



# RedBox Configuration Guide

Application Note

CONFIDENTIAL

**AN1296**  
Rev. APPL-2024.06  
2024-06-28

---

## Table of Contents

1. Introduction .....	3
2. Limitations .....	3
3. RedBox Operating Modes .....	3
3.1. PRP-SAN .....	3
3.2. HSR-SAN .....	5
3.3. HSR-PRP .....	6
3.4. HSR-HSR .....	9
4. NodesTable and ProxyNodeTable .....	11
4.1. NodesTable .....	12
4.2. ProxyNodeTable .....	12
5. Supervision Frame Handling .....	13
6. RedBox Port Interfaces .....	16
7. Command Line Interface .....	18
7.1. Configuration .....	18
7.2. Status and Statistics .....	21
8. Examples .....	41
8.1. PRP-SAN .....	41
8.2. HSR-SAN .....	43
8.3. HSR-PRP .....	44
8.4. HSR-HSR .....	49

---

# 1. Introduction

This guide provides instructions on configuring Redundancy Box (RedBox) functionality on Microchip-based switches with RedBox support running IStax.

A RedBox is a device designed to connect PRP, HSR, or single-threaded networks.

PRP (Parallel Redundancy Protocol) and HSR (High-availability Seamless Redundancy) are defined by International Electrotechnical Commission (IEC) in the 62439-3 standard, edition 4, 2021-12. This will be referred to as "the standard" in this guide.

A corrigendum (IEC 62439-3:2021/COR1:2023) was issued in 2023, primarily detailing the translation of supervision frames in HSR-PRP RedBoxes. This will be referred to as "the corrigendum" in this guide.

## 2. Limitations

Please note that the current version of the IStax software does not support simultaneous operation of RedBox functionality and PTP on the same ports.

## 3. RedBox Operating Modes

A RedBox can function in one of the following four modes:

- PRP-SAN
- HSR-SAN
- HSR-PRP
- HSR-HSR

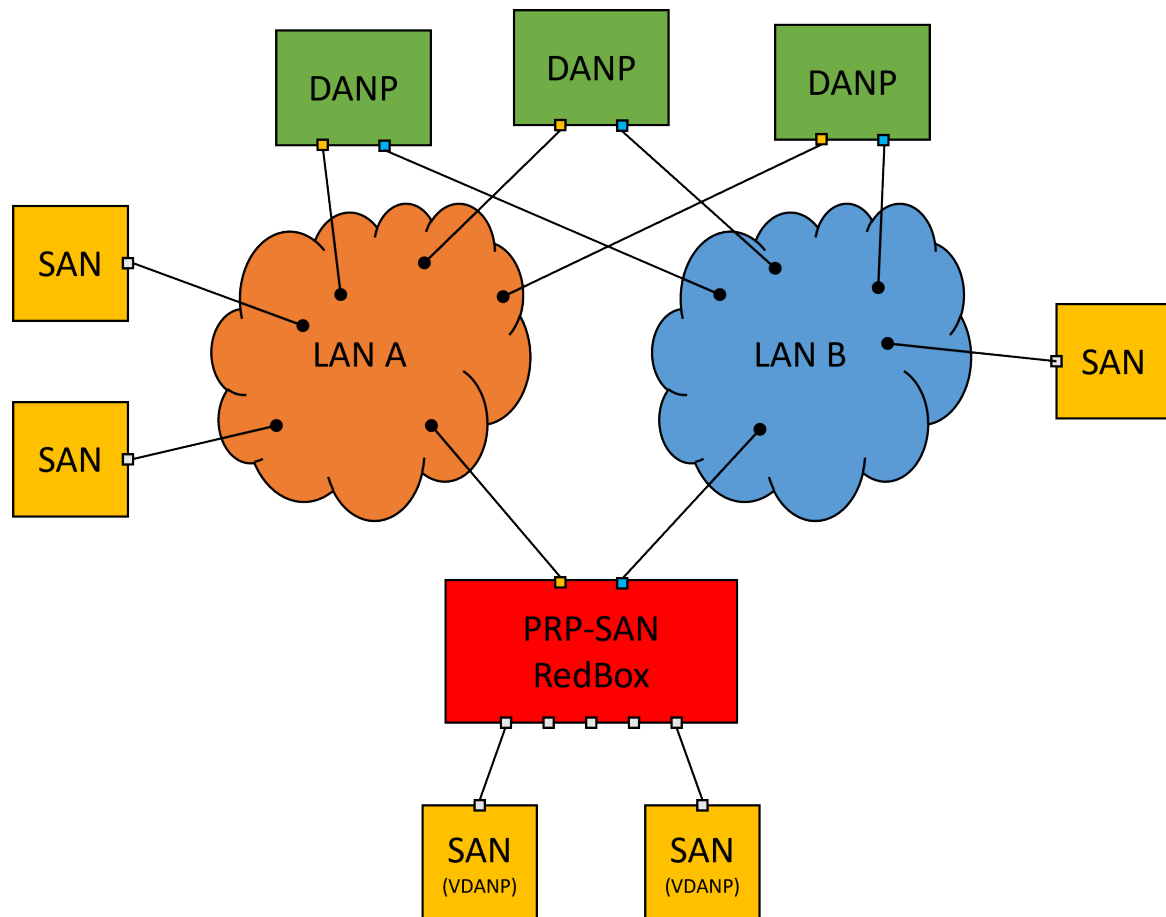
The standard refers to several optional HSR sub-modes. However, the IStax software only supports Mode H, which will be further explained in the subsequent HSR sections of this guide.

### 3.1. PRP-SAN

PRP is designed to offer fail-safe redundancy in Ethernet networks, ensuring immediate recovery post any failures.

Other redundancy protocols like STP, MRP, and ERPS require network reconfiguration (typically to unblock a blocked port) upon failure and recovery from failure before traffic flows again. Depending on protocol, recovery may take from a few milliseconds to several seconds.

PRP introduces redundancy at the node level by connecting two network interfaces (ports) to two separate and disjoint parallel networks, LAN A and LAN B. These two ports are referred to as Link Redundancy Entity (LRE) ports. The LRE port connected to LAN A is known as Port A, while the one connected to LAN B is termed Port B.



*Figure 1. A PRP-SAN RedBox connected to a PRP network*

Nodes that are connected to both LAN A and LAN B are termed Doubly Attached Nodes (DANs), and within a PRP network, these DANs are referred to as DANPs.

Singly Attached Nodes (SANs) are standard Ethernet devices with only one port connected to either LAN A or LAN B - or to the RedBox. When utilized as depicted in the figure, the RedBox is configured as a RedBox in PRP-SAN mode.

A PRP-SAN RedBox is also a DANP that connects to both LAN A and LAN B. In addition to this, it has one or more ports - on its bridge-side - that are simple network ports that may connect other SANs to the PRP network. The interface that connects Port A and Port B with the bridge-side is referred to as the interlink port or Port C.

All DANPs and all RedBoxes in the network can communicate, and will not lose connection if either of their LRE ports goes down.

SANs connected directly to LAN A or LAN B can only communicate with other nodes connected to the respective LAN.

SANs connected to the RedBox' bridge-side are known as Virtual DANs (VDANs) and in the PRP-case, they are called VDANPs. VDANPs can communicate with all SANs and all DANs in both LAN A and LAN B via the PRP-SAN RedBox.

A DANP sends the same frame simultaneously through its two LRE ports towards the destination node. A Redundancy Control Trailer (also known as Redundancy Check Trailer; RCT) containing a sequence number (RCT.SeqNr) is added to each frame copy

to help the destination node distinguish between duplicate frames. The first error-free frame copy that arrives at the destination DANP node or destination PRP RedBox gets the RCT removed and the destination DANP node sends the frame to its higher layers. If the destination node is a RedBox, the frame is forwarded through the interlink port to the bridge-side of the RedBox.

A DANP does not forward frames between its two LRE ports. When or if the second copy arrives, it is detected as a duplicate and gets discarded, provided it arrives before a configurable duplicate discard timer times out.

A RedBox forwards frames with an RCT added on behalf of the connected VDANPs and keeps track of each VDANPs next RCT.SeqNr.

A SAN connected to LAN A or LAN B only receives one copy, and since a SAN is RCT unaware, it will forward the frame as is to its higher layers. This will not affect protocols running on the SAN, because most (if not all) protocols are designed to ignore frame data beyond the frame's payload.

A VDANP also only receives one frame copy, since the RedBox removes duplicates - and strips the RCT.

### 3.2. HSR-SAN

HSR is designed to function in a ring topology, ensuring zero recovery time. An HSR ring is composed of DANs each equipped with two ring ports, as depicted in the subsequent figure. These specific DANs are referred to as DANHs.

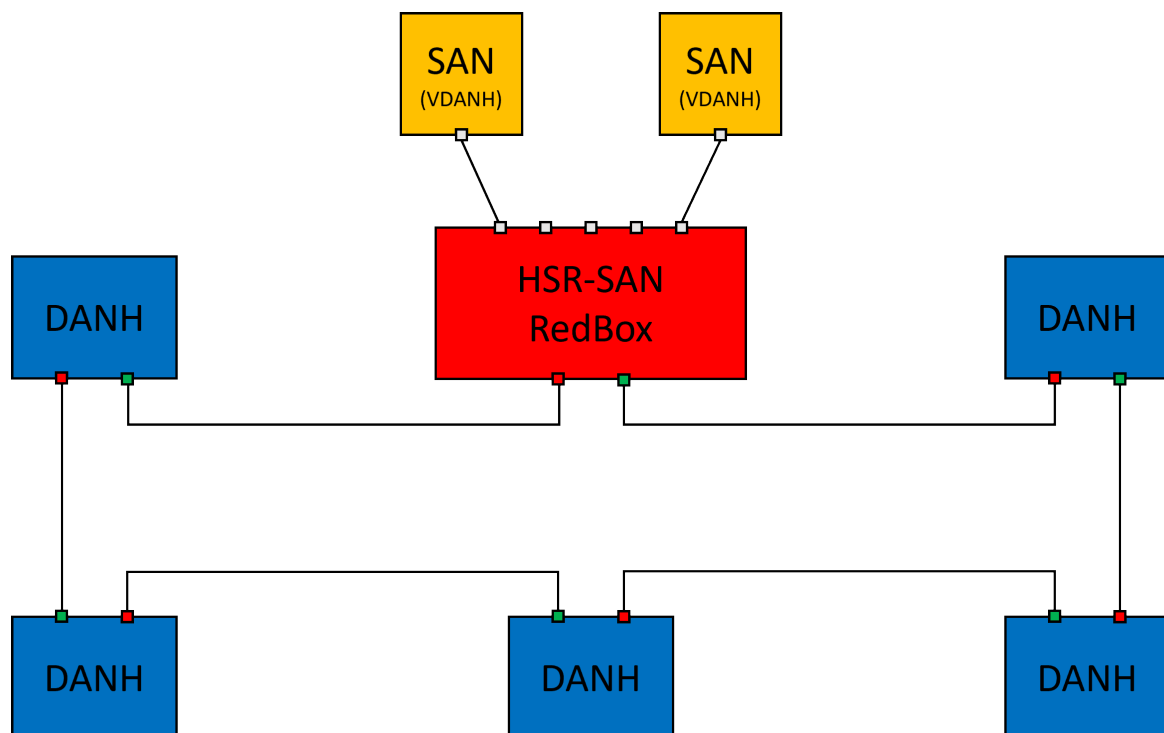


Figure 2. An HSR-SAN RedBox connected to an HSR ring

All devices within the HSR ring must be capable of processing an HSR tag on frames received from the ring and to add the HSR tag to all frames sent to the ring. This implies that a SAN cannot be directly connected to the HSR ring. Instead, it must be

connected through a RedBox on the RedBox's bridge-side. Such SANs are recognized as Virtual DANHs (VDANHs), as the RedBox is responsible for managing the pushing and popping of HSR tags on their behalf.

The 6-byte HSR tag, which includes a sequence number (HSR.SeqNr), is inserted by DANHs and RedBoxes. Each RedBox maintains a sequence number for every attached VDANH.

A DANH simultaneously sends the same frame through both LRE ports, resulting in two frame copies circulating both clockwise and counter-clockwise in the ring.

A unicast frame, whose final destination is a node within the ring, travels in both directions around the ring until it reaches the destination node. The destination node forwards the first error-free frame copy to its higher layers or, in the case of a RedBox, to the bridge-side after removing the HSR tag. The frame is not forwarded to the other LRE port. If or when the second copy arrives, it is identified as a duplicate and discarded, provided it arrives before the configurable duplicate discard timer expires.

A unicast frame, whose final destination is not a node within the ring, is forwarded by every node in the ring until it reaches the originating node, where it is dropped.

A multicast or broadcast frame is forwarded by each node as there can be multiple consumers of this frame. Therefore, such frames always reach the originating node and are dropped.

### 3.3. HSR-PRP

The integration of HSR and PRP allows for the connection of a PRP network with an HSR ring. This specific topology necessitates the parallel use of two RedBoxes, as depicted in the subsequent figure.

