OPEN NETWORKING FOUNDATION

Core Information Model (CoreModel)

TR-512.5

Resilience (Protection, Restoration and Recovery)

Version 1.2
September 20, 2016

OpenFlow

ONF Document Type: Technical Recommendation
ONF Document Name: Core Information Model version 1.2

Disclaimer

Open Networking Foundation
2275 E. Bayshore Road, Suite 103, Palo Alto, CA 94303
www.opennetworking.org

Table of Contents

List of Figures

Document History

Version	Date	Description of Change
1.0	March 30, 2015	Initial version of the base document of the "Core Information Model" fragment of the ONF Common Information Model (ONF-CIM).
1.1	November 24, 2015	Version 1.1
1.2	September 20, 2016	Version 1.2 [Note Version 1.1 was a single document whereas 1.2 is broken into a number of separate parts]

1 Introduction

This document is an addendum to the TR-512_v1.2 ONF Core Information Model and forms part of the description of the ONF-CIM. For general overview material and references to the other parts refer to TR-512.1 ONF Core IM - Overview.

1.1 References

For a full list of references see TR-512.1.

1.2 Definitions

For a full list of definition see TR-512.1.

1.3 Conventions

See TR-512.1 for an explanation of:

- UML conventions
- Lifecycle Stereotypes
- Diagram symbol set

1.4 Viewing UML diagrams

Some of the UML diagrams are very dense. To view them either zoom (sometimes to 400%), open the associated image file (and zoom appropriately) or open the corresponding UML diagram via Papyrus (for each figure with a UML diagram the UML model diagram name is provided under the figure or within the figure).

1.5 Understanding the figures

Figures showing fragments of the model using standard UML symbols as well as figures illustrating application of the model are provided throughout this document. Many of the application-oriented figures also provide UML class diagrams for the corresponding model fragments (see TR-512.1 for diagram symbol sets). All UML diagrams depict a subset of the relationships between the classes, such as inheritance (i.e. specialization), association relationships (such as aggregation and composition), and conditional features or capabilities. Some UML diagrams also show further details of the individual classes, such as their attributes and the data types used by the attributes.

2 Introduction to the Resilience Model

The focus of this document is the modeling of resilience in the ONF-CIM.

This document:

- Introduces the resilience model structure
- Describes the key classes of the resilience model

- Explains the attributes of the resilience model
- Shows how the model deals with various resilience schemes
- Explains how the specification model describes resilience schemes (protection etc.)
- Highlights work in progress to further advance the resilience model

The resilience model builds on aspects of the Core Network Model related to Termination and Forwarding described in TR-512.2 ONF Core IM - Forwarding and Termination and related to Topology TR-512.4 ONF Core IM - Topology. Resilience capability and other specification considerations are covered in TR-512.7 ONF Core IM - Specification.

A data dictionary that sets out the details of all classes, data types and attributes is also provided (TR-512.8).

3 Resilience model detail

3.1 Resilience Pattern

The resilience model unifies a number of apparently different traditional model approaches that are used for various different resilience schemes (see [ITU-T 808.1]). The resilience model focus is the FcSwitch which represents the forwarding selector and which enables changes of forwarding to achieve resilience. The model also represents the control element of the resilience control loop that monitors behavior, assesses that behavior identifying necessary configuration changes and applies those configuration changes to make the necessary adjustments to Forwarding so as to achieve the intended resilience.

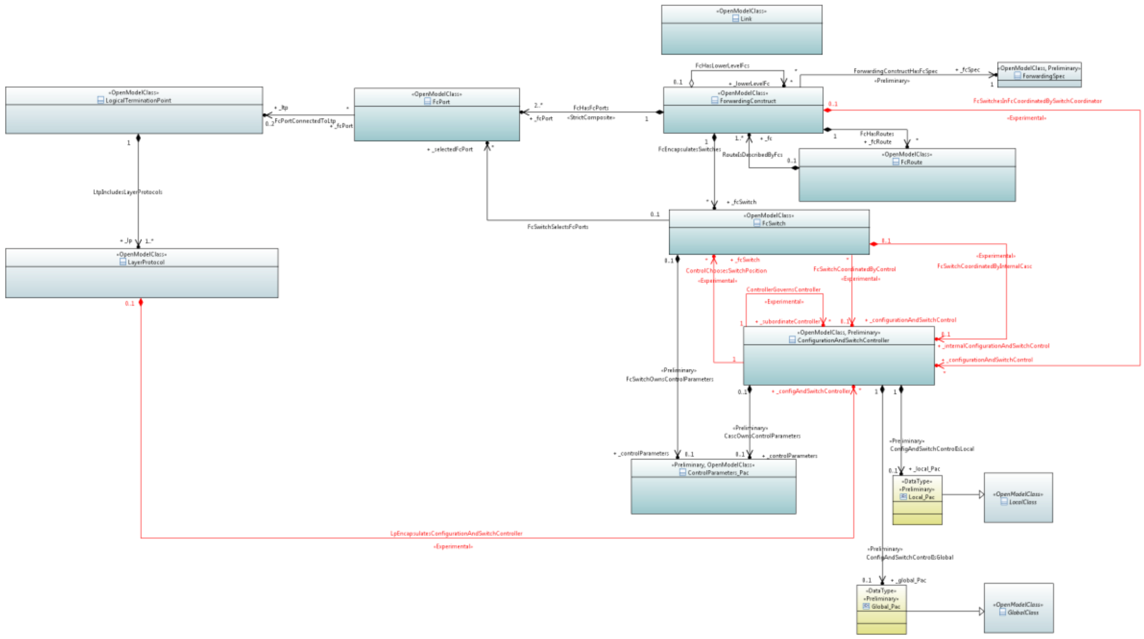
Some resilience schemes require combinations of control elements and switches. A particular pattern of combination of control elements and switches along with forwarding of control messages fully describe each scheme. This single uniform approach replaces the various traditional approaches (e.g. in some traditional representations a protection group is used, the protection group is replaced by one or more control elements in the new model).

3.2 Resilience Model

The figure below shows the key classes involved in protection and the associations between them. The majority of this model was present in the previous release. Key updates are:

- Upgrading of parts of the model to preliminary
- Addition of protection attributes discussed later
- Addition of mechanism for naming of key parts (shown in yellow below)

Note that the Link is shown solely because it gains parameters related to resilience that are applicable when a link is forming part of a route.



CoreModel diagram: Resilience-Pattern

Figure 3-1 Basic resilience pattern

The key classes present in the model that specifically support resilience are described in the following sections. The naming/identification classes, Local_Pac and Global_Pac, are discussed in section 3.5 – Naming the ConfigurationAndSwitchController on page 17. The FcSpec class is included as it will be used to express the structure of the resilience scheme of the ForwardingConstructs, this is described in detail in TR-512.7 ONF Core IM - Specification. See also TR-512.2 ONF Core IM - Forwarding and Termination for an explanation of some key classes in the figure.

The associations shown in red are experimental.

3.2.1 ConfigurationAndSwitchController

Qualified Name: CoreModel::CoreNetworkModel::ObjectClasses::Resilience::ConfigurationAndSwitchController

Represents the capability to control and coordinate switches, to add/delete/modify FCs and to add/delete/modify LTPs/LPs so as to realize a protection scheme.

This class is Preliminary.

3.2.2 ControlParameters_Pac

Qualified Name: CoreModel::CoreNetworkModel::ObjectClasses::Resilience::ControlParameters_Pac

A list of control parameters to apply to a switch.

This class is Preliminary.

3.2.3 FcRoute

Qualified Name: CoreModel::CoreNetworkModel::ObjectClasses::Resilience::FcRoute

Each instance of an FC Route (FcRoute) class models an individual route of an FC. The route is an alternative view of the internal structure of the FC to FC aggregation (see FcHasLowerLeverFcs association). There are cases where a route is the most appropriate representation and cases where the aggregation approach is the most appropriate representation. The route of an FC object is represented by a list of FCs at a lower level with the implicit understanding that unmodelled link connections are interleaved between the lower level FCs. Note that depending on the service supported by an FC, the FC can have multiple routes. The FcRoute is also applicable where an NE level ForwardingDomain may be decomposed into subordinate ForwardingDomains. Applies to both virtual and real NE cases.

Inherits properties from:

- LocalClass

3.2.4 FcSwitch

Qualified Name: CoreModel::CoreNetworkModel::ObjectClasses::Resilience::FcSwitch

The FcSwitch class models the switched forwarding of traffic (traffic flow) between FcPorts and is present where there is protection functionality in the FC. If an FC exposes protection (having two or more FcPorts that provide alternative identical inputs/outputs), the FC will have one or more associated FcSwitch objects to represent the alternative flow choices visible at the edge of the FC. The FC switch represents and defines a protection switch structure encapsulated in the FC. Essentially performs one of the functions of the Protection Group in a traditional model. It associates to 2 or more FcPorts each playing the role of a Protection Unit. One or more protection, i.e. standby/backup, FcPorts provide protection for one or more working (i.e. regular/main/preferred) FcPorts where either protection or working can feed one or more protected FcPort. The switch may be used in revertive or non-revertive (symmetric) mode. When in revertive mode it may define a waitToRestore time. It may be used in one of several modes including source switch, destination switched, source and destination switched etc (covering cases such as 1+1 and 1:1). It may be locked out (prevented from switching), force switched or manual switched. It will indicate switch state and change of state. The switch can be switched away from all sources such that it becomes open and hence two coordinated switches can both feed the same LTP so long as at least one of the two is switched away from all sources (is "open"). The ability for a Switch to be "high impedance" allows bidirectional ForwardingConstructs to be overlaid on the same bidirectional LTP where the appropriate control is enabled to prevent signal conflict. This ability allows multiple alternate routes to be present that otherwise would be in conflict.

