

## **PREFACE**

### **SEGMENT ROUTING POC**

---

#### **1. Technical Details**

- a) Assumptions**
- b) Device Platform Used for Testing**
- c) NSO Version**
- d) NED Version**
- e) NSO Packages**
- f) Devices List**
- g) Migration Strategy or Steps**

#### **2. Use Cases and Model Representations**

##### **1. Brown Field Deployment Topology**

- i. Model: SR-infrastructure**
- ii. Configuration of model in NSO**

##### **2. Configuration of segment routing on brownfield deployments using sr-infrastructure**

- i. Model: Segment Routing**
- ii. Configuration of model in NSO**
- iii. Addition of New Router with same instance-name in the Existing Brownfield Topology**
- iv. Configuration of model in NSO**
- v. Device Modifications**

##### **3. Addition of New Routers to Brown Field Deployment Topology**

- i. Model: Segment Routing**
- ii. Configuration of model in NSO**
- iii. Device Modifications**

##### **4. Configuration of Topology Independent Loop Free Alternative**

- i. Topology**
- ii. Model: TI-LFA**

- iii. Configuration in NSO using custom instance (brownfield) and all interfaces
- iv. Device Modifications
- v. Configuration in NSO using brown field and specifying fewer interfaces
- vi. Device Modifications

## 5. Segment Routing Mapping Server

- i. Model: SR-MS
- ii. Configuration of model in NSO

## 1. TECHNICAL DETAILS

---

### a. Assumptions:

- Router isis instance-name: We assume that this instance name already exists and is active for a customer deployment as we sought to migrate.
- Interfaces under Router isis instance name should be already present. It may not have ti-lfa configurations, which nso would tend to configure on them.
- Loopback used for segment routing is assumed to be already present on the device.
- Compass Service Models/Configurations that get pushed are platform independent. They should work on all the platforms that run ios-xr.

### b. Device Platform Used for Testing

```
RP/0/RP0/CPU0:PE4_Zermat#show version
Thu Jun 21 21:51:49.783 UTC
Cisco IOS XR Software, Version 6.5.1.24I
Copyright (c) 2013-2018 by Cisco Systems, Inc.
```

#### Build Information:

```
Built By   : ahoang
Built On   : Tue May 29 13:04:05 PDT 2018
Build Host : iox-ucs-025
Workspace  : /auto/iox-ucs-025-san2/prod/6.5.1.24I.SIT_IMAGE/ncs5500/ws
Version    : 6.5.1.24I
Location   : /opt/cisco/XR/packages/
Label      : 6.5.1.24I
```

```
cisco NCS-5500 () processor
System uptime is 1 week 6 days 16 hours 36 minutes
```

## TECHNICAL DETAILS

---

### RP/0/RP0/CPU0:PE4\_Zermat#show platform

Thu Jun 21 21:52:02.701 UTC

Node	Type	State	Config state
0/RP0/CPU0	NCS-5502-SE(Active)		IOS XR RUN NSHUT
0/RP0/NPU0	Slice	UP	
0/RP0/NPU1	Slice	UP	
0/RP0/NPU2	Slice	UP	
0/RP0/NPU3	Slice	UP	
0/RP0/NPU4	Slice	UP	
0/RP0/NPU5	Slice	UP	
0/RP0/NPU6	Slice	UP	
0/RP0/NPU7	Slice	UP	
0/FT0	NC55-2RU-FAN-FW		OPERATIONAL NSHUT
0/FT1	NC55-2RU-FAN-FW		OPERATIONAL NSHUT
0/FT2	NC55-2RU-FAN-FW		OPERATIONAL NSHUT
0/PM1	NC55-2KW-ACFW		OPERATIONAL NSHUT
0/PM3	NC55-2KW-ACFW		OPERATIONAL NSHUT

RP/0/RP0/CPU0:PE4\_Zermat#exit

- c. **NSO Version:** 4.6 or above
- d. **NED :** There is no official ned being used. A Netconf NED named Prouter-ned is part of packages for IOSXR Devices . NED version 1.0.
- e. **NSO Packages**

```
admin@ncs> show packages package oper-status up
```

NAME	UP
disable-ldp	X
prouter-ned	X
resource-manager	X
sr	X
sr-ms	X
ti-fla	X

## TECHNICAL DETAILS

---

### f. Devices list (netsim netconf devices)

**admin@ncs> show devices list**

NAME	ADDRESS	DESCRIPTION	NED ID	ADMIN STATE
P-0	127.0.0.1	-	netconf	unlocked
P-1	127.0.0.1	-	netconf	unlocked
P-2	127.0.0.1	-	netconf	unlocked
P-3	127.0.0.1	-	netconf	unlocked
P-4	127.0.0.1	-	netconf	unlocked

### g. Migration Strategy or Steps

Customer is expected to run a LDP/RSVP Networks.