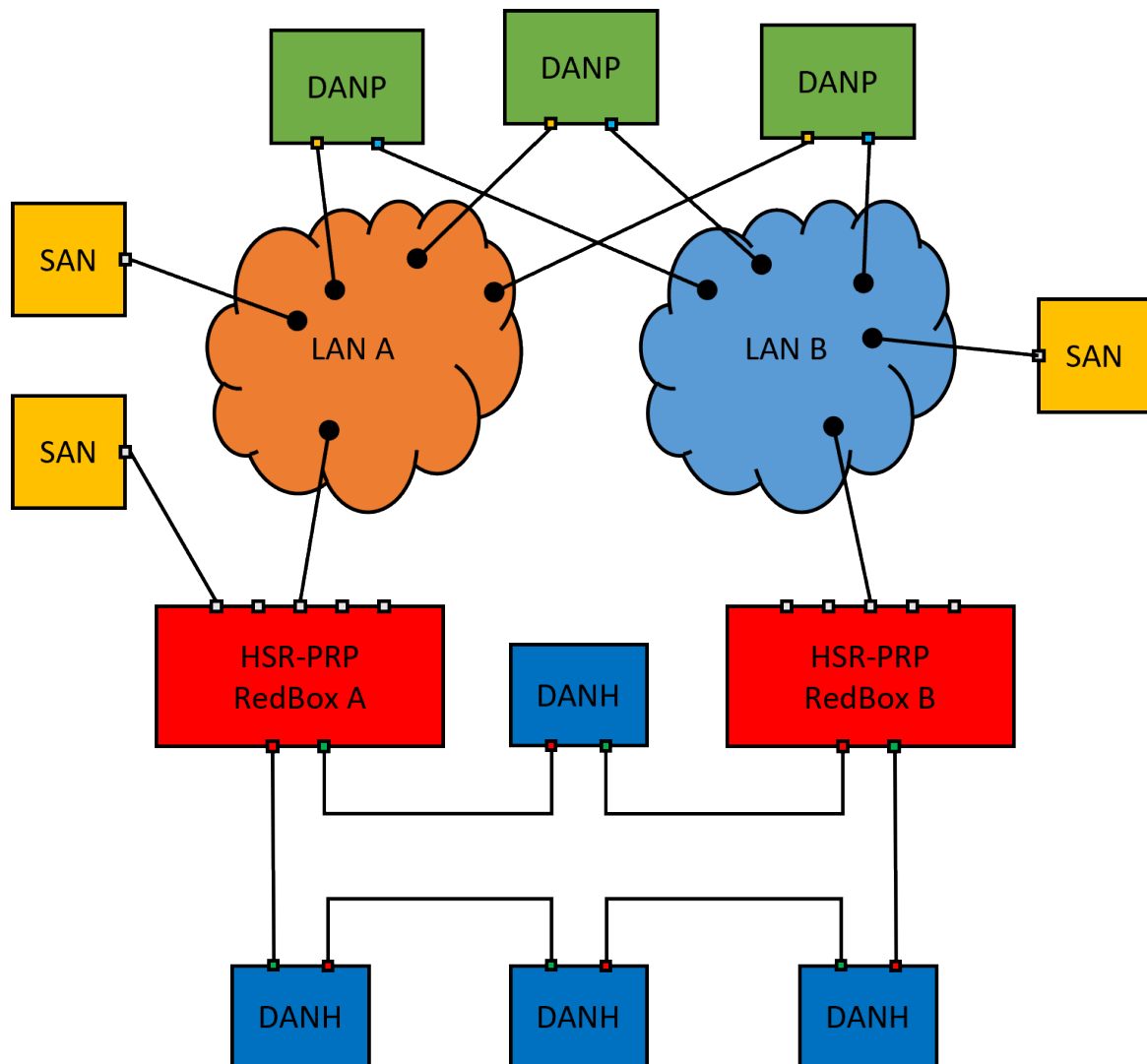


Figure 3. Two HSR-PRP RedBoxes interconnect a PRP network and an HSR ring

RedBox A connects to LAN A via a bridge-side port and to the HSR ring through its two LRE ports. Similarly, RedBox B connects to LAN B through a bridge-side port and to the HSR ring via its two LRE ports.

HSR-tagged traffic from the HSR ring, when forwarded to the interlink port, undergoes a transformation of the HSR tag to an RCT before it is injected towards the bridge-side of the RedBox.

In the same manner, PRP-tagged traffic from the PRP network, when forwarded through the interlink port towards the HSR ring, has the RCT transformed into an HSR tag with identical HSR.SeqNr.

If the traffic in the PRP network is not PRP-tagged (originating from a SAN), the RedBox maintains and inserts an HSR tag and sequence number on behalf of the SAN before forwarding it to the HSR ring.

To prevent traffic originating from the PRP network from being re-injected into the PRP network through the other RedBox, each HSR frame carries a 4-bit PathId (HSR.PathId), which identifies the PRP network from which the frame originated.

The PathId is a concatenation of a 3-bit NetId (HSR.NetId) and a 1-bit LanId (HSR.LanId). The two RedBoxes connected to the same PRP network must be configured with the same NetId, a number ranging from 1 to 7. 0 is reserved and used by true DANHs and by RedBoxes in HSR-SAN mode.

The LanId identifies the RedBox (A or B) that injected the frame into the HSR ring. LanId 0 is used by RedBox A and LanId 1 is used by RedBox B. The LanId is included for supervision purposes on the HSR ring.

A RedBox in HSR-PRP mode does not forward a frame carrying its own HSR.NetId to its interlink. This process is referred to as NetId filtering.

An RCT also contains a PathId (RCT.PathId). Frames with an RCT on LAN A always carry LanId 0 and frames with an RCT on LAN B always carry LanId 1. As per the standard, the NetId is always 5 in PRP networks, so that the combined NetId and LanId in hexadecimal notation becomes 0xA (0b1010) for frames on LAN A and 0xB (0b1011) for frames on LAN B.

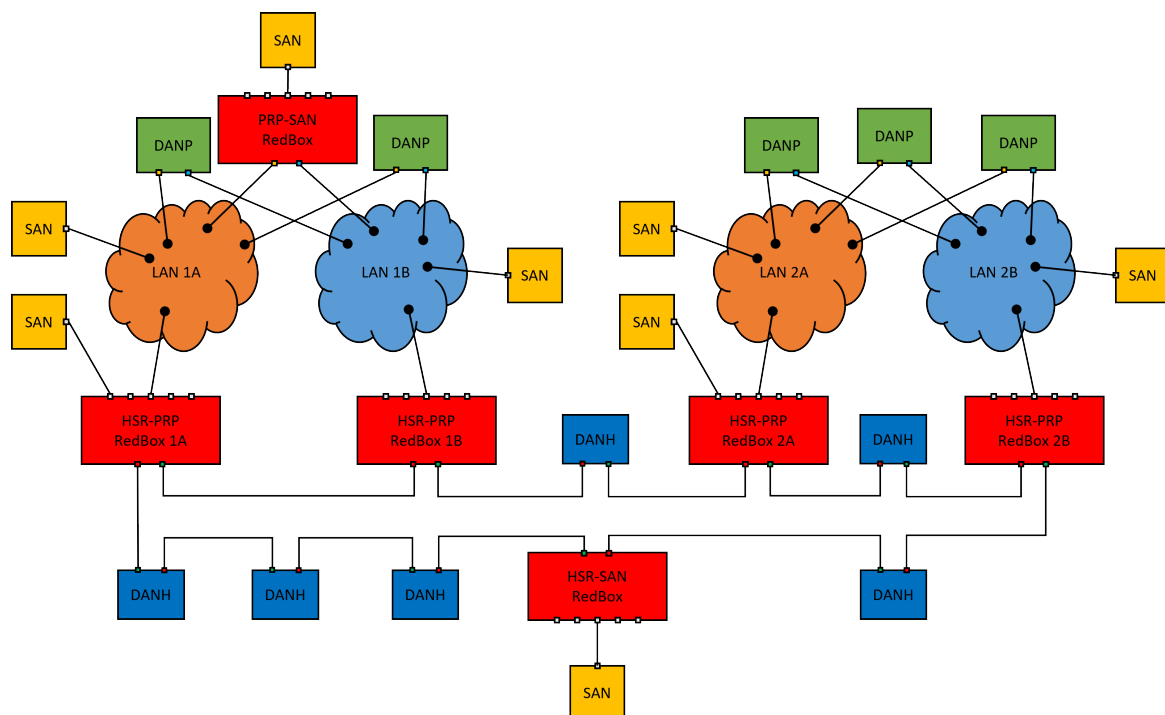
Frames from the HSR ring forwarded by the RedBox towards the interlink have the RCT inserted with a LanID corresponding to the configured LanId.

An HSR-PRP RedBox performs duplicate-discard on both the interlink and the LRE ports to prevent the same frame from being forwarded out the same RedBox port multiple times.

### 3.3.1. Connecting Multiple PRP Networks to a Single HSR Ring

Utilizing the RedBox types described above, a wide range of network topologies can be constructed.

The subsequent figure illustrates an example of connecting two PRP networks to a single HSR ring.





*Figure 4. Multiple PRP networks connected to a single HSR ring*

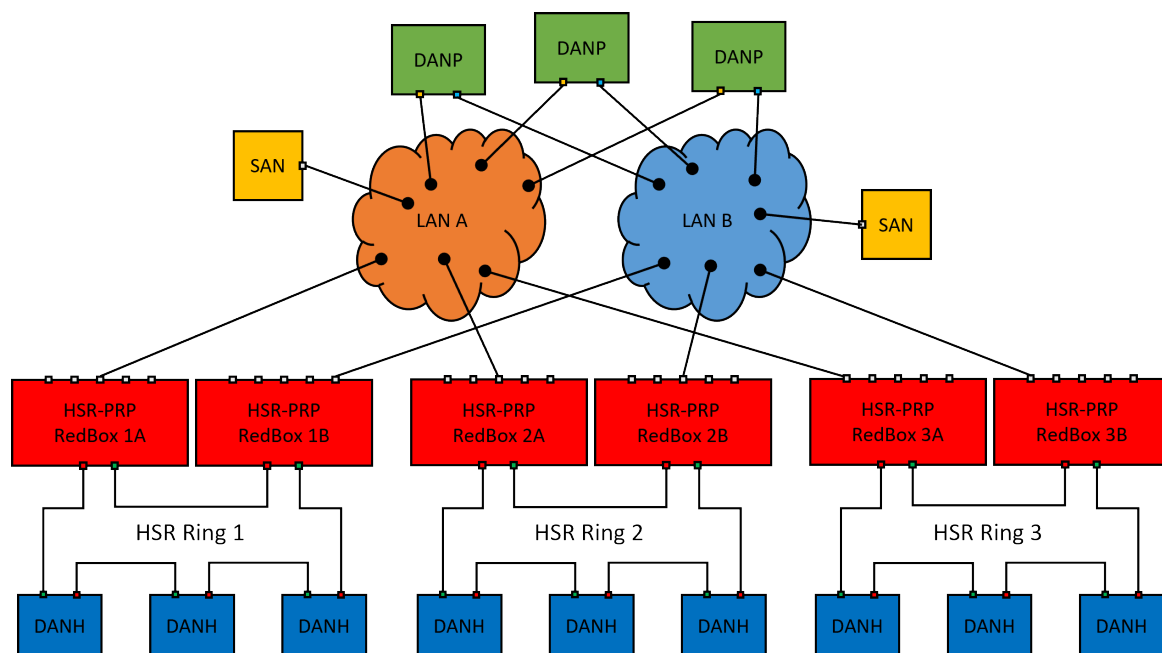
The 3-bit NetId facilitates the connection of up to seven distinct PRP networks to the same HSR ring.

For illustrative purposes, two additional RedBoxes, one in PRP-SAN mode and another in HSR-SAN mode, are included in the diagram.

### 3.3.2. Connecting a PRP Network to Multiple HSR Rings

A PRP network can be connected to an unlimited number of HSR rings. However, these rings cannot be interconnected, neither by QuadBoxes (refer to the next section) nor by another PRP network, as this would result in loops.

All RedBoxes in the subsequent figure may utilize the same NetId.

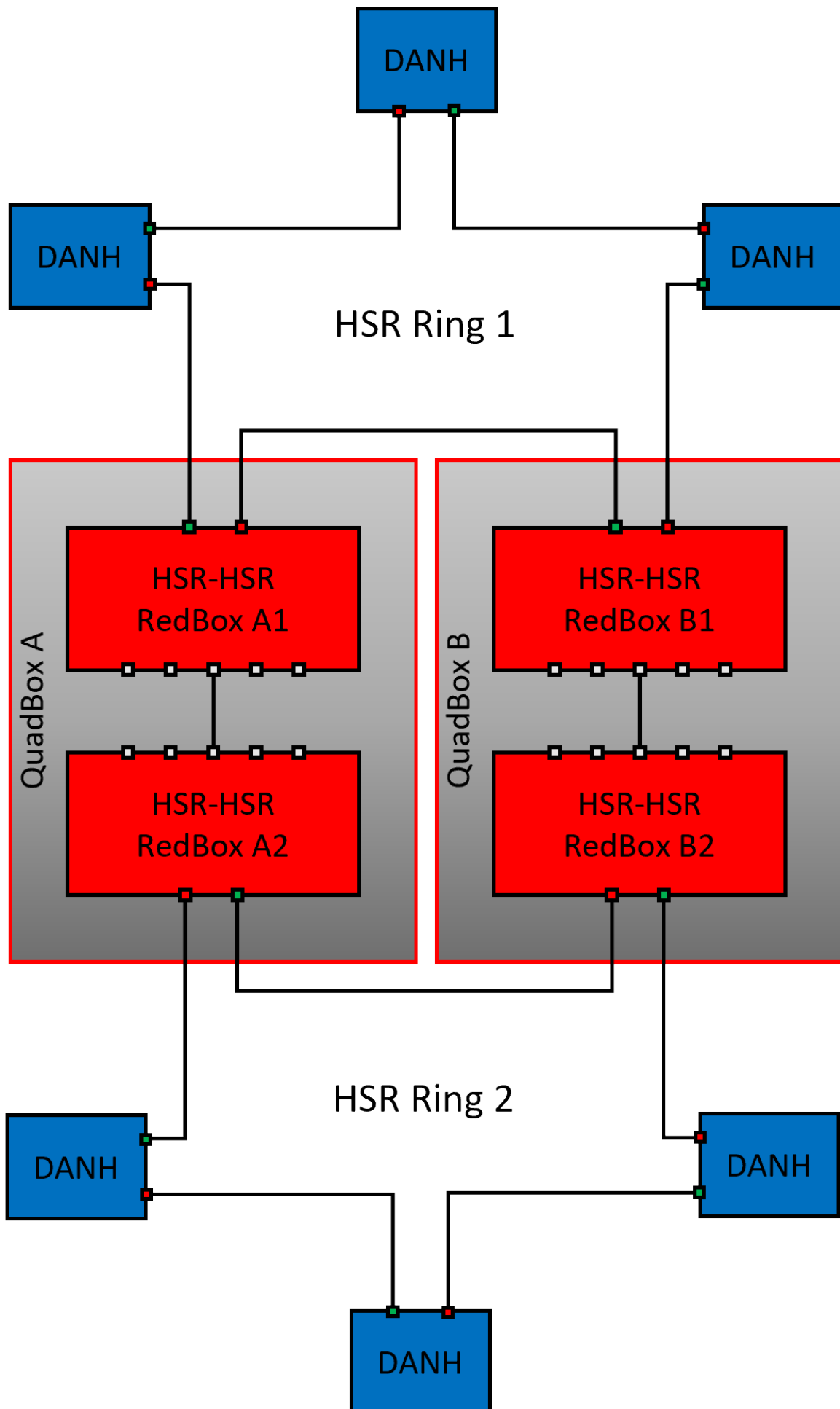


*Figure 5. Single PRP network connected to multiple HSR rings*

## 3.4. HSR-HSR

In HSR-HSR mode, traffic is HSR-tagged not only on the LRE ports but also on the interlink port.