Inherits properties from:

- LocalClass

3.3 Further discussion

3.3.1 Embedding the ConfigurationAndSwitchController

The controller of switching that may be embedded in the FC or, as discussed, may be external to the FC. This allows it to coordinate switching of several FCs (as per traditional protection group) and further can cause the creation/deletion of FC and hence is part of the continuum of management-control. This model fragment offers flexibility in the way the FcSwitch gains its ControlParameters and provides an instantiable ConfigurationAndSwitchController that can be positioned with an appropriate scope of control for any particular case. The ConfigurationAndSwitchController can provide the control parameter to the FcSwitch or the FcSwitch can reference a profile (also available to the ConfigurationAndSwitchController). The ConfigurationAndSwitchController can be contained in the FcSwitch, can be contained in the FC and referenced by the FcSwitch or can be contained in some ConfigurationGroup entity with a scope greater than the FC and the FcSwitch.

3.3.2 An Open FcSwitch

The figure below (see section 4 Explanatory figures on page 19 for an explanation of the figure symbol set) shows an example of multiple open switches showing both legal and illegal settings.

The figure assumes a circuit switched technology and shows four cases of an NE with a protected signal flow to one client LTP (green) supported by an LTP (purple) bound to a physical port (on the left of each diagram). The cases highlighted are the two normal states of switches in the upper two diagrams, a transient state in lower left and an illegal state in lower right where the Configuration and Switch Controller (C&SC) has failed.

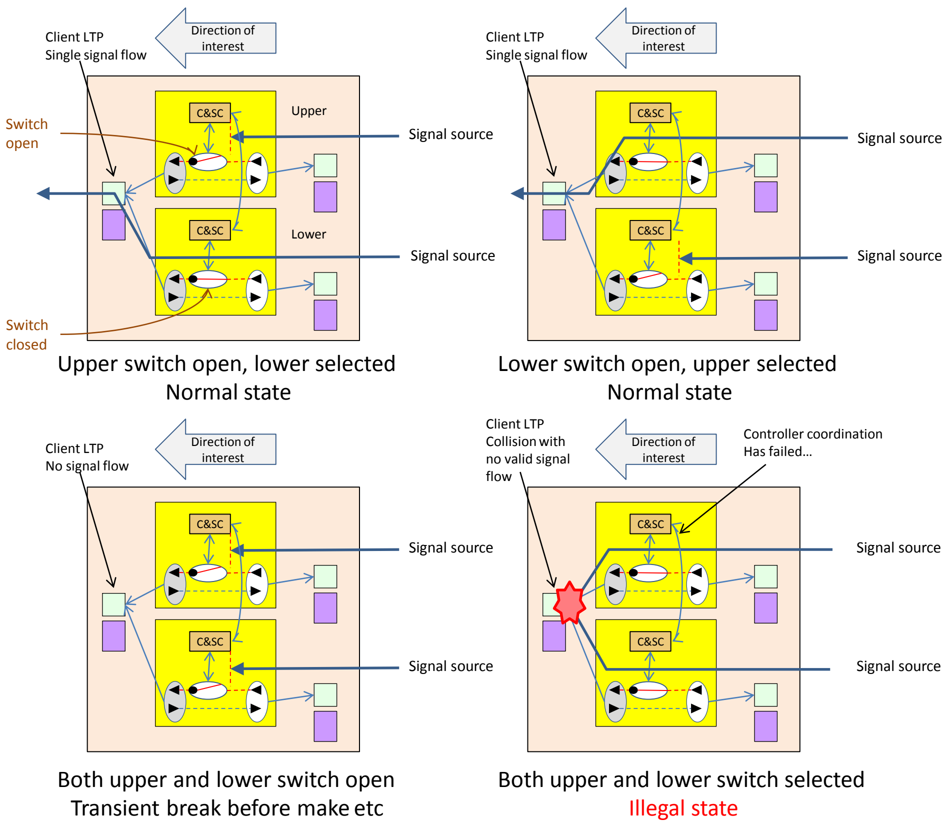


Figure 3-2 Multiple open switch case with one client LTP

3.3.3 Sharing FcPorts and switch orientation convention

The diagrams in the figure below (in dotted red ellipses) illustrate usage of a mix of output and input switches (designated by "o" and "i" respectively). The modelling orientation convention is that the switch common is on the sharing FcPort if there is only one sharing FcPort (hence in some cases mixed ingress/egress switches are used). If there are two sharing FcPorts, or no sharing FcPorts the convention is that the input switch (default) is used unless there is specific complexity that can only be resolved with output switches.

See also Figure 4-9 Showing detail of a single ended view of 1+1 and 1:1 switches in a route context on page 26 and Figure 4-11 Nodal controller peering in a route context on page 28 for more details on the specific case of use.

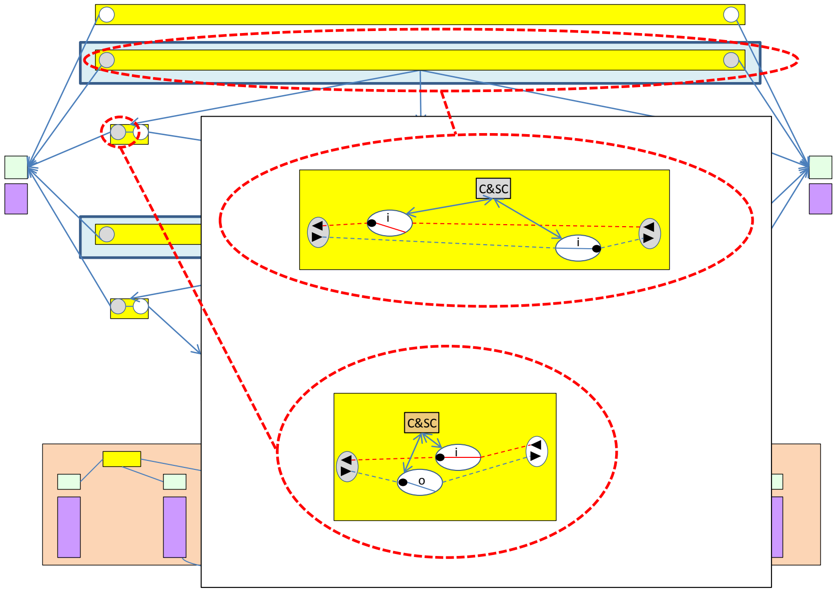
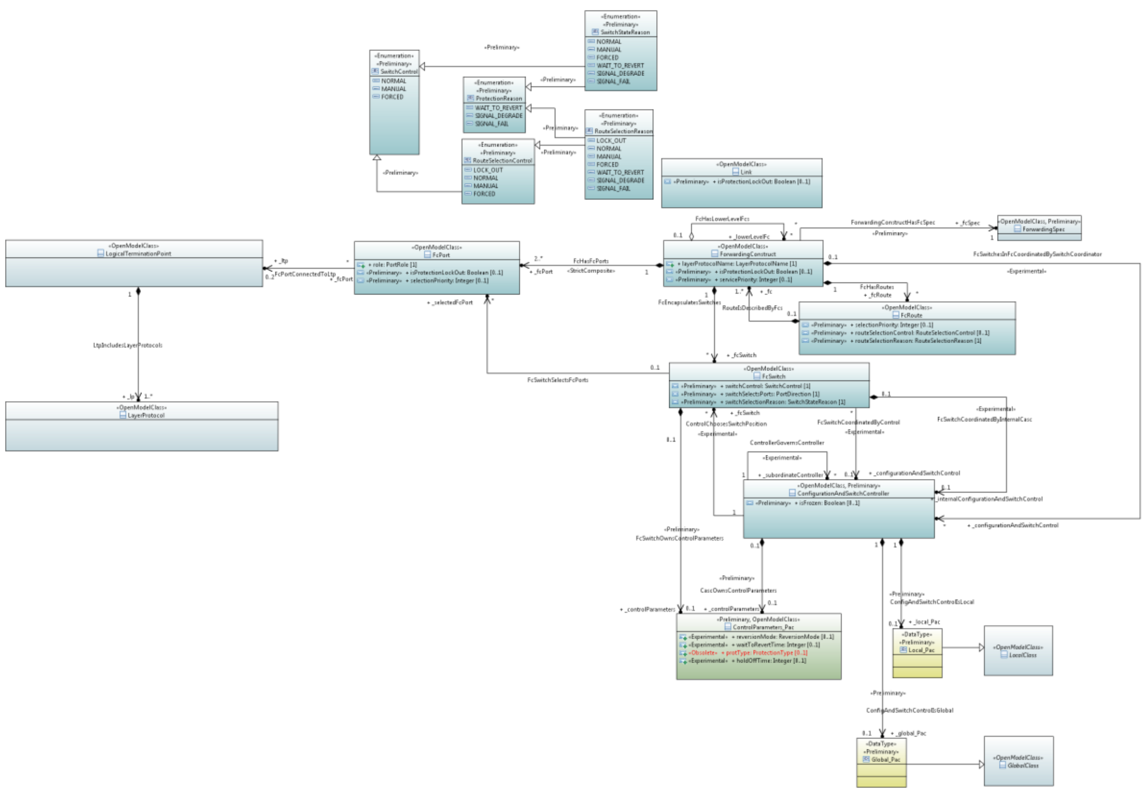


Figure 3-3 Sharing FcPorts and switch orientation convention

3.4 Resilience Attributes

The figure below highlights the key attributes of the resilience model.



