APPENDIX A

LISP Host Mobility Deployment Best Practices

This appendix presents some design best practices and recommendations when deploying the LISP Host Mobility solution between data center sites equipped with Nexus 7000 switches. When not specified otherwise, the assumption is that the recommendation applies to both LISP Host Mobility deployment models (with Extended Subnet and Across Subnets).

LISP and MTU Considerations

Figure 2-2 displayed how 36 extra bytes are added when encapsulating an IP packet to be sent via LISP across an IPv4 transport infrastructure. Given the original IP header of the packet, the consequence is that the largest IP payload that can be sent without requiring any fragmentation is (1500 – 36 – 20) = 1444 Bytes.

The current behavior on Nexus 7000 is to drop IP packets larger than 1444 Bytes, without performing LISP encapsulation. If the DF bit is set, the xTR will also generate an ICMP Destination Unreachable message (type 3, code 4) with a code meaning "fragmentation needed and DF set" and will send it back to the source of the packet (as specified in the original IP header).

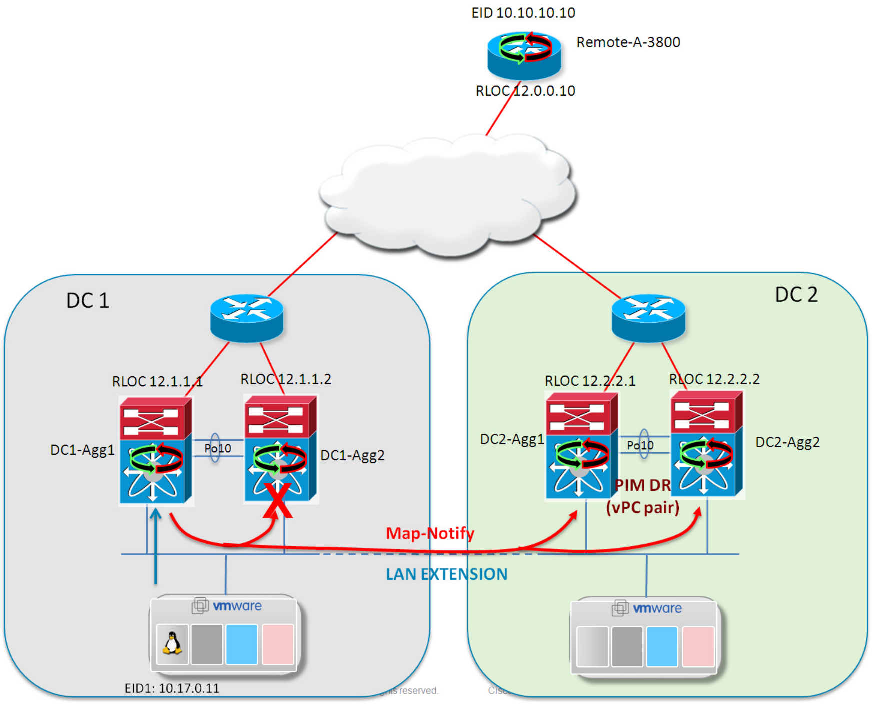
The behavior of packets being dropped independently of the available MTU of the L3 links connecting the xTR to the L3 domain. This means that even if Jumbo frame support is configured on these interfaces, the xTR would not perform the LISP encapsulation and simply discard the traffic. As a consequence, it is required to ensure that the source of the traffic can adjust the MTU based on the received ICMP message, or that the original MTU of the servers is set lower than 1444 Bytes.

LISP Host Mobility with Extended Subnet and PIM Interaction

When deploying LISP Host Mobility with Extended Subnet, it is important to pay attention to a specific interaction between LISP and PIM. It may happen that the L3 interface (SVI) where LISP mobility is enabled also requires having PIM enabled. This is for example the case if L3 multicast traffic was sourced or received on that subnet even before deploying LISP mobility.

With LAN extension deployed between DC sites, we end up having multiple PIM enabled routers sitting on the same subnet and only one of them (or a pair when vPC is deployed) is elected as PIM Designated Router (DR). With the current NX-OS implementation only the PIM DR is capable of receiving and punting to the LISP process the Map-notify-group message generated by an xTR after an EID discovery. This would create problems in the scenarios below.

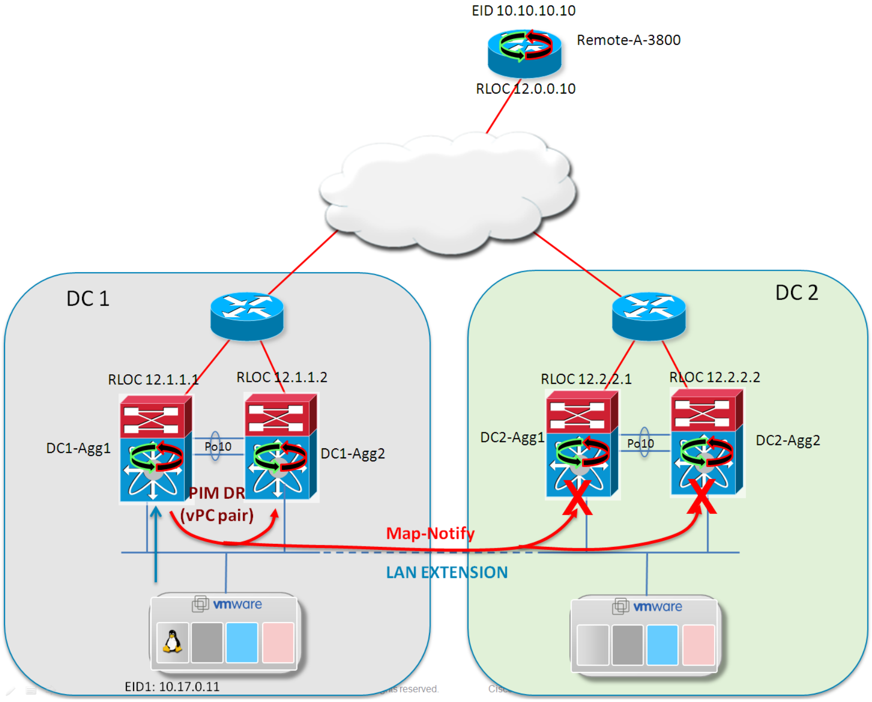
Figure A-1 PIM DR Generating the Map-Notify Message