Two RedBoxes in HSR-HSR mode can thus be connected back-to-back to form a so-called QuadBox, which is used to connect two separate HSR rings. In the following figure, two QuadBoxes operate in parallel to provide redundancy between the two HSR rings.



*Figure 6. Connection of two HSR rings with two QuadBoxes*

A QuadBox may or may not be located on the same physical device. The connection between the two RedBoxes forming a QuadBox is simply considered a "wire". If it is located on the same physical device, the two interlink ports should be the only ones in the same VLAN.

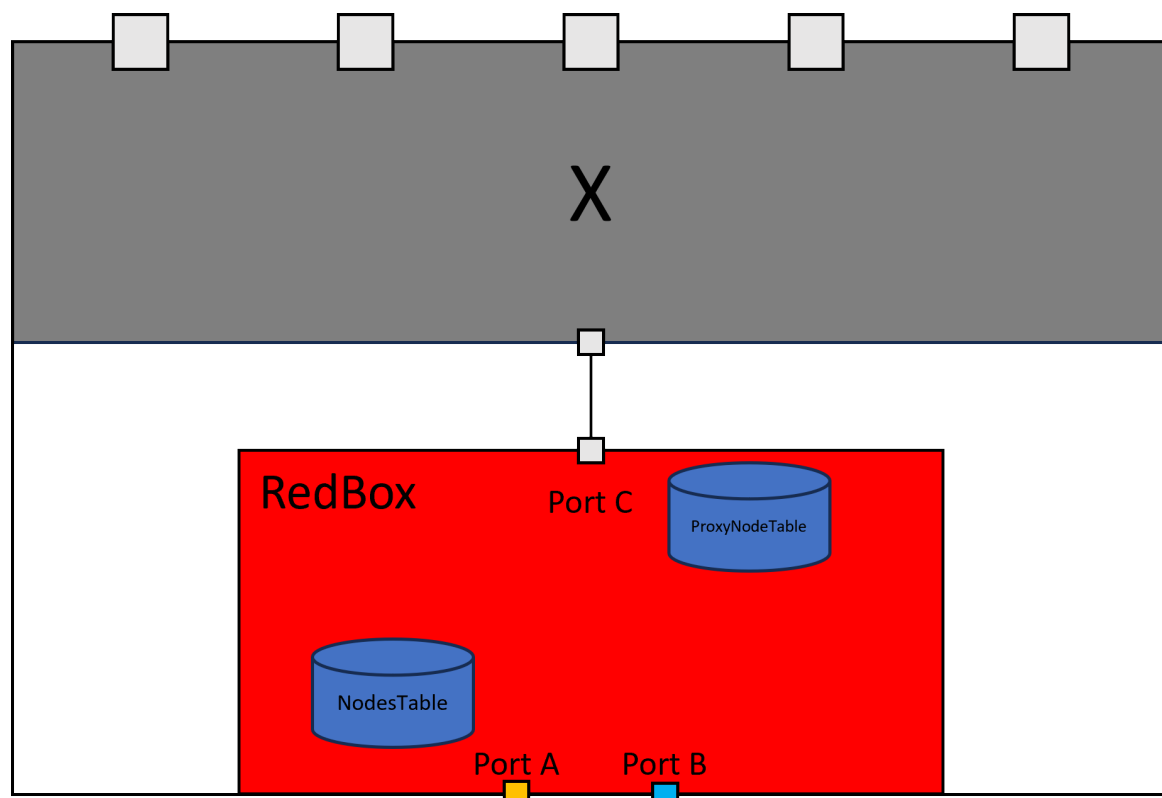
The QuadBoxes may or may not be configured with the same NetId (> 0). DANHs send with NetId 0, which is never NetId filtered by a RedBox. A frame sourced by a DANH in HSR Ring 1 and destined to a DANH in HSR ring 2 will therefore enter both RedBox A1 and B1 in two copies. Each of the RedBoxes will duplicate discard one of the copies, so that only one frame will be sent to the interlink of RedBox A2 and one to RedBox B2, both of which will generate two copies sent to HSR Ring 2.

Note that although the two RedBoxes have sent four copies of the frame to the HSR ring, each node on the ring will still only receive two copies, because the HSR-HSR RedBoxes perform duplicate discard also on their LRE ports, which prevents the same frame from being forwarded out the same LRE port multiple times. For example, the frame copy sent by RedBox A2 clockwise (through A2's Port B) will be filtered by RedBox B2 and not forwarded to RedBox B2's Port B if RedBox B2 has already sent this <SMAC, HSR.SeqNr> tuple towards its Port B.

Frame copies that travel the entire ring without being removed by a destination DANH are filtered by RedBox A2 and B2.

## 4. NodesTable and ProxyNodeTable

The following figure zooms in on a single RedBox. Besides the two LRE ports, Port A and Port B, there is an interlink towards the bridge-side of the RedBox, Port C.



### *Figure 7. NodesTable and ProxyNodeTable*

The RedBox contains two tables, the NodesTable, which keeps track of source MAC addresses (SMACs) seen in frames received on the LRE ports, and the ProxyNodeTable, which keeps track of SMACs seen in frames received on the interlink port.

Let's delve deeper into these two tables.

## 4.1. NodesTable

To manage properties of nodes on the LRE-side, a RedBox maintains a NodesTable.

The NodesTable tracks SMACs seen in frames received on LRE ports. Among other things, it contains a field that indicates whether a node is a SAN or a DAN.

In HSR modes, this field is not utilized, as all nodes on an HSR ring are DANs by definition.

In PRP-SAN mode, however, frames coming from the bridge-side and destined to nodes in the PRP network marked as DANs in the NodesTable get an RCT appended to the frame and the frame is transmitted to both LRE ports. If a node is marked as a SAN in the NodesTable, another field indicates which of the two LRE ports the node was seen on. A frame destined to the SAN only exits the LRE port it was seen on and will not contain an RCT.

Frames with a multicast or broadcast destination MAC address (DMAC) or with a DMAC not present in the NodesTable are handled as DANs.

A MAC address in the NodesTable can be marked as a DAN in two ways:

- If a frame with the same <SMAC, RCT.SeqNr> is seen on both LRE ports and the frame contains an RCT, the entry gets hardware-modified to a DAN.
- If a valid supervision frame that mentions the MAC arrives on an LRE port, software updates the entry to becoming a DAN.

Frames arriving on an LRE port where the entry in the NodesTable is marked as a DAN must contain a valid RCT or they will be discarded by the RedBox. For a PRP node, an RCT is only considered valid if the LanId corresponds to the LanId of the LRE port the frame was received on.

In all modes, a NodesTable entry ages out after a certain amount of time of inactivity (the NodesTable age time) from that MAC address and if no supervision frames mentioning that SMAC have been received on an LRE port.

Besides the SAN/DAN flag, an entry in the NodesTable contains per LRE-port counters for the number of data frames it has received from a given node. In PRP-SAN mode, it also contains per LRE-port counters for the number of frames it has received with a wrong LanId.

## 4.2. ProxyNodeTable

The ProxyNodeTable tracks SMACs received in frames arriving on the interlink port from the bridge-side.

In addition to dynamically added entries, it contains two locked entries:

- The RedBox' own MAC address. This is added as a DAN
- The bridge's management MAC address. This is added as a SAN.

In HSR-HSR mode, the ProxyNodeTable is not populated with dynamic entries.

In PRP-SAN and HSR-SAN mode, dynamically added entries are always marked as SANs.

In HSR-PRP mode, an entry is marked as a SAN by hardware. Software changes this to a DAN if a supervision frame mentioning the entry is received on the interlink from the PRP network side.

Once an entry is marked as a DAN (in HSR-PRP mode), frames from that SMAC must contain a valid RCT, and the LanId must match that of the RedBox, or the frame will be discarded. RBNTBD: This is not the case, it seems!

A dynamic entry ages out after a certain amount of time of inactivity (the ProxyNodeTable age time) from that MAC address and - in HSR-PRP mode - if no supervision frames mentioning that SMAC have been received on the interlink port.

Besides the SAN/DAN flag, an entry in the ProxyNodeTable contains counters for the number of data frames it has received from a given node and the number of frames it has received with a wrong LanId (HSR-PRP mode, only), provided the entry is marked as a DAN, which it only can if a supervision frame mentioning it has arrived in the PRP network.

## 5. Supervision Frame Handling

Supervision frames are generated by DANs (hereunder RedBoxes) to notify other DANs connected to the PRP network or HSR ring about their or a VDAN's presence.

In HSR-SAN and HSR-HSR modes, supervision frames primarily serve informational purposes. However, in PRP-SAN and HSR-PRP modes, they play an active role in marking a specific node as a DAN in the NodesTable and ProxyNodeTable, respectively.

A supervision frame is a multicast frame transmitted to DMAC 01-15-4E-00-01-xx, where xx is configurable. The EtherType of a supervision frame is 0x88FB.

Each supervision frame contains its own sequence number, which increments by one for each originated supervision frame. This sequence number is disregarded upon reception.

Supervision frames include one or two Type/Length Values (TLVs), TLV1 and optionally a TLV2. Each TLV contains a MAC address and a TLV Type.

Supervision frames sent to a PRP network (PRP-SAN or HSR-PRP) have a TLV1 type of 20 or 21. A value of 20 is used when the DANP is in Duplicate Discard Mode and 21 is used when it is in Duplicate Accept mode. Duplicate Accept mode is not supported by the IStax software, so only TLV1.Type == 20 is transmitted by Microchip RedBoxes.

Supervision frames sent to an HSR ring (all HSR modes) have a TLV1 type of 23.

TLV2 is only used when the supervision frame is originated by a RedBox. In that case, TLV2's Type is 30 and TLV2's MAC address is the MAC address of the RedBox.

If TLV2 is not present, TLV1's MAC address is a DANP or a DANH. The RedBox itself transmits supervision frames without TLV2 only for its own MAC address.

If TLV2 is present, TLV1's MAC address is that of a VDAN.

In PRP-SAN, HSR-SAN, and HSR-PRP mode, the IStax software regularly polls the ProxyNodeTable and originates proxied supervision frames at a configurable interval towards the LRE ports on behalf of the detected SANs (VDANs).

In HSR-PRP mode, supervision frames received from the PRP network will cause the IStax software to stop sending proxied supervision frames on behalf of both TLV1.MAC and TLV2.MAC (if present), as they appear to send their own.

In HSR-PRP mode, the IStax software also originates supervision frames towards the PRP network for the RedBox's own MAC address, only.

In PRP-SAN and HSR-SAN mode, any supervision frames received from the bridge-side are discarded by the RedBox at the interlink port, but it will be bridged normally in the SAN network.

In HSR-HSR mode, supervision frames are hardware forwarded between interlink and LRE ports, that is, in both directions, as any other data frame.

The standard's corrigendum states that in HSR-PRP mode, the network operator may choose to let supervision frames received from the PRP network be forwarded as they are to the HSR ring and vice versa.

This may lead to problems if not all nodes on the HSR ring are PRP supervision frame aware (TLV1.Type is 20 or 21) or if not all DANs on the PRP network are HSR supervision frame aware.

Therefore, a PRP-to-HSR and an HSR-to-PRP configuration option allows for having the IStax software forward supervision frames and change TLV1.Type to the value expected within the HSR ring and PRP network, respectively. The corrigendum also states that if the original supervision frame did not include a TLV2, the RedBox must add a TLV2 with its own MAC and replace the supervision sequence number with the RedBox's own supervision sequence number. The supervision frame is always originated with the RedBox MAC as the frame's SMAC.

Whenever a supervision frame is received by the IStax software it undergoes validation before it is actively used:

- DMAC must be 01-15-4e-00-01-xx, where xx is don't care
- TLV1 must be present and come first
- TLV1.Length must be correct
- TLV1.MAC must be a unicast MAC address
- TLV1.MAC must not be the RedBox's own MAC address
- TLV1.MAC must not be the device's management MAC address
- TLV1.Type must be any of the three valid types

- If TLV2 is present:
  - TLV2 must follow immediately after TLV1
  - TLV2.Length must be correct
  - TLV2.MAC must be a unicast MAC address
  - TLV2.MAC must not be the RedBox's own MAC address
  - TLV2.MAC must not be the device's management MAC address
  - TLV2.Type must be 30
- A null-TLV must follow immediately after TLV1 or TLV2 (if present)
- If received from PRP network:
  - RCT must be present
  - RCT.LanId must be correct
  - RCT.PRPSuffix must be 0x88FB
- If received from HSR ring:
  - HSR tag must be present
  - HSR.NetId must not be our own (HSR-PRP and HSR-HSR).