CoreModel diagram: Resilience-KeyAttributes

Figure 3-4 Key resilience attributes

3.4.1 ConfigurationAndSwitchController

Table 1: Attributes for ConfigurationAndSwitchController

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
switchRule	Experimental	A sketch of the presence of complex rules governing the switch behavior.
isFrozen	Preliminary	Temporarily prevents any switch action to be taken and, as such, freezes the current state. Until the freeze is cleared, additional near-end external commands are rejected and fault condition changes and received APS messages are ignored. All administrative controls of any aspect of protection are rejected.
isCoordinatedSwitchingBothEnds	Experimental	The C&SC is operating such that switching at both ends of each flow acorss the FC is coordinated at both ingress and egress ends.
_fcSwitch	Experimental	The switch being controlled.
_controlParameters	Preliminary	The control parameters to be applied if local parameters are used rather than profiles
_profileProxy	Experimental	Applied profiles.
_local_Pac	Preliminary	See referenced class
_global_Pac	Preliminary	See referenced class
_subordinateController	Experimental	A C&SC that is fully or partially subordinate this C&SC. A peer is considered as partially subordinate in that the peer will respond to requests for action from this C&SC but will also make requests for action to be carried out by this C&SC. Where there is a peer relationship each controller in the peering will see the other controller as subordinate.
_cascSpec	Experimental	See referenced class

3.4.2 ControlParameters_Pac

Table 2: Attributes for ControlParameters_Pac

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
reversionMode	Experimental	Indcates whether the protection scheme is revertive or non-revertive.
waitToRevertTime	Experimental	If the protection system is revertive, this attribute specifies the time, in minutes, to wait after a fault clears on a higher priority (preferred) resource before reverting to the preferred resource.
protType	Obsolete	Indicates the protection scheme that is used for the ProtectionGroup.

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
holdOffTime	Experimental	This attribute indicates the time, in milliseconds, between declaration of signal degrade or signal fail, and the initialization of the protection switching algorithm.
_networkSchemeSpecification	Experimental	See referenced class

3.4.3 FcPort

Table 3: Attributes for FcPort

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
role		Each FcPort of the FC has a role (e.g., working, protection, protected, symmetric, hub, spoke, leaf, root) in the context of the FC with respect to the FC function.
fcPortDirection		The orientation of defined flow at the FcPort.
isProtectionLockOut	Preliminary	The resource is configured to temporarily not be available for use in the protection scheme(s) it is part of. This overrides all other protection control states including forced. If the item is locked out then it cannot be used under any circumstances. Note: Only relevant when part of a protection scheme.
selectionPriority	Preliminary	The preference priority of the resource in the protection scheme for a particular FC. The lower the value the higher the priority. A lower value of selection priority is preferred If two resources have the same value they are of equal priory. There is no preference between equal priorities. If a resource with the lowest value selection priority fails then the next lowest value available (may be the same value) is picked. Hence on failure of the current resource the next best available will be selected. If there are several equal values the choice is essentially arbitrary). If the scheme is revertive then when a resource of higher priority than the currently selected resource recovers it will be selected. This is equivalent to working/protection but allows for all static scheme types with n:m capability. In simple schemes 0 = working and 1 = protecting.
isInternalPort	Experimental	The FcPort is not exposed and cannot have associated LTPs. This form of FcPort is used to enable chaining of FcSwitches or FcRoutes in complex network protection scenarios.
_ltp		The FcPort may be associated with more than one LTP when the FcPort is bidirectional and the LTPs are unidirectional. Multiple Ltp - Bidirectional FcPort to two Uni Ltps Zero Ltp - BreakBeforeMake transition - Planned Ltp not yet in place - Off-network LTP referenced through other mechanism
_fcRouteFeedsFcPortEgress	Experimental	Identifies which route(s) currently actively forward to the FcPort to exit the FC to an LTP (or for an internal FcPort to propagate to the next internal switch/route).

3.4.4 FcRoute

Table 4: Attributes for FcRoute

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
selectionPriority	Preliminary	The preference priority of the resource in the protection scheme for a particular FC. The lower the value the higher the priority. A lower value of selection priority is preferred If two resources have the same value they are of equal priory. There is no preference between equal priorities. If a resource with the lowest value selection priority fails then the next lowest value available (may be the same value) is picked. Hence on failure of the current resource the next best available will be selected. If there are several equal values the choice is essentially arbitrary). If the scheme is revertive then when a resource of higher priority than the currently selected resource recovers it will be selected. This is equivalent to working/protection but allows for all static scheme types with n:m capability. In simple schemes 0 = working and 1 = protecting.
routeSelectionControl	Preliminary	Degree of administrative control applied to the route selection.
routeSelectionReason	Preliminary	The reason for the current route selection.
_fc		The list of FCs describing the route of an FC. In most cases the FcRoute has 2 or more FCs however there are some cases where a Route with one FC is valid.

3.4.5 FcSwitch

Table 5: Attributes for FcSwitch

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
holdOffTime	Obsolete	Moved to ControlParameter_Pac... This attribute indicates the time, in seconds, between declaration of unacceptable quality of signal on the currently selected FcPort, and the initialization of the protection switching algorithm.
protType	Obsolete	Indicates the protection scheme that is used for the ProtectionGroup.
reversionMode	Obsolete	Moved to ControlParameter_Pac... This attribute whether or not the protection scheme is revertive or non-revertive.
switchControl	Preliminary	Degree of administrative control applied to the switch selection.
switchSelectsPorts	Preliminary	Indicates whether the switch selects from ingress to the FC or to egress of the FC, or both.
switchSelectionReason	Preliminary	The reason for the current switch selection.
waitToRestoreTime	Obsolete	Moved to ControlParameter_Pac and changed to waitToRevert... If the protection system is revertive, this attribute specifies the amount of time, in seconds, to wait after the preferred FcPort returns to an acceptable state of

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
		operation (e.g. a fault has cleared) before restoring traffic to that preferred FcPort.
_selectedFcPort		Indicates which points are selected by the switch. Depending on the switch spec (via Fcspec) - more than one FcPort can be selected at any one time (e.g. egress switch, ingress packet switch) - zero FcPorts can be selected. For an ingress switch this indicates that the switch common (egress) is "high impedance" .
_profileProxy	Experimental	Provides a set of predefined values for switch control in place of the direct values available via the FcSwitch or via _configurationAndSwitchControl.
_configurationAndSwitchControl	Experimental	A ConfigurationAndSwitchController encapsulated in the FcSwitch that controls the FcSwitch alone.
_internalConfigurationAndSwitchControl	Experimental	A switch controller encapsulated in the FcSwitch.
_controlParameters		See referenced class

3.4.6 ForwardingConstruct

Table 6: Attributes for ForwardingConstruct

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
layerProtocolName		The layerProtocol at which the FC enables the potential for forwarding.
forwardingDirection		The directionality of the ForwardingConstruct. Is applicable to simple ForwardingConstructs where all FcPorts are BIDIRECTIONAL (the ForwardingConstruct will be BIDIRECTIONAL) or UNIDIRECTIONAL (the ForwardingConstruct will be UNIDIRECTIONAL). Is not present in more complex cases.
isProtectionLockOut	Preliminary	The resource is configured to temporarily not be available for use in the protection scheme(s) it is part of. This overrides all other protection control states including forced. If the item is locked out then it cannot be used under any circumstances. Note: Only relevant when part of a protection scheme.
servicePriority	Preliminary	Relevant where "service" FCs are competing for server resources. Used to determine which signal FC is allocated resource. The priority of the "service" with respect to other "services". Lower numeric value means higher priority. Covers cases such as preemptable.
_lowerLevelFc		An FC object supports a recursive aggregation relationship such that the internal construction of an FC can be exposed as multiple lower level FC objects (partitioning). Aggregation is used as for the FD to allow changes in hierarchy. FC aggregation reflects FD aggregation. The FC represents a Cross-Connection in an NE. The Cross-Connection in an NE is not necessarily the lowest level of FC partitioning.

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
_fcRoute		An FC object can have zero or more routes, each of which is defined as a list of lower level FC objects describing the flow across the network.
_fcPort		The association of the FC to LTPs is made via FcPorts (essentially the ports of the FC).
_fcSwitch		If an FC exposes protection (having two FcPorts that provide alternative identical inputs/outputs), the FC will have one or more associated FcSwitch objects. The arrangement of switches for a particular instance is described by a referenced FcSpec
_configurationAndSwitchControl	Experimental	Reference to a ConfigurationAndSwitchController that coordinates switches encapsulated in the FC. The controller coordinates multiple switches in the same FC.
_fcSpec	Preliminary	References the specification that describes the capability and internal structure of the FC (e.g. The arrangement of switches for a particular instance is described by a referenced FcSpec). The specification allows interpretation of FcPort role and switch configurations etc.
_supportedLink	Experimental	An FC that spans between LTPs that terminate the LayerProtocol usually supports one or more links in the client layer.

3.5 Foldaway of complexity – Naming the ConfigurationAndSwitchController

Where there is one switch controller in a context (e.g. a switch or an FC itself) and where the controller relates to the context entity by composition it is reasonable to fold the controller into the context entity.