In Figure A-1, EID 10.17.0.11 is discovered in DC1, where the PIM DR is deployed (both xTRs are DR since vPC is used in this example to connect to the access layer devices). The discovering xTR generates the Map-Notify message and sends it to all other xTRs via the extended LAN connection. Only the local peer xTR is able to punt that frame to the LISP process, whereas the two xTRs in DC2 will not be able to do so, with the end result that no Null0 entry will be added to their routing tables (or no valid /32 entry will be removed if the EID was previously located in DC2).

A variation of the same problem is shown in Figure A-2:

Figure A-2 PIM DR Receiving the Map-Notify Message



In this case, the PIM DR is deployed in the DC2 site where the EID is not discovered. The end result is that xTRs in DC2 will be able to receive and process the Map-Notify message, but that won't be the case for the second xTR in DC1 (DC1-Agg2), creating inconsistent information in the dynamic EID tables of the two xTRs in that site.

Note This issue does not apply to LISP Across Subnet Mode deployments when leveraging vPC to connect the xTRs to the edge switches, since both xTRs devices in each site always perform the DR function.

CSCtz22163 has been opened to track this issue, which will be fixed in 6.2 NX-OS release. In the meantime a workaround is available, consisting in the definition of a PIM enabled loopback interface on each xTR configured to join each Map-Notify multicast group specified in the LISP Host Mobility configuration:

On all LISP DC xTRs

```
interface loopback 1
 ip address a.b.c.d/32
 ip pim sparse-mode
 ip igmp join-group <map-notify-group1>
 ip igmp join-group <map-notify-group2>
 ip igmp join-group <map-notify-group3>
```

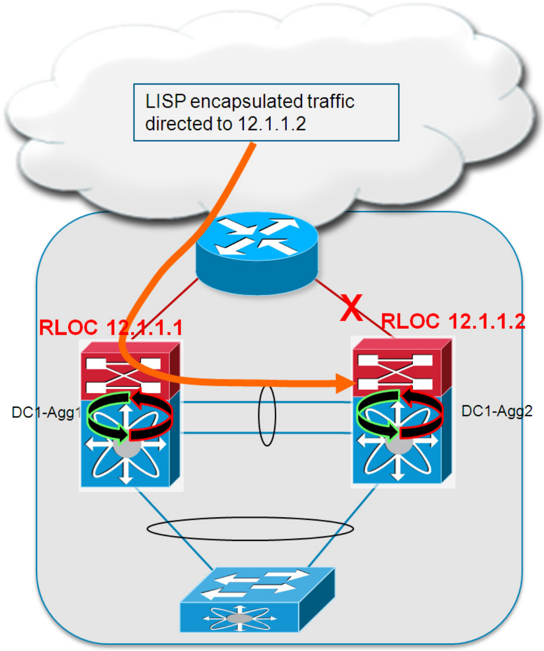
The configuration of the loopback interface ensures that every map-notify message received by the xTR device will always be sent to the LISP process, independently from the fact that the device is operating as DR or not for the extended subnet.

Note It is possible to leverage for this purpose the same loopback already defined as RLOC.

Establishing L3 Peering between LISP DC xTR Devices

A dedicated L3 link must be used between the xTRs deployed at the aggregation layer to establish L3 peering. Also, the link must terminate on M1-32 cards. This specific design consideration is highlighted in Figure A-3.

Figure A-3 Routing LISP Encapsulated Traffic between xTRs

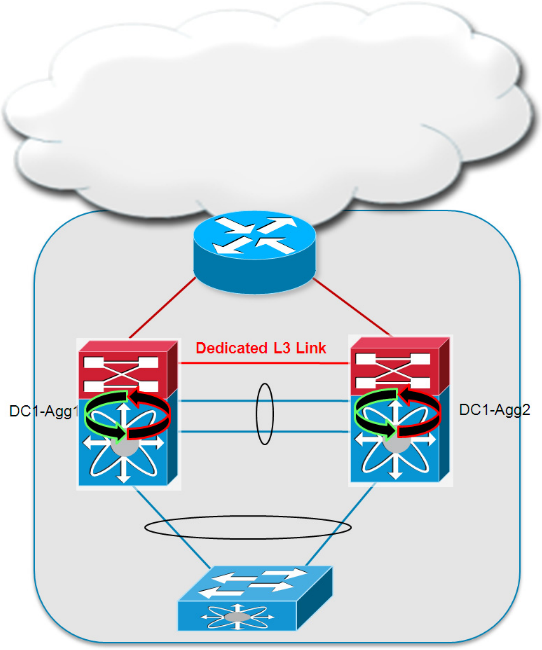


A LISP encapsulated flow is arriving into DC1 directed to the RLOC 12.1.1.2 identifying DC1-Agg2 xTR. If that xTR loses connectivity to the L3 DC core, the only remaining L3 path from the core to the RLOC is via the transit link connecting the two xTRs. The transit link is usually a vPC peer-link configured as a L2 trunk and it is common practice to leverage a dedicate VLAN to establish a L3 peering (SVI-to-SVI) between the xTRs. When LISP encapsulated traffic is re-routed across the transit link, two scenarios are possible:

- The transit link is implemented with interfaces belonging to M1-32 linecards: in this case, the traffic cannot be LISP de-capsulated causing the black holing of all the traffic destined to 12.1.1.2.
- The transit link is implemented with interfaces belonging to F1 linecards: in this case the traffic is de-capsulated in SW once it reaches DC1-Agg2.