Any supervision frame that does not follow these rules is discarded by the IStax software. Hardware-forwarding of supervision frames is based on the EtherType only, so in that case there is no guarantee that all these rules are obeyed.

Moreover, a valid supervision frame may be filtered and not used by the IStax software because of one or more of the following reasons:

- The supervision frame was received on an LRE Port and
  - Port C is blocked for some reason or
  - Port C's VLAN ingress filtering is enabled, but Port C is not a member of the classified VLAN
- If TLV2.MAC is present and is either the RedBox's own MAC address or the switch's management MAC address or TLV2 is not present and TLV1.MAC is either of the two MAC addresses
- The supervision frame was received on a port in PRP mode but did not contain a valid RCT
- The supervision frame was received on a port in HSR mode but did not contain a valid HSR tag
- RedBox is in HSR-PRP mode and supervision frame received on LRE port and software HSR-to-PRP translation is enabled, but no ports in the PRP network are members of the classified VLAN.

Upon reception of a valid and non-filtered supervision frame on an LRE port, software adds both TLV1.MAC and TLV2.MAC (if present) to the NodesTable as DANs. It may happen that a supervision frame received from another RedBox mentions a TLV1.MAC not already in the NodesTable, because that MAC address is silent. In that case the entry will be created. For that reason silent entries can be kept alive by the mere reception of supervision frames that "mentions" these entries. Moreover, if PRP-to-

HSR or HSR-to-PRP supervision frame translation is active in some nodes in the PRP network or on the HSR ring, the frame's SMAC may not be the same as TLV2.MAC, so TLV2.MAC may also not pre-exist in the NodesTable.

## 6. RedBox Port Interfaces

Microchip switches with RedBox support have a predetermined mapping between RedBox instance numbers and port interfaces.

LAN969x devices are capable of supporting up to five RedBox instances, numbered from 1 to 5. The diagram below illustrates the possible port interfaces that can be assigned to each RedBox instance.

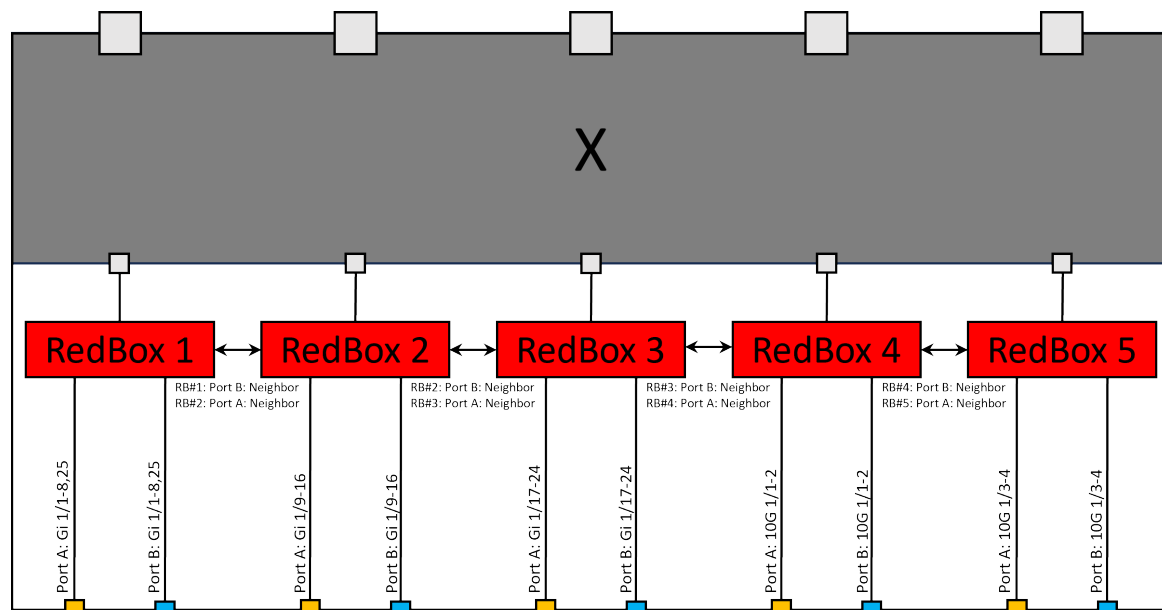


Figure 8. RedBox-to-Port-Interface mappings for LAN969x switches

RedBox instances can be interconnected in a daisy-chain configuration using an internal connection. For example, Port B of RedBox #1 can be linked to Port A of RedBox #2 through an internal connection within the chip. This eliminates the need for an external port interface to connect Port B of RedBox #1 to Port A of RedBox #2.

However, only adjacent RedBox instances can be daisy-chained. RedBox #1 cannot daisy-chain its Port A interface, and similarly, the farthest RedBox (in this case, RedBox #5) cannot daisy-chain its Port B interface.

Daisy-chaining of RedBoxes is used to connect multiple RedBox instances to the same HSR-ring. RedBoxes that daisy chain cannot be configured in PRP-SAN mode.

To avoid loop formations, the interlinks of daisy-chained RedBox instances must not share the same VLANs.

The Command Line Interface (CLI) can be used to display the configurable interfaces for each RedBox instance:



```
# show redbox interfaces
Instance Interfaces
```

```
-----
      1 Gi 1/1-8,25
      2 Gi 1/9-16
      3 Gi 1/17-24
      4 10G 1/1-2
      5 10G 1/3-4
```

This information can also be displayed per interface rather than per RedBox instance:

```
# show redbox interfaces sort-by-interface
Interface Instance
```

```
-----
Gi 1/1          1
Gi 1/2          1
Gi 1/3          1
Gi 1/4          1
Gi 1/5          1
Gi 1/6          1
Gi 1/7          1
Gi 1/8          1
Gi 1/9          2
Gi 1/10         2
Gi 1/11         2
Gi 1/12         2
Gi 1/13         2
Gi 1/14         2
Gi 1/15         2
Gi 1/16         2
Gi 1/17         3
Gi 1/18         3
Gi 1/19         3
Gi 1/20         3
Gi 1/21         3
Gi 1/22         3
Gi 1/23         3
Gi 1/24         3
10G 1/1         4
10G 1/2         4
10G 1/3         5
10G 1/4         5
Gi 1/25         1
```

From the perspective of the bridge, the interlink port (Port C) is one of the configured physical port interfaces, as shown in the table below.

Port A	Port B	Port C
Physical Port	Physical Port	Port A. Port B is unconnected.

Physical Port	Neighbor	Port A
Neighbor	Physical Port	Port B
Neighbor	Neighbor	Invalid combination of Port A and Port B

The unconnected port is unconnected in the sense that it is not connected to the switch core; it is connected directly to the RedBox. This means that port-specific parameters such as link speed and link status are still applicable to unconnected ports, but frames are not bridged directly towards or from such ports.

## 7. Command Line Interface

The command line interface (CLI) offers commands for configuring a RedBox, showing status, showing and clearing statistics, NodesTable and ProxyNodeTable contents.

### 7.1. Configuration

One RedBox instance represents a RedBox in either of the previously described modes. When an instance is disabled, it has no impact on the frames passing through the switch.

Upon enabling a RedBox instance, a sanity check on the provided parameters will be performed and if the combination of parameters lies within the acceptable range, the RedBox instance will start to operate.

In order to create a RedBox instance, use the following syntax in CLI configuration mode:

```
redbox <inst>
```

Where

<code>inst</code>	RedBox instance number
-------------------	------------------------

Likewise, in order to delete one or all RedBox instances, use the following syntax in CLI configuration mode:

```
no redbox {<inst> | all}
```

Where

<code>inst</code>	Delete a particular RedBox instance
<code>all</code>	Delete all Redbox instances

The syntax for RedBox configuration level CLI command is:

```
admin-state {enable | disable}
[no] duplicate-discard-age-time <10-10000>
[no] lan-id {a | b}
[no] mode {prp-san | hsr-san | hsr-prp | hsr-hsr}
[no] net-id <1-7>
[no] nodes-table-age-time <1-65>
[no] port-a interface [<port_type_id> | neighbor]
[no] port-b interface [<port_type_id> | neighbor]
[no] proxy-node-table-age-time <1-65>
[no] supervision-dmac-lsb <uint8>
[no] supervision-frame-interval <1-60>
[no] supervision-translate-prp-to-hsr
[no] supervision-translate-hsr-to-prp
[no] supervision-vlan {native | <vlan_id>} [pcp <0-7>]
```

Where:

admin-state	Enable or disable this RedBox instance
duplicate-discard-age-time	Number of milliseconds before an entry in the
duplicate-discard table	times out.
	Use the no-form to set it to its default of 40
milliseconds.	
lan-id	The LanId is used to filter frames from an HSR ring
towards the PRP	network. It must be 'a' for the RedBox connecting
	to LAN A and 'b' for
	the RedBox connecting to LAN B.
	Use the no-form to set it to its default ('a').
	This option is only available in HSR-PRP and
HSR-HSR modes.	
mode	Set the mode of this RedBox.
	Use the no-form to set it to its default (PRP-SAN)
net-id	If a frame arriving on an LRE port has a NetId
identical to this one,	it gets filtered and not forwarded to the interlink
port, but may get	forwarded to the other LRE port. In HSR-HSR mode,
frames arriving on	the interlink are forwarded to the LRE ports while
translating the	NetId in the incoming frame to the NetId of the
RedBox.	
	Use the no-form to set it to its default (1).
	This option is only available in HSR-PRP and
HSR-HSR modes.	
nodes-table-age-time	Number of seconds without activity before a remote
node is removed	from the NodesTable.
	Use the no-form to set it to its default (60
seconds).	
port-a	Assign an interface to port A.
	Use the no-form to unassign port A's interface
(requires the RedBox to	be disabled).
port-b	Assign an interface to port B.
	Use the no-form to unassign port B's interface
(requires the RedBox to	be disabled).
proxy-node-table-age-time	Number of seconds without activity before a proxy
node is removed from	the ProxyNodeTable.
	Use the no-form to set it to its default (60
seconds).	
	This option is not available in HSR-HSR mode.
supervision-dmac-lsb	Set the least significant byte used in the
destination MAC address	(01-15-4e-00-01-xx) of generated PRP/HSR
supervision frames.	
	Use the no-form to set it to its default (0x00).
supervision-frame-interval	Number of seconds between transmission of
supervision frames.	

<p>seconds).</p> <p>supervision-translate-prp-to-hsr</p> <p>frames received on the</p> <p>on the HSR ring.</p> <p>supervision-translate-hsr-to-prp</p> <p>frames received on the</p> <p>the PRP network.</p> <p>supervision-vlan</p> <p>tag used in</p> <p>interlink port's Port VLAN</p> <p>on the interlink port,</p> <p>their defaults</p>	<p>Use the no-form to set it to its default (2</p> <p>The RedBox will software-translate supervision</p> <p>PRP network to HSR supervision frames and transmit</p> <p>Use the no-form to hardware-forward such frames.</p> <p>This option is only available in HSR-PRP mode.</p> <p>The RedBox will software-translate supervision</p> <p>HSR ring to PRP supervision frames and transmit on</p> <p>Use the no-form to hardware-forward such frames.</p> <p>This option is only available in HSR-PRP mode.</p> <p>Set the VLAN ID and PCP value of a possible VLAN</p> <p>supervision frames. Use 'native' to use the</p> <p>ID. If the resulting VLAN is configured as tagged</p> <p>the supervision frame will be transmitted tagged.</p> <p>Use the no-form to set the VLAN ID and PCP value to</p> <p>('native' and 7, respectively).</p>
---	---

## 7.2. Status and Statistics

There are several kinds of status and statistics available in CLI EXEC mode.

```
show redbox [<instances>] {status | statistics | nodes-table [supervision] [filter]
| proxy-node-table [filter]} [details]
```

Where

<instances> statistics for.	List of RedBox instances to show status, table contents or statistics for.
status	If left out, all created RedBox instances will be shown. Show status.
RedBox with a status	If [details] is left out, it shows one line per created overview.
instance is given.	If [details] is included, more info about each RedBox
statistics	Show statistics.
RedBox with a statistics	If [details] is left out, it shows one line per created summary.
RedBox instance.	If [details] is included, it shows detailed statistics per
nodes-table	Show NodesTable contents.
addresses currently present	If [details] is left out, it shows the number of MAC in the NodesTable.
shows one line per MAC	If [details] is included and [supervision] is left out, it address with data frame statistics.
one line per MAC address	If both [details] and [supervision] is included, it shows with supervision frame statistics.
'Rx Wrong LAN'.	If [filter] is included it only shows entries with non-zero
proxy-node-table	Show ProxyNodeTable contents.
addresses currently present	If [details] is left out, it shows the number of MAC in the ProxyNodeTable.
with data and supervision	If [details] is included it shows one line per MAC address frame statistics.
'Rx Wrong LAN'.	If [filter] is included it only shows entries with non-zero
details	As out lined above.

Finally, there's a suite of clear commands, also accepted in CLI EXEC mode:

```
clear redbox [<instances>] {statistics | nodes-table | proxy-node-table}
```

Where

<instances>	List of RedBox instances to clear tables or statistics for.
tables or statistics	If left out, all created RedBox instances will have their cleared.
statistics	Clear statistics.
nodes-table	Clear the NodesTable.
proxy-node-table	Clear dynamic (unlocked) entries of the ProxyNodeTable.

The following sections contain examples and explanations of the output of these show and clear commands.

### 7.2.1. Status

After configuration of a RedBox, the `show redbox [<instances>] status` command should be used to see if everything is well.