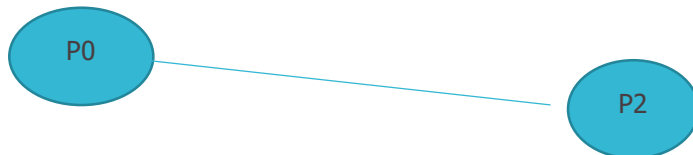
Migration done in form of small steps. These are actually the name of NSO Packages and needs in the configured in the order described below after verification at each step. Verification is left to the user at moment. The verification is to ensure that the configuration is effective in the network and could be seen as part of the device operational.

- **SR** : prefix-sid are assigned to devices.
- **TI-LFA** ( Topology Independent – Loop Free Alternative) : every node/link can be configured a backup path. Hence on a node failure it needs to reconverge so a backup path is configured.
- **SR-MS** (Segment Routing – Mapping Server ) : Label path - switch between LDP and SR networks. A network could both SR and LDP. To ensure that the switching works across this two kinds of networks a label path is statically assigned. I.e. for this ldp prefix use this sr prefix. Idea is to advertise ldp prefixes into sr.
- **Disable-ldp** (Label Distribution Protocol)

## USE CASES AND MODELS REPRESENTATION

---

1. Brown Field Deployment Topology (Example): Assume the following P Routers has the same instance-name and loopback.



All the P-Routers that share the same instance name and loopback can be configured in the infrastructure with this details. If there is a probability that most of the routers in the network would have same details, then the sr-infrastructure configuration can be re-used and pointed back to it. This saves User time to re-enter same details again and again for each router.

### a. Model: SR-infrastructure

```
module: infrastructure
  +--rw sr-infrastructure!
    +--rw sr-global-block-pools* [name]
      | +--rw name -> /ralloc:resource-pools/idalloc:id-pool/name
      +--rw instance-name?      string
        +--rw loopback?         uint32
```

### b. Configuration of the model in NSO

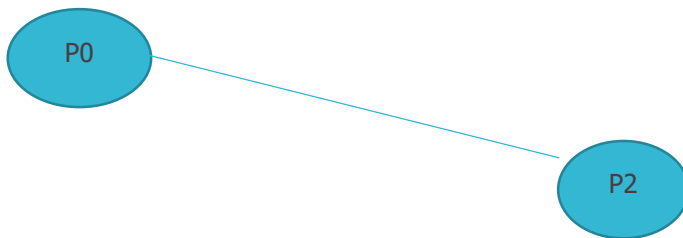
```
admin@ncs% set resource-pools id-pool sr-pool range start 16000 end 23999
[ok][2018-07-30 21:00:57]
admin@ncs% set sr-infrastructure instance-name CORE loopback 0 sr-global-block-pools
sr-pool
[ok][2018-07-30 21:01:44]

admin@ncs% commit
```

## USE CASES AND MODEL REPRESENTATIONS

---

### 2. CONFIGURATION OF SEGMENT ROUTING ON BROWNFIELD DEPLOYMENTS USING SR-INFRASTRUCTURE



#### Model: Segment Routing

module: sr

augment /ncs:services:

+--rw sr\* [name]

+--rw name string

+--rw router\* [device-name]

+--rw device-name -> /ncs:devices/device/name

+--rw prefix-preference

| +--rw (prefix-choice)?

| | +--:(auto-assign-prefix-sid)

| | | +--rw auto-assign-prefix-sid? empty

| | +--:(assign-prefix-sid)

| | | +--rw assign-prefix-sid? uint16

+--rw instance-preference

+--rw (instance-choice)?

+--:(use-sr-infrastructure)

| +--rw use-sr-infrastructure? empty

+--:(custom-instance)

+--rw custom-instance

+--rw instance-name? string

+--rw loopback? uint32

## USE CASES AND MODEL REPRESENTATIONS

---

### Configuration of the model in NSO

```
services {
+  sr ACME-west {
+    router P-0 {
+      prefix-preference {
+        auto-assign-prefix-sid;
+      }
+      instance-preference {
+        use-sr-infrastructure;
+      }
+    }
+    router P-2 {
+      prefix-preference {
+        auto-assign-prefix-sid;
+      }
+      instance-preference {
+        use-sr-infrastructure;
+      }
+    }
+  }
}
```

In the above scenario; we are asking NSO to configure Router P-0 and P-2. As part of the configuration NSO will auto assign prefix sid as indicated in the model configuration prefix-preference “auto-assign-prefix-sid”. NSO uses a resource allocator to pick IDs from a pool. These pool is already defined in the SR-INFRASTRUCTURE model. If no pool is assigned, NSO will result into error as it is missing the details of the pool. Whenever this option auto-assign-prefix-sid is selected NSO will pick up next available ID from the pool until the pool is exhausted.