- The context entity gains the controller attributes
- Any reference to the controller becomes a reference to the context entity

Where there are several switch controllers in a context but where those controllers do NOT need to be referenced in any way from outside the context entity it is reasonable to fold the controllers into a data structure within the context entity

- The context entity gains a structure of multiple controller attribute blocks
- The controller "instance" is resolved by position in the structure
- It is NOT POSSIBLE to reference the controller from outside the context entity

Where there are several switch controllers in a context and/or where those controllers need to be referenced from outside the context it is not possible to fold the controllers into the context entities but the entities representing the controllers can have a relative identification (localId) within the scope of the identifier for the context

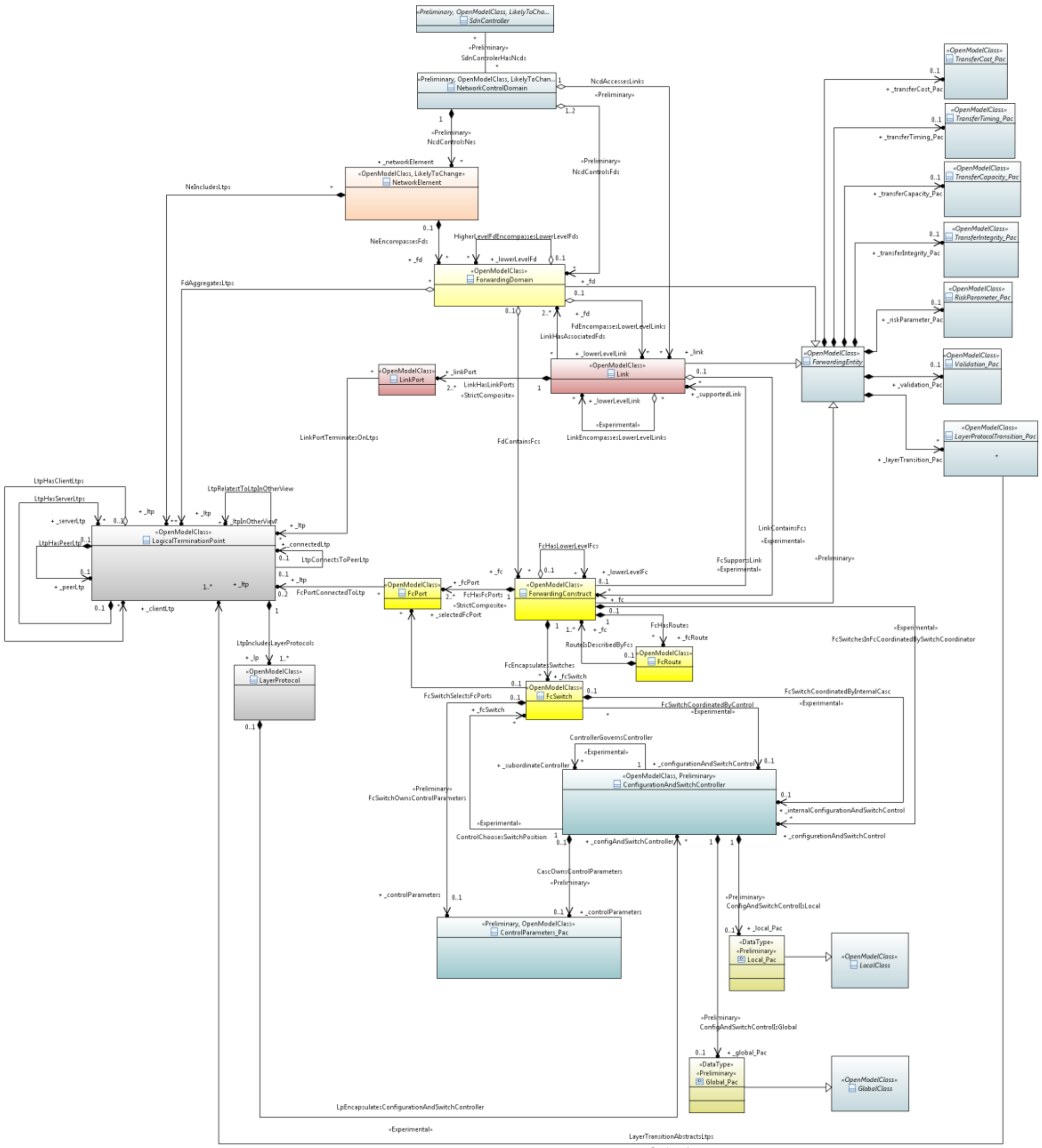
- References are via an address with contextId and localId as elements

Where the switch controller is not in any stable context then it must have a UUID and can be directly referenced via that UUID. The ConfigurationAndSwitchController can be:

- Embedded in an FcSwitch, a local class, essentially as a _PAC with no need for ids etc.
- Embedded in an FC, a global class, essentially as a local class with need for only relative ids etc.
- Stand alone as a global class with need for a UUID

Hence it is necessary to use a mechanism that allows the class to have a variable id strategy. This is achieved using conditional composition rather than inheritance (this approach has only been applied here but may be relevant for other cases in the model).

3.6 Overview of model with resilience



CoreModel diagram: Resilience-FullSkeleton

Figure 3-5 Classes of model including topology and resilience

4 Explanatory figures

This section provides figures that illustrate the application of the model to represent resilience schemes. The section builds up from simple protection schemes to complex network resilience schemes.

For an explanation of the symbol set being used in the figures see section 1.3 Conventions on page 6 and below.

4.1 Key to diagrams

The following diagram highlights the symbols used for various classes in the resilience model.

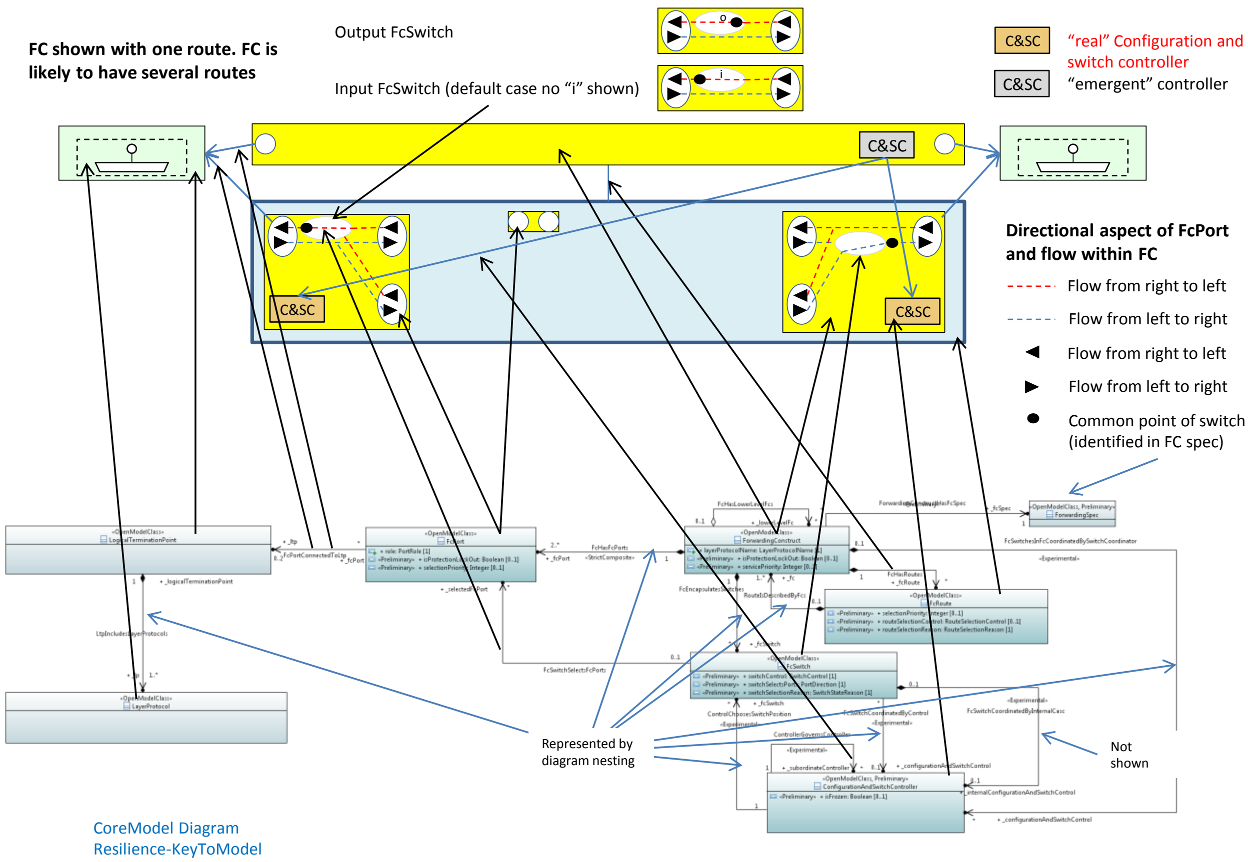


Figure 4-1 Instance diagram key

4.2 1?1 cases

This section deals with basic 1+1 and 1:1 cases and shows how they can be represented. The abbreviation 1?1 has been used where the description is common between both 1+1 and 1:1.