Here's an example of showing status. For the sake of this example, five RedBoxes are configured in four different modes. Two of them connect internally, thereby saving an external port interface.

```
# show redbox status
Inst Oper. State Mode   Port A   Port B   Port C   Warnings Notifications
-----
  1 Active      PRP-SAN Gi 1/1    Gi 1/2    Gi 1/1    YES!      No
  2 Inactive
  3 Active      HSR-SAN Gi 1/17   Neighbor  Gi 1/17    No        YES!
  4 Active      HSR-PRP Neighbor  10G 1/2   10G 1/2   YES!      YES!
  5 Active      HSR-HSR 10G 1/3   10G 1/4   10G 1/3    No        No
```

The first column shows the RedBox instance number. RedBoxes that aren't created won't be displayed.

The next column shows the operational state. An active instance is administratively (configuration-wise) enabled. An inactive instance is administratively disabled.

The third, fourth, and fifth columns shows the configured mode and the configured port interfaces. The combination of the two LRE ports gives the interlink port (Port C), as described in Finding the Interlink Port table. This is displayed in column six.

The last two columns are important to understand. The **Warnings** column states whether there are configurational warnings that need to be fixed for the RedBox instance to work optimally or work at all.

The **Notifications** column holds runtime conditions that may affect the operation of a RedBox.

To catch the reader's eye, warnings and notifications are written in capitals followed by an exclamation mark.

These two columns are described in more details in the next sections.

### Configurational Warnings

It is easy to make configurational mistakes. The IStax software attempts to disclose the most obvious, such as wrong VLAN or port configurations. It is important to note that the detected warnings do not constitute the complete list of possible configuration mistakes.

With offset in the previous example, we can see that RedBox instance #1 and #4 have configurational warnings, whereas the remaining don't.

Let's focus on the warnings of instance #1 by selecting this instance in `show` command and then add the `details` keyword to the command:

```
# show redbox 1 status details
Instance:                1
Mode:                    PRP-SAN
Operational State:       Active
Configurational Warnings: The MTU is too high on at least one of the LRE ports
(max is 2000) <- Here
                        Interlink port is not member of the supervision
frame VLAN ID           <- Here
NodesTable/ProxyNodeTable full: No

-----
Interface      Port A      Port B      Port C
-----
Link           Gi 1/1      Gi 1/2      Gi 1/1
Wrong LAN Rx   Up          Up          -
Non-HSR-tagged Rx No          No          No
```

As can be seen from the lines annotated with `<- Here`, there are two configurational warnings that should be fixed. To do so, consult the following table, which lists possible warnings, their reason, and how to fix them.

Number	Configurational Warning	Reason	Remedy
1	The MTU is too high on at least one of the LRE ports (max is 2000)	RCTs and HSR tags contain a 12-bit field called <code>LSDUsize</code> , which contains the size of the frame starting after a possible VLAN tag and subsequent EtherType and ending just before the frame's FCS. This means that the protocol itself doesn't support frames larger than roughly 4 KBytes. The RedBox hardware itself only supports frames of 2000 bytes (from DMAC up to and including FCS), so the MTU on the LRE ports must be restricted to 2000 bytes.	Enter interface configuration mode and set MTU on LRE ports to 2000 or lower. For example: # configure terminal (config)# interface GigabitEthernet 1/1,2 (config-if)# mtu 2000



2	The MTU is too high on at least one non-LRE port. Frames larger than 1994 cannot traverse the HSR/PRP network	The MTU is too high on non-LRE ports that have VLAN memberships in common with the RedBox's interlink port. The MTU must be 6 bytes smaller than the supported 2000 bytes (see warning #1 above), because the RedBox must be able to push an HSR tag or add an RCT to the frame before it leaves the LRE ports.	Enter interface configuration mode and set MTU on non-LRE ports that have VLAN memberships in common with the RedBox to 1994 or lower. For example: <pre># configure terminal (config)# interface GigabitEthernet 1/ 3-22 (config-if)# mtu 1994</pre>
3	Interlink port must use C-tags	The IStax software only supports setting up C-tags (not S-tags or custom S-tags) in the RedBox hardware. So if the interlink port is configured as an S-tagged port, the RedBox hardware will e.g. HSR tag frames wrongly (the HSR tag will be inserted after the SMAC instead of after the VLAN tag).	Only hybrid VLAN mode, allows for configuring the interlink port's egress tag type to something other than C-tags, so that must be the reason for this warning. We need to change it to a C-port: <pre># configure terminal (config)# interface GigabitEthernet 1/1 (config-if)# switchport hybrid port-type c-port</pre>

4	Interlink port is not member of the supervision frame VLAN ID	When the interlink port is not a member of the supervision frame VLAN ID, the IStax software will not be able to transmit supervision frames towards the LRE ports through the RedBox.	There can be multiple reasons for this warning. For example, if the configured supervision frame VLAN ID is <code>native</code> , the interlink port must be a member of the port's VLAN ID (PVID). If then the port is in trunk or hybrid VLAN mode, the native VLAN must be set to one of the allowed VLANs. If the port is in access VLAN mode, the access VLAN must be defined globally. If the configured supervision frame VLAN ID is not <code>native</code> : If the interlink port is in trunk or hybrid VLAN mode, the supervision frame VLAN ID must be amongst the allowed VLANs, or if in access VLAN mode, the VLAN must be defined globally.
5	The neighbor RedBox is not configured	Port A or port B of this RedBox is configured to use the neighboring RedBox rather than a physical port, but the neighboring RedBox is not configured or created.	Create and configure the neighboring RedBox or configure this RedBox to use a physical port rather than a neighbor.
6	The neighbor RedBox is not active	Port A or port B of this RedBox is configured to use the neighboring RedBox rather than a physical port, but the neighboring RedBox is not enabled.	Enable the neighboring RedBox or configure this RedBox to use a physical port rather than a neighbor.

7	The neighbor's port A is not configured as a RedBox neighbor	This RedBox has configured its port B to use the neighboring RedBox, but the neighboring RedBox' port A is not configured to use its neighbor.	Configure this RedBox to use a physical port or the neighboring RedBox to use us as its neighbor.
8	The neighbor's port B is not configured as a RedBox neighbor	This RedBox has configured its port A to use the neighboring RedBox, but the neighboring RedBox' port B is not configured to use its neighbor.	See warning #7 above.
9	The neighbor's interlink port has coinciding VLAN memberships with this RedBox's interlink port	If this RedBox is daisy-chained with the neighboring RedBox, the two RedBoxes' interlink ports should not have any VLAN memberships in common. If they have it might result in loops in the network.	Make sure the two RedBoxes' interlink ports don't have any VLAN memberships in common.
10	Interlink port has spanning tree enabled	The Spanning Tree Protocol (STP; both MSTP and RSTP) must be disabled on the interlink port. If STP is enabled, and it decides to block the interlink port, no frames will be forwarded between the bridge-side and the LRE ports.	Disable spanning tree on the interlink port, e.g.: # configure terminal (config)# interface GigabitEthernet 1/1 (config-if)# no spanning-tree

## Notifications

Several runtime conditions may affect the operation of a RedBox. Some are grave, such as link down on one of the LRE ports, and some are less serious, such as the connection of a SAN directly to an HSR ring.

These conditions are known as notifications, and can give rise to a JSON notification or a an SNMP trap.

Continuing the `show redbox status` example from above, we can see that RedBox instance #3 and #4 have active notifications, whereas the remaining don't.

Let's see the details of instance #3:

```
# show redbox 3 status details
Instance:                3
Mode:                    HSR-SAN
Operational State:       Active
Configurational Warnings: None
NodesTable/ProxyNodeTable full: No          <- Here

-----
Interface      Port A      Port B      Port C
-----
Link           Gi 1/17    Neighbor    Gi 1/17
Wrong LAN Rx   DOWN!      Up          Up          <- Here
Non-LSR-tagged Rx No         No         No         <- Here
Non-LSR-tagged Rx No         No         -          <- Here
```

The lines marked with <- Here are notification lines. In this output, there is only one active notification, which is link down on Port A.

If a notification is not applicable for a given port in the current RedBox mode, a dash is printed.

Possible notifications are listed below.

Number	Configurational Warning	Reason	Remedy
1	NodesTable/ProxyNodeTable full	In hardware, these two tables are combined into one single with 4096 entries. If the limit is reached, this notification is raised. It will disappear as soon as at least one entry is free in the combined table.	Limit the number of nodes in the network.
2	Link	Shows the link status on Port A and Port B. LRE ports that are configured to use the neighboring RedBox are always up. The interlink port (Port C) is also always up, although it is represented by a physical port, which may be down.	Get link on the port.

3	Wrong LAN Rx	<p>A frame with wrong RCT.LanId is received on a port in a PRP mode.</p> <p>In PRP-SAN mode, Port A expects to receive frames with RCT.LanId = 0 and Port B expects to receive frames with RCT.LanId = 1 (if the frames contain an RCT).</p> <p>In HSR-PRP mode, Port C expects to receive frames with RCT.LanId equal to the LanId configured on the RedBox. If a DANP happens to send frames with a valid RCT with a wrong LanId, this also counts.</p> <p>The notification is based on reading of RedBox statistics, so it may take up to 10 seconds after it really occurred for the notification to appear. It disappears again after roughly 20 seconds without changes in the wrong LAN counters.</p>	<p>In PRP-SAN mode, go through the NodesTable and locate the MAC address, where the Rx Wrong LAN field is non-zero. Such entries can be shown with the filter option to show redbox nodes-table. Then reconfigure or change connections for that node.</p> <p>In HSR-PRP mode, go through the ProxyNodeTable and locate the MAC address, where the Rx Wrong LAN field is non-zero. Such entries can be shown with the filter option to show redbox proxy-node-table. Then reconfigure or change connections for that node.</p>
---	--------------	--	--

4	Non-HSR-tagged Rx	A frame without HSR tag was received on one of the LRE ports with the RedBox in any HSR mode. This can never occur on the interlink port, so that port is marked with a dash.	Unfortunately, there is no field in the NodesTable that tells that a particular MAC address is the sinner, but since the standard says that frames received on an LRE port in HSR mode without HSR tag must be discarded, we know that the sinner is connected directly to either Port A or Port B, since it cannot have travelled longer in the ring. The notification is based on reading of RedBox statistics, so it may take up to 10 seconds after it really occurred for the notification to appear. It disappears again after roughly 20 seconds without changes in the wrong LAN counters.
---	-------------------	---	--

### 7.2.2. Statistics

With outset in the `show redbox status` example let's get an overview of the number of frames ingressing and egressing the RedBoxes.

# show redbox statistics					
Port A		Port B		Port C	
Inst	Rx	Tx	Rx	Tx	Rx
1	758	7398	0	7398	
7398	758				
2	Inactive				
3	0	0	0	4395	
4395	0				
4	4395	0	0	0	
0	4389				
5	0	0	0	0	
0	0				

For each created RedBox instance, there are three sets of Rx and Tx counters - one for Port A, one for Port B, and one for Port C.

Administratively disabled instances are shown as **Inactive**.

Rx and Tx are seen from the perspective of the RedBox itself. E.g. "Port C Tx", means frames (received on an LRE port and) transmitted towards the bridge-side of the RedBox.

Add the **details** keyword to see detailed statistics for one or all instances, e.g.

```
# show redbox 1 statistics details
Instance: 1
```

Counter	Port A		Port B		Port C	
	Rx	Tx	Rx	Tx	Rx	Tx
Tagged		0	372	0	371	
0	0					
Untagged		89	0	0	0	
376	89					
BPDUs		0	0	0	0	
0	0					
Own		0	-	0	-	
0	-					
Wrong LAN		0	-	0	-	
0	-					
Zero Duplicates		-	0	-	0	
-	0					
One Duplicate		-	0	-	0	
-	0					
Two or more Duplicates		-	0	-	0	
-	0					
PRP-DD Supervision		0	360	0	362	
0	0					
PRP-DA Supervision		0	0	0	0	
0	0					
HSR Supervision		0	0	0	0	
0	0					
Erroneous Supervision		0	-	0	-	
0	-					
Filtered Supervision		0	-	0	-	
0	-					

Each of the counters are described in more details here:

Counter	Description
Tagged	<p>If the port in question is in one of the HSR-modes, this indicates the number of HSR-tagged frames received/transmitted on the port.</p> <p>If the port in question is in PRP mode, this indicates the number of frames received/transmitted on the port with an RCT.</p>

Untagged	<p>If the port in question is in one of the HSR-modes, this indicates the number of frames received/transmitted on the port without an HSR tag.</p> <p>If Rx Untagged is non-zero, this gives rise to a "Non-HSR-tagged Rx" notification. See Section 7.2.1.2. In .e.g HSR-SAN mode, Port C's Tx Untagged counter counts, because the RedBox removes an HSR tag before sending the frame towards the interlink.</p> <p>If the port in question is in PRP mode, this indicates the number of frames received/transmitted on the port without an RCT.</p>
BPDUs	Number of BPDUs (frames with DMAC = 01-80-C2-00-00-0x) received/transmitted.
Own	Number of frames received whose SMAC matches a MAC address in the ProxyNodeTable.
Wrong LAN	Number of frames received with wrong LanId. This is only applicable if the port is in PRP mode. If non-zero, this gives rise to a "Wrong LAN Rx" notification. See Section 7.2.1.2.
Zero Duplicates One Duplicate Two or More Duplicates	<p>These are duplicate discard counters and count for every port to which the first frame copy was transmitted, how many duplicates (zero, one or more than one) the RedBox has discarded.</p> <p>The counters are updated once the duplicate discard timer for the &lt;SMAC, HSR/RCT.SeqNr&gt; tuple expires (after the configured duplicate-discard-age-time milliseconds).</p>
PRP-DD Supervision PRP-DA Supervision HSR Supervision	<p>Indicates the number of supervision frames received/transmitted on a given port.</p> <p>PRP-DD Supervision counts when a supervision frame's TLV1.Type is 20.</p> <p>PRP-DA Supervision counts when a supervision frame's TLV1.Type is 21.</p> <p>HSR Supervision counts when a supervision frame's TLV1.Type is 23.</p> <p>If receiving supervision frames with other TLV1.Type values, Erroneous Supervision counts.</p> <p>Notice that the IStax software never transmits PRP-DA supervision frames, because this mode is not supported. See also Section 5.</p>
Erroneous Supervision	If a supervision frame received by the IStax software does not fulfil the guidelines outlined in Supervision Frame Handling, it will be discarded, and won't be counted in the PRP-DD, PRP-DA, or HSR Supervision counters.
Filtered Supervision	Number of non-erroneous received supervision frames that were not used by the IStax software because of one or more of the reasons mentioned in Supervision Frame Handling



All the supervision frame counters are software-based.