## USE CASES AND MODEL REPRESENTATIONS

---

### TEMPLATE

```
<config-template xmlns="http://tail-f.com/ns/config/1.0">
  <devices xmlns="http://tail-f.com/ns/ncs">
    <device>
      <name>{$DEVICENAME}</name>
      <config>
        <isis xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-clns-isis-cfg">
          <instances>
            <instance>
              <instance-name>{$INSTANCE-NAME}</instance-name>
              <srgb>
                <lower-bound>{$SRGB-START}</lower-bound>
                <upper-bound>{$SRGB-END}</upper-bound>
              </srgb>
              <afs>
                <af>
                  <af-name>ipv4</af-name>
                  <saf-name>unicast</saf-name>
                  <af-data>
                    <segment-routing>
                      <prefix-sid-map>
                        <advertise-local/>
                      </prefix-sid-map>
                      <mpls>ldp</mpls>
                    </segment-routing>
                    <mpls>
                      <router-id>
                        <interface-name>Loopback{$LOOPBACK}</interface-name>
                      </router-id>
                      <level>
                        <level2>true</level2>
                      </level>
                    </mpls>
                  </af-data>
                </af>
              </afs>
              <interfaces>
                <interface>
                  <interface-name>Loopback{$LOOPBACK}</interface-name>
                  <interface-afs>
                    <interface-af>
                      <af-name>ipv4</af-name>
                      <saf-name>unicast</saf-name>
                      <interface-af-data>
                        <prefix-sid>
                          <type>absolute</type>
                          <value>{$PREFIX-SID}</value>
                          <php>enable</php>
                          <explicit-null>disable</explicit-null>
                          <nflag-clear>disable</nflag-clear>
                        </prefix-sid>
                      </interface-af-data>
                    </interface-af>
                  </interface-afs>
                </interface>
              </interfaces>
            </instance>
          </instances>
        </isis>
      </config></device></devices></config-template>
```

## USE CASES AND MODEL REPRESENTATIONS

---

### TEMPLATE VARIABLES

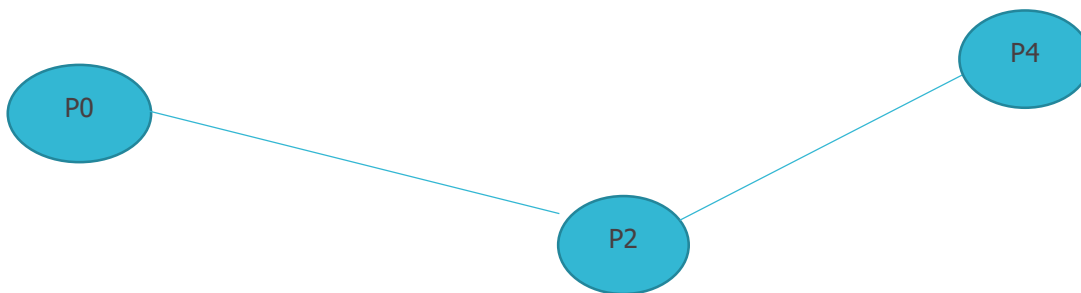
Variable name	Value
\$INSTANCE_NAME	SR-INFRASTRUCTURE INSTANCE-NAME OR USER DEFINED
\$SRGB-START	SR-INFRASTRUCTURE RESOURCE POOL START RANGE VALUE
\$SRGB-END	SR-INFRASTRUCTURE RESOURCE POOL END RANGE VALUE
\$LOOPBACK	LOOPBACK ID
\$PREFIX-SID	NSO ALLOCATED OR USER PROVIDED

## USE CASES AND MODEL REPRESENTATIONS

---

### Addition of New Router with same instance –name in the Existing Brownfield

**Topology :** Administrator can choose to pick a new sid prefix for this newly added router as described below.



### Model : Segment Routing

```
module: sr
augment /ncs:services:
  +--rw sr* [name]
  +--rw name string
  +--rw router* [device-name]
    +--rw device-name -> /ncs:devices/device/name
    +--rw prefix-preference
      | +--rw (prefix-choice)?
      |   +--:(auto-assign-prefix-sid)
      |   | +--rw auto-assign-prefix-sid? empty
      |   +--:(assign-prefix-sid)
      |   +--rw assign-prefix-sid? uint16
    +--rw instance-preference
      +--rw (instance-choice)?
      +--:(use-sr-infrastructure)
      | +--rw use-sr-infrastructure? empty
      +--:(custom-instance)
      +--rw custom-instance
        +--rw instance-name? string
        +--rw loopback? uint32
```

## USE CASES AND MODEL REPRESENTATIONS

---

In the above scenario, Users wishes to have his own choice of prefix-sid which he/she enters and User also wishes to use the same sr-infrastructure details (instance name and loopback) as the device has it. In this case NSO will not auto assign prefix sid from a pool. It is to the user onus to ensure that NSO has not already assigned the id he/she enters.

