediately before the ABR/LBR is reached. The penultimate router can determine that the ABR/LBR will forward the packet out of area/level and, in that case, the penultimate router can remove the MRT marking but still forward the packet along the MRT next-hop to reach the ABR. For instance, in Figure 4, if node H fails, node E has to put traffic towards prefix p onto the red MRT. But since node D knows that ABR1 will use a best from another area, it is safe for D to remove the MRT marking and just send the packet to ABR1 still on the red MRT but unmarked. ABR1 will use the shortest path in Area 10.

In all cases for ISIS and most cases for OSPF, the penultimate router can determine what decision the adjacent ABR will make. The one case where it can't be determined is when two ASBRs are in different non-backbone areas attached to the same ABR, then the ASBR's Area ID may be needed for tie-breaking (prefer the route with the largest OPSF area ID) and the Area ID isn't announced as part of the ASBR link-state advertisement (LSA). In this one case, suboptimal forwarding along the MRT in the other area would happen. If this is a realistic deployment scenario, OSPF extensions could be considered.

## 9. Issues with Area Abstraction

MRT fast-reroute provides complete coverage in a area that is 2-connected. Where a failure would partition the network, of course, no alternate can protect against that failure. Similarly, there are ways of connecting multi-homed prefixes that make it impractical to protect them without excessive complexity.

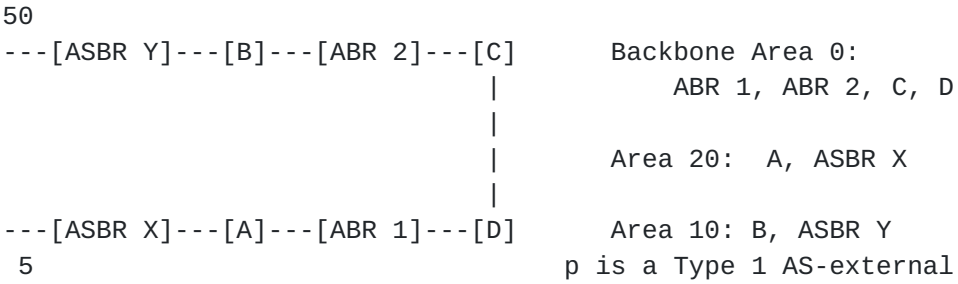



Figure 5: AS external prefixes in different areas

Consider the network in Figure 5 and assume there is a richer connective topology that isn't shown, where the same prefix is announced by ASBR X and ASBR Y which are in different non-backbone areas. If the link from A to ASBR X fails, then an MRT alternate

could forward the packet to ABR 1 and ABR 1 could forward it to D, but then D would find the shortest route is back via ABR 1 to Area 20. The only real way to get it from A to ASBR Y is to explicitly tunnel it to ASBR Y.

Tunnelling to the backup ASBR is for future consideration. The previously proposed PHP approach needs to have an exception if BGP policies (e.g. BGP local preference) determines which ASBR to use. Consider the case in Figure 6. If the link between A and ASBR X (the preferred border router) fails, A can put the packets to p onto an MRT alternate, even tunnel it towards ASBR Y. Node B, however, must not remove the MRT marking in this case, as nodes in Area 0, including ASBR Y itself would not know that their preferred ASBR is down.

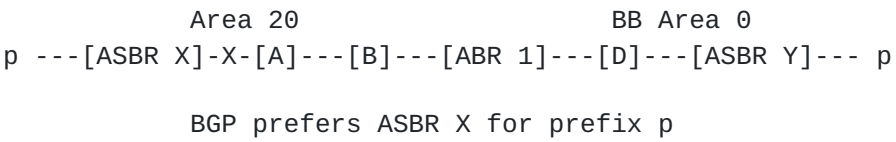



Figure 6: Failure of path towards ASBR preferred by BGP

The fine details of how to solve multi-area external prefix cases, or identifying certain cases as too unlikely and too complex to protect is for further consideration.

## 10. Partial Deployment and Islands of Compatible MRT FRR routers

A natural concern with new functionality is how to have it be useful when it is not deployed across an entire IGP area. In the case of MRT FRR, where it provides alternates when appropriate LFAs aren't available, there are also deployment scenarios where it may make sense to only enable some routers in an area with MRT FRR. A simple example of such a scenario would be a ring of 6 or more routers that is connected via two routers to the rest of the area.

First, a computing router S must determine its local island of compatible MRT fast-reroute routers. A router that has common forwarding mechanisms and common algorithm and is connected to either to S or to another router already determined to be in S's local island can be added to S's local island.

Destinations inside the local island can obviously use MRT alternates. Destinations outside the local island can be treated like a multi-homed prefix with caveats to avoid looping. For LDP

labels including both destination and topology, the routers at the borders of the local island need to originate labels for the original FEC and the associated MRT-specific labels. Packets sent to an LDP label marked as blue or red MRT to a destination outside the local island will have the last router in the local island swap the label to one for the destination and forward the packet along the outgoing interface on the MRT towards a router outside the local island that was represented by the proxy-node.

