MAC NE in n:N FAILURE PROTECTION

INVENTOR:

JASON W. RUPE

Description

All forms of MAC NE (RMD, RMC, RPD) could be designed and configured to allow full pass through or to act as a PHY repeater if instructed to do so. By specifying messaging to the MAC NEs from the MAC Manager or Core, and/or a peer to peer signaling method between MAC NEs, a system of n redundant spares protecting N actives can be established. Then, if there is failure of one active NE, there is a close enough backup that can serve to replace it, if the failed NE can fail with pass through. In simplest forms, they can be set in 1:1 protection; but there may be situations where 1:N or even n:N are possible.

This relates to the non-provisional Network Grafting (61126) and the FDX Amp for Trees to Mesh idea entered previously.

If properly configured, the load could be shared in such a way that one physical MAC NE could handle PHY duties while one or more share MAC duties in a redundant way. This could be done so that all manners of MAC NE may be mixed in a network with load and resource sharing at upper layers, with or without PHY redundancy.

Back haul to a core backup MAC or PHY device could be enabled too, as an architecture option.

Method Flow:

MAC NE (remote) N+1 is the protect remote, protecting remotes 1 to N. One of the N remotes fails. In this flow, the protect remove is in the core, but doesn't need to be. It can even be one that provides service and load shares. Say arbitrarily that remove 1 fails. Detecting loss of US or a signal of remote failure from the interface port (for examples), and detection that the remote has gone into pass through mode (which may be inferred or assumed alternately), the core routes the optical traffic to the protect remote and instructs it to take over processing. If the failed remote can continue to convert optical to RF, then the protect remote will handle any processing needed, package as DOCSIS over optical or whichever agreed protocol is determined, which is then handled by the crippled remote. If the failed remote is not capable of this translation, then the core sends the optical signal to the protect remote which handles the remote duties now, and sends an RF signal properly boosted, via coax connection to the crippled remote. The crippled remote in pass through will do nothing if needed, or may amplify the signal if able and useful to do so. Signals continue in one or both directions this way.

In the case of an active load share, the active remote serves as the protect remote as described above without change, and simply handles the new load and its previous load.

If the remotes are branches from the core, then routing will be from core to remote, back to core, and out to the crippled remote. If they are in a ring, then the routing is the same as in the fully functional case except that the protect remote is not passed through but traffic drops there and then continues on the ring (drop and continue) to the crippled remote. If the remotes are optically branched from the core but coax between, then the traffic continues optically to the protect remote, then gets translated to RF to continue to the crippled remote location or out directly to the CMs protected. If optical and coax are branched out in parallel to the remotes then then the protect remote may connect via RF or optical as needed to the CMs or the crippled remote for pass through. Some of the possible configurations are in the attached drawing.

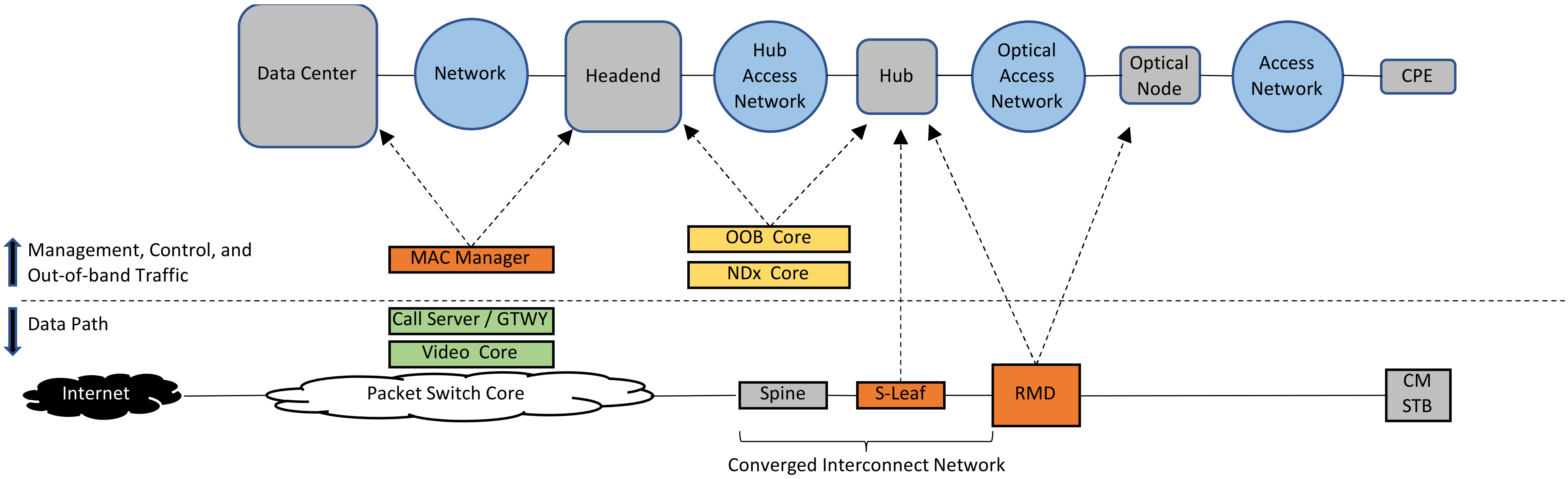
10G needs reliability options but coax plant is a tree, so you are limited in what protection mechanisms can be implemented. Relying on, and perhaps expanding on, our Network Grafting patent, we can envision MAC NE who function in a way to allow protection of these active elements, and even shared protection for efficiency.