The following shows how to clear statistics for all RedBoxes in one go:

```
# clear redbox statistics
```

And only for instance #1 and #4:

```
# clear redbox 1,4 statistics
```

### 7.2.3. NodesTable Contents

First, to get an overview of the contents of the NodesTable, issue the following command without any keywords.

```
# show redbox nodes-table
Inst Mode      MAC Addresses Wrong LAN
-----
 1 PRP-SAN           5 YES!
 2 HSR-SAN          212 -
 3 Inactive
 4 HSR-PRP           26 -
 5 HSR-HSR           65 -
```

One line per RedBox that contains the instance number, the current mode and the number of MAC addresses in the NodesTable is shown. The last column is only relevant in PRP-SAN mode, and reads **No** if all DANPs are connected correctly to the PRP network, **YES!** otherwise.

To see details for a particular RedBox, add the **details** keyword to the command. In the following example, only details for RedBox #1 are shown.

```
# show redbox 1 nodes-table details
```

			Rx		Last Seen		Rx Wrong LAN		
			-----		-----		-----		
Inst	MAC Address	Node Type	Forward	Port A	Port B	Port A	Port B	Port A	Port B
1	00-00-00-00-01-00	DANP	Both	5793	5793	1	1		
0	0								
1	00-00-00-00-01-01	SAN	Port A	6132	0	17	-		
0	0								
1	00-00-00-00-01-02	SAN	Port B	0	133	-	53		
0	0								
1	00-00-00-00-01-03	VDANP	Both	0	0	-	-		
0	0								
1	00-00-00-00-01-04	DANP-RedBox	Both	13	13	2	2		
13	13								

In this constructed example, we show the NodesTable contents of a RedBox in PRP-SAN mode. The listing contains one line per entry, sorted by MAC address.

The contents of each column is described in the following:

Name	Description
Inst	The RedBox instance this row relates to.
MAC Address	The MAC address this row relates to.
Node Type	<p>The IStax software detects and names the node after the RedBox' mode of operation and reception of supervision frames on the LRE ports that relate to that node. See Section 4.1 and Section 5 for a description of hardware's and software's use of the NodesTable.</p> <p>Also, below, is a table that describes how the IStax software names nodes based on the RedBox mode and whether a supervision frame that mentions a given node has been received or not.</p>
Forward	<p>This field is only relevant in PRP-SAN mode and a dash ( - ) is shown in other modes.</p> <p>It indicates whether hardware will forward frames destined to this MAC address on Port A, Port B (SANs) or both (DANs).</p>
Rx Port A Rx Port B	Indicates per LRE port the number of frames received with this MAC address as SMAC. A SAN sends to only one LRE port, whereas DANs send to both.
Last Seen Port A Last Seen Port B	<p>Number of seconds ago this MAC address was last seen as SMAC on this LRE port. This field is only valid if the number of received frames is non-zero. Otherwise a dash ( - ) is shown.</p> <p>Remember from Section 5 that an entry can be added by software if a supervision frame is received on that entry's behalf although no data frames have been received from that entry.</p> <p>The value is a rough estimate. Hardware only has a limited number of levels of Last Seen , so the IStax software attempts to tune this to a better granularity based on the first time software sees this entry and its current hardware level.</p>
Rx Wrong LAN Port A Rx Wrong LAN Port B	<p>This field is only relevant in PRP-SAN mode and a dash ( - ) is shown in other modes.</p> <p>It counts if a DANP's RCT.LanId indicates LAN-B, but the frame is received on Port A and vice versa.</p> <p>If this is non-zero, the DANP (or the RedBox itself) is wrongly connected to the PRP network and a notification will be raised (see Section 7.2.1.2).</p>

Hardware knows whether a given node is a SAN or a DAN. The first encounter of a new MAC address causes hardware to add it as a SAN (in HSR modes, this doesn't matter, because there is no such thing as a SAN on the HSR ring). If the same frame is received on both Port A and Port B, hardware automatically updates the entry to a DAN.

Software may refine what it finds in the NodesTable by the reception of supervision frames on the LRE ports. Supervision frames always contain a TLV1, and if the supervision frame is sent by a RedBox on behalf of a SAN, it also contains a TLV2 with that RedBox' MAC address. This information is used to build the Node Type displayed in the **Node Type** column.

The following table summarizes the Node Types that can be shown based on whether supervision frames have been received and whether that supervision frame had a TLV2 and whether the MAC address stems from TLV1.MAC or TLV2.MAC.

<b>Supervision frame received</b>	<b>H/W Type</b>	<b>PRP-SAN mode</b>	<b>HSR-xxx mode</b>	<b>Notes</b>
No	SAN	SAN	DANH	
No	DAN	DANP	DANH	
TLV1.MAC, TLV2 not present	DAN	DANP	DANH	1, 2
TLV1.MAC, TLV2 present	DAN	VDANP	VDANH	1, 2, 3
TLV2.MAC, TLV2 present	DAN	DANP-RedBox	DANH-RedBox	1, 2

**Note 1:**

If a supervision frame is received with a TLV2 at any point in time since this MAC address was added to the NodesTable, that MAC will continue being displayed as a DANx-RedBox, whether the same MAC appears in a TLV1 later on. The reason for this is that a RedBox may send both proxy supervision frames, in which case the RedBox' MAC address appears in TLV2, and it may send its own DANx supervision frames, in which case the RedBox' MAC address appears in TLV1, and there's no TLV2. So to avoid shifting the type back and forth, we make the DANx-RedBox type sticky.

**Note 2:**

The hardware type is set by software to DAN.

**Note 3:**

It is impossible to detect the real origin of a given VDAN node.

As stated in the table, the IStax software names the nodes in the NodesTable after the RedBox's mode (PRP-SAN or any HSR mode). However, a given VDANx node may not really be of the specified type (VDANP or VDANH).

As an example, consider an HSR ring, where we are attached with a RedBox in any HSR mode. Suppose another RedBox in HSR-PRP mode is connected to the same ring. Nodes connected behind that other RedBox are really VDANPs (unless yet another HSR-PRP RedBox connects another HSR ring) and not VDANHs as stated in the table. A similar argument can be made for a RedBox in PRP-SAN mode.

The vigilant reader will notice that the VDANP from the example hasn't sent any frames itself, so it must have been added to the NodesTable because another RedBox has sent a supervision frame with the VDANP in TLV1.MAC (and the RedBox itself as TLV2.MAC). The RedBox must be the only node marked as a RedBox in the table.

Another thing to notice is that this RedBox seems to be wrongly connected to the PRP network, since "Rx Wrong LAN" counts for both Port A and Port B.

Showing both data and supervision frame details in the same table would be prohibitive for print width reasons, so supervision frame reception details are shown with a slightly different command:

```
# show redbox 1 nodes-table supervision
```

Inst	MAC Address	Node Type	Forward	Rx		Last Seen		Last Type	
				Port A	Port B	Port A	Port B	Port A	Port B
1	00-00-00-00-01-00	DANP	Both	10	10	12	12	PRP-DD	PRP-DD
1	00-00-00-00-01-01	SAN	Port A	-	-	-	-	-	-
1	00-00-00-00-01-02	SAN	Port B	-	-	-	-	-	-
1	00-00-00-00-01-03	VDANP	Both	13	13	2	2	HSR	HSR
1	00-00-00-00-01-04	DANP-RedBox	Both	13	13	2	2	HSR	HSR

The command shows all MAC addresses in the NodesTable whether or not supervision frames have been received.

Instead of data frames received on an LRE port, the **Rx** columns show the number of supervision frames received on an LRE port.

The **Last Seen** column shows the number of seconds since the last supervision frame was received mentioning this MAC address. SANs never send supervision frames and are marked with a dash ( - ).

The **Last Type** shows the TLV1.Type held in the last received supervision frame.

Continuing the previous example, not only is the DANP-RedBox connected wrongly, it also transmits supervision frames with TLV1.Type set to HSR instead of PRP-DD or PRP-DA.

Finally, both `show redbox nodes-table details` and `show redbox nodes-table supervision` take an additional keyword that can be used to quickly find entries that have non-zero **Rx Wrong LAN** counters (relevant in PRP-SAN mode only):

```
# show redbox nodes-table details filter
```

Inst	MAC Address	Node Type	Forward	Rx		Last Seen		Rx Wrong LAN	
				Port A	Port B	Port A	Port B	Port A	Port B
1	00-00-00-00-01-04	DANP-RedBox	Both	13	13	2	2		

The following shows how to clear the contents of all RedBoxes' NodesTable in one go:

```
# clear redbox nodes-table
```

And only for instance #1 and #4:

```
# clear redbox 1,4 nodes-table
```

### 7.2.4. ProxyNodeTable Contents

Let's begin with the overview:

```
# show redbox proxy-node-table
Inst Mode    MAC Addresses Wrong LAN
-----
 1 PRP-SAN           3 -
 2 HSR-SAN           5 -
 3 Inactive
 4 HSR-PRP           2 YES!
 5 HSR-HSR           2 -
```

One line per RedBox that contains the instance number, the current mode and the number of MAC addresses in the NodesTable is shown. The last column is only relevant in HSR-PRP mode, and reads **No** if all DANPs are connected correctly to the PRP network, **YES!** otherwise.

The layout of the detailed contents of the ProxyNodeTable is very similar to that of the NodesTable, but instead of showing data for two LRE ports, there is only one - the interlink port. This means that data and supervision info are shown in the same detailed table. To see details for a particular RedBox, add the **details** keyword to the command. In the following constructed example, only details for RedBox #2 are shown.

```
# show redbox 2 proxy-node-table details
                                     Data                Supervision
-----
Inst MAC Address      Node Type  Rx      Last Seen  Rx Wrong LAN Rx      Tx      Last
Seen  Last Type
-----
 2 00-00-00-00-02-00  VDANH      534258      0          -          -      1008
-
 2 00-00-00-00-02-01* VDANH      31          45          -          -      30308
-
 2 00-00-00-00-02-02* DANH-RedBox 56          10          -          -      30308
-
 2 00-00-00-00-02-03  VDANH      327         43          -          -      12
-
 2 00-00-00-00-02-04  VDANH      80          33          -          -      188
-
-
```

In this example, we show the ProxyNodeTable contents of a RedBox in HSR-SAN mode. The listing contains one line per entry, sorted by MAC address.

The contents of each column is described in the following:

Name	Description
Inst	The RedBox instance this row relates to.
MAC Address	The MAC address this row relates to. MAC addresses marked with a * are added by the RedBox itself. In this example, the first one is for the switch's management MAC address. The management MAC address is a VDANH in HSR-SAN mode. The second one is the RedBox' own MAC address. These two addresses can never age out and are not cleared when clearing the ProxyNodeTable.
Node Type	The IStax software detects and names the node after the RedBox' mode of operation and reception of supervision frames on the interlink port in HSR-PRP mode. See Section 4.2 and Section 5 for a description of hardware's and software's use of the ProxyNodeTable. Also, below, is a table that describes how the IStax software names nodes based on the RedBox mode and whether a supervision frame that mentions a given node has been received or not.
Data Rx	Indicates the number of frames received with this MAC address as SMAC on the interlink port.
Data Last Seen	Number of seconds ago this MAC address was last seen as SMAC on the interlink port. This field is only valid if the number of received frames is non-zero. Otherwise a dash ( - ) is shown. Remember from Section 5 that an entry can be added by software if a supervision frame is received on that entry's behalf although no data frames have been received from that entry. The value is a rough estimate. Hardware only has a limited number of levels of Last Seen , so the IStax software attempts to tune this to a better granularity based on the first time software sees this entry and its current hardware level.
Data Rx Wrong LAN	This field is only relevant in HSR-PRP mode and a dash ( - ) is shown in other modes. It counts if a DANP's RCT.LanId indicates LAN-B, but the frame is received on Port A and vice versa. If this is non-zero, the DANP (or the RedBox itself) is wrongly connected to the PRP network and a notification will be raised (see Section 7.2.1.2).
Supervision Rx	Only relevant in HSR-PRP and HSR-HSR mode. In other modes, it is shown as a dash ( - ). It indicates the number of supervision frames received from the bridge-side of the RedBox mentioning this MAC address.

Supervision Tx	Number of proxied supervision frames transmitted towards the LRE ports on behalf of VDANx and the RedBox' own MAC address (DANx-RedBox). In HSR-PRP mode, the RedBox also transmits supervision frames towards the PRP network for the RedBox itself, only. These are also counted here.
Supervision Last Seen	Number of seconds ago a supervision frame mentioning this MAC address was received on the interlink port. This field is only relevant in HSR-PRP and HSR-HSR mode. In other modes, it is shown as a dash ( - ). If it is more than the ProxyNodeTable age time seconds ago a supervision frame was received for this MAC address and no data frames have been received in the meanwhile, the entry ages out.
Supervision Last Type	This shows the TLV1.Type held in the last received supervision frame. If no supervision frame has been received or the RedBox is not in HSR-PRP or HSR-HSR mode, a dash is shown.