Both scenarios above are obviously undesirable, so the recommended workaround is to leverage a dedicated routed interface (or routed port-channel) to establish the L3 peering between xTRs, as shown in Figure A-4.

Figure A-4 Leveraging a Dedicated L3 Connection Between xTR Devices



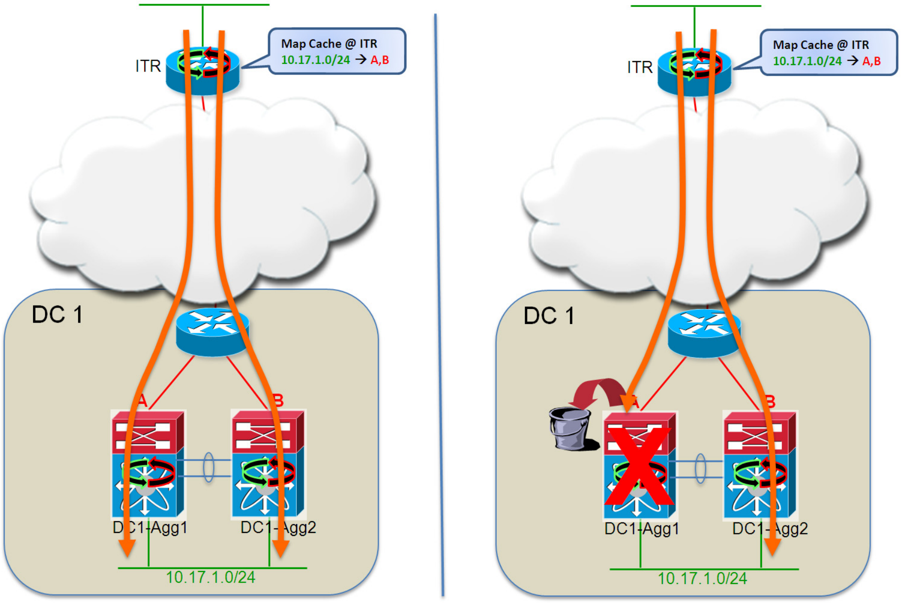
Note Since this link can be used to send/receive LISP encapsulated traffic, it is mandatory to leverage M1-32 interfaces. No other interfaces (other M1 modules, F2, M2) are supported for this function.

Dealing with an ETR Failure Scenario

Given the fact that communication between ITR and ETR happens in an overlay fashion, one important thing to consider is how to detect a remote (and indirect) ETR failure, to avoid the black holing of the traffic.

Figure A-5 highlights the problem, focusing on traffic flows exchanged between a remote ITR and two DC ETRs (the same considerations apply to the opposite direction).

Figure A-5 ETR Failure Scenario



On the left side we notice how traffic is normally load-balanced across both DC ETRs, leveraging the mapping information on the ITR associating the DC EID subnet to the two RLOCs A and B. As previously mentioned, the load balancing is happening assuming that the priority and weight parameters configured on the DC xTRs are matching. In that case, load balancing is performed on a per-flow basis, depending on the L2, L3 and L4 parameters of the original flow.

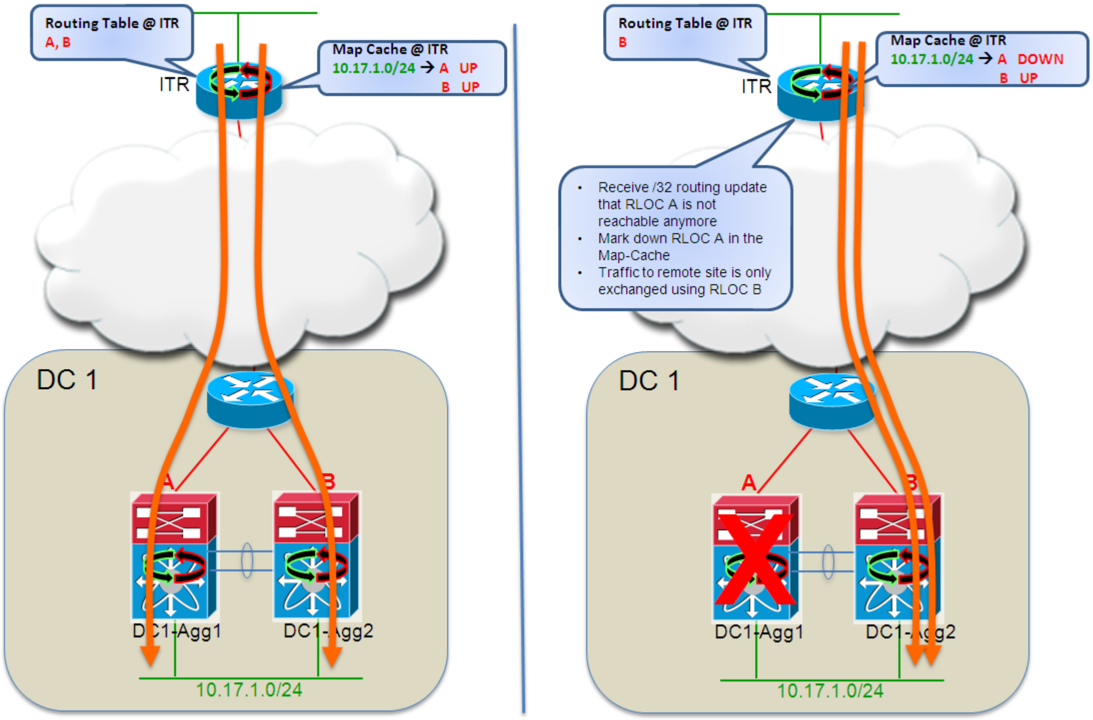
On the right we show the traffic behavior after the failure of one of the DC xTRs: since the map-cache information on the ITR remains valid by default for 24 hours (this value could be tune down to 15 minutes if needed), the ITR keeps using the RLOC A to send traffic to the 10.17.1.0/24 subnet, causing the black-holing of the traffic.