Abstract

All forms of MAC NE (RMD, RMC, RPD) could be designed and configured to allow full pass through or to act as a PHY repeater if instructed to do so. By specifying messaging to the MAC NEs from the MAC Manager or Core, and/or a peer to peer signaling method between MAC NEs, a system of n redundant spares protecting N actives can be established. Then, if there is failure of one active NE, there is a close enough backup that can serve to replace it, if the failed NE can fail with pass through. In simplest forms, they can be set in 1:1 protection; but there may be situations where 1:N or even n:N are possible.

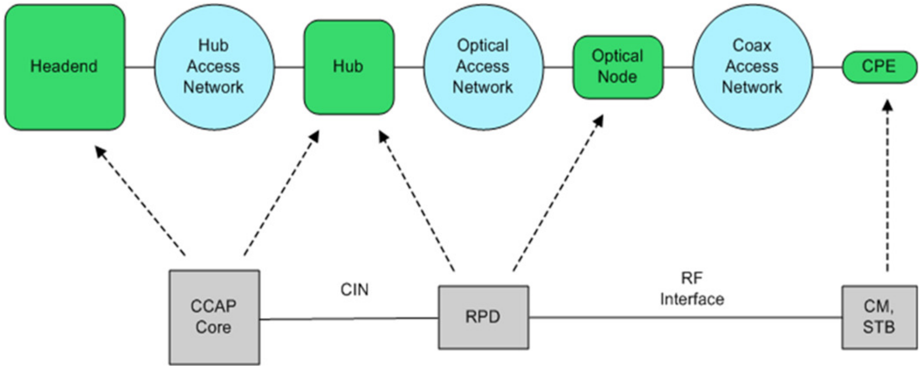
MAC NE in n:N failure protection

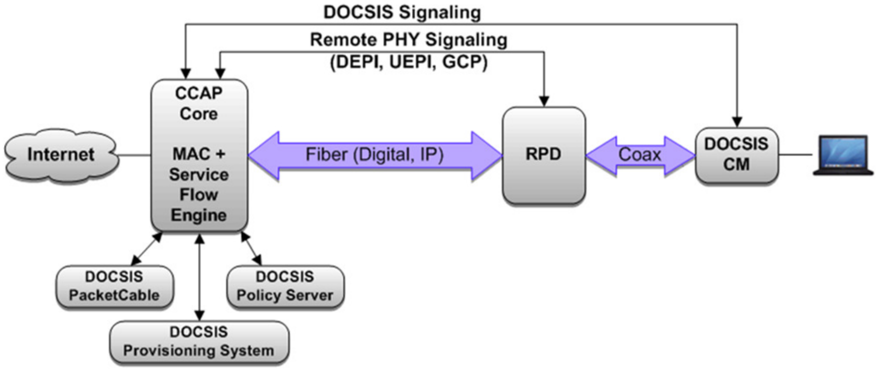
High Level FMA Architecture

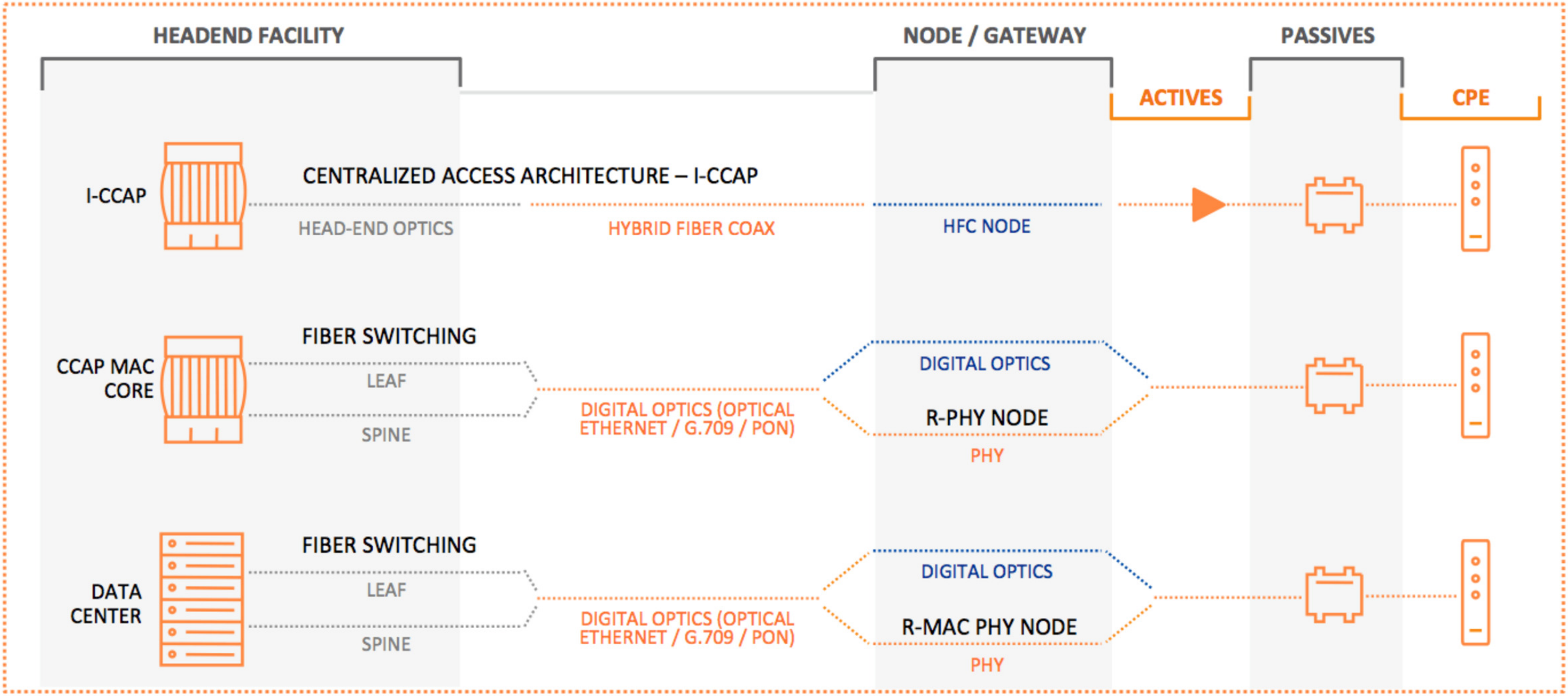


Orange Boxes Comprise CCAP Functions

High Level RPHY Architecture






HEADEND FACILITY
NODE / GATEWAY
PASSIVES
ACTIVES
CPE
I-CCAP
CENTRALIZED ACCESS ARCHITECTURE – I-CCAP
HEAD-END OPTICS
HYBRID FIBER COAX
HFC NODE
CCAP MAC CORE
FIBER SWITCHING
LEAF
SPINE
DIGITAL OPTICS (OPTICAL ETHERNET / G.709 / PON)
DIGITAL OPTICS
R-PHY NODE
PHY
DATA CENTER
FIBER SWITCHING
LEAF
SPINE
DIGITAL OPTICS (OPTICAL ETHERNET / G.709 / PON)
DIGITAL OPTICS
R-MAC PHY NODE
PHY

