

# SGI Multi-Rail Unit Test Plan

- Overview
- Unit Test Plan
  - Local Network Configuration
    - In-Range UT
    - Out-of-Range UT
    - Error UT
  - Remote Peer Configuration
    - Expected Behavior
    - In-Range UT
    - Out-of-Range UT
    - Error UT
  - Policy Configuration
    - In-Range UT
    - Error UT
  - General Configuration
  - Functional Requirements
    - Interface Selection and Message Sending Requirements
    - Dynamic NID Discovery
    - Debugging Requirements
    - Network interface Health
    - Backwards Compatibility Requirements
  - Performance Requirements
  - Misc Error Scenarios

## Overview

The Unit Test Plan (UTP) will follow the same section breakdown as the Requirements in the Scope & Requirement Document.

The following types of tests shall be included where it makes sense.

1. In-Range UT - these are the test cases which cover normal operations.
2. Out-of-Range UT - these are the test cases which cover out of range scenarios:
  - a. border cases
  - b. race conditions
  - c. unexpected events
    - i. EX: Tearing down an active Network Interface
3. Error UT
  - a. Error parameters
  - b. Error Conditions
    - i. network goes down unexpectedly
    - ii. Wire gets disconnected, etc

Performance Testing cases will be a separate section in this document.

## Unit Test Plan

Configuration tests should be done through the DLC direct interface, as well as the YAML interface.

### Local Network Configuration

#### In-Range UT

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
cfg-020	cfg-005, cfg-010, cfg-015, cfg-045, cfg-055, cfg-060, cfg-065	UT-0005	<ul style="list-style-type: none"><li>• Configure 3 NIDs on the same TCP network.</li><li>• Show the NIDs</li></ul>
		UT-0010	<ul style="list-style-type: none"><li>• Configure 3 NIDs on the same IB network</li><li>• Show the NIDs</li></ul>

		UT-0015	<ul style="list-style-type: none"> <li>• Configure 3 NIDs on the same TCP/IB Network</li> <li>• Show the NIDs</li> <li>• Delete 1 NID from the TCP/IB Network</li> <li>• Show the NIDs</li> </ul>
		UT-0020	<ul style="list-style-type: none"> <li>• Configure 2 NIDs on tcp0/o2ib0</li> <li>• Configure 2 NIDs on tcp1/o2ib1</li> <li>• Show the NIDs</li> <li>• Delete 1st NID from tcp0</li> <li>• Delete 2nd NID from tcp0</li> <li>• Show NIDs <ul style="list-style-type: none"> <li>◦ No more tcp 0 should exist</li> <li>◦ o2ib0 should be unaffected</li> </ul> </li> </ul>
cfg-025	cfg-005, cfg-010, cfg-015, cfg-045, cfg-055, cfg-060, cfg-065	UT-0025	<ul style="list-style-type: none"> <li>• Configure the system to have 4 CPTs <ul style="list-style-type: none"> <li>◦ options libcfs cpu_npartitions=4 cpu_pattern="0[0] 1[1] 2[2] 3[3]"</li> </ul> </li> <li>• Configure 2 NIDs on tcp0 <ul style="list-style-type: none"> <li>◦ NID 1 should be on CPTs 0, 3</li> <li>◦ NID 2 should be on CPTs 1, 2</li> </ul> </li> <li>• Show NIDs <ul style="list-style-type: none"> <li>◦ proper CPT association should be displayed</li> </ul> </li> </ul>
cfg-035	cfg-040, cfg-045, cfg-055, cfg-060, cfg-065	UT-0030	<ul style="list-style-type: none"> <li>• Configure the system to have 4 CPTs <ul style="list-style-type: none"> <li>◦ options libcfs cpu_npartitions=4 cpu_pattern="0[0] 1[1] 2[2] 3[3]"</li> </ul> </li> <li>• Configure 3 NIDs on tcp0 <ul style="list-style-type: none"> <li>◦ NID 1 should be on CPTs 0, 3</li> <li>◦ NID 2 should be on CPTs 1, 2</li> <li>◦ NID 3 should be on all CPTs</li> </ul> </li> <li>• Show NIDs <ul style="list-style-type: none"> <li>◦ proper CPT association should be displayed</li> <li>◦ NID 3 should exist on all CPTs</li> </ul> </li> </ul>
		UT-0035	<ul style="list-style-type: none"> <li>• Configure 1st NID on tcp0 using the legacy ip2nets parameter from DLC</li> <li>• Show NIDs</li> </ul>
		UT-0040	<ul style="list-style-type: none"> <li>• Configure 1st NID on tcp*/o2ib* in the following ip2nets form: <ul style="list-style-type: none"> <li>◦ tcp(&lt;eth intf&gt;[&lt;cpt&gt;] &lt;pattern&gt;</li> </ul> </li> <li>• Show NIDs to ensure that the interface has been added to the correct CPTs</li> </ul>
		UT-0045	<ul style="list-style-type: none"> <li>• Configure 1st NID on tcp*/o2ib* in the following ip2nets form: <ul style="list-style-type: none"> <li>◦ tcp(&lt;eth intf&gt;, &lt;eth intf&gt;, ...)[&lt;cpt&gt;] &lt;pattern&gt;</li> <li>◦ [&lt;cpt&gt;] can have only one value</li> </ul> </li> <li>• Show NIDs to ensure that the interface has been added to the correct CPTs</li> </ul>
		UT-0050	<ul style="list-style-type: none"> <li>• Configure 1st NID on tcp*/o2ib* in the following ip2nets form: <ul style="list-style-type: none"> <li>◦ tcp(&lt;eth intf&gt;[&lt;cpt&gt;], &lt;eth intf&gt;[&lt;cpt&gt;], ...) &lt;pattern&gt;</li> </ul> </li> <li>• Show NIDs to ensure that the interface has been added to the correct CPTs</li> </ul>
cfg-060	cfg-065	UT-0055	<p>Go through the following Inetctl commands and exercise their parameters:</p> <ul style="list-style-type: none"> <li>• net</li> <li>• set num_range</li> </ul>