For IP in IP encapsulations, remote destinations' loopback addresses for the MRTs cannot be used, even if they were available. Instead, the MRT loopback address of the router attached to a proxy-node, which represents destinations outside the local island, can be used. Packets sent to the router's MRT loopback address will have their outer IP header removed and will need to be explicitly forwarded along the outgoing interface on the MRT towards a router outside the local island that was represented by the proxy-node. This behavior requires essentially remembering the MT-ID indicated by the outer IP address. An alternate option would be to advertise different loopback addresses to be associated with the proxy-node; the outer IP address would still be removed but it would indicate the outgoing interface to use and no lookup would be necessary on the internal IP address while maintaining MT-ID context.

A key question is which routers outside the MRT island can packets be forwarded to so that they are not forwarded back into the MRT island. An example of the necessary network graph transformations are given in Figure 7. There are two parts to the computation. First, the MRT island is collapsed into a single node; this assumes that the cost of transiting the MRT island is nothing and is pessimistic but allows for simpler computation. Then, for each destination (other than the MRT island), the routers adjacent to the MRT island are checked to see if they are loop-free with respect to the MRT island and the destination. The two loop-free neighbors of the MRT island that are closest to the destination are selected. Then, a graph of just the MRT island is augmented with proxy-nodes that are attached via the outgoing interfaces to the selected loop-free neighbors. Finally, the MRTs rooted at each proxy-node are computed on that augmented MRT island graph. Essentially, the MRT island must have a loop-free neighbor to be able to have an alternate.

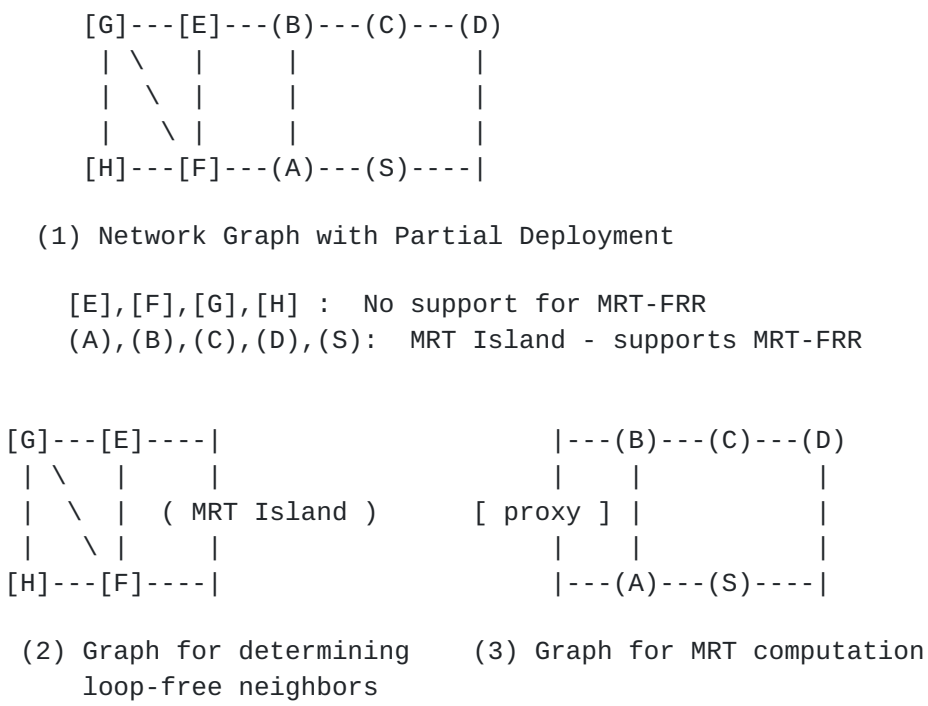



Figure 7: Computing alternates to destinations outside the MRT Island

Naturally, there are more complicated options to improve coverage, such as connecting multiple MRT islands across tunnels, but it is not clear that the additional complexity is necessary.

## 11. Network Convergence and Preparing for the Next Failure

After a failure, MRT detours ensure that packets reach their intended destination while the IGP has not reconverged onto the new topology. As link-state updates reach the routers, the IGP process calculates the new shortest paths. Two things need attention: micro-loop prevention and MRT re-calculation.

### 11.1. Micro-forwarding loop prevention and MRTs

As is well known[RFC5715], micro-loops can occur during IGP convergence; such loops can be local to the failure or remote from the failure. Managing micro-loops is an orthogonal issue to having alternates for local repair, such as MRT fast-reroute provides.

There are two possible micro-loop prevention mechanism discussed in [RFC5715]. The first is Ordered FIB [I-D.ietf-rtgwg-ordered-fib]. The second is Farside Tunneling which requires tunnels or an alternate topology to reach routers on the farside of the failure.

Since MRTs provide an alternate topology through which traffic can be sent and which can be manipulated separately from the SPT, it is possible that MRTs could be used to support Farside Tunneling. Details of how to do so are outside of this document.