### Configuration of the model in NSO

```
services {
+  sr ACME-west {
+    router P-4 {
+      prefix-preference {
+        assign-prefix-sid 22004;
+      }
+      instance-preference {
+        use-sr-infrastructure;
+      }
+    }
+  }
+ }
}
```

### DEVICE MODIFICATIONS

```
admin@ncs% request services sr ACME-west get-modifications
cli {
  local-node {
    data devices {
      device P-0 {
        config {
          clns-isis-cfg:isis {
            instances {
```

```

instance CORE {
+     srgb {
+
+         lower-bound 16000;
+         upper-bound 23999;
+     }
    afs {
+     af ipv4 unicast {
+         af-data {
+             segment-routing {
+                 prefix-sid-map {
+                     advertise-local;
+                 }
+                 mpls ldp;
+             }
+             mpls {
+                 router-id {
+                     interface-name Loopback0;
+                 }
+                 level {
+                     level2 true;
+                 }
+             }
+         }
+     }
+ }
    interfaces {
        interface Loopback0 {
            interface-afs {
+                interface-af ipv4 unicast {
+                    interface-af-data {
+                        prefix-sid {
+                            type absolute;
+                            value 16000;
+                            php enable;
+                            explicit-null disable;
+                            nflag-clear disable;

```



```

+             level2 true;
+             }
+         }
+     }
+ }
+ interfaces {
+     interface Loopback0 {
+         interface-afs {
+             interface-af ipv4 unicast {
+                 interface-af-data {
+                     prefix-sid {
+                         type absolute;
+                         value 16001;
+                         php enable;
+                         explicit-null disable;
+                         nflag-clear disable;
+                     }
+                 }
+             }
+         }
+     }
+     running;
+ }
+ running;
device P-4 {
    config {
        clns-isis-cfg:isis {
            instances {
+                instance CORE {
+                    srgb {
+                        lower-bound 16000;
+                        upper-bound 23999;
+                    }
+                    afs {
+                        af ipv4 unicast {
+                            af-data {

```

```

+         segment-routing {
+             prefix-sid-map {
+                 advertise-local;
+             }
+             mpls ldp;
+         }
+         mpls {
+             router-id {
+                 interface-name Loopback0;
+             }
+             level {
+                 level2 true;
+             }
+         }
+     }
+ }
+ interfaces {
+     interface Loopback0 {
+         interface-afs {
+             interface-af ipv4 unicast {
+                 interface-af-data {
+                     prefix-sid {
+ +                 type absolute;
+ +                 value 22004;
+ +                 php enable;
+ +                 explicit-null disable;
+ +                 nflag-clear disable;
+                     }
+                 }
+             }
+         }
+         running;
+     }
+ }
+ running;
+ }