## Out-of-Range UT

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
------------------------	--------------------------	--------------	-----------------------

cfg-020	cfg-005, cfg-010, cfg-015	UT-0060	<ul style="list-style-type: none"> <li>• Configure 32 NIDs on the same TCP/IB Network</li> <li>• Show the NIDs</li> </ul>
		UT-0065	<ul style="list-style-type: none"> <li>• Configure 32 NIDs on the same TCP/IB Network</li> <li>• Show the NIDs.</li> <li>• Delete 32 NIDs on the same TCP/IB Network</li> </ul>
		UT-0070	<ul style="list-style-type: none"> <li>• Configure NID A, B and C on tcp0/o2ib0 Network</li> <li>• Configure NID A and B on tcp1/o2ib1 Network</li> <li>• Show the NIDs <ul style="list-style-type: none"> <li>◦ Configuration should succeed. NIs can exist on different networks</li> </ul> </li> </ul>
cfg-060	cfg-065	UT-0075	<p>Go through the following Inetctl commands and exercise their parameters, by providing out of range values:</p> <ul style="list-style-type: none"> <li>• net</li> <li>• set num_range</li> </ul>
		UT-0080	<ul style="list-style-type: none"> <li>• Don't configure any LNet modprobe.</li> <li>• Load LNet where there exists only one commissioned IB interface with IPoIB configured</li> <li>• a TCP network should be created with that IB interface</li> <li>• Configure an o2ib network with the same IB interface</li> <li>• Now you should have two interfaces with exactly the same IB</li> </ul>

## Error UT

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
		UT-0090	<ul style="list-style-type: none"> <li>• Configure a non-existent NID on tcp0</li> <li>• Configuration should fail with INVALID PARAMETER</li> </ul>
		UT-0095	<ul style="list-style-type: none"> <li>• Configure the system to have 4 CPTs <ul style="list-style-type: none"> <li>◦ options libcfs cpu_npartitions=4 cpu_pattern="0[0] 1[1] 2[2] 3[3]"</li> </ul> </li> <li>• Configure 3 NIDs on tcp0 <ul style="list-style-type: none"> <li>◦ NID 1 should be on CPTs 0, 4</li> <li>◦ NID 2 should be on CPTs 1, 2</li> <li>◦ NID 3 should be on all CPTs</li> </ul> </li> <li>• Show NIDs <ul style="list-style-type: none"> <li>◦ NID 1 should fail since no CPT 4</li> </ul> </li> </ul>
		UT-0096	<ul style="list-style-type: none"> <li>• Configure 1st NID on tcp*/o2ib* in the following ip2nets form: <ul style="list-style-type: none"> <li>◦ tcp(&lt;eth intf&gt;, &lt;eth intf&gt;, ...)[&lt;cpt, cpt&gt;] &lt;pattern&gt;</li> </ul> </li> <li>• Configuration should fail with syntax error</li> </ul>
		UT-0100	<p>Go through the following Inetctl commands and exercise their parameters, by providing error values:</p> <ul style="list-style-type: none"> <li>• net <ul style="list-style-type: none"> <li>◦ Valid net values are: tcp, o2ib, gni</li> <li>◦ Provide any garbage. Return value should be BAD PARAM</li> </ul> </li> <li>• set num_range <ul style="list-style-type: none"> <li>◦ valid range is any positive value.</li> <li>◦ Provide a negative value. Return value should be BAD PARAM</li> </ul> </li> </ul>
		UT-0105	<p>Delete a non-existent network</p> <p>Should return -EINVAL</p>
		UT-0110	<p>Delete a non-existent NID on tcp/o2ib</p> <p>Should return -EINVAL</p>

## Remote Peer Configuration

## Expected Behavior

- A peer can be added by specifying a list of NIDs
  - The first NID shall be used as the primary NID. The rest of the NIDs will be added under the primary NID
- A peer can be added by explicitly specifying the key NID, and then by adding a set of other NIDs, all done through one API call
- If a key NID already exists, but it's not an MR NI, then adding that Key NID from DLC shall convert that NI to an MR NI
- If a key NID already exists, and it is an MR NI, then re-adding the Key NID shall have no effect
- if a Key NID already exists as part of another peer, then adding that NID as part of another peer shall fail
- if a NID is being added to a peer NI and that NID is a non-MR, then that NID is moved under the peer and is made to be MR capable
- if a NID is being added to a peer and that NID is an MR NID and part of another peer, then the operation shall fail
- if a NID is being added to a peer and it is already part of that Peer then the operation is a no-op.

## In-Range UT

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
cfg-070		UT-0115	<ul style="list-style-type: none"> <li>• add a new peer with only 1 NID</li> </ul>
cfg-070		UT-0120	<ul style="list-style-type: none"> <li>• add a new peer with only 1 NID</li> <li>• add more nids to that peer</li> </ul>
cfg-070		UT-0125	<ul style="list-style-type: none"> <li>• add a new peer with multiple NIDs</li> </ul>
cfg-070		UT-0130	<ul style="list-style-type: none"> <li>• add a new peer with only 1 NID</li> <li>• Delete that NID</li> </ul>
cfg-070		UT-0131	<ul style="list-style-type: none"> <li>• add a new peer with multiple NIDs</li> <li>• delete the primary NI of the peer</li> <li>• The entire peer should be deleted.</li> </ul>
cfg-070		UT-0135	<ul style="list-style-type: none"> <li>• add a new peer with multiple NIDs</li> <li>• Delete each NID one at a time until the peer is removed</li> </ul>
cfg-070		UT-0140	<ul style="list-style-type: none"> <li>• add a new peer with multiple NIDs</li> <li>• Delete all NIDs but primary NID only.</li> <li>• Re-add multiple NIDs one at a time.</li> </ul>
cfg-070		UT-0145	<ul style="list-style-type: none"> <li>• add a new peer with multiple NIDs</li> <li>• Delete all NIDs but primary NID.</li> <li>• Re-add multiple NIDs in one shot.</li> </ul>
cfg-075		UT-0150	<ul style="list-style-type: none"> <li>• add a new peer with multiple NIDs on different networks</li> </ul>