To avoid this issue, it is required to dynamically update the ITR map-cache information, so that RLOC A can be marked as unusable and all the traffic can be directed to RLOC B. Three mechanisms can currently be leveraged for this purpose: they are described below with relative recommendation and deployment considerations.

1. Leveraging specific RLOC prefix updates

   The basic assumption for this method is that specific /32 RLOC prefixes can be exchanged between ITR and ETR, leveraging the deployed routing protocol (IGP or BGP), as shown in Figure A-6.

Figure A-6 Marking RLOC Unreachable after Routing Update



In this case, the failure of the DC xTR would trigger a specific routing withdrawal for RLOC A from the ITR routing table. Once the ITR receives the routing update, it can immediately mark the RLOC as DOWN in the map-cache, causing the recovery of the original flow via RLOC B. It is important to clarify how this mechanism can currently be leveraged if the ITR has the specific /32 routing information relative to the remote RLOC in its routing table. This means that we need to ensure that the remote RLOC address is not part of an aggregate subnet advertised from the DC site toward the remote location and that specific /32 prefixes can be injected into the core of the network.

The advantages of this solution are:

- Achieve very fast (sub-second) traffic recovery when an IGP is deployed between ITR and ETRs.
- Dynamic solution that does not require any specific configuration.

For what concerns the drawbacks:

- Convergence may be slower when deploying BGP as control plane between ITR and ETRs.
- Requires being able to inject specific /32 prefixes associated to the RLOCs into the core of the network. This is usually not a feasible option when connecting to a Service Provider network.
- Currently works only if no default route is present in the routing table of the remote ITR. With a default route, even receiving the specific routing update would not cause marking the RLOC as unreachable in the ITR map-cache.

2. Enabling RLOC probing

A second mechanism to detect the failure of a remote ETR consists in enabling RLOC probing on the ITR. This can be done with the simple configuration shown below:

NX-OS

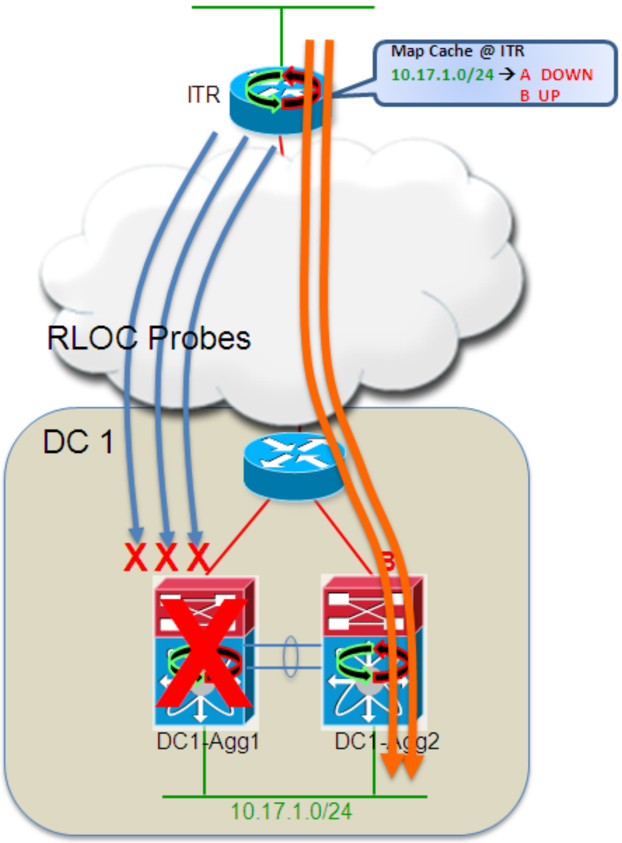
```
DC1-Agg1(config)# lisp loc-reach-algorithm rloc-probing
```

IOS

```
router lisp
  loc-reach-algorithm rloc-probing
```

Once RLOC probing is enabled, periodic control plane messages are sent from the ITR to the RLOC IP addresses associated to the EIDs in the local map-cache. If an RLOC probe does not get a response (for example because of the failure of the ETR), the ITR then tries to send two more probes (at 1 seconds interval) before declaring the RLOC unreachable and mark it down in its local map-cache (Figure A-7).