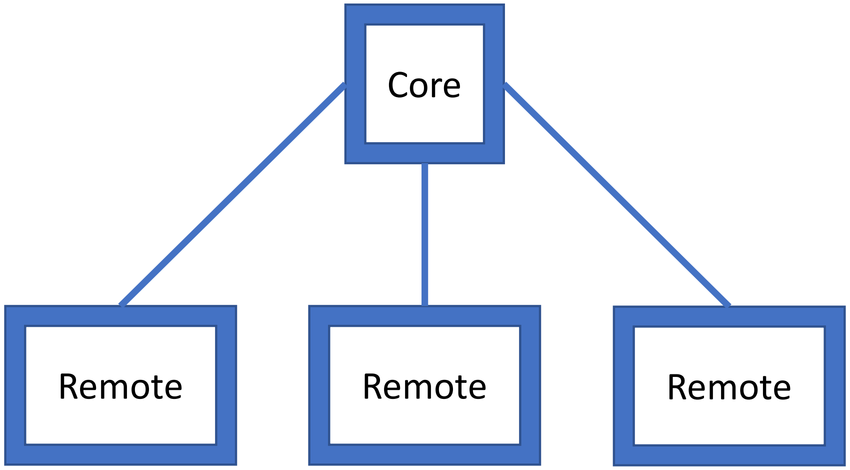
Components

- Remote
  - Remote PHY – only PHY function at remote
  - Remote MAC PHY = Flexible MAC Architecture – MAC and PHY function at remote

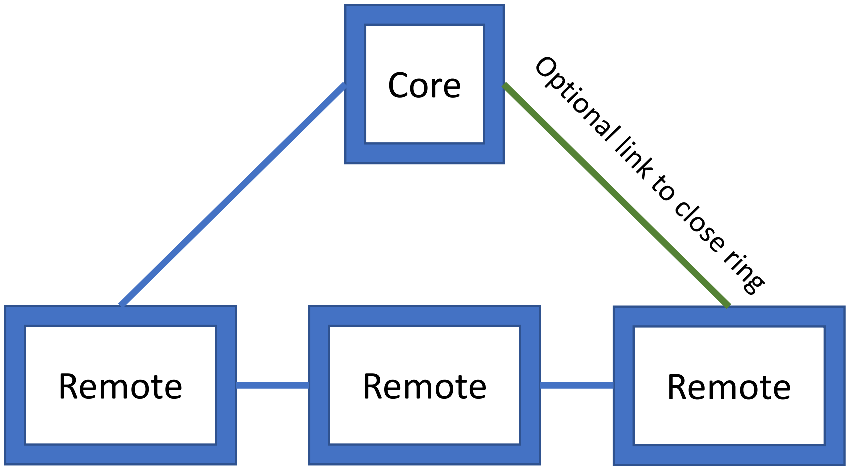
- Core
  - Remote PHY – MAC core – PNM and management servers in the core
  - Remote MAC PHY – data center in core – includes a MAC Manager function

Configurations

- Tree



- Ring (open or closed)

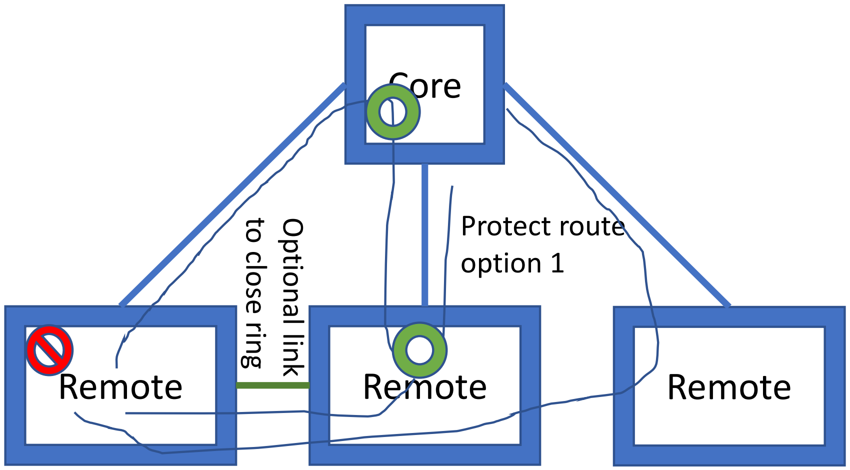


Can be more than one remote to a core.
Can use redundant cores to remotes too, by duplicating links in the tree or adding a second core to a ring