```



```

    }
  }
}
}
resource-pools {
  id-pool sr-pool {
+   allocation ACME-west-P-0 {
+     username admin;
+     allocating-service /services/sr:sr[name='ACME-west'];
+     request {
+       sync false;
+     }
+   }
+   allocation ACME-west-P-2 {
+     username admin;
+     allocating-service /services/sr:sr[name='ACME-west'];
+     request {
+       sync false;
+     }
+   }

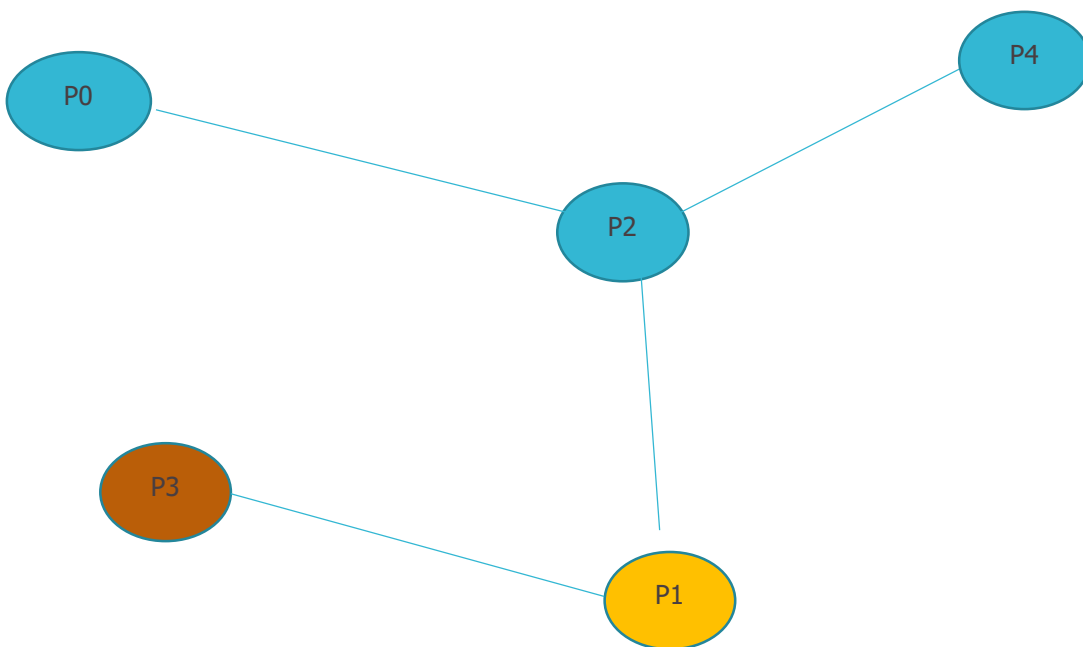
+   allocation ACME-west-P-3 {
+     username admin;
+     allocating-service /services/sr:sr[name='ACME-west'];
+     request {
+       sync false;

```

## USE CASES AND MODEL REPRESENTATIONS

---

3. **Addition of new routers to Brown Field Deployment Topology(Example):** A new set of brown field P Routers has been brought into the existing topology that has different instance name and loopback.



Administrator can now use a different instance name and loopback depending on the P-Router configuration as shown below.

## USE CASES AND MODEL REPRESENTATIONS

---

### Model: Segment Routing

module: sr

augment /ncs:services:

+--rw sr\* [name]

+--rw name string

+--rw router\* [device-name]

+--rw device-name -> /ncs:devices/device/name

+--rw prefix-preference

| +--rw (prefix-choice)?

| | +--:(auto-assign-prefix-sid)

| | | +--rw auto-assign-prefix-sid? empty

| | +--:(assign-prefix-sid)

| | | +--rw assign-prefix-sid? uint16

+--rw instance-preference

+--rw (instance-choice)?

+--:(use-sr-infrastructure)

| +--rw use-sr-infrastructure? empty

+--:(custom-instance)

+--rw custom-instance

+--rw instance-name? string

+--rw loopback? uint32

### Configuration of Model in NSO

```
admin@ncs% show | compare
```

```
services {
```

```
+ sr ACME-west {
```

```
+   router P-1 {
```

```
+     prefix-preference {
```

```
+       assign-prefix-sid 22001;
```

```
+     }
```

```
+     instance-preference {
```

```
+       custom-instance {
```

## USE CASES AND MODEL REPRESENTATIONS

---

```
+         instance-name foobar;
+         loopback 1;
+     }
+ }
+ router P-3 {
+     prefix-preference {
+         auto-assign-prefix-sid;
+     }
+     instance-preference {
+         custom-instance {
+             instance-name westzone;
+             loopback 1;
+         }
+     }
+ }
+ }
```

In this scenario, user has brought in two different routers as part of existing topology. Each of the routers shares a different instance-name and loopback that is different from one defined in the SR-infrastructure. In addition the user wishes to let NSO assign a prefix-sid on one of the Routers for whatsoever reason while on the other Router User wants to self assign a prefix sid id. NSO will only assign one id as requested, for the other router NSO will just take the user input and assign to the router.