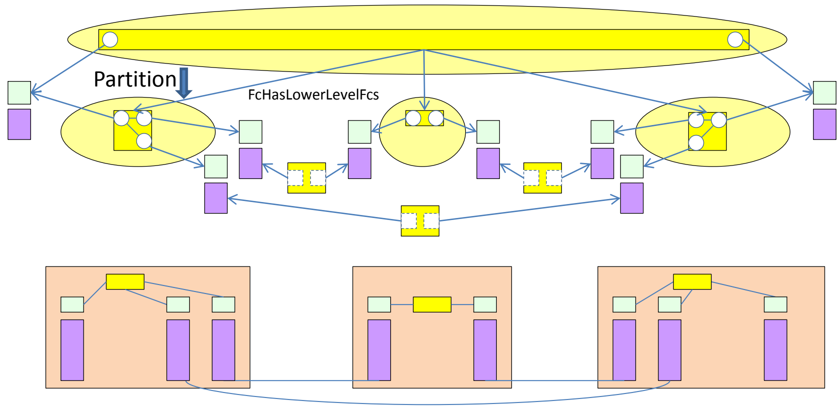


Figure 4-2 Simple summary example of 1?1 cases (represented via partition)

The figure above shows a simple summary example of a 1?1 case in a basic network with three NEs. Clearly this can be generalized further to be in a rule form. A specific solution can include zero or more NEs on either path[1]. The end-end FC is partitioned into subordinate (i.e. is an aggregation of the subordinate parts via FcHasLowerLevelFcs). The scheme may involve signalling.

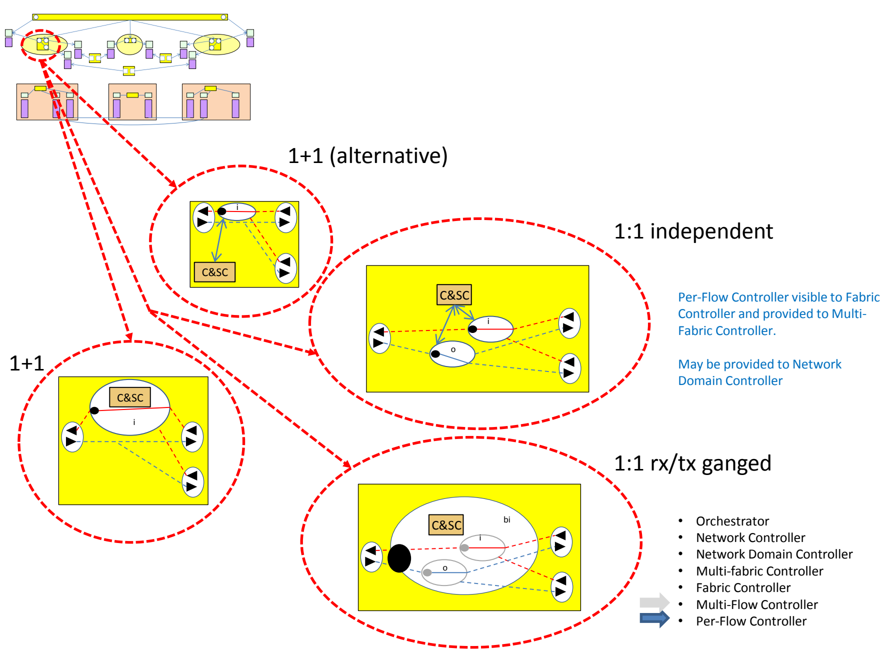


Figure 4-3 Showing detail of a single ended view of 1+1 and 1:1 switches

The figure above shows a nodal view and highlights ConfiguraionAndSwitchControllers (C&SC) encapsulated in the FcSwitch in some cases and in FC in others. The encapsulation chosen depends upon the scope of control of the C&SC. The encapsulation is via

[1] Note that there is work in progress to develop scheme specs that will provide a rule based view of the scheme.

FcSwitchCoordinatedByInternalControl when in the FcSwitch and FcSwitchesInFcCoordinatedBySwitchCoordinator when in the FC.

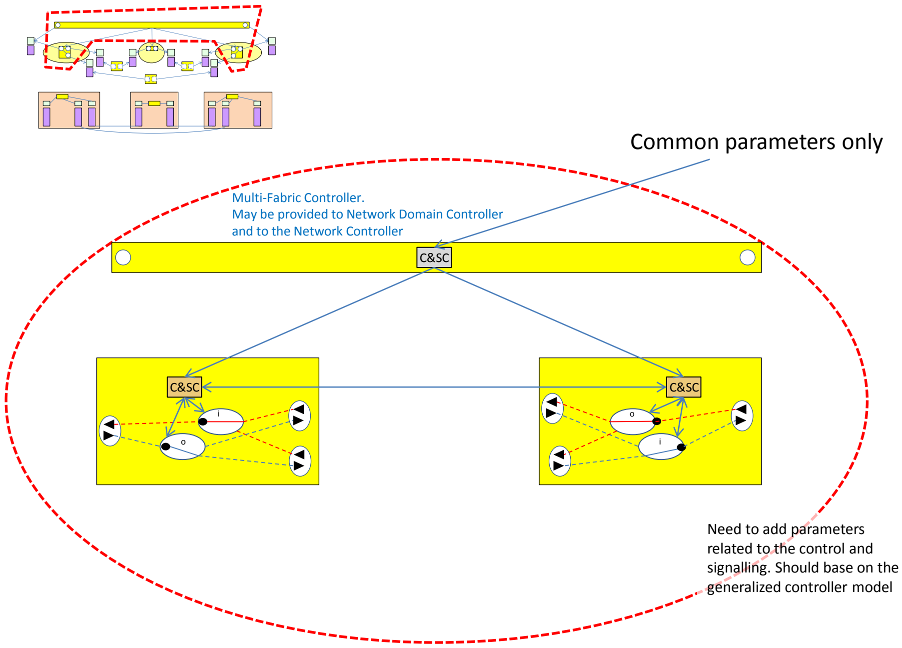


Figure 4-4 Showing an emergent abstract controller in a 1:1 case

The figure above shows a case of 1:1 independent switching (where the two directions of traffic are switched independently). The figure assumes that there is a distributed control solution (where the C&SCs in the FCs signal each other) and highlights an emergent C&SC which does not actually exist in the real control solution but which can be expressed to collect together parameters that should be set to the same value at both ends. In the network the coordination occurs through peer signaling. Above the network the SDN controller may realize the coordination[2].

[2] This recognition of levels of control from the most basis local two state switch controller through the various levels shown here and on two ring controllers and the SDN controller peer-hierarchy is a manifestation of and a validation of the concept of the Management-Control Continuum. Representation of the Management-Control Continuum will be further explored in the next release.

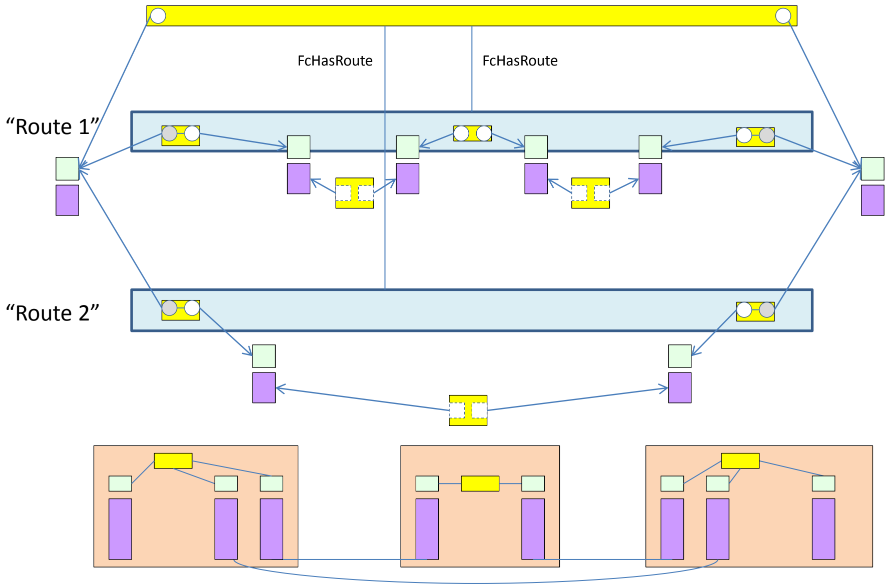


Figure 4-5 Showing a basic route based representation of the 1?1 protection scheme

The figure above shows an alternative representation of the 1?1 to that shown in Figure 4-2 Simple summary example of 1?1 cases on page 21. In the representation above two FcRoutes are used to represent the two alternative flows across the network. It should be noted that the FCs at the ends of each route are associated with the same LTPs and are only not conflicting because of the switches that they encapsulate (which when appropriately coordinated can ensure that only one FC is feeding the LTP at any time). The FcPort that can be switched off, i.e. be open, to ensure conflict can be avoided are depicted in grey (an output FcPort that can be switched off can share an LTP with another similar output FcPort and hence is called a sharing FcPort in this document). The FcSpec would identify the port via the switch configuration definition.

The figure below shows an alternative, slightly more verbose, representation of the 1?1 protection using two levels of route whether the top level routes have FCs that have the same span and the end-end FC[3]

The FCs of the route are contained in the route via the RouteIsDescribedByFc composition association and hence are not members of an FD. The FCs are used to represent flow and are defined in terms of the LTPs they reference in the context of the Route. The FD if visible would still have the FCs as shown in Figure 4-2 Simple summary example of 1?1 cases.

[3] This pattern of "decomposition" of the FC into two parallel FCs is also used when the FC is representing a Control Plane Call and when there is a need to combine two unidirectional FCs into a bidirectional FC. In these cases the decomposition takes place via the FcHasLowerLevelFcs association and the FCs are members of an FD via the FdContainsFc association.