Protection examples

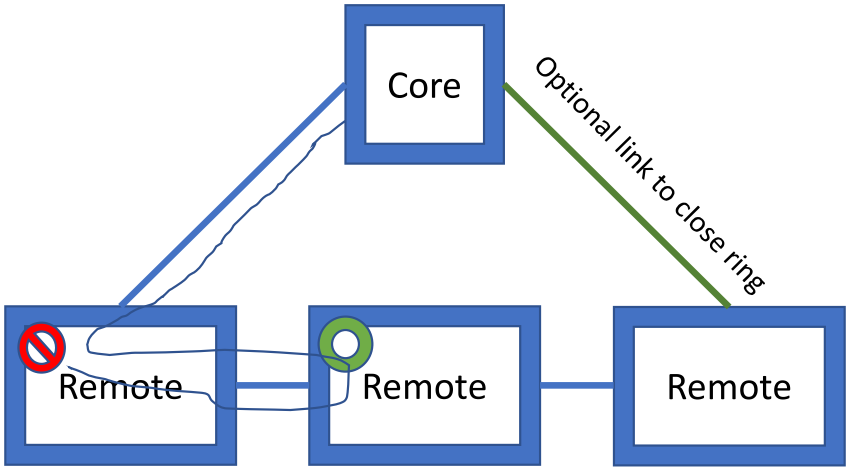
Protecting a function in a remote requires connecting traffic, likely tunneled in some encapsulation, back to a protecting function in another remote or core, which then handles the failed function, then connecting through the remote with the failed function to pass through and keep traffic functioning. If all functions fail, then failed remote goes to pass through, and traffic from core to failed remote must hairpin through the protect remote functions first.

- Tree



Protect route
option 2 & 3, if optional link available

- Ring (open or closed)



Protect route may or may not require communication with core depending on how implemented.

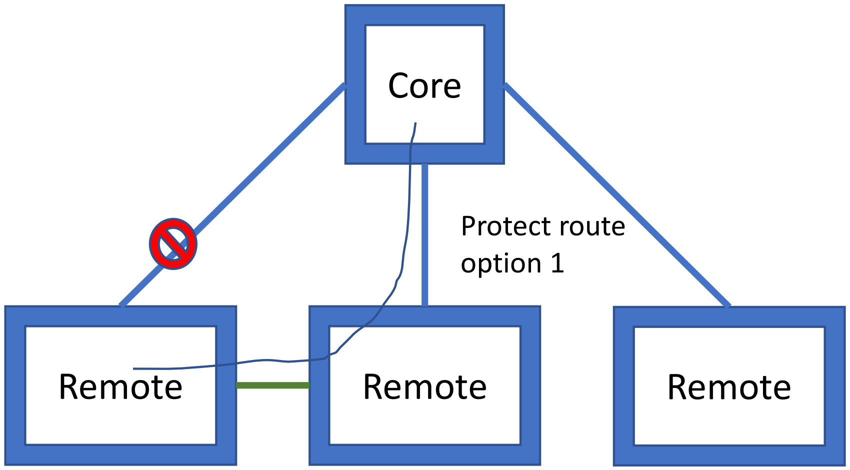
🚫 = failed function

○ = backup function (potential locations if more than one shown)