## USE CASES AND MODEL REPRESENTATIONS

---

### TEMPLATE

```
<config-template xmlns="http://tail-f.com/ns/config/1.0">
  <devices xmlns="http://tail-f.com/ns/ncs">
    <device>
      <name>{$DEVICENAME}</name>
      <config>
        <isis xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-clns-isis-cfg">
          <instances>
            <instance>
              <instance-name>{$INSTANCE-NAME}</instance-name>
              <srgb>
                <lower-bound>{$SRGB-START}</lower-bound>
                <upper-bound>{$SRGB-END}</upper-bound>
              </srgb>
              <afs>
                <af>
                  <af-name>ipv4</af-name>
                  <saf-name>unicast</saf-name>
                  <af-data>
                    <segment-routing>
                      <prefix-sid-map>
                        <advertise-local/>
                      </prefix-sid-map>
                      <mpls>ldp</mpls>
                    </segment-routing>
                    <mpls>
                      <router-id>
                        <interface-name>Loopback{$LOOPBACK}</interface-name>
                      </router-id>
                      <level>
                        <level2>>true</level2>
                      </level>
                    </mpls>
                  </af-data>
                </af>
              </afs>
              <interfaces>
                <interface>
                  <interface-name>Loopback{$LOOPBACK}</interface-name>
                  <interface-afs>
                    <interface-af>
                      <af-name>ipv4</af-name>
                      <saf-name>unicast</saf-name>
                      <interface-af-data>
                        <prefix-sid>
                          <type>absolute</type>
                          <value>{$PREFIX-SID}</value>
                          <php>enable</php>
                          <explicit-null>disable</explicit-null>
                          <nflag-clear>disable</nflag-clear>
                        </prefix-sid>
                      </interface-af-data>
                    </interface-af>
                  </interface-afs>
                <running/>
              </interface>
            </interfaces>
          <running/>
        </instance>
      </instances>
    </isis>
  </config></device></devices></config-template>
```

## USE CASES AND MODEL REPRESENTATIONS

---

### TEMPLATE VARIABLES

Variable name	Value
\$INSTANCE_NAME	SR-INFRASTRUCTURE INSTANCE-NAME OR USER DEFINED
\$SRGB-START	SR-INFRASTRUCTURE RESOURCE POOL START RANGE VALUE
\$SRGB-END	SR-INFRASTRUCTURE RESOURCE POOL END RANGE VALUE
\$LOOPBACK	LOOPBACK ID
\$PREFIX-SID	NSO ALLOCATED OR USER PROVIDED

### DEVICE MODIFICATIONS

```
admin@ncs% request services sr ACME-west get-modifications
```

```
cli {  
  local-node {  
    data devices {  
      device P-1 {  
        config {  
          clns-isis-cfg:isis {  
            instances {  
              + instance foobar {  
                + srgb {  
                + lower-bound 16000;              }  
            }  
          }  
        }  
      }  
    }  
  }  
}
```

```

+         upper-bound 23999;
+     }
+     afs {
+         af ipv4 unicast {
+             af-data {
+                 segment-routing {
+                     prefix-sid-map {
+                         advertise-local;
+                     }
+                 }
+                 mpls ldp;
+             }
+             mpls {
+                 router-id {
+                     interface-name Loopback1;
+                 }
+                 level {
+                     level2 true;
+                 }
+             }
+         }
+     }
+ }
+ interfaces {
+     interface Loopback1 {
+         interface-afs {
+             interface-af ipv4 unicast {
+                 interface-af-data {
+                     prefix-sid {
+                         type absolute;
+                         value 22001;
+                         php enable;
+                         explicit-null disable;
+                         nflag-clear disable;
+                     }
+                 }
+             }
+         }
+     }
+ }

```





```

+         interface Loopback1 {
+             interface-afs {
+                 interface-af ipv4 unicast {
+                     interface-af-data {
+                         prefix-sid {
+                             type absolute;
+                             value 16002;
+                             php enable;
+                             explicit-null disable;
+                             nflag-clear disable;
+                         }
+                     }
+                 }
+             }
+         }
+         running;

```

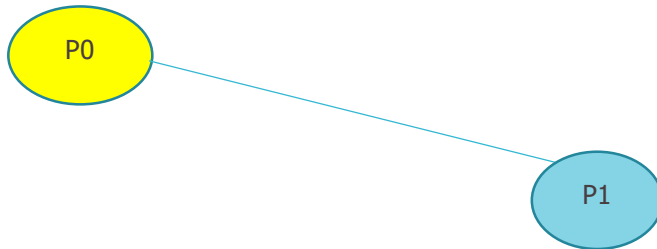
```

resource-pools {
  id-pool sr-pool {

    +   allocation ACME-west-P-3 {
    +     username admin;
    +     allocating-service /services/sr:sr[name='ACME-west'];
    +     request {
    +       sync false;
    +     }
    +   }
  }
}

```

### 4. Configuration of Topology Independent Loop Free Alternative



Model

```
module: ti-lfa
augment /ncs:services:
  +--rw ti-lfa* [name]+--rw address-family?      string
  +--rw router* [device-name]
  +--rw device-name      -> /ncs:devices/device/name
  +--rw instance-name-preference
  | +--rw (instance-name-choice)?
  |   +--:(use-sr-infrastructure)
  |   | +--rw use-sr-infrastructure? empty
  |   +--:(custom-instance)
  |     +--rw custom-instance
  |       +--rw instance-name? string
  +--rw interface-preference
  +--rw (interfaces)?
  +--:(all-interfaces)
  | +--rw all-interfaces? empty
  +--:(select-interface)
  +--rw select-interface* [interface-type interface-id]
  +--rw interface-type  int-type
  +--rw interface-id    int-id
```

## USE CASES AND MODEL REPRESENTATIONS

---

The above model provides two alternatives to configure TI-LFA. One is to allow user to cherry pick interfaces of the device on which TI-FLA needs to be configured, the other is to configure all the interfaces.