Figure A-7 Use of RLOC Probing to Detect EID Failure

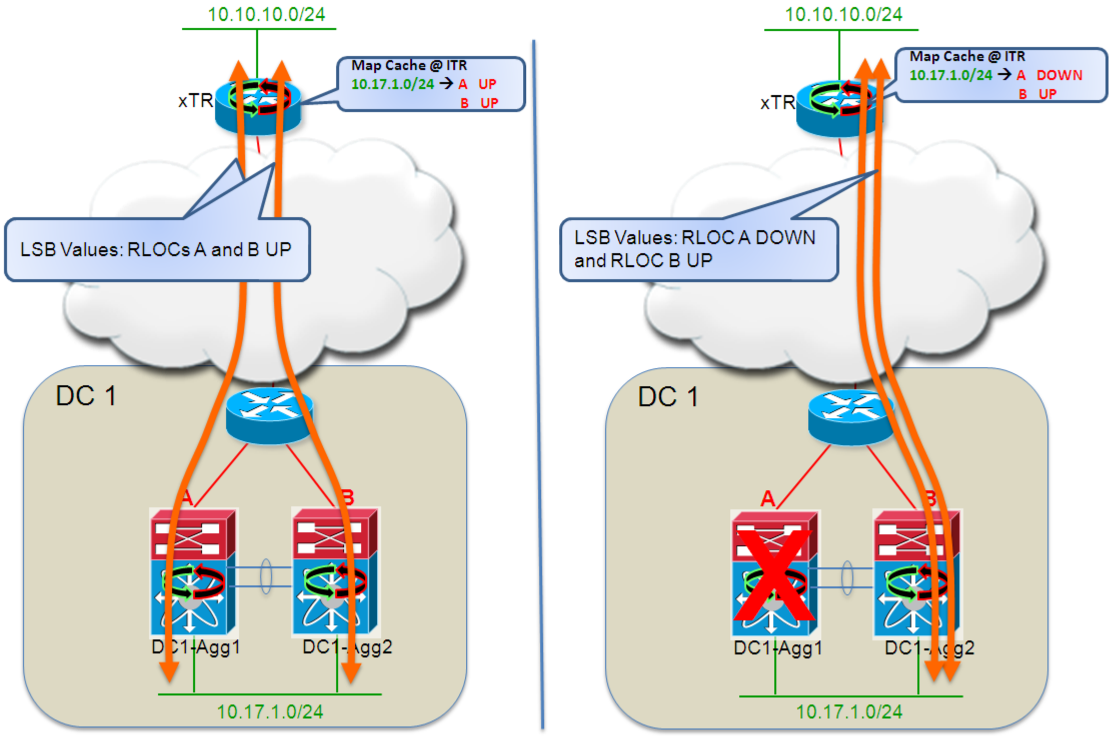


Before enabling RLOC probing, it is important to consider the scalability implications in medium/large LISP deployment, so this option should be used carefully. In addition to that, the 60 seconds RLOC probing period is currently not configurable and this may cause up to 60 seconds outage for the traffic flows originally destined to the failed ETR.

3. Leveraging LSB bits in the LISP header of encapsulated packets

The last mechanism available to deal with the failure of an ETR consists in leveraging specific information contained in the LISP header of encapsulated frames.

Figure A-8 Use of LSB Bits to Communicate an RLOC Failure



As shown on the left side of Figure A-8, when both ETRs are up and running, the packets they generate destined to the remote EID subnet 10.10.10.0/24 contain in the LSB portion of the header the information that both DC1 RLOCs are available. The remote ETR that keeps load-balancing traffic toward both RLOCs A and B receive this information.

When one of the DC LISP devices fail, the peer device receives a routing update about the now missing RLOC and as a result changes the information in the LSB bits of the traffic directed to the remote ETR to inform it that only RLOC B is now up and running. The remote ETR receives the packet and leverages the LSB bits information to mark down the corresponding RLOC in the map-cache, allowing for traffic recovery via RLOC B.

The advantage of this solution is that it is enabled by default on the data plane and does not require any specific consideration on the routing protocol deployment side.

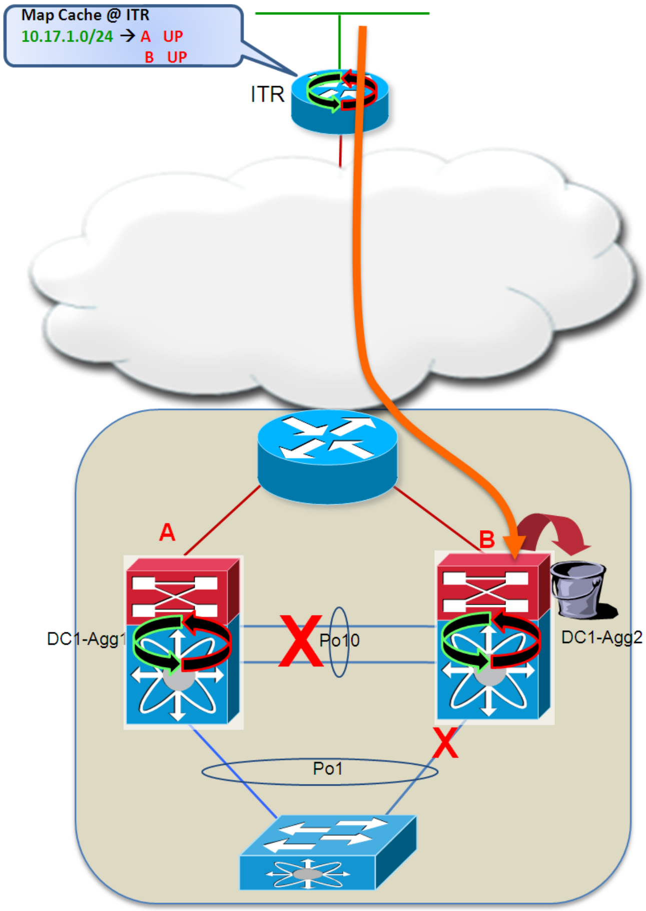
Some of the drawbacks are:

- It assumes that bidirectional flows are established between remote LISP devices, since the LSB notification is always associated to data plane traffic.
- Current Nexus 7000 LISP enabled HW (M1-32 linecards) do not allow to react to the changes applied by a remote xTR to the LSB bits value, making impossible to recover the flows destined to the remote xTR. Future HW support with F3 linecard will solve this problem.

Handling vPC Peer-Link Failure

When the vPC Peer-Link between LISP DC xTRs fails, the xTR operating as vPC secondary brings down all the vPC member interfaces, causing a complete isolation from the L2 domain. However, connectivity to the L3 domain is still available and as a consequence traffic from a remote xTR/PxTR could still being sent to the RLOC of the DC `xTR, causing traffic black holing.

Figure A-9 vPC Peer-Link Failure



To avoid the traffic black-holing, it is required that the ITR marks as down RLOC B in its map-cache once the DC xTR loses connectivity to the L2 domain. This can currently be achieved in two different ways:

1. Enabling RLOC probing on the ITR, so that when the next probe is sent to DC1-Agg2 ETR, the response will communicate the fact that no connectivity to any EID is available on that ETR and the ITR will mark as down the corresponding RLOC B. As previously mentioned, the enablement of RLOC probing should be done carefully because of the scalability implications it may have, especially in large scale deployments leveraging many remote ITRs.
2. The second solution is more a workaround leveraging a simple Embedded Event Manager (EEM) applet on the Nexus 7000 to bring down the RLOC on the secondary vPC device once the vPC Peer-Link fails.