## Out-of-Range UT

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
		UT-0155	<ul style="list-style-type: none"> <li>• add a new peer with 32 NIDs</li> </ul>
		UT-0160	<ul style="list-style-type: none"> <li>• delete a peer and all its 32 NIDs</li> </ul>
		UT-0165	<ul style="list-style-type: none"> <li>• load Inet</li> <li>• Inetctl Inet configure</li> <li>• add 2 or more peers on a non-local network</li> <li>• delete peer 1</li> <li>• delete peer 2</li> </ul>

		UT-0170	<ul style="list-style-type: none"> <li>• load Inet</li> <li>• Inetctl Inet configure</li> <li>• add 2 or more peers on a non-local network</li> <li>• Inetctl Inet unconfigure</li> <li>• lustre_rmmod</li> </ul>
		UT-0171	<ul style="list-style-type: none"> <li>• load Inet</li> <li>• Inetctl Inet configure</li> <li>• add 2 or more peers on tcp1 (non-local)</li> <li>• check that refcount = 2 (1 for hashlist &amp; 1 for remote list)</li> <li>• check credits are not set</li> <li>• add tcp1 network</li> <li>• Check refcount 1 (remote list refcount removed)</li> <li>• check credits are set</li> </ul>
		UT-0172	<ul style="list-style-type: none"> <li>• load Inet</li> <li>• Inetctl Inet configure</li> <li>• add 2 or more peers on tcp1 (primary peer ni)</li> <li>• add 2 or more peers on tcp2</li> <li>• add tcp 1 and tcp 2 networks</li> <li>• remove the tcp1 network</li> <li>• check that the entire peer is removed</li> </ul>
		UT-0173	<ul style="list-style-type: none"> <li>• same steps as above</li> <li>• remove a tcp2 network</li> <li>• check that all peers on that network are removed.</li> </ul>
		UT-0175	<ul style="list-style-type: none"> <li>• startup Inet</li> <li>• startup traffic</li> <li>• add a peer ni on a non-local network</li> <li>• add a local network for that peer</li> <li>• Send traffic over that peer_ni</li> </ul>
		UT-0176	<ul style="list-style-type: none"> <li>• startup Inet</li> <li>• add tcp1 network</li> <li>• add peers on tcp1 network</li> <li>• check they are multi-rail</li> <li>• run traffic</li> <li>• delete the peers</li> <li>• peers should be recreated because of traffic and they should be non-mr</li> </ul>

## Error UT

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
cfg-070		UT-0180	<ul style="list-style-type: none"> <li>• add more than 32 NIDs</li> </ul>
cfg-070		UT-0185	<ul style="list-style-type: none"> <li>• add a peer with multiple NIDs</li> <li>• delete a non-existent peer NID from the peer identified by key-NID</li> </ul>
cfg-080	snd-065	UT-0190	<ul style="list-style-type: none"> <li>• add peer 1 with NIDs A, B and C</li> <li>• add peer 2 with NIDs D, C and E <ul style="list-style-type: none"> <li>◦ Adding NID C should fail</li> </ul> </li> </ul>

## Policy Configuration

### In-Range UT

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
------------------------	--------------------------	--------------	-----------------------

cfg-090		UT-0195	<ul style="list-style-type: none"> <li>Set the NUMA range to 0</li> <li>The NI closest to the message memory NUMA will be picked.</li> </ul>
cfg-090		UT-0200	<ul style="list-style-type: none"> <li>Increase the NUMA range step by step</li> <li>Note that more NIs are picked when sending</li> </ul>
cfg-090	snd-025	UT-0205	<ul style="list-style-type: none"> <li>Set the NUMA range to a large value</li> <li>start traffic</li> <li>NIs are picked in round robin</li> </ul>

## Error UT

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
cfg-090		UT-0210	<ul style="list-style-type: none"> <li>Set the NUMA range to &lt; 0</li> <li>This should be rejected</li> </ul>

## General Configuration

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
cfg-170		UT-0215	<ul style="list-style-type: none"> <li>Configure multiple NIs</li> <li>Configure multiple Peers with multiple NIDs</li> <li>set NUMA range value</li> <li>Dump the YAML configuration</li> <li>use the YAML configuration file to delete all configuration</li> <li>use the YAML configuration file to reconfigure the node.</li> </ul>

## Functional Requirements

### Interface Selection and Message Sending Requirements

Note to test NUMA proximity, you can use python psutil to bind a process to a specific CPU then execute a write/read operation to the FS on that CPU.

The CPU distances can be acquired from `/proc/sys/net/cpu_partition_distance`

The NUMA cpu list can be acquired from `/sys/devices/system/node/node*/cpulist`

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description	Behavior of Note
snd-005		UT-0220	<ul style="list-style-type: none"> <li>Configure 3 NIs with equidistant NUMA distance</li> <li>Send three or more messages</li> <li>Dump statistics on each NI to verify that each NI was used to send messages</li> </ul>	
snd-010	snd-015	UT-0225	<ul style="list-style-type: none"> <li>Configure 3 NIs closer to different NUMA nodes</li> <li>dump the NI statistics</li> <li>Verify that each NI has the correct device CPT</li> </ul>	
snd-020		UT-0230	<ul style="list-style-type: none"> <li>Configure 3 NIs with different NUMA distances</li> <li>Send messages</li> <li>Confirm through statistics that messages are being sent over the nearest NI (NUMA wise)</li> </ul>	