The naming of the nodes ( **Node Type** ) depends on the currently configured mode and - in HSR-PRP mode - whether supervision frames have been received mentioning the MAC address. The following table summarizes the Node Types that can be shown. The "Own MAC" column is "Yes" when it's one of the two MAC addresses the RedBox has added itself, "No" otherwise.

To understand the table, let's take an example: A supervision frame with both TLV1 and TLV2 present is received, and it does not mention one of the RedBox' own MAC addresses. This is row 5 (TLV1.MAC, TLV2 Present) and row 6 (TLV2.MAC, TLV2 Present).

Supervision frames received on the interlink are discarded in PRP-SAN and HSR-SAN modes and hardware forwarded in HSR-HSR mode and have no influence on the naming of nodes in these modes (marked as "N/A" in the table).

Therefore, only HSR-PRP is of interest. Since the supervision frame contains both a TLV1 and a TLV2, it must have been sent by a RedBox. RedBoxes that can be attached to the PRP network are PRP-SAN and HSR-PRP RedBoxes. So we know that TLV1.MAC (row 5) is a VDANP (if sent by a PRP-SAN RedBox) or VDANH (if sent by a HSR-PRP RedBox). Both are shown as VDANPs, since we cannot really distinguish the two kinds. TLV2.MAC (row 6) is the PRP-SAN or HSR-PRP RedBox itself, which is always displayed as a DANP-RedBox.

Had TLV2 not been present, it would have been originated by a real DANP (row 4) or a DANH behind an HSR-PRP RedBox without HSR-to-PRP supervision frame translation enabled.

The mere reception of such supervision frames in HSR-PRP mode causes the IStax software to mark both TLV1.MAC and TLV2.MAC as DANs in the hardware ProxyNodeTable.

<b>Supervision frame received</b>	<b>Own MAC</b>	<b>H/W Type</b>	<b>PRP-SAN mode</b>	<b>HSR-SAN mode</b>	<b>HSR-PRP mode</b>	<b>HSR-HSR mode</b>	<b>Notes</b>
No	Yes	SAN	VDANP	VDANH	VDANP	VDANH	1
No	Yes	DAN	DANP-RedBox	DANH-RedBox	DANH-RedBox	DANH-RedBox	2
No	No	SAN	VDANP	VDANH	VDANP	N/A	3
TLV1.MAC, TLV2 not present	No	DAN	N/A	N/A	DANP	N/A	3, 4, 5, 6, 7
TLV1.MAC, TLV2 present	No	DAN	N/A	N/A	VDANP	N/A	3, 4, 5, 6, 7
TLV2.MAC, TLV2 present	No	DAN	N/A	N/A	DANP-RedBox	N/A	3, 5, 6, 7

**Note 1:**

This is the switch's management MAC address.

**Note 2:**

This is the RedBox' MAC address. In HSR-PRP mode, the RedBox is shown as a DANH-RedBox, because it connects to the HSR ring, but it could also have been shown as a DANP-RedBox, because it also connects to the PRP network.

**Note 3:**

The ProxyNodeTable is not used in HSR-HSR mode, except for our own two MAC addresses.

**Note 4:**

In HSR-PRP mode, it is impossible to detect the real origin of a given DANP or VDANP node.

As an example, consider a PRP network, where another RedBox in HSR-PRP mode translates HSR supervision frames arriving on its LRE ports to PRP supervision frames before sending them to the PRP network. When this supervision frame arrives at our RedBox, it looks like any other supervision frame transmitted by e.g. a PRP-SAN RedBox connected to the PRP network.

**Note 5:**

Supervision frames arriving on the interlink port are only used in HSR-PRP mode.

**Note 6:**

Reception of a supervision frame causes the entry to be changed to a DAN.

**Note 7:**

If a supervision frame is received with a TLV2 at any point in time since this MAC address was added to the ProxyNodeTable, that MAC will continue being displayed as a DANP-RedBox, whether the same MAC appears in a TLV1 later on. The reason



for this is that a RedBox may send both proxy supervision frames, in which case the RedBox' MAC address appears in TLV2, and it may send its own DANP supervision frames, in which case the RedBox' MAC address appears in TLV1, and there's no TLV2. So to avoid shifting the type back and forth, we make the DANP-RedBox type sticky.

In HSR-PRP mode, the Rx Wrong LAN may count if a DANP is connected wrongly to the PRP network. To quickly identify such nodes, you may add the `filter` keyword to the command, i.e.

```
# show redbox proxy-node-table details filter
```

Data					Supervision				
Inst	MAC	Address	Node Type	Rx	Last Seen	Rx Wrong LAN	Rx	Tx	Last
Seen	Last	Type							
4	00-00-00-00-04-00		DANP	1999	10	1	3	0	
17	PRP-DD								

This continues the example, where only RedBox #4 is in HSR-PRP mode, so that is the only contributor to this.

Lastly, to clear all RedBoxes' ProxyNodeTable in one go, do:

```
# clear redbox proxy-node-table
```

To clear only instance #1 and #4:

```
# clear redbox 1,4 proxy-node-table
```

Remember that clearing the ProxyNodeTable only clears the dynamic MAC addresses, not the RedBox' two own.

## 8. Examples

This section provides configuration examples.

The examples take base in some of the figures previously presented.

### 8.1. PRP-SAN

Figure 1 showed a PRP-SAN RedBox connected to a PRP network through the LRE ports and two SANs through the bridge-side (SAN-side).

In the following example, we assign GigabitEthernet 1/1 (Gi 1/1) to Port A and Gi 1/2 to Port B of RedBox #1 and Gi 1/3,4 connect the two SANs (VDANPs) to the RedBox' bridge-side.

Command	Purpose
---------	---------

# <b>configure terminal</b>	Enter configuration mode.
(config)# <b>vlan 2</b> (config-vlan)# <b>interface</b> <b>GigabitEthernet 1/1-2</b> (config-if)# <b>switchport access vlan 2</b> (config-if)# <b>no spanning-tree</b> (config-if)# <b>mtu 2000</b>	Create VLAN 2 and make the LRE ports members of this VLAN. This is not strictly necessary, but to ease debugging, it is good to separate traffic in the setup from traffic from other ports. As described in the Section 7.2.1.1 section, spanning tree must be disabled and there are restrictions on the MTU.
(config-if)# <b>interface</b> <b>GigabitEthernet 1/3-4</b> (config-if)# <b>switchport access vlan 2</b> (config-if)# <b>mtu 1994</b>	We also make the non-LRE ports in our setup members of VLAN 2, and make sure that frames beyond 1994 bytes of size cannot enter these ports. We leave STP enabled on these ports in order to prevent loops on the SAN side of the RedBox.
(config-if)# <b>redbox 1</b>	Enter configuration mode of RedBox #1, which is the one that supports the two LRE ports we have chosen in the setup.
(config-redbox)# <b>mode prp-san</b>	Configure the RedBox mode to PRP-SAN. This is the default mode and is not necessary explicitly to type.
(config-redbox)# <b>port-a interface</b> <b>GigabitEthernet 1/1</b> (config-redbox)# <b>port-b interface</b> <b>GigabitEthernet 1/2</b>	Configure Port A and Port B according to the setup.
(config-redbox)# <b>admin-state enable</b>	We leave the remaining configuration options at their defaults and enable this RedBox. Not until this line is executed will the RedBox configuration be applied to hardware.

To summarize, here's a list of commands, where default and irrelevant commands are omitted.

```

vlan 2

interface GigabitEthernet 1/1
 switchport access vlan 2
 no spanning-tree
 mtu 2000

interface GigabitEthernet 1/2
 switchport access vlan 2
 no spanning-tree
 mtu 2000

interface GigabitEthernet 1/3
 switchport access vlan 2
 mtu 1994

interface GigabitEthernet 1/4
 switchport access vlan 2
 mtu 1994

redbox 1
 mode prp-san
 port-a interface GigabitEthernet 1/1
 port-b interface GigabitEthernet 1/2
 admin-state enable

```

Now, let's check the status of the RedBox to see if there are any configurational warnings or notifications.

```

# show redbox 1 status
Inst Oper. State Mode Port A Port B Port C Warnings Notifications
-----
1 Active PRP-SAN Gi 1/1 Gi 1/2 Gi 1/1 No No

```

It's all good - no configurational warnings and no notifications.

## 8.2. HSR-SAN

With basis in Figure 2, let's configure an HSR-SAN RedBox by re-using the RedBox from the previous example: Gi 1/1 is Port A, Gi 1/2 is Port B, Gi 1/3,4 are connected to the two SANs (VDANHs).

This is achieved simply by changing the mode of RedBox #1:

Command	Purpose
# configure terminal	Enter configuration mode.
(config)# redbox 1 (config-redbox)# mode hsr-san	Change the existing RedBox' mode from PRP-SAN to HSR-SAN

This gives the following running-config, where default and irrelevant commands are omitted.

```
vlan 2

interface GigabitEthernet 1/1
  switchport access vlan 2
  no spanning-tree
  mtu 2000

interface GigabitEthernet 1/2
  switchport access vlan 2
  no spanning-tree
  mtu 2000

interface GigabitEthernet 1/3
  switchport access vlan 2
  mtu 1994

interface GigabitEthernet 1/4
  switchport access vlan 2
  mtu 1994

redbox 1
  mode hsr-san
  port-a interface GigabitEthernet 1/1
  port-b interface GigabitEthernet 1/2
  admin-state enable
```

### 8.3. HSR-PRP

Configuration of the HSR-PRP RedBox from Figure 3 is almost identical to the configuration of the PRP-SAN and HSR-SAN RedBox from the previous examples, so instead let's turn towards Figure 4.

Implementing HSR-PRP RedBox 1A and HSR-PRP RedBox 1B in the same switch chip would break redundancy, so in this example, we will configure HSR-PRP RedBox 1A and the PRP-SAN RedBox shown at the top of the figure on the same switch chip.

The following illustration re-arranges Figure 4 and annotates port interfaces.

The number in parentheses preceded by a hash tag is the RedBox instance number.

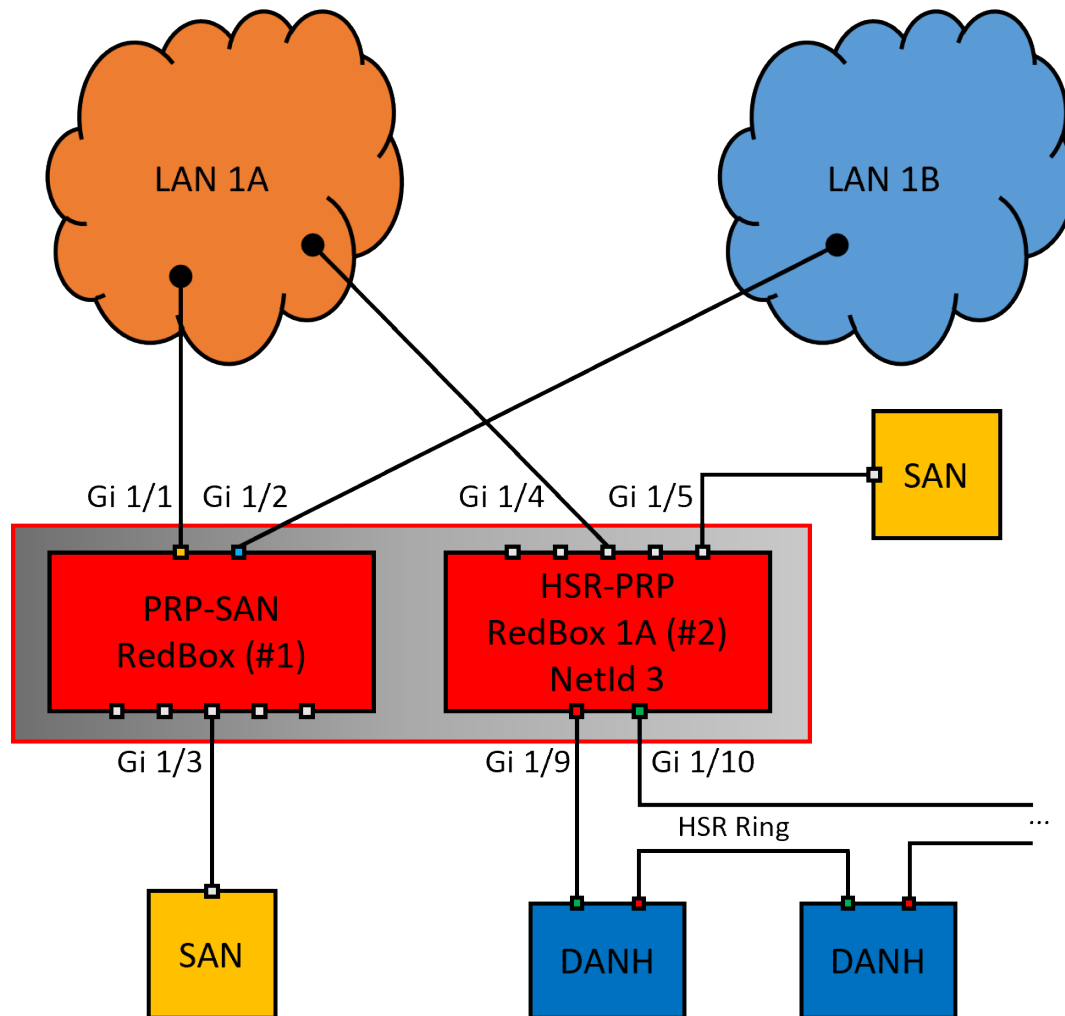


Figure 9. One PRP-SAN and one HSR-PRP RedBox on the same switch chip

Since the two RedBoxes are located on the same chip, we must prevent frames on the bridge-side to cross directly between the two RedBoxes. For instance, frames ingressing Gi 1/3 may not egress Gi 1/4 directly, because that will defeat the whole idea of RedBoxes. Therefore, we assign VLAN 2 to ports in RedBox #1 and VLAN 3 to ports in RedBox #2.