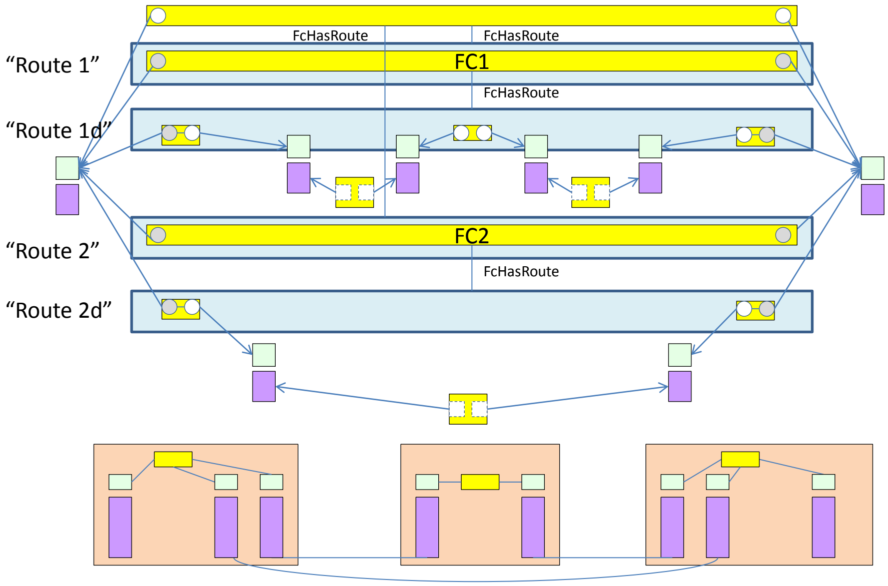


Figure 4-6 Showing a two level route based representation of the 1?1 protection scheme

The figure below shows the preferred route based representation which is a hybrid of the two where the FC of a route is described in terms of FCs via the FcHasLowerLevelFcs such that the lower level (nodal) FCs are in the context of FDs via the FdContainsFcs aggregation (a usual partition).

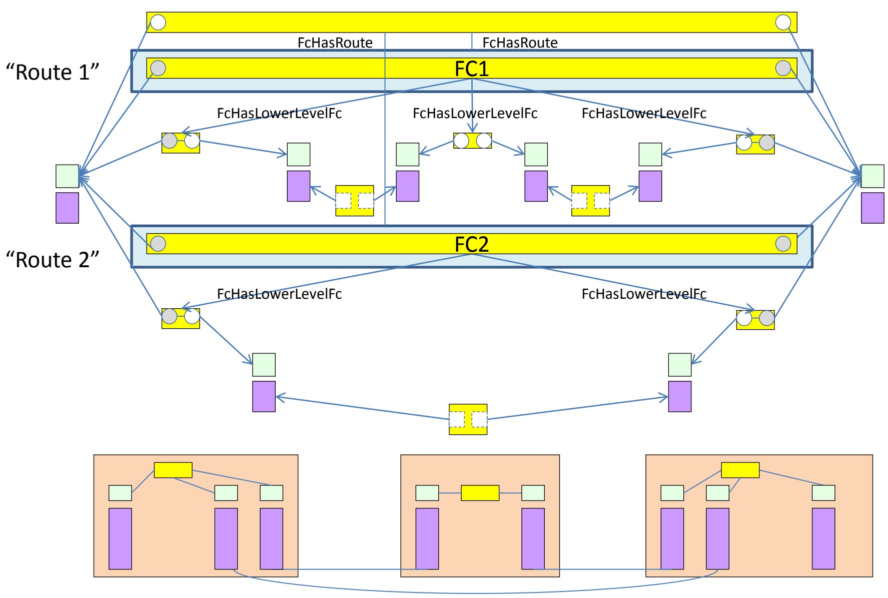


Figure 4-7 Showing the preferred route based representation of the 1?1 protection scheme

This approach for the 1?1 case, which may involve signalling, with a decomposition then partition is used in following diagrams.

The figure below shows the ConfigurationAndSwitchController (C&SC) positions and their associations (ControllerGovernsSubordinateController). The figure shows a number of potential emergent controllers as well as some real controllers assuming a distributed control scheme.

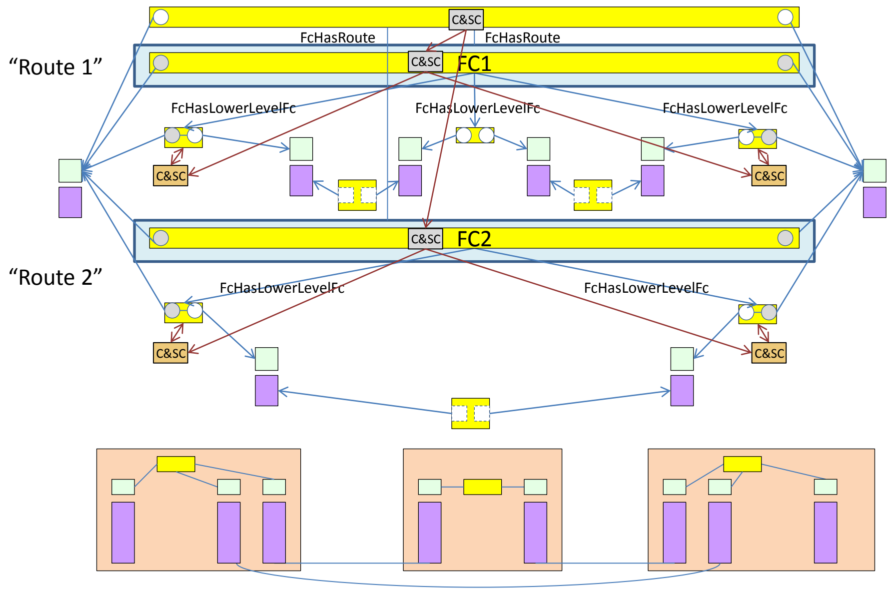


Figure 4-8 Route based representation of the 1?1 protection scheme showing C&SCs

The figure below shows a nodal view for one route and highlights ConfiguraionAndSwitchControllers (C&SC) encapsulated in the FcSwitch in some cases and in FC in others. The encapsulation chosen depends upon the scope of control of the C&SC. The encapsulation is via FcSwitchCoordinatedByInternalControl when in the FcSwitch and FcSwitchesInFcCoordinatedBySwitchCoordinator when in the FC.

Some of the diagrams in the figure below (in dotted red ellipses) use a mixture of output and input switches (designated by "o" and "i" respectively). The modelling orientation convention is covered in section 3.3.3 Sharing FcPorts on page 11.

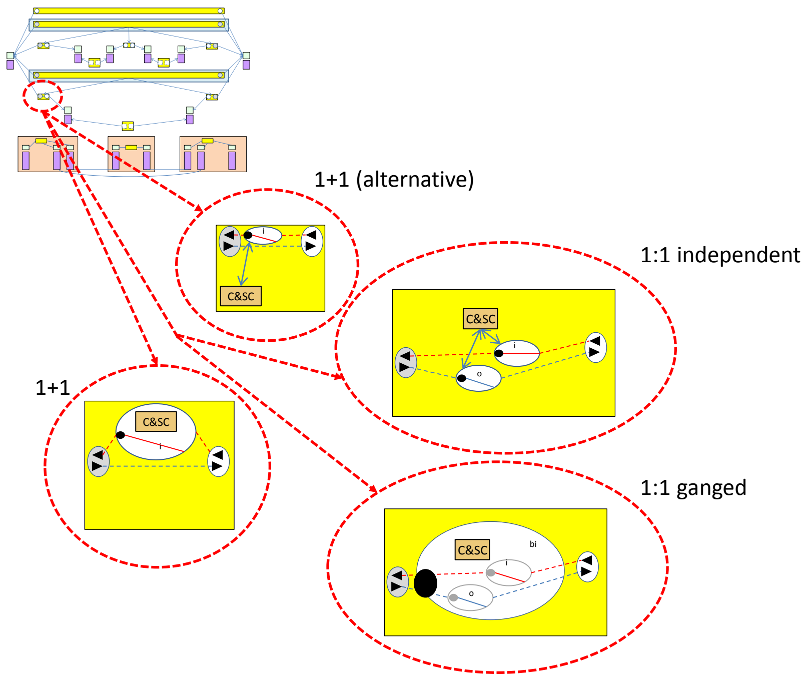


Figure 4-9 Showing detail of a single ended view of 1+1 and 1:1 switches in a route context

The figure below shows the interaction between the C&SCs of the FCs of the two routes. The interaction is via a balanced dual form of the ControllerGovernsController[4] association used to indicate a peer relationship. Setting values for one controller will affect the values in the peer.

Rules for the effect need to be stated in the spec. If aspects of the peering can be disabled this would lead to attributes to control those aspects.