### Configuration of Model in NSO

The following example uses a custom instance name while specifying all interfaces to be configured.

```
services {
+  ti-lfa ACME {
+    address-family ipv6;
+    router P-0 {
+      instance-name-preference {
+        custom-instance {
+          instance-name core;
+        }
+      }
+      interface-preference {
+        all-interfaces;
+      }
+    }
+  }
}
```

In this example, NSO uses the instance name provided that will be configured while NSO will also read all the interfaces under this instance name and configure the TI-LFA on each of these interfaces minus the Loopback interface.

## USE CASES AND MODEL REPRESENTATIONS

---

### TEMPLATE

```
<config-template xmlns="http://tail-f.com/ns/config/1.0">
  <devices xmlns="http://tail-f.com/ns/ncs">
    <device>
      <name>{$DEVICE}</name>
      <config>
        <isis xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-clns-isis-cfg">
          <instances>
            <instance>
              <instance-name>{$INSTANCE-NAME}</instance-name>
              <interfaces>
                <interface>
                  <interface-name>{$INTERFACE-NAME}{ $INTERFACE-
ID}</interface-name>
                  <interface-afs>
                    <interface-af>
                      <af-name>{$ADDRESS-FAMILY}</af-name>
                      <saf-name>unicast</saf-name>
                      <interface-af-data>
                        <interface-frr-table>
                          <frartilfa-types>
                            <frartilfa-type>
                              <level>not-set</level>
                            </frartilfa-type>
                          </frartilfa-types>
                          <frr-types>
                            <frr-type>
                              <level>not-set</level>
                              <type>per-prefix</type>
                            </frr-type>
                          </frr-types>
                        </interface-frr-table>
                      </interface-af-data>
                    </interface-af>
                  </interface-afs>
                <running/>
              </interface>
            </interfaces>
          <running/>
        </instance>
      </instances>
    </isis>
  </config>
</device>
</devices>
</config-template>
```

## USE CASES AND MODEL REPRESENTATIONS

---

### TEMPLATE VARIABLES

TEMPLATE VARIABLE	VALUE
\$INSTANCE-NAME	SR-INFRASTRUCUTRE OR USER PROVIDED
\$INTERFACE-NAME	USER PROVIDER OR NSO FINDS IT QUERYING DEVICE
\$INTERFACE-ID	USER PROVIDED OR NSO FINDS IT QUERYING DEVICE
ADDRESS-FAMILY	IPV4 OR IPV6

### DEVICE MODIFICATIONS

```
admin@ncs% request services ti-lfa ACME get-modifications
cli {
  local-node {
    data devices {
      device P-0 {
        config {
          clns-isis-cfg:isis {
            instances {
              instance core {
                interfaces {
                  interface Bundle-Ether511 {
                    interface-afs {
+                     interface-af ipv6 unicast {
+                     interface-af-data {
+                     interface-frr-table {
+                     frr-types {
```

```

+             frr-type not-set {
+                 type per-prefix;
+             }
+         }
+         frtilfa-types {
+             frtilfa-type not-set;
+         }
+     }
+ }
+     running;
+ }
interface TenGigabitEthernet2/1/1 {
    interface-afs {
+         interface-af ipv6 unicast {
+             interface-af-data {
+                 interface-frr-table {
+                     frr-types {
+                         frr-type not-set {
+                             type per-prefix;
+                         }
+                     }
+                     frtilfa-types {
+                         frtilfa-type not-set;
+                     }
+                 }
+             }
+         }
+     }
+     running;
+ }
interface TenGigabitEthernet2/1/2 {
    interface-afs {
+         interface-af ipv6 unicast {
+             interface-af-data {
+                 interface-frr-table {

```

```
+          frr-types {
+            frr-type not-set {
+              type per-prefix;
+            }
+          }
+          frtilfa-types {
+            frtilfa-type not-set;
+          }
+        }
+      }
+    }
+  }
+  running;
+}
+  running;
+}
```

## USE CASES AND MODEL REPRESENTATIONS

---

The following example uses an instance name defined in the sr-infrastructure while user cherry picks few interfaces to be configured that are part of the instance-name. User will not be able to provide a loopback.

```
admin@ncs% show | compare
services {
+  ti-lfa ACME {
+    address-family ipv4;
+    router P-0 {
+      instance-name-preference {
+        use-sr-infrastructure;
+      }
+      interface-preference {
+        select-interface Bundle-Ether 611;
+        select-interface Bundle-Ether 711;
+        select-interface Bundle-Ether 811;
+      }
+    }
+  }
}
[ok][2018-07-30 22:07:22]

[edit]
admin@ncs%
```

In this case NSO will send configure the interfaces specified above with ti-fla configuration.