The specific applet that allows achieving this result is shown below.

```
event manager applet Suspend_LISP
  event syslog pattern "Peer-link going down, suspending all vPCs on secondary"
  action 1.0 cli conf t
  action 2.0 cli interface lo1
  action 2.1 cli shut
  action 9.0 syslog msg Suspend LISP after vPC Peer-link failure
```

The applet above shuts down the loopback interface used as RLOC (Loopback 1) once the vPC secondary device notices that the peer-link has failed.

In a similar fashion, the RLOC loopback is reactivated when the peer-link recovers:

```
event manager applet Reactivate_LISP
  event syslog pattern "vPC restore timer expired, reiniting vPCs"
  action 0.5 cli sleep 120
  action 1.0 cli conf t
  action 2.0 cli interface lo1
  action 2.1 cli no shut
  action 9.0 syslog msg Reactivate LISP after vPC Peer-link recovery
```

Notice how an artificial delay of 2 minutes is added to ensure that the recovering xTR has enough time to receive EID information from the peer xTR (via Map-notify messages).

LISP and Services (FW, SLB) Integration Considerations

Integration of network services, such as Firewalls (FW) and Server Load Balancing (SLB) devices, in a LISP-enabled architecture currently represent an important design challenge. The main problem is that these devices usually work in a stateful fashion (i.e. maintain information about the “state” of each specific traffic flow that traverses them), so specific attention needs to be paid when leveraging LISP to steer flows between data center sites, to avoid the creation of asymmetric traffic patterns.

FW and LISP xTR Positioning

The first thing to consider when discussing the relative positioning of a FW and the LISP xTR is that currently the FW must be deployed “south” of the LISP device. This is mandatory to allow the enforcement of security policies on original IP packets and it is the consequence of the UDP encapsulation that the xTR performs on the original IP packets.

Note When deploying the FW “north” of the LISP xTR, the only policy enforcement allowed is a stateless ACL permitting UDP traffic on port 4341 (LISP encapsulated traffic).

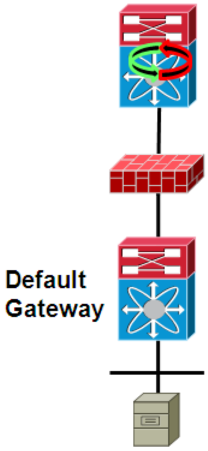
There are two possible design options for the deployment of the FW south of the xTR:

Option 1: FW in routed mode between the default gateway and the LISP xTR

The deployment model (Figure A-10) positioning the FW in routed mode north of the default gateway has become very popular and has been validated and documented as part of the Virtualized Multi-Service Data Center (VMDC) designs.

Note For more information on VMDC designs please refer to the link below:
http://www.cisco.com/en/US/solutions/ns340/ns414/ns742/ns743/ns1050/landing_vmdc.html

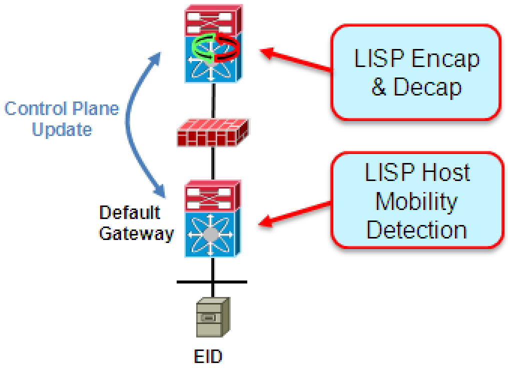
Figure A-10 FW Deployed between Default Gateway and LISP xTR



Positioning a FW between the default gateway and the LISP xTR implies that the EID dynamic detection can't happen anymore on the first L3 hop device directly connected to the EID subnet. This means that for EID discovery to happen, the IP packet generated from the EID must be routed north of the default gateway, traverse the FW and reach the xTR on top. Because of this, the recommendation is to wait for deploying this model until a new functionality (internally named "LISP Multi-Hop Host Mobility") will become available.

As shown in Figure A-11, with Multi-Hop Host Mobility it will be possible to separate two LISP functions: the dynamic EID detection will remain on the default gateway device, whereas the EID prefix registration and LISP HW encapsulation/decapsulation will be performed by the device north of the FW. A control plane channel will be established between the two devices to communicate the information of the discovery of the EID from the first-hop router to the LISP xTR.

Figure A-11 Multi-Hop Mobility

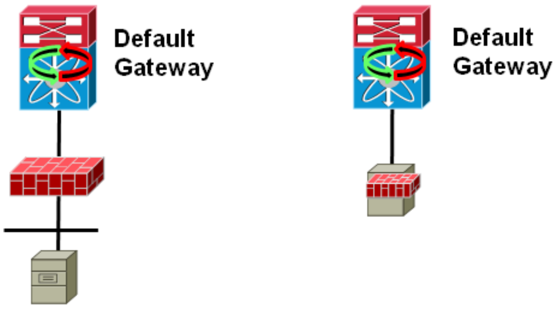


The LISP Multi-Hop Host Mobility functionality will definitely provide much more flexibility for positioning the LISP xTR inside the Data Center. More detailed information will be added to this document once the feature is released and validated (at the time of writing of this document the plan is to have it by Q1CY13).

Option 2: FW in transparent mode or Virtual Firewall (VSG)

This second approach is shown in Figure A-12.

Figure A-12 FW in Transparent Mode and VSG



In this case the FW will be deployed in L2 (transparent or bridged) mode or, in a virtualized type of deployment, leveraging the Virtual Services Gateway (VSG) functionality available with Nexus1000v.