snd-020		UT-0235	<ul style="list-style-type: none"> <li>• Configure 2 NIs with different NUMA distances</li> <li>• Send messages</li> <li>• Confirm through statistics that messages are being sent over the nearest NI (NUMA wise)</li> <li>• add another NI which is close NUMA wise than the current nearest</li> <li>• confirm through statistics that messages are not being sent over the newly added NI</li> </ul>	
snd-030		UT-0240	<ul style="list-style-type: none"> <li>• Configure 3 NIs, one EDR, one FDR and one QDR</li> <li>• set the NUMA range to a large value so all NIs are considered through RR</li> <li>• start traffic</li> <li>• monitor statistics on each NI.</li> <li>• Confirm that EDR is preferred until it becomes saturated, then FDR is selected then QDR</li> </ul>	
snd-030		UT-0245	<ul style="list-style-type: none"> <li>• Configure 3 NIs</li> <li>• set the NUMA range to a large value so all NIs are considered through RR</li> <li>• start traffic</li> <li>• monitor statistics on each NIs to confirm all are being used.</li> <li>• Remove one of the NIs</li> <li>• Confirm that that NI is no longer used for new messages</li> <li>• Confirm that the other 2 NIs are being used.</li> <li>• No messages should be dropped.</li> </ul>	
snd-035		UT-0250	<ul style="list-style-type: none"> <li>• Configure 3 NIs</li> <li>• Configure a peer with 3 NIDs</li> <li>• Send messages to the peer</li> <li>• Confirm through statistics that peer NIDs are being used based on their available credits.</li> </ul>	
snd-040		UT-0255	<ul style="list-style-type: none"> <li>• Configure 3 NIs which are not equidistant all on the same network</li> <li>• configure a peer with 3 NIDs all on the same network</li> <li>• start traffic</li> <li>• Confirm closest NUMA NI is being used</li> <li>• Confirm peer NIDs are being used</li> <li>• set NUMA range to a large value</li> <li>• Confirm all NIs are being used</li> <li>• Confirm no change in traffic pattern to the peers</li> </ul>	
snd-045	snd-070	UT-0260	<ul style="list-style-type: none"> <li>• Configure NIs A, B and C</li> <li>• Configure the peer with the same NIDs</li> <li>• Send 1 message which requires a response from NI A</li> <li>• Confirm that responses are being sent to the same NI</li> </ul>	
snd-050		UT-0265	<ul style="list-style-type: none"> <li>• Configure NIs A, B and C</li> <li>• Configure the peer with the same NIDs</li> <li>• Send 1 message which requires a response from NI A</li> <li>• bring down NI A</li> <li>• confirm that response is sent to one of the other configured NIDs</li> </ul>	
snd-050		UT-0270	<ul style="list-style-type: none"> <li>• Configure an MR system</li> <li>• Start traffic</li> <li>• monitor traffic is being sent to all configured peers</li> <li>• bring down one of the peer NIDs</li> <li>• monitor traffic is no longer sent to that peer NID</li> <li>• no messages should be dropped</li> </ul>	

snd-050	snd-060, snd-075	UT-0275	<ul style="list-style-type: none"> <li>• Configure an MR system</li> <li>• Start traffic</li> <li>• monitor traffic is being sent to all configured peers</li> <li>• bring down one of the peer NIDs</li> <li>• monitory traffic is no longer sent to that peer NID</li> <li>• bring up the peer NID again</li> <li>• monitor traffic is being sent to it again</li> </ul>	<p>When a peer NID is removed then added or if a new peer NID is added, then it's peer_ni-&gt;lpni_seq number will start off at 0.</p> <p>In the case of credits being == then the newly added peer will always be picked until it's lpni_seq number catches up with the other peer NIs sequence numbers. See code below.</p> <pre> } else if (lpni-&gt;lpni_txcredits == best_lpni_credits) {     /*      * The best peer found so far     and the current peer      * have the same number of     available credits let's      * make sure to select     between them using Round      * Robin     */     if (best_lpni) {         if (best_lpni- &gt;lpni_seq &lt;= lpni-&gt;lpni_seq)             continue;     } </pre> <p>This behavior will manifest itself in low bandwidth environment. In high bandwidth environment it is likely that the credits in the selection algorithm will be different and the peer_NI will be picked according to credits.</p> <p>Another scenario to consider is when the lpni_seq number wraps. In low bandwidth environment this could cause the peer NI which wrapped to be picked until it catches up with the other peer NIs sequence numbers.</p> <p>This in itself might not be significant enough, but does raise the question of the benefit of having a seq number to start with. Does it give much of a functional advantage, or having the credits criteria enough.</p> <p>The same issue is present with local NI sequence numbers.</p>
snd-055	snd-060, snd-075	UT-0280	<ul style="list-style-type: none"> <li>• Configure an MR system</li> <li>• Start traffic</li> <li>• monitor traffic is being sent to all configured peers</li> <li>• bring down one of the peer NIDs</li> <li>• monitory traffic is no longer sent to that peer NID</li> <li>• bring down all peer NIDs</li> <li>• message should fail.</li> </ul>	
snd-055	snd-060, snd-075, snd-085	UT-0285	<ul style="list-style-type: none"> <li>• Configure an MR system</li> <li>• Start traffic</li> <li>• monitor traffic is being sent to all configured peers over all NIs</li> <li>• bring down the local NIs one by one</li> <li>• note traffic is migrated to the NIs still up, until no NIs are left then messages are dropped</li> </ul>	
snd-055	snd-060, snd-075, snd-085	UT-0290	<ul style="list-style-type: none"> <li>• Configure an MR system</li> <li>• Start traffic</li> <li>• monitor traffic is being sent to all configured peers over all NIs</li> <li>• bring down the local NIs one by one</li> <li>• note traffic is migrated to the NIs still up, until no NIs are left then messages are dropped</li> <li>• bring up the NIs again and confirm that NIs are being reused.</li> </ul>	
snd-055	snd-060, snd-075	UT-0295	<ul style="list-style-type: none"> <li>• Configure two networks tcp and o2ib</li> <li>• Configure nodes to have multiple interfaces on each of the networks</li> <li>• start traffic over the o2ib network</li> <li>• o2ib should be used</li> <li>• bring down the o2ib network</li> <li>• traffic should migrage to the tcp network.</li> <li>• no traffic should be dropped.</li> </ul>	

snd-080		UT-0300	<ul style="list-style-type: none"> <li>• Configure an MR system</li> <li>• bring down an NI</li> <li>• confirm that the show info shows the NI as down</li> </ul>	
snd-080		UT-0305	<ul style="list-style-type: none"> <li>• TODO: how do we test device failure?</li> </ul>	
		UT-0310	<ul style="list-style-type: none"> <li>• Configure an MR system</li> <li>• Configure peers via DLC</li> <li>• Run traffic</li> <li>• Delete one of the peer_nis we're sending to via DLC</li> <li>• Traffic going over that peer_ni should continue but no more traffic should use that NI</li> </ul>	
		UT-0315	<ul style="list-style-type: none"> <li>• Configure an MR system</li> <li>• Configure peers via DLC</li> <li>• Run traffic</li> <li>• Delete one of the peer_nis we're sending to via DLC</li> <li>• Bring that peer_ni back</li> <li>• Note traffic stops and starts on that peer with no traffic loss</li> <li>• Repeat the deletion and reconfiguration of the peer_ni</li> </ul>	
		UT-0320	<ul style="list-style-type: none"> <li>• Configure an MR system</li> <li>• Configure peers via DLC</li> <li>• Run traffic</li> <li>• Delete the entire peer</li> <li>• The peer should be recreated on the next message, but it won't be MR capable.</li> </ul>	

## Dynamic NID Discovery

The unit tests for peer NID discovery depend on `lnetctl ping` not triggering discovery. To force discovery, use `lnetctl discover`. Note that some of the tests require DLC configuration to include non-existing peer NIDs. These nids are marked with a \*.