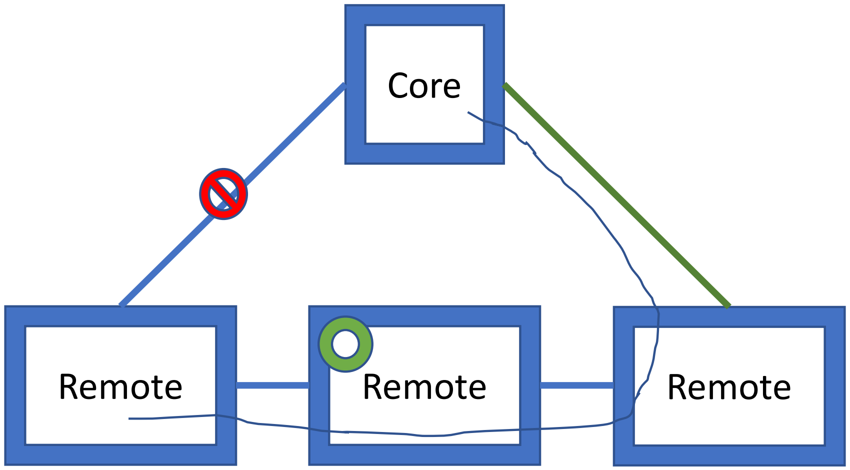
Protection examples

Protecting a link failure is simpler as a reroute is sufficient.

- Tree



- Ring (open or closed)



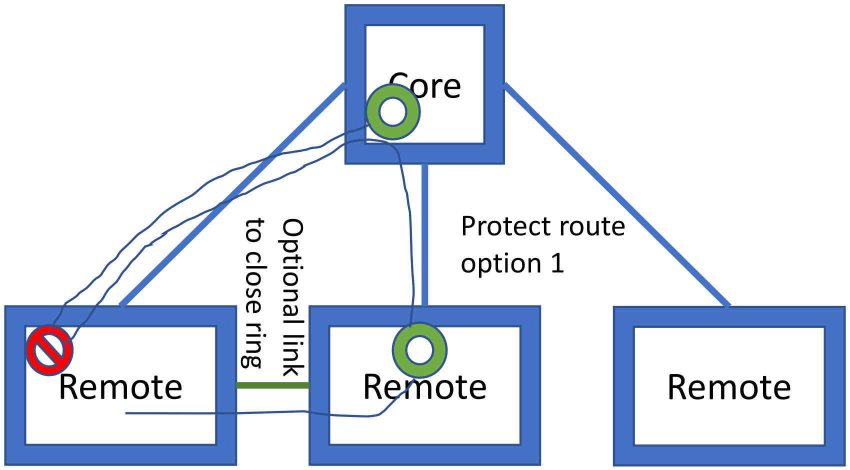
🚫 = failed function (link in this case)

○ = backup function (potential locations if more than one shown)

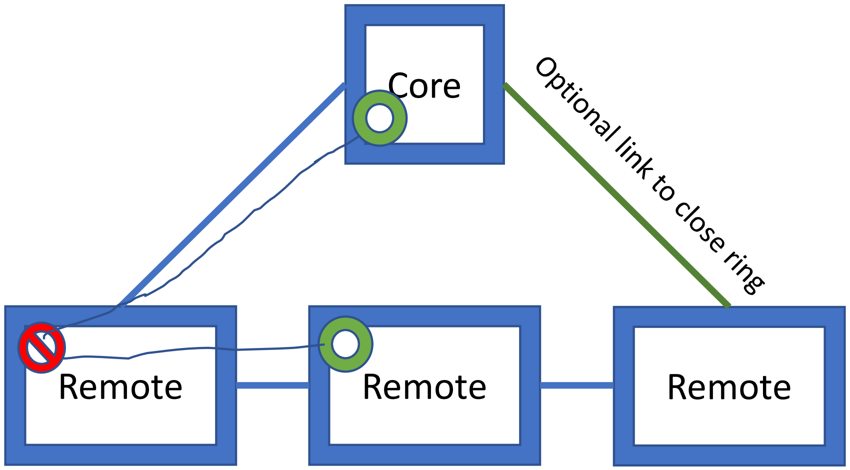
Protection examples

State can be redundantly maintained in the working and protect functions so that if only state is lost it can be shared to allow the failed function to gain the backup state and recover. Likewise, if firmware or software is lost in one remote (fully or partially), the protect function can be used to peer-to-peer share the backup sw/fw.

- Tree



- Ring (open or closed)



Protect route may or may not require communication with core depending on how implemented.

🚫 = failed function

○ = backup function (potential locations if more than one shown)