### 11.2. MRT Recalculation

When a failure event happens, traffic is put by the PLRs onto the MRT topologies. After that, each router recomputes its shortest path tree (SPT) and moves traffic over to that. Only after all the PLRs have switched to using their SPTs and traffic has drained from the MRT topologies should each router install the recomputed MRTs into the FIBs.

At each router, therefore, the sequence is as follows:

1. Receive failure notification

2. Recompute SPT

3. Install new SPT

4. Recompute MRTs

5. Wait configured period for all routers to be using their SPTs and traffic to drain from the MRTs.

6. Install new MRTs.

While the recomputed MRTs are not installed in the FIB, protection coverage is lowered. Therefore, it is important to recalculate the MRTs and install them as quickly as possible.

The installation of the MRTs can be staged such that the affected or broken MRTs are updated first and then the unbroken.

## 12. Acknowledgements

The authors would like to thank Hannes Gredler, Jeff Tantsura, Ted Qian, Kishore Tiruveedhula, Santosh Esale, Nitin Bahadur, Harish Sitaraman and Raveendra Torvi for their suggestions and review.

## 13. IANA Considerations

This doument includes no request to IANA.

## 14. Security Considerations

This architecture is not currently believed to introduce new security concerns.

## 15. References

### 15.1. Normative References

[I-D.enyedi-rtgwg-mrt-frr-algorithm]
Enyedi, G., Atlas, A. and A. Csaszar, "Algorithms for computing Maximally Redundant Trees for IP/LDP Fast-Reroute", draft-enyedi-rtgwg-mrt-frr-algorithm-01 (work in progress), March 2012.

[RFC5286] Atlas, A. and A. Zinin, "Basic Specification for IP Fast Reroute: Loop-Free Alternates", RFC 5286, September 2008.

[RFC5714] Shand, M. and S. Bryant, "IP Fast Reroute Framework", RFC 5714, January 2010.

### 15.2. Informative References

[EnyediThesis]
Enyedi, G., "Novel Algorithms for IP Fast Reroute", Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics Ph.D. Thesis, February 2011, <http://timon.tmit.bme.hu/theses/thesis_book.pdf>.

[I-D.atlas-rtgwg-mrt-mc-arch]
Atlas, A., Kebler, R., Wijnands, I., Csaszar, A., and G. Envedi, "An Architecture for Multicast Protection Using Maximally Redundant Trees", draft-atlas-rtgwg-mrt-mc-arch-00 (work in progress), March 2012.

[I-D.ietf-mpls-ldp-multi-topology]
Zhao, Q., Fang, L., Zhou, C., Li, L., and N. So, "LDP Extensions for Multi Topology Routing", draft-ietf-mpls-ldp-multi-topology-03 (work in progress), March 2012.

[I-D.ietf-rtgwg-ipfrr-notvia-addresses]
Bryant, S., Previdi, S., and M. Shand, "IP Fast Reroute Using Not-via Addresses", draft-ietf-rtgwg-ipfrr-notvia-addresses-08 (work in

progress), December 2011.

[I-D.ietf-rtgwg-lfa-applicability]
Filsfils, C. and P. Francois, "LFA applicability in SP networks", draft-ietf-rtgwg-lfa-applicability-06 (work in progress), January 2012.

[I-D.ietf-rtgwg-ordered-fib]
Shand, M., Bryant, S., Previdi, S., and C. Filsfils, "Loop-free convergence using oFIB", draft-ietf-rtgwg-ordered-fib-05 (work in progress), April 2011.

[LFARevisited]
Retvari, G., Tapolcai, J., Enyedi, G., and A. Csaszar, "IP Fast ReRoute: Loop Free Alternates Revisited", Proceedings of IEEE INFOCOM , 2011, <http://opti.tmit.bme.hu/~tapolcai/papers/retvari2011lfa_infocom.pdf>.

[LightweightNotVia]
Enyedi, G., Retvari, G., Szilagyi, P., and A. Csaszar, "IP Fast ReRoute: Lightweight Not-Via without Additional Addresses", Proceedings of IEEE INFOCOM , 2009, <http://mycite.omikk.bme.hu/doc/71691.pdf>.

[RFC5715] Shand, M. and S. Bryant, "A Framework for Loop-Free Convergence", RFC 5715, January 2010.

## Authors' Addresses

Alia Atlas (editor)
Juniper Networks
10 Technology Park Drive
Westford, MA 01886
USA

Email: akatlas@juniper.net

Robert Kebler
Juniper Networks
10 Technology Park Drive
Westford, MA 01886
USA

Email: rkebler@juniper.net

Gabor Sandor Enyedi
Ericsson
Konyves Kalman krt 11.
Budapest 1097
Hungary

Email: Gabor.Sandor.Enyedi@ericsson.com

Andras Csaszar
Ericsson
Konyves Kalman krt 11
Budapest 1097
Hungary

Email: Andras.Csaszar@ericsson.com

Maciek Konstantynowicz
Cisco Systems

Email: maciek@bgp.nu

Russ White
Cisco Systems

Email: russwh@cisco.com

Mike Shand

Email: mike@mshand.org.uk