Tests with discovery enabled.

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040	UT-DD-CFG-0001	<ul style="list-style-type: none"> <li>• turn on lnet discovery</li> <li>• run <code>lnetctl discover</code></li> <li>• ensure that peers are discovered</li> </ul>
		UT-DD-CFG-0002	<ul style="list-style-type: none"> <li>• turn off lnet discovery</li> <li>• run <code>lnetctl discover</code></li> <li>• ensure we get some valid error back</li> </ul>
		UT-DD-CFG-0003	<ul style="list-style-type: none"> <li>• show the status of lnet discovery (if it's on or off)</li> </ul>
		UT-DD-CFG-0004	<ul style="list-style-type: none"> <li>• Make sure lnet discovery is configurable via YAML</li> </ul>
		UT-DD-CFG-0005	<ul style="list-style-type: none"> <li>• configure <code>lnet_max_interfaces</code> from command line</li> <li>• Show <code>lnet_max_interfaces</code> from command line and ensure it's set</li> </ul>
		UT-DD-CFG-0006	<ul style="list-style-type: none"> <li>• configure <code>lnet_max_interfaces</code> from YAML</li> <li>• show <code>lnet_max_interfaces</code> from YAML and ensure it's set</li> </ul>

	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040	UT-DD-EN-0001	<p>Basic functionality 1-1: discovery of an MR peer via its primary.</p> <ul style="list-style-type: none"> <li>• MR Node with interfaces N1, N2</li> <li>• MR Peer with interfaces P1, P2, P3</li> <li>• Ping P1 from node</li> <li>• Ping P2 from node</li> <li>• Verify that node sees two different peers: P1, P2.</li> <li>• Discover P1 from node</li> <li>• Verify that node sees one MR peer with three NIDS: P1, P2, P3.</li> <li>• Verify that peer sees node as one MR peer with two NIDS: N1, N2.</li> </ul>
	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040	UT-DD-EN-0002	<p>Basic functionality 1-2: discovery of an MR peer via a secondary.</p> <ul style="list-style-type: none"> <li>• MR Node with interfaces N1, N2</li> <li>• MR Peer with interfaces P1, P2, P3</li> <li>• Ping P1 from node</li> <li>• Ping P2 from node</li> <li>• Verify that node sees two different peers: P1, P2.</li> <li>• Discover P2 from node</li> <li>• Verify that node sees one MR peer with three NIDS: P1, P2, P3.</li> <li>• Verify that peer sees node as one MR peer with two NIDS: N1, N2.</li> </ul>
	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040	UT-DD-EN-0003	<p>Basic functionality 1-3: discovery of an MR peer via a tertiary.</p> <ul style="list-style-type: none"> <li>• MR Node with interfaces N1, N2</li> <li>• MR Peer with interfaces P1, P2, P3</li> <li>• Ping P1 from node</li> <li>• Ping P2 from node</li> <li>• Verify that node sees two different peers: P1, P2.</li> <li>• Discover P3 from node</li> <li>• Verify that node sees one MR peer with three NIDS: P1, P2, P3.</li> <li>• Verify that peer sees node as one MR peer with two NIDS: N1, N2.</li> </ul>
dyn-020	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040, dyn-055	UT-DD-EN-0004	<p>Basic functionality 1-4: implicit discovery of an MR peer</p> <ul style="list-style-type: none"> <li>• MR Node with interfaces N1, N2</li> <li>• MR Peer with interfaces P1, P2, P3</li> <li>• Force some filesystem traffic between node and peer.</li> <li>• Verify that node sees one MR peer with three NIDS: P1, P2, P3.</li> <li>• Verify that peer sees node as one MR peer with two NIDS: N1, N2.</li> </ul>
	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040	UT-DD-EN-0005	<p>Basic functionality 1-5: discovery of an MR peer with &gt; 16 interfaces. (This test exercises the code path that resizes the push buffers.)</p> <ul style="list-style-type: none"> <li>• MR Node with interfaces N1, N2, ..., N17</li> <li>• MR Peer with interfaces P1, P2, ..., P17</li> <li>• Ping P1 from node.</li> <li>• Discover P1 from node</li> <li>• Verify that node sees one MR peer with all NIDS: P1, P2, ..., P17</li> <li>• Verify that peer sees node as one MR peer with all NIDS: N1, N2, ..., N17.</li> </ul>
	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040	UT-DD-EN-0006	<p>Compatibility 2-1: discovery of a non-MR peer via its primary.</p> <ul style="list-style-type: none"> <li>• MR Node with interface N1, N2.</li> <li>• Non-MR Peer with interfaces P1, P2, P3.</li> <li>• Ping P1 from node.</li> <li>• Ping P2 from node.</li> <li>• Verify that node sees two different peers: P1, P2.</li> <li>• Discover P1 from node</li> <li>• Verify that node sees one non-MR peer with three NIDS: P1, P2, P3.</li> <li>• Verify that peer sees one as one peer with one NID: N1.</li> </ul>
	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040	UT-DD-EN-0007	<p>Compatibility 2-2: discovery of a non-MR peer via a secondary.</p> <ul style="list-style-type: none"> <li>• MR Node with interface N1, N2.</li> <li>• Non-MR Peer with interfaces P1, P2, P3.</li> <li>• Ping P1 from node.</li> <li>• Ping P2 from node.</li> <li>• Verify that node sees two different peers: P1, P2.</li> <li>• Discover P2 from node</li> <li>• Verify that node sees one non-MR peer with three NIDS: P1, P2, P3.</li> <li>• Verify that peer sees one as one peer with one NID: N1.</li> </ul>