[4] Note that this association is Experimental

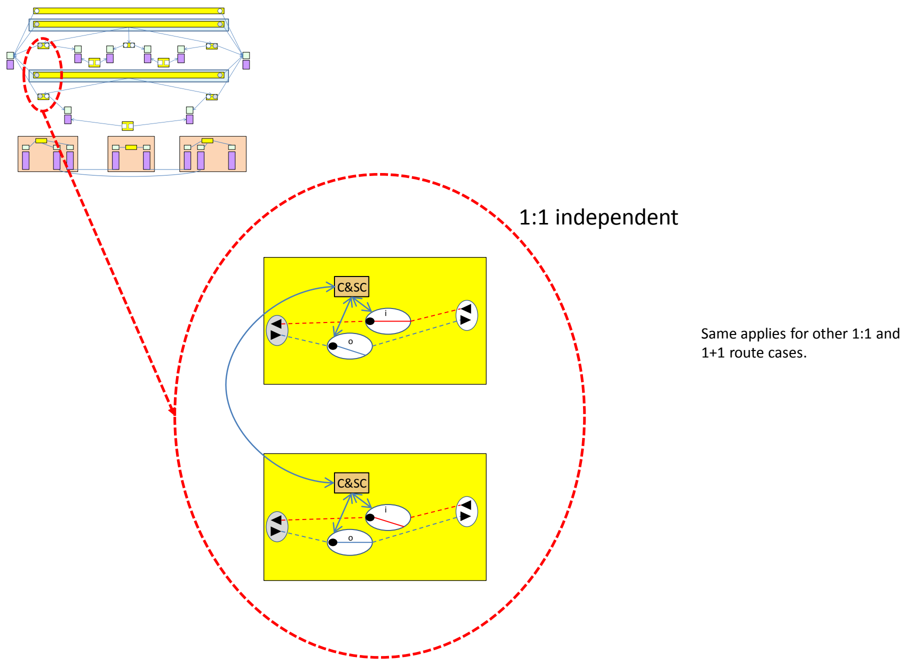


Figure 4-10 Single ended view of 1:1 switches in a route context with peer C&SC coordination

The figure below shows the controller peering between routes (emergent as the scheme is assumed to be a distributed control scheme) and also the emergent control in the end-end FC. It is proposed that the orientation convention is that input switch is preferred when ambiguous, see section 3.3.3 Sharing FcPorts on page 11 for further details.

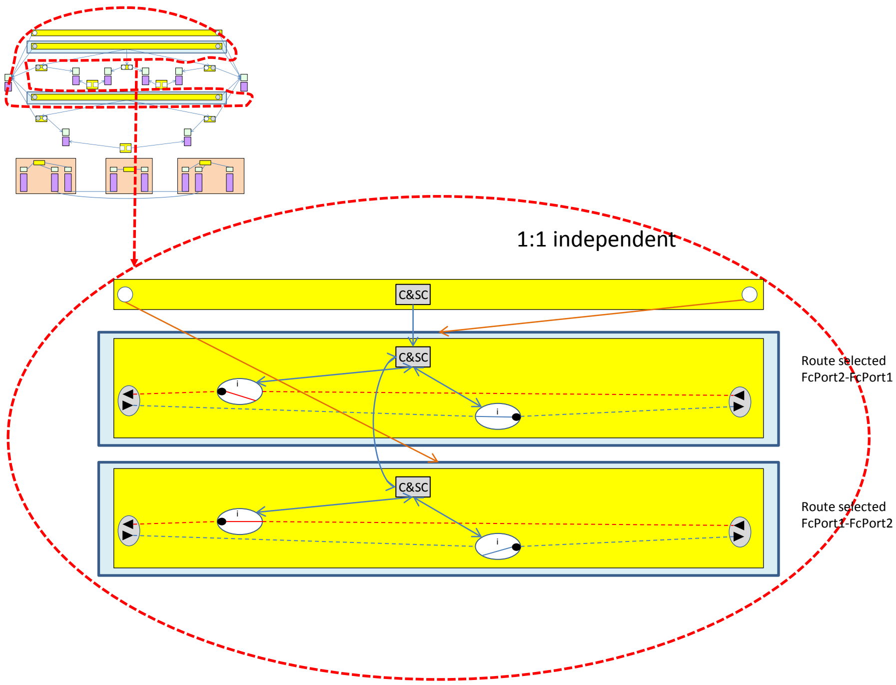


Figure 4-11 Nodal controller peering in a route context

4.3 1?1 open protection cases

The figures in this section are similar to those in the previous section.

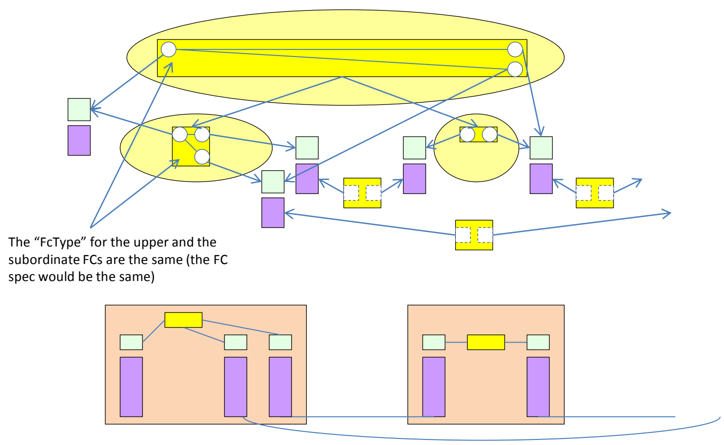


Figure 4-12 Simple summary example of open 1?1 cases

The figure above shows a simple summary example of an open 1?1 case (e.g. where only one end of the recovery scheme is within the scope of the SDN controller) in a basic network with three NEs. Clearly this can be generalized further to be in a rule form.

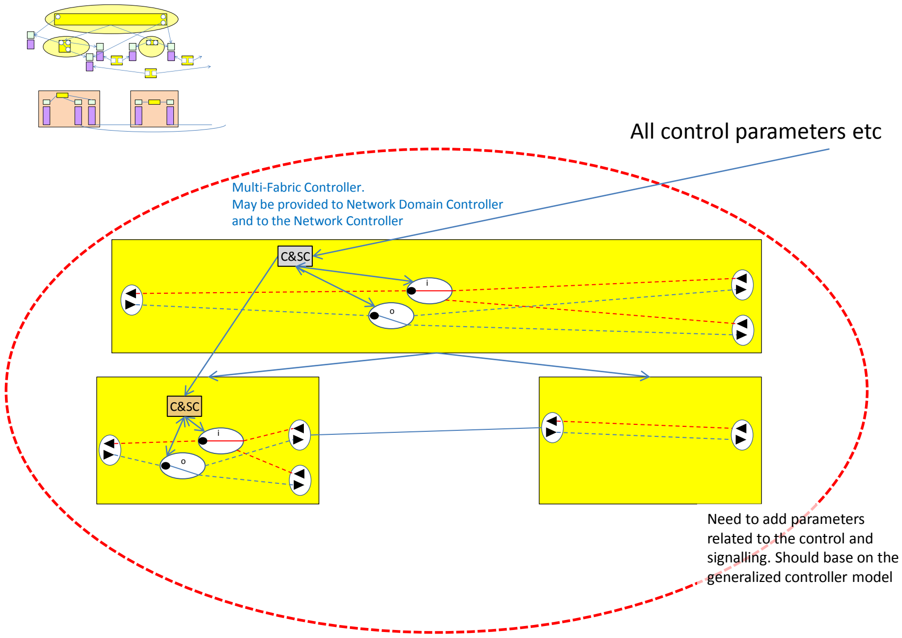


Figure 4-13 Showing an emergent abstract controller in an open 1?1

The figure above shows a case of 1:1 independent switching (where the two directions of traffic are switched independently). The figure assumes that there is a distributed control solution (where the C&SCs in the FCs signal each other) and highlights an emergent C&SC which does not actually exist in the real control solution but which can be expressed to collect together parameters that should be set to the same value at both ends. In the network the coordination occurs through peer signaling where the peer signaling is between C&SCs one of which is outside this view. Above the network the SDN controller may realize the coordination but to do this it will itself need to have communication with network peers (SDN controllers or other management-control entities) that control the off-network end(s) of the protection scheme.

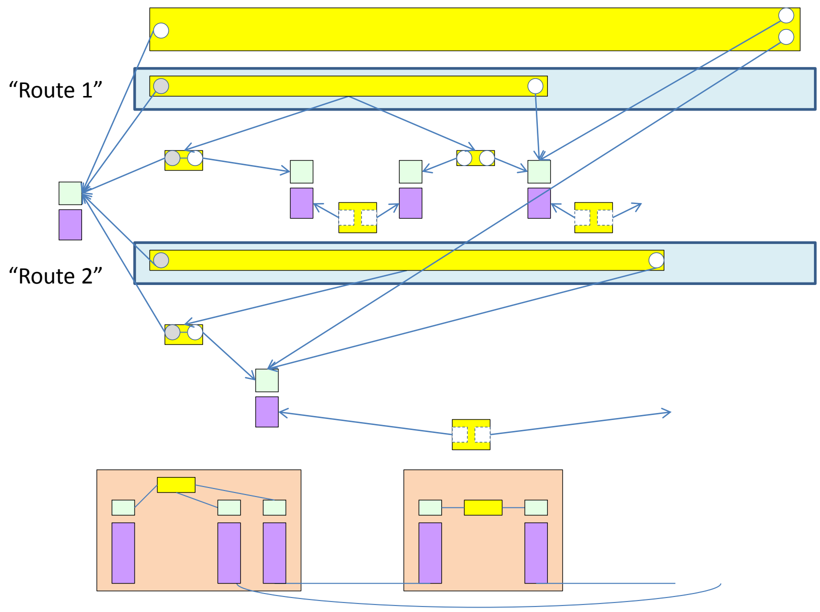


Figure 4-14 Simple summary example of open 1?1 cases showing route approach

The figure below shows the preferred route based representation which is a hybrid of the two where the FC of a route is described in terms of FCs via the FcHasLowerLevelFcs such that the lower level (nodal) FCs are in the context of FDs via the FdContainsFcs aggregation (a usual partition).