We pick NetId 3 for the HSR ring.

Command	Purpose
# <b>configure terminal</b>	Enter configuration mode.
(config)# <b>vlan 2,3</b>	Create two VLANs. We use VLAN 2 for the PRP-SAN RedBox and VLAN 3 for the HSR-PRP RedBox.

<pre>(config-vlan)# interface GigabitEthernet 1/1,2 (config-if)# switchport access vlan 2 (config-if)# no spanning-tree (config-if)# mtu 2000</pre>	Make the PRP-SAN RedBox's LRE ports members of VLAN 2, disable spanning tree and make restrictions on the MTU.
<pre>(config-if)# interface GigabitEthernet 1/3 (config-if)# switchport access vlan 2 (config-if)# mtu 1994</pre>	Make the PRP-SAN RedBox's bridge-side port member of VLAN 2 and restrict the MTU
<pre>(config-if)# interface GigabitEthernet 1/9,10 (config-if)# switchport access vlan 3 (config-if)# no spanning-tree (config-if)# mtu 2000</pre>	Make the HSR-PRP RedBox's LRE ports members of VLAN 3, disable spanning tree and make restrictions on the MTU.
<pre>(config-if)# interface GigabitEthernet 1/4,5 (config-if)# switchport access vlan 3 (config-if)# mtu 1994</pre>	Make the HSR-PRP RedBox' bridge-side ports members of VLAN 3 and restrict the MTU.
<pre>(config-if)# redbox 1</pre>	Enter configuration mode of the PRP-SAN RedBox.
<pre>(config-redbox)# mode prp-san</pre>	Configure the RedBox mode to PRP-SAN.
<pre>(config-redbox)# port-a interface GigabitEthernet 1/1 (config-redbox)# port-b interface GigabitEthernet 1/2</pre>	Configure Port A and Port B according to the setup.
<pre>(config-redbox)# admin-state enable</pre>	Enable the PRP-SAN RedBox.
<pre>(config-redbox)# redbox 2</pre>	Enter configuration mode of the HSR-PRP RedBox.
<pre>(config-redbox)# mode hsr-prp</pre>	Configure the RedBox mode to HSR-PRP.
<pre>(config-redbox)# port-a interface GigabitEthernet 1/9 (config-redbox)# port-b interface GigabitEthernet 1/10</pre>	Configure Port A and Port B according to the setup.

(config-redbox)# <b>lan-id a</b>	The HSR-PRP RedBox connects to LAN A on the PRP network side. This is the default and will not be shown in running-config.
(config-redbox)# <b>net-id 3</b>	The HSR-PRP RedBox connects to the HSR ring with NetId 3.
(config-redbox)# <b>admin-state enable</b>	Enable the HSR-PRP RedBox.

That's it. This boils down to the following running-config:

```

interface GigabitEthernet 1/1
  switchport access vlan 2
  no spanning-tree
  mtu 2000

interface GigabitEthernet 1/2
  switchport access vlan 2
  no spanning-tree
  mtu 2000

interface GigabitEthernet 1/3
  switchport access vlan 2
  mtu 1994

interface GigabitEthernet 1/4
  switchport access vlan 3
  mtu 1994

interface GigabitEthernet 1/5
  switchport access vlan 3
  mtu 1994

interface GigabitEthernet 1/9
  switchport access vlan 3
  no spanning-tree
  mtu 2000

interface GigabitEthernet 1/10
  switchport access vlan 3
  no spanning-tree
  mtu 2000

redbox 1
  mode prp-san
  port-a interface GigabitEthernet 1/1
  port-b interface GigabitEthernet 1/2
  admin-state enable
!
redbox 2
  mode hsr-prp
  port-a interface GigabitEthernet 1/9
  port-b interface GigabitEthernet 1/10
  net-id 3
  admin-state enable

```

And gives the following status:

```

# show redbox status
Inst Oper. State Mode   Port A   Port B   Port C   Warnings Notifications
-----
  1 Active   PRP-SAN Gi 1/1   Gi 1/2   Gi 1/1   No       YES!
  2 Active   HSR-PRP Gi 1/9   Gi 1/10  Gi 1/9   No       YES!

```



There are no configurational warnings, so from that perspective the configuration is good.

There are, however, notifications, which can be seen with `show redbox status details`, but these are irrelevant for the configuration.

## 8.4. HSR-HSR

In this example we will build the configuration of QuadBox A and B from Figure 6.

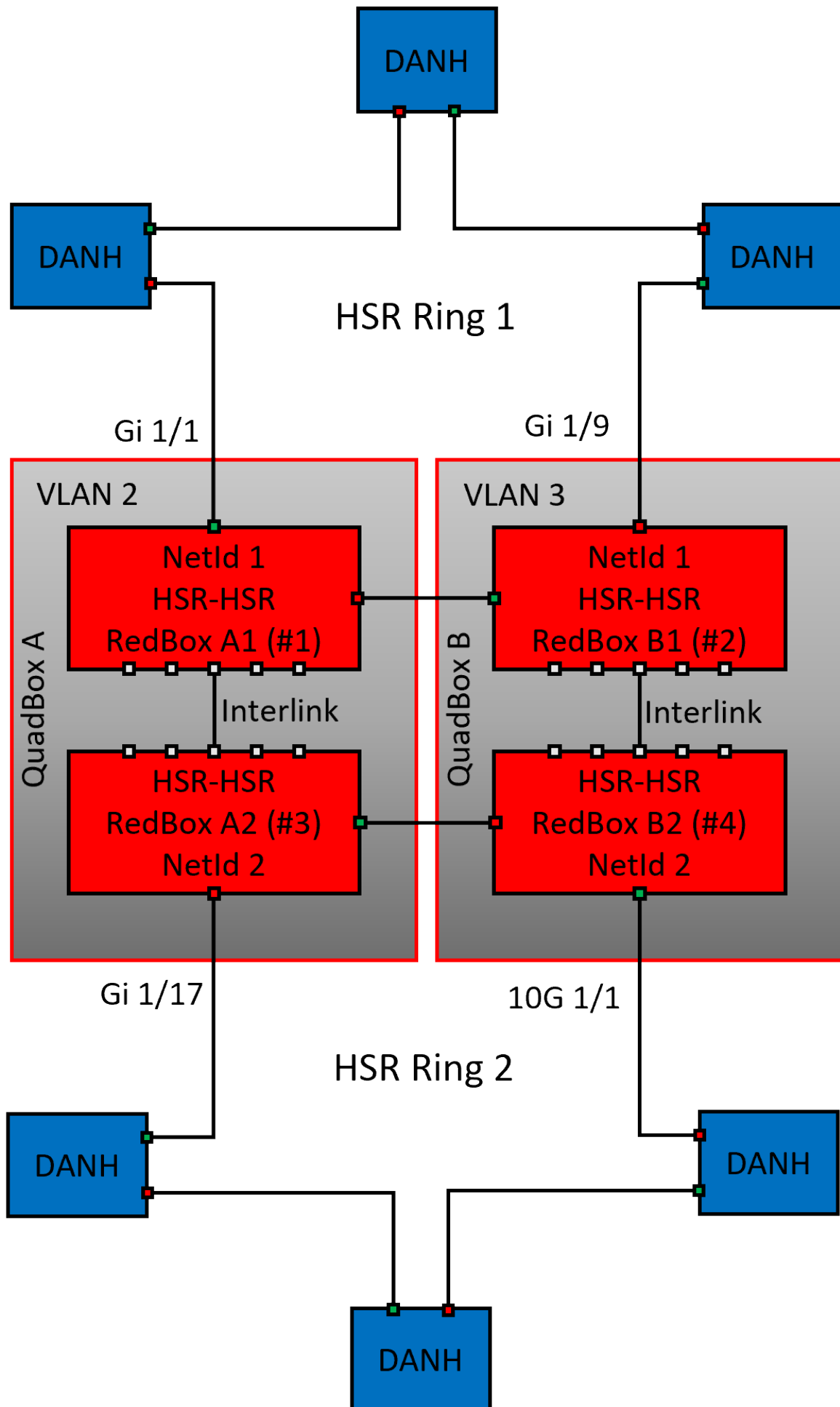
Although not recommended from a redundancy point of view, we will let all four RedBoxes reside on the same switch chip.

In real-world scenarios, it is recommended to have four different chips or at least two, where QuadBox A is located on one chip, and QuadBox B on another.

The following figure shows how the four RedBoxes will be connected. The number in parentheses preceded by a hash tag is the RedBox instance number. RedBox A1 connects internally to RedBox B1 and similarly for RedBox A2 and B2.

The connection between the two RedBoxes inside a QuadBox is the interlink. Since the two QuadBoxes are located on the same chip, and we can't let frames on the bridge-sides cross from QuadBox A to QuadBox B, they must be VLAN separated, so we let QuadBox A run in VLAN 2 and QuadBox B run in VLAN 3.

We assign NetId 1 to HSR Ring 1 and NetId 2 to HSR Ring 2.



*Figure 10. All four RedBoxes are located on the same switch chip*

The configuration recipe is pretty much the same as for the other examples:

Command	Purpose
<code># configure terminal</code>	Enter configuration mode.
<code>(config)# vlan 2,3</code>	Create two VLANs. We use VLAN 2 for QuadBox A and VLAN 3 for QuadBox B.
<code>(config-vlan)# interface GigabitEthernet 1/1,17 (config-if)# switchport access vlan 2 (config-if)# no spanning-tree (config-if)# mtu 2000</code>	Make QuadBox A's LRE ports and thereby interlink members of VLAN 2, disable spanning tree and make restrictions on the MTU.
<code>(config-if)# interface GigabitEthernet 1/9 10GigabitEthernet 1/1 (config-if)# switchport access vlan 3 (config-if)# no spanning-tree (config-if)# mtu 2000</code>	Make QuadBox B's LRE ports members and thereby interlink of VLAN 3, disable spanning tree and make restrictions on the MTU.
<code>(config-if)# redbox 1</code>	Enter configuration mode of RedBox A1.
<code>(config-redbox)# mode hsr-hsr</code>	Configure the RedBox mode to HSR-HSR.
<code>(config-redbox)# port-a interface GigabitEthernet 1/1 (config-redbox)# port-b interface neighbor</code>	Configure Port A as a physical interface and Port B to connect internally to RedBox #2 (RedBox B1).
<code>(config-redbox)# net-id 1</code>	RedBox A1 connects to an HSR ring identified by NetId 1. This is the default and will not be shown in the running-config.
<code>(config-redbox)# admin-state enable</code>	Enable RedBox A1.
<code>(config-redbox)# redbox 2 (config-redbox)# mode hsr-hsr</code>	Enter configuration mode of RedBox B1 and configure it for HSR-HSR mode.
<code>(config-redbox)# port-a interface neighbor (config-redbox)# port-b interface GigabitEthernet 1/9</code>	Configure Port A to connect internally to RedBox A1 and Port B to connect to a physical interface.

(config-redbox)# <b>net-id 1</b>	RedBox B1 also connects to the HSR ring with NetId 1.
(config-redbox)# <b>admin-state enable</b>	Enable RedBox B1.
(config-redbox)# <b>redbox 3</b> (config-redbox)# <b>mode hsr-hsr</b>	Enter configuration mode of RedBox A2 and configure it for HSR-HSR mode.
(config-redbox)# <b>port-a interface GigabitEthernet 1/17</b> (config-redbox)# <b>port-b interface neighbor</b>	Configure Port A as a physical interface and Port B to connect internally to RedBox #4 (RedBox B2).
(config-redbox)# <b>net-id 2</b>	RedBox A2 connects to an HSR ring identified by NetId 2.
(config-redbox)# <b>admin-state enable</b>	Enable RedBox A2.
(config-redbox)# <b>redbox 4</b> (config-redbox)# <b>mode hsr-hsr</b>	Enter configuration mode of RedBox B2 and configure it for HSR-HSR mode.
(config-redbox)# <b>port-a interface neighbor</b> (config-redbox)# <b>port-b interface 10GigabitEthernet 1/1</b>	Configure Port A to connect internally to RedBox A2 and Port B to connect to a physical interface.
(config-redbox)# <b>net-id 2</b>	RedBox B2 also connects to the HSR ring with NetId 2.
(config-redbox)# <b>admin-state enable</b>	Enable RedBox B2.

This gives the following running-config, where irrelevant commands are left out:

```
vlan 2,3

interface GigabitEthernet 1/1,17
 switchport access vlan 2
 no spanning-tree
 mtu 2000

interface GigabitEthernet 1/9 10GigabitEthernet 1/1
 switchport access vlan 3
 no spanning-tree
 mtu 2000

redbox 1
 mode hsr-hsr
 port-a interface GigabitEthernet 1/1
 port-b interface neighbor
 admin-state enable

redbox 2
 mode hsr-hsr
 port-a interface neighbor
 port-b interface GigabitEthernet 1/9
 admin-state enable

redbox 3
 mode hsr-hsr
 port-a interface GigabitEthernet 1/17
 port-b interface neighbor
 net-id 2
 admin-state enable

redbox 4
 mode hsr-hsr
 port-a interface neighbor
 port-b interface 10GigabitEthernet 1/1
 net-id 2
 admin-state enable
```

The status is the following:

```
# show redbox status
Inst Oper. State Mode   Port A      Port B      Port C      Warnings Notifications
-----
  1 Active   HSR-HSR Gi 1/1    Neighbor    Gi 1/1      No          YES!
  2 Active   HSR-HSR Neighbor   Gi 1/9      Gi 1/9      No          YES!
  3 Active   HSR-HSR Gi 1/17    Neighbor    Gi 1/17     No          YES!
  4 Active   HSR-HSR Neighbor   10G 1/1     10G 1/1     No          YES!
```

The notifications are irrelevant here, but should be studied in a real setup.