## USE CASES AND MODEL REPRESENTATIONS

---

### DEVICE MODIFICATIONS

admin@ncs% request services ti-lfa ACME get-modifications

```
cli {
  local-node {
    data devices {
      device P-0 {
        config {
          clns-isis-cfg:isis {
            instances {
              instance CORE {
                interfaces {
                  interface Bundle-Ether611 {
                    interface-afs {
+                     interface-af ipv4 unicast {
+                     interface-af-data {
+                     interface-frr-table {
+                     frr-types {
+                       frr-type not-set {
+                         type per-prefix;
+                       }
+                     }
+                     frrtifa-types {
+                       frrtifa-type not-set;
+                     }
+                   }
+                 }
+               }
+             }
+           }
+         }
+       }
+     }
+   }
+   running;
+ }
+ interface Bundle-Ether711 {
+   interface-afs {
+     interface-af ipv4 unicast {
+     interface-af-data {
```

```

+         interface-frr-table {
+             frr-types {
+                 frr-type not-set {
+                     type per-prefix;
+                 }
+             }
+             frtilfa-types {
+                 frtilfa-type not-set;
+             }
+         }
+     }
+     running;
+ }
interface Bundle-Ether811 {
    interface-afs {
+         interface-af ipv4 unicast {
+             interface-af-data {
+                 interface-frr-table {
+                     frr-types {
+                         frr-type not-set {
+                             type per-prefix;
+                         }
+                     }
+                     frtilfa-types {
+                         frtilfa-type not-set;
+                     }
+                 }
+             }
+         }
+     }
+ }

```

## USE CASES AND MODEL REPRESENTATIONS

---

### 5. SEGMENT ROUTING MAPPING SERVER

#### a. Model : SR-MS

```
module: sr-ms
augment /ncs:services:
+--rw sr-ms* [name]

+--rw name                string
+--rw address-family?     string
+--rw ipv4-address?       inet:ipv4-address
+--rw prefix-length?      uint16
+--rw first-sid-value?    uint32
+--rw number-of-allocated-sids? uint32
+--rw router* [device-name]
+--rw device-name -> /ncs:devices/device/name
```

The above model takes in various input parameters from user and does template replacement. NO NSO specific logic is done.

#### b. Configuration of Model in NSO

```
services {
+  sr-ms ACME {
+    address-family ipv4;
+    ipv4-address 192.168.10.1;
+    prefix-length 24;
+    first-sid-value 16001;
+    number-of-allocated-sids 80;
+    router P-0;
+  }
}
```

## USE CASES AND MODEL REPRESENTATIONS

---

### c. TEMPLATES

```
<config-template xmlns="http://tail-f.com/ns/config/1.0"
  servicepoint="sr-ms">
  <devices xmlns="http://tail-f.com/ns/ncs">
    <device>
      <name>{/router/device-name}</name>
      <config>
        <sr xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-segment-routing-ms-cfg">
          <mappings>
            <mapping>
              <af>{/address-family}</af>
              <ip>{/ipv4-address}</ip>
              <mask>{/prefix-length}</mask>
              <sid-start>{/first-sid-value}</sid-start>
              <sid-range>{/number-of-allocated-sids}</sid-range>
            </mapping>
          </mappings>
          <enable/>
        </sr>
      </config>
    </device>
  </devices>
</config-template>
```

### TEMPLATE VARIABLES

TEMPLATE VARIABLE	VALUE
DEVICE-NAME	USER PROVIDED
ADDRESS-FAMILY	USER PROVIDED ipv4 or ipv6
IPV4-ADDRESS	USER PROVIDED
PREFIX-LENGTH	USER PROVIDED
FIRST SID VALUE	USER PROVIDED

NUMBER OF ALLOCATED-SIDS	USER PROVIDED
--------------------------	---------------

#### d. DEVICE MODIFICATIONS

admin@ncs% request services sr-ms ACME get-modifications

```
cli {
  local-node {
    data devices {
      device P-0 {
        config {
          segment-routing-ms-cfg:sr {
            mappings {
+           mapping ipv4 192.168.10.1 24 {
+             sid-start 16001;
+             sid-range 80;
+           }
+         }
+       enable;
+     }
+   }
+ }
+ }
+ }
+ }
}
```