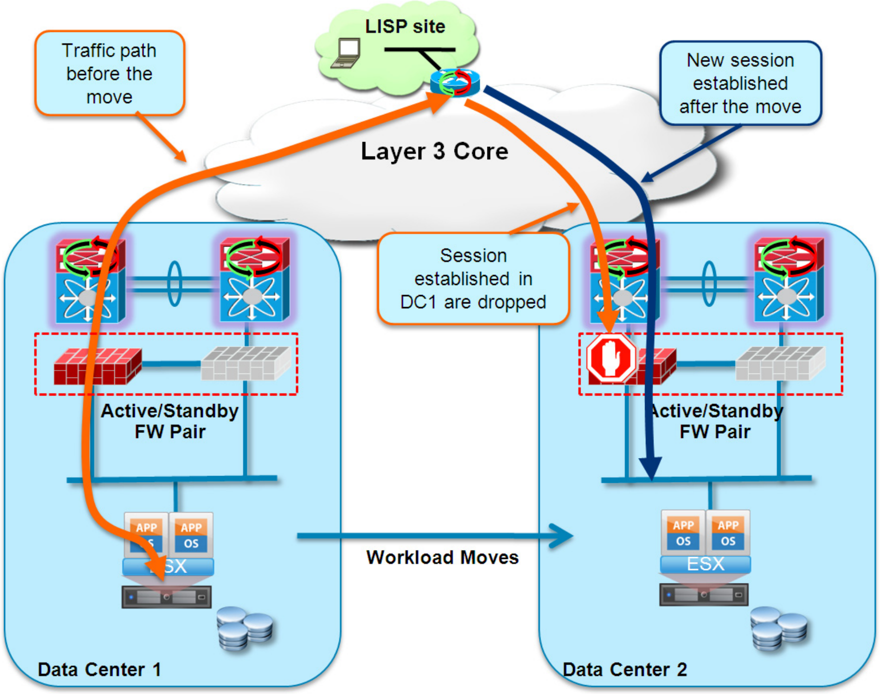
Note For more information about VSG and Nexus1000v deployment please refer to the document below: http://www.cisco.com/en/US/docs/solutions/Enterprise/Data_Center/DCI/4.0/EMC/EMC.pdf

In both models above, the LISP xTR device remains co-located on the first-hop router (default-gateway) so all the deployment considerations and configuration steps previously discussed remain valid. The only additional step required for the dynamic discovery of the EID is to ensure that traffic generated by this device can flow across the FW (physical or virtual) to reach the LISP xTR upstream.

When looking at an end-to-end Data Center Interconnect (DCI) architecture leveraging physical FW devices, there are two possible deployment models.

The first one, shown in Figure A-13, leverages an independent pair of Active/Standby FW nodes in each data center location.

Figure A-13 Active/Standby FW Pair in each DC Site



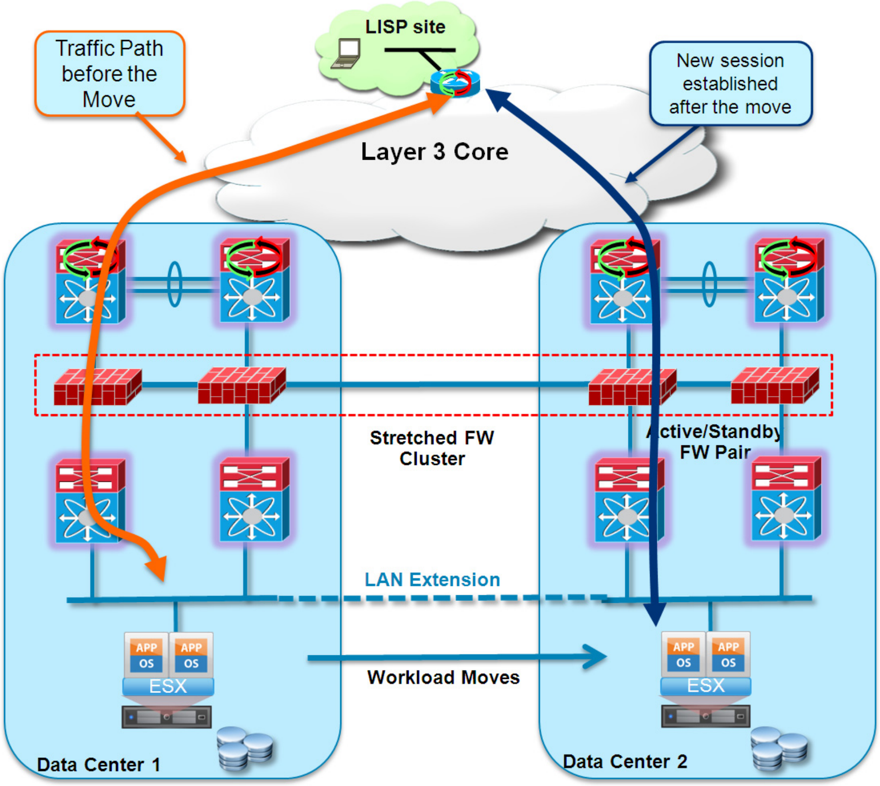
The two pair of devices function in a complete stateless fashion across sites, which means the following sequence of events happens to establish client-server communication before and after a workload move:

- The client-server communication is initially established via the active FW in DC1, since LISP direct the traffic to that site based on the workload location.
- Once the workload moves, LISP starts steering the client-server flows to DC2.
- TCP traffic flows initially established via DC1 are going to be dropped, since the active FW in DC2 does not have any state information for these pre-established sessions.
- New TCP based client-server sessions will be established via the optimized path, creating new state information in the active FW in DC2.

Because of this specific behavior, this deployment model is usually positioned for cold migration scenarios (like Disaster Recovery), where it is normal to re-establish new sessions after the workload migration to a secondary site.