	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040	UT-DD-EN-0008	<p>Compatibility 2-3: discovery of a non-MR peer via a tertiary.</p> <ul style="list-style-type: none"> <li>• MR Node with interface N1, N2.</li> <li>• Non-MR Peer with interfaces P1, P2, P3.</li> <li>• Ping P1 from node.</li> <li>• Ping P2 from node.</li> <li>• Verify that node sees two different peers: P1, P2.</li> <li>• Discover P3 from node</li> <li>• Verify that node sees one non-MR peer with three NIDS: P1, P2, P3.</li> <li>• Verify that peer sees one as one peer with one NID: N1.</li> </ul>
dyn-020	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040, dyn-055	UT-DD-EN-0009	<p>Compatibility 2-4: implicit discovery of an MR peer</p> <ul style="list-style-type: none"> <li>• MR Node with interfaces N1, N2</li> <li>• MR Peer with interfaces P1, P2, P3</li> <li>• Force some filesystem traffic between node and peer.</li> <li>• Verify that node sees one non-MR peer with three NIDS: P1, P2, P3.</li> <li>• Verify that peer sees node as one peer with one NID: N1</li> </ul>
dyn-060	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040	UT-DD-EN-0010	<p>Interaction with DLC 3-1: DLC overrides Discovery of MR peer</p> <ul style="list-style-type: none"> <li>• MR node with interface N1</li> <li>• MR peer with interface P1, P2, P3</li> <li>• DLC configure MR peer on node with interfaces P1, P2, P4*.</li> <li>• Discover P1 from node.</li> <li>• Verify that node sees one MR peer with three NIDS: P1, P2, P4*.</li> <li>• Verify presence of error messages on node (error code is <code>-EPERM</code>): <ul style="list-style-type: none"> <li>◦ Error adding NID P3 to peer P1: -1</li> <li>◦ Error deleting NID P3 from peer P1: -1</li> </ul> </li> </ul>
dyn-060	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040	UT-DD-EN-0011	<p>Interaction with DLC 3-2: DLC overrides Discovery of non-MR peer</p> <ul style="list-style-type: none"> <li>• MR node with interface N1</li> <li>• non-MR peer with interface P1, P2, P3</li> <li>• DLC configure non-MR peer on node with interfaces P1, P2, P4*.</li> <li>• Discover P1 from node.</li> <li>• Verify that node sees one non-MR peer with three NIDS: P1, P2, P4*.</li> <li>• Verify presence of error messages on node (error code is <code>-EPERM</code>): <ul style="list-style-type: none"> <li>◦ Error adding NID P3 to peer P1: -1</li> <li>◦ Error deleting NID P3 from peer P1: -1</li> </ul> </li> </ul>
dyn-060	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040, dyn-065	UT-DD-EN-0012	<p>Interaction with DLC 3-3: DLC overrides Discovery of MR peer with primary conflict</p> <ul style="list-style-type: none"> <li>• MR node with interface N1</li> <li>• MR peer with interface P1, P2, P3</li> <li>• DLC configure MR peer on node with interfaces P2, P3, P4*.</li> <li>• Discover P2 from node.</li> <li>• Verify that node sees one MR peer with three NIDS: P2, P3, P4*.</li> <li>• Verify presence of error message on node (error code is <code>-EEXIST</code>): <ul style="list-style-type: none"> <li>◦ Primary NID error P2 versus P1: -17</li> </ul> </li> </ul>
dyn-060	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040, dyn-065	UT-DD-EN-0013	<p>Interaction with DLC 3-4: DLC overrides Discovery of non-MR peer with primary conflict</p> <ul style="list-style-type: none"> <li>• MR node with interface N1</li> <li>• non-MR peer with interface P1, P2, P3</li> <li>• DLC configure non-MR peer on node with interfaces P2, P3, P4*.</li> <li>• Discover P2 from node.</li> <li>• Verify that node sees one MR peer with three NIDS: P2, P3, P4*.</li> <li>• Verify presence of error message on node (error code is <code>-EEXIST</code>): <ul style="list-style-type: none"> <li>◦ Primary NID error P2 versus P1: -17</li> </ul> </li> </ul>

dyn-060	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040	UT-DD-EN-0014	<p>Interaction with DLC 3-5: "push MR bit" exception to DLC overrides Discovery</p> <ul style="list-style-type: none"> <li>• MR node with interface N1</li> <li>• MR peer with interface P1, P2, P3</li> <li>• DLC configure non-MR peer on node with interfaces P1, P2, P4*.</li> <li>• Discover N1 from peer.</li> <li>• Verify that node sees one MR peer with three NIDS: P1, P2, P4*.</li> <li>• Verify presence of error message on node (error code is -EEXIST): <ul style="list-style-type: none"> <li>◦ Push says P1 is Multi-Rail, DLC says not</li> <li>◦ Error adding NID P3 to peer P1: -1</li> <li>◦ Error deleting NID P3 from peer P1: -1</li> </ul> </li> </ul>
dyn-060	dyn-005, dyn-015, dyn-025, dyn-030, dyn-035, dyn-040, dyn-065	UT-DD-EN-0015	<p>Interaction with DLC 3-6: "push MR bit" exception to DLC overrides Discovery</p> <ul style="list-style-type: none"> <li>• MR node with interface N1</li> <li>• MR peer with interface P1, P2, P3</li> <li>• DLC configure non-MR peer on node with interfaces P2, P3, P4*.</li> <li>• Discover N1 from peer.</li> <li>• Verify that node sees one MR peer with three NIDS: P2, P3, P4*.</li> <li>• Verify presence of error message on node (error code is -EEXIST): <ul style="list-style-type: none"> <li>◦ Push says P2 is Multi-Rail, DLC says not</li> <li>◦ Primary NID error P2 versus P1: -17</li> </ul> </li> </ul>

Tests with discovery disabled. Note that disabling discovery does not fully disable it. The MR capable node will continue to process pushes, and if there is a problem with a push it will ping the originator to obtain the information.