The figure below shows the interaction between the C&SCs of the FCs of the two routes. The interaction is the same as discussed earlier for Figure 4-10 Single ended view of 1:1 switches in a route context with peer C&SC coordination on page 27.

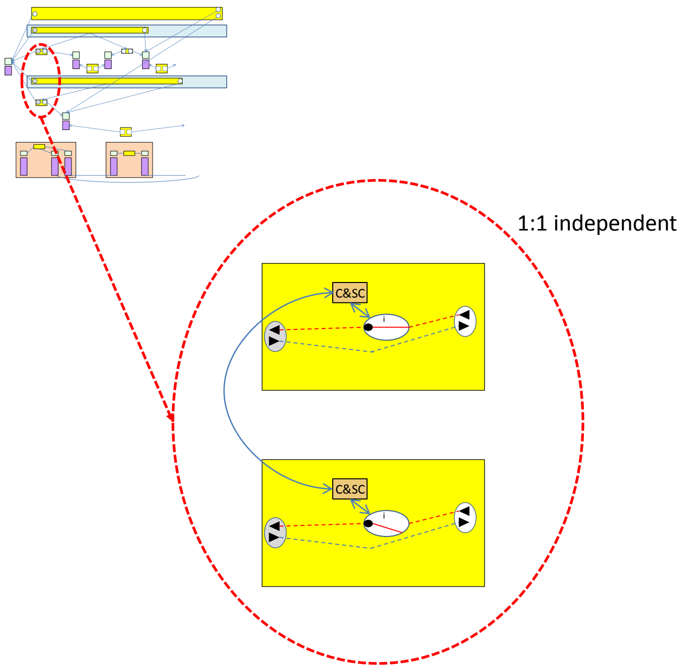


Figure 4-15 Single ended view of open 1:1 switches in a route context with peer C&SC coordination

The figure below shows the controller peering between routes (emergent as the scheme is assumed to be a distributed control scheme) and also the emergent control in the end-end FC.

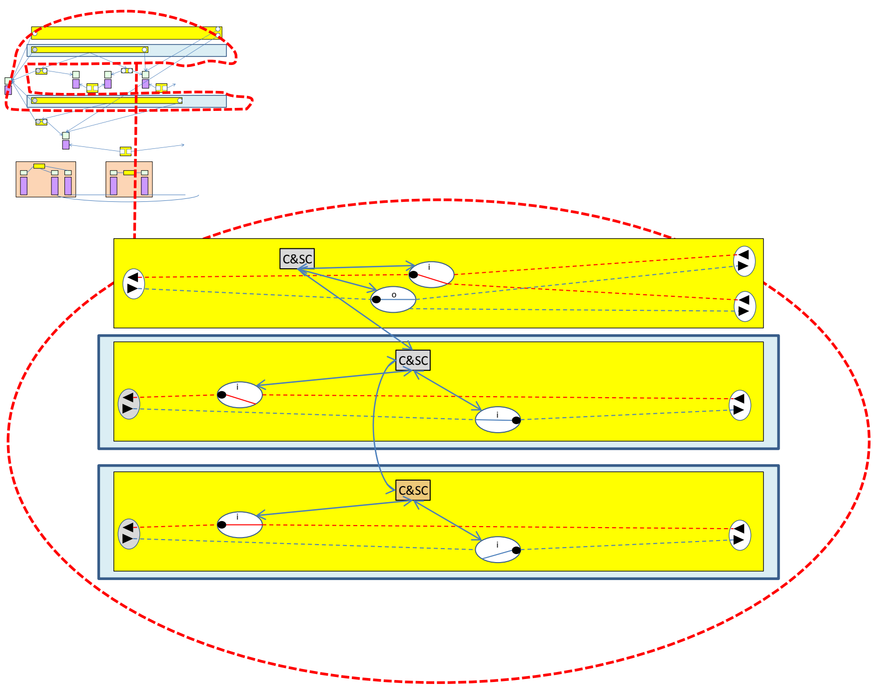


Figure 4-16 Nodal controller peering in a route context

4.4 1:N Cases

This section deals with basic 1:N cases and shows how they can be represented.

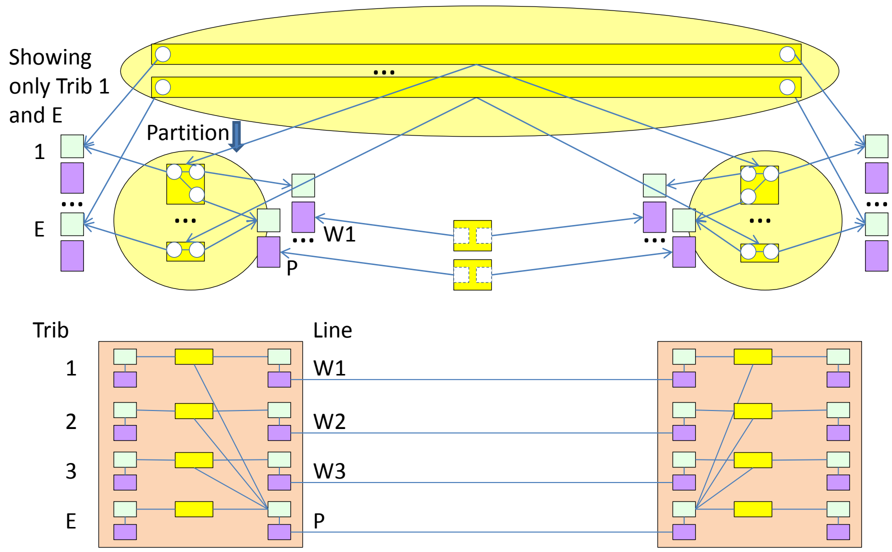


Figure 4-17 Simple summary example of 1:N cases (represented via partition)

The figure above shows a simple summary example of a 1:N case in a basic network. As shown in the detailed NE view at the bottom of the figure, the scheme provides protection to three traffic signals (1, 2 and 3) and also provides a lower grade path for "Extra Traffic" (E). The traffic signals 1, 2 and 3 normally each use a dedicated "Worker"[5] paths (W1-W3 (numbered to match the traffic signal numbers)). The Protection path (P) provides an alternative for any one of the Workers. The "Extra Traffic", E, uses the protection path, P, when it is not needed to protect any of W1-W3. Clearly this can be generalized further to be in a rule form. A specific solution can include one more traffic paths.

The FC view shows only one of the traffic paths (1) and the "Extra Traffic" (E). The end-end FC representing the traffic path is partitioned into subordinate (i.e. is an aggregation of the subordinate parts via FcHasLowerLevelFcs) as is the "Extra Traffic" path (E). It should be noted that the nodal FC from E to P and the FC from 1 to W1 and P use the same LTP at P. The apparent conflict is resolved by the C&SC (not shown). The scheme will involve signalling.

[5] The term "Worker" means normal path for particular traffic

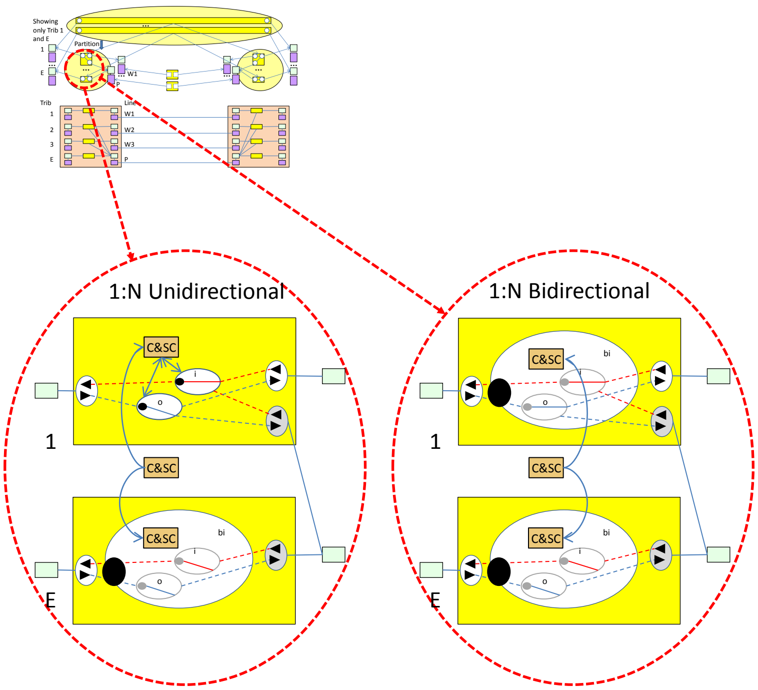


Figure 4-18 Showing detail of a single ended view of 1:N line system

The figure above shows a nodal view and highlights ConfiguraionAndSwitchControllers (C&SC) encapsulated in the FcSwitch in some cases and in FC in others. The encapsulation chosen depends upon the scope of control of the C&SC. The encapsulation is via FcSwitchCoordinatedByInternalControl when in the FcSwitch and FcSwitchesInFcCoordinatedBySwitchCoordinator when in the FC.

In the case of 1:N with Extra Traffic it is necessary for the switch of the Extra Traffic to be coordinated with the switches for protection of the main traffic (and likewise for the switches of each of the main traffic signals to be coordinated). It is assumed here that there is a real C&SC that carries out that coordination. The C&SCs encapsulated in the FCs/FcSwitches are assumed subordinate and hence the ControllerGovernsController association is one way from the independent C&SC to the C&SCs in the FCs/FcSwitches.