For live mobility scenarios, it is instead more appropriate to use a second model leveraging a FW cluster stretched between DC locations, as shown in Figure A-14.

Figure A-14 FW Cluster Stretched between DC Sites

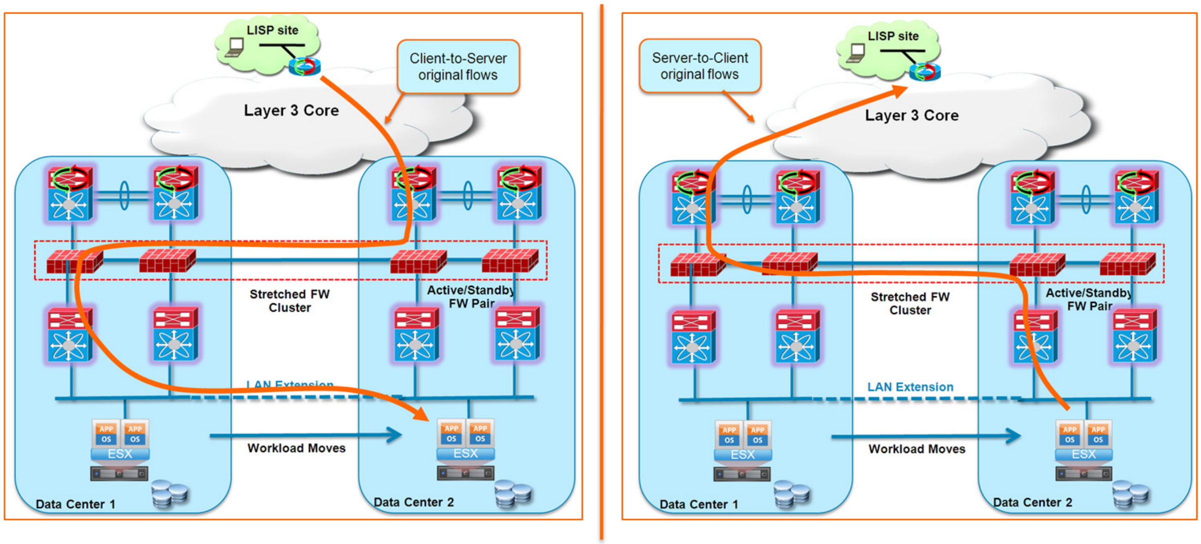


At the time of writing of this document, the only clustering functionality available for the FW is stretching an active/standby pair between sites. However, multi-node cluster support for Cisco ASA platform is scheduled to be released released in 2HCY12, allowing to cluster together up to eight FW nodes.

Figure A-14 highlights how new sessions can be established via one of the FW cluster nodes deployed in DC2. When stretching an Active/Standby pair between sites, this behavior can be achieved leveraging two FW contexts, one active in DC1 (and standby in DC2) and the other active in DC2 (and standby in DC1).

Differently from the scenario previously discussed, this deployment model also allows to preserve previously established sessions, at the price of creating a suboptimal traffic path (Figure A-15).

Figure A-15 Maintaining Established Traffic Flows



The behavior shown above is due to the fact that traffic flows initially established via the FW node in DC1, needs to keep going through that node to be maintained alive. This requires an intra-cluster redirection happening at L2 between one of the nodes in DC2 and the original node in DC1.

Note When deploying an Active/Standby pair of nodes with a pair of Active/Active contexts (one in each site), the behavior above is achieved leveraging the Asymmetric Routing functionality. For more information, refer to the "Configuring Asymmetric Routing Support" section of "Configuring Failover" at: http://www.cisco.com/en/US/customer/docs/security/asa/asa70/configuration/guide/failover.html

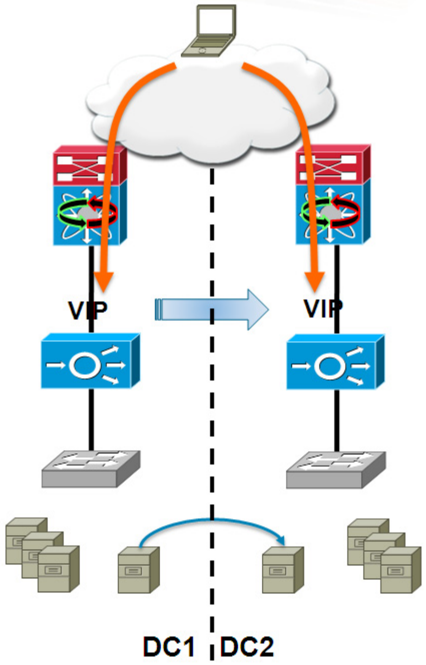
Notice that to support the behavior highlighted above, it is usually mandatory to extend L2 connectivity between DC sites. This is required both for intra-cluster traffic flows redirection and for allowing live workload mobility.

SLB and LISP xTR Positioning

The final consideration around LISP and services integration is regarding the introduction of Server Load Balancers (SLBs). The main thing to keep in mind in this case is the fact that all client sessions directed to a load-balanced server-farm are connecting to the VIP of the load-balancer. That means that it is the VIP of the SLB that plays the role of the EID in this scenario.

The immediate consequence is that the move between DC sites of a workload belonging to the server-farm would go unnoticed from a LISP perspective, since the VIP would remain anchored to the old location. The use case is then shifting from workload migration to server-farm migration, where the goal becomes the move of the VIP of the SLB once the entire balanced server-farm (or at least the majority of it) is migrated, as shown in Figure A-16.

Figure A-16 LISP and SLB Integration



The easiest way to migrate a load-balancer VIP is by leveraging a manual procedure. This is usually implemented in Disaster Recovery scenarios. In other cases, it may be useful to have a more dynamic way to move the VIP, based on the actual move of real-server belonging to the server-farm. More information about this mechanism (and integration of this functionality with orchestration tools) will be added in future releases of this paper.