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
	dyn-005, dyn-025, dyn-030	UT-DD-DIS-0001	<p>Discovery disabled 4-1: discovery of an MR peer via its primary</p> <ul style="list-style-type: none"> <li>• MR Node with interfaces N1, N2</li> <li>• MR Peer with interfaces P1, P2, P3</li> <li>• Discovery is disabled on Node</li> <li>• Ping P1 from node</li> <li>• Ping P2 from node</li> <li>• Verify that node sees two different peers: P1, P2.</li> <li>• Discover P1 from node</li> <li>• Verify that node sees two different peers: P1, P2.</li> <li>• Verify that peer sees node as one peer with one NID: N1</li> </ul>
	dyn-005, dyn-025, dyn-030	UT-DD-DIS-0002	<p>Discovery disabled 4-2: discovery of an MR peer via a secondary</p> <ul style="list-style-type: none"> <li>• MR Node with interfaces N1, N2</li> <li>• MR Peer with interfaces P1, P2, P3</li> <li>• Discovery is disabled on Node</li> <li>• Ping P1 from node</li> <li>• Ping P2 from node</li> <li>• Verify that node sees two different peers: P1, P2.</li> <li>• Discover P2 from node</li> <li>• Verify that node sees two different peers: P1, P2.</li> <li>• Verify that peer sees node as one peer with one NID: N1</li> </ul>
	dyn-005, dyn-025, dyn-030	UT-DD-DIS-0003	<p>Discovery disabled 4-3: discovery of an MR peer via a tertiary.</p> <ul style="list-style-type: none"> <li>• MR Node with interfaces N1, N2</li> <li>• MR Peer with interfaces P1, P2, P3</li> <li>• Discovery is disabled on Node</li> <li>• Ping P1 from node</li> <li>• Ping P2 from node</li> <li>• Verify that node sees two different peers: P1, P2.</li> <li>• Discover P3 from node</li> <li>• Verify that node sees three different peers: P1, P2, P3.</li> <li>• Verify that peer sees node as one peer with one NID: N1</li> </ul>
	dyn-005, dyn-025, dyn-030	UT-DD-DIS-0004	<p>Discovery disabled 4-4: implicit discovery of an MR peer</p> <ul style="list-style-type: none"> <li>• MR Node with interfaces N1, N2</li> <li>• MR Peer with interfaces P1, P2, P3</li> <li>• Discovery is disabled on both Node and Peer</li> <li>• Force some filesystem traffic between node and peer.</li> <li>• Verify that node sees one peer with one NID: P1.</li> <li>• Verify that peer sees node as one peer with one NID: N1.</li> </ul>

dyn-020	dyn-005, dyn-025, dyn-030, dyn-055	UT-DD-DIS-0005	Discovery disabled 4-5: implicit discovery of an MR peer.  (This test shows that if discovery is enabled on either node or peer, it happens on both.)  <ul style="list-style-type: none"> <li>• MR Node with interfaces N1, N2</li> <li>• MR Peer with interfaces P1, P2, P3</li> <li>• Discovery is disabled on Node</li> <li>• Force some filesystem traffic between node and peer.</li> <li>• Verify that node sees one MR peer with NIDs: P1, P2, P3</li> <li>• Verify that peer sees node as one peer with NIDs: N1, N2.</li> </ul>
dyn-020	dyn-005, dyn-025, dyn-030, dyn-055	UT-DD-DIS-0006	Discovery disabled 4-6: implicit discovery of an MR peer, > 16 interfaces.  (This test shows that if discovery is enabled on either node or peer, it happens on both, including retries required because buffers need to be extended.)  <ul style="list-style-type: none"> <li>• MR Node with interfaces N1, N2, ..., N17</li> <li>• MR Peer with interfaces P1, P2, ..., P17</li> <li>• Discovery is disabled on Node</li> <li>• Force some filesystem traffic between node and peer.</li> <li>• Verify that node sees one MR peer with all NIDs: P1, P2, ..., P17</li> <li>• Verify that peer sees node as one MR peer with all NIDs: N1, N2. ... , N17.</li> </ul>
dyn-060	dyn-005, dyn-025, dyn-030	UT-DD-DIS-0007	Disabled with DLC 5-1: DLC overrides Discovery of MR peer  <ul style="list-style-type: none"> <li>• MR node with interface N1</li> <li>• MR peer with interface P1, P2, P3</li> <li>• Discovery is disabled on node.</li> <li>• DLC configure MR peer on node with interfaces P1, P2, P4*.</li> <li>• Discover P1 from node.</li> <li>• Verify that node sees one MR peer with three NIDs: P1, P2, P4*.</li> <li>• No error messages should show up.</li> </ul>
dyn-060	dyn-005, dyn-025, dyn-030	UT-DD-DIS-0008	Disabled with DLC 5-2: DLC overrides Discovery of non-MR peer  <ul style="list-style-type: none"> <li>• MR node with interface N1</li> <li>• non-MR peer with interface P1, P2, P3</li> <li>• Discovery is disabled on node.</li> <li>• DLC configure non-MR peer on node with interfaces P1, P2, P4*.</li> <li>• Discover P1 from node.</li> <li>• Verify that node sees one non-MR peer with three NIDs: P1, P2, P4*.</li> <li>• No error messages should show up.</li> </ul>
dyn-060	dyn-005, dyn-025, dyn-030	UT-DD-DIS-0009	Disabled with DLC 5-3: DLC overrides Discovery of MR peer with primary conflict  <ul style="list-style-type: none"> <li>• MR node with interface N1</li> <li>• MR peer with interface P1, P2, P3</li> <li>• Discovery is disabled on node.</li> <li>• DLC configure MR peer on node with interfaces P2, P3, P4*.</li> <li>• Discover P2 from node.</li> <li>• Verify that node sees one MR peer with three NIDs: P2, P3, P4*.</li> <li>• No error messages should show up.</li> </ul>
dyn-060	dyn-005, dyn-025, dyn-030	UT-DD-DIS-0010	Disabled with DLC 5-4: DLC overrides Discovery of non-MR peer with primary conflict  <ul style="list-style-type: none"> <li>• MR node with interface N1</li> <li>• non-MR peer with interface P1, P2, P3</li> <li>• Discovery is disabled on node.</li> <li>• DLC configure non-MR peer on node with interfaces P2, P3, P4*.</li> <li>• Discover P2 from node.</li> <li>• Verify that node sees one MR peer with three NIDs: P2, P3, P4*.</li> <li>• No error messages should show up.</li> </ul>
dyn-060	dyn-005, dyn-025, dyn-030	UT-DD-DIS-0011	Disabled with DLC 5-5: "push MR bit" exception to DLC overrides Discovery  <ul style="list-style-type: none"> <li>• MR node with interface N1</li> <li>• MR peer with interface P1, P2, P3</li> <li>• Discovery is disabled on node.</li> <li>• DLC configure non-MR peer on node with interfaces P1, P2, P4*.</li> <li>• Discover N1 from peer.</li> <li>• Verify that node sees one MR peer with three NIDs: P1, P2, P4*.</li> <li>• Verify presence of error message on node (error code is -EPerm): <ul style="list-style-type: none"> <li>◦ Push says P1 is Multi-Rail, DLC says not</li> <li>◦ Error adding NID P3 to peer P1: -1</li> <li>◦ Error deleting NID P3 from peer P1: -1</li> </ul> </li> </ul>

dyn-060	dyn-005, dyn-025, dyn-030, dyn-065	UT-DD-DIS-0012	<p>Disabled with DLC 5-6: "push MR bit" exception to DLC overrides Discovery</p> <ul style="list-style-type: none"> <li>• MR node with interface N1</li> <li>• MR peer with interface P1, P2, P3</li> <li>• Discovery is disabled on node.</li> <li>• DLC configure non-MR peer on node with interfaces P2, P3, P4*.</li> <li>• Discover N1 from peer.</li> <li>• Verify that node sees one MR peer with three NIDS: P2, P3, P4*.</li> <li>• Verify presence of error message on node (error code is -EXIST): <ul style="list-style-type: none"> <li>◦ Push says P2 is Multi-Rail, DLC says not</li> <li>◦ Primary NID error P2 versus P1: -17</li> </ul> </li> </ul>
---------	------------------------------------	----------------	---

## Debugging Requirements

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
dbg-005	dbg-010, dbg-015, dbg-020, dbg-025, dbg-030, dbg-035, dbg-080	UT-0325	<ul style="list-style-type: none"> <li>• dump per NI statistics <ul style="list-style-type: none"> <li>◦ transmitted</li> <li>◦ received</li> <li>◦ dropped</li> <li>◦ timeouts</li> <li>◦ state</li> </ul> </li> </ul>
dbg-040	dbg-080, dbg-095	UT-0330	<ul style="list-style-type: none"> <li>• configure multiple NIs</li> <li>• run traffic</li> <li>• dump stats on all NIs</li> </ul>
dbg-040	dbg-080	UT-0335	<ul style="list-style-type: none"> <li>• configure multiple NIs</li> <li>• run traffic</li> <li>• dump stats on all NIs</li> <li>• Filter on specific NID</li> </ul>
dbg-045	dbg-080	UT-0340	<ul style="list-style-type: none"> <li>• dump LNet level statistics</li> </ul>
dbg-050	dbg-080, dbg-100	UT-0345	<ul style="list-style-type: none"> <li>• configure multiple peers</li> <li>• start traffic</li> <li>• dump per peer statistics</li> </ul>
dbg-110		UT-0350	<ul style="list-style-type: none"> <li>• configure multiple NIs</li> <li>• toggle their state from ACTIVE to DOWN</li> <li>• confirm that state change is being printed to console.</li> </ul>
dbg-115		UT-0355	<ul style="list-style-type: none"> <li>• configure an MR system</li> <li>• start traffic</li> <li>• bring down an NI</li> <li>• confirm that messages indicating that another NI/peer is being used is printed.</li> </ul>
dbg-120		UT-0360	<ul style="list-style-type: none"> <li>• Configure an MR system</li> <li>• run traffic</li> <li>• stop traffic</li> <li>• dump NI statistics</li> <li>• dump peer statistics</li> <li>• dump LNet level statistics</li> <li>• zero out stats</li> <li>• dump all statistics above to confirm they've been zeroed out.</li> </ul>

## Network interface Health

## Backwards Compatibility Requirements

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
------------------------	--------------------------	--------------	-----------------------

bck-025		UT-0365	<ul style="list-style-type: none"> <li>• Configure an MR client</li> <li>• Configure an MR OSS</li> <li>• Configure a non-MR MDS/MGS</li> <li>• create an FS</li> <li>• Run IO tests</li> <li>• Make sure that MR clients/OSS integrates seamlessly in the system.</li> </ul>
---------	--	---------	---

## Performance Requirements

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
		UT-0370	<p>Testing reconnects. In Large clusters it is possible that servers might need to handle a burst of client connects.</p> <p>The performance of such scenarios needs to be quantified.</p>

## Misc Error Scenarios

Primary Requirement ID	Secondary Requirement ID	Unit Test ID	Unit Test Description
			<ul style="list-style-type: none"> <li>• mount an File System</li> <li>• delete all networks from the MDS</li> </ul>
			<ul style="list-style-type: none"> <li>• mount an File System</li> <li>• delete all networks from the MDS</li> </ul>
			<ul style="list-style-type: none"> <li>• mount an File System</li> <li>• delete all networks from the MDS</li> </ul>
			<ul style="list-style-type: none"> <li>• mount an FS</li> <li>• reboot the MDS</li> <li>• mount the MDS again</li> </ul>
			<ul style="list-style-type: none"> <li>• mount an FS</li> <li>• reboot the OSS</li> <li>• mount the OSS again</li> </ul>
			<ul style="list-style-type: none"> <li>• mount an FS</li> <li>• reboot the Client</li> <li>• mount the Client again</li> </ul>
			<ul style="list-style-type: none"> <li>• mount an FS with no modprobe.conf configured</li> <li>• the tcp network with a default tcp interface should be configured</li> <li>• add the same interface again via DLC</li> <li>• This operation should be a no-op</li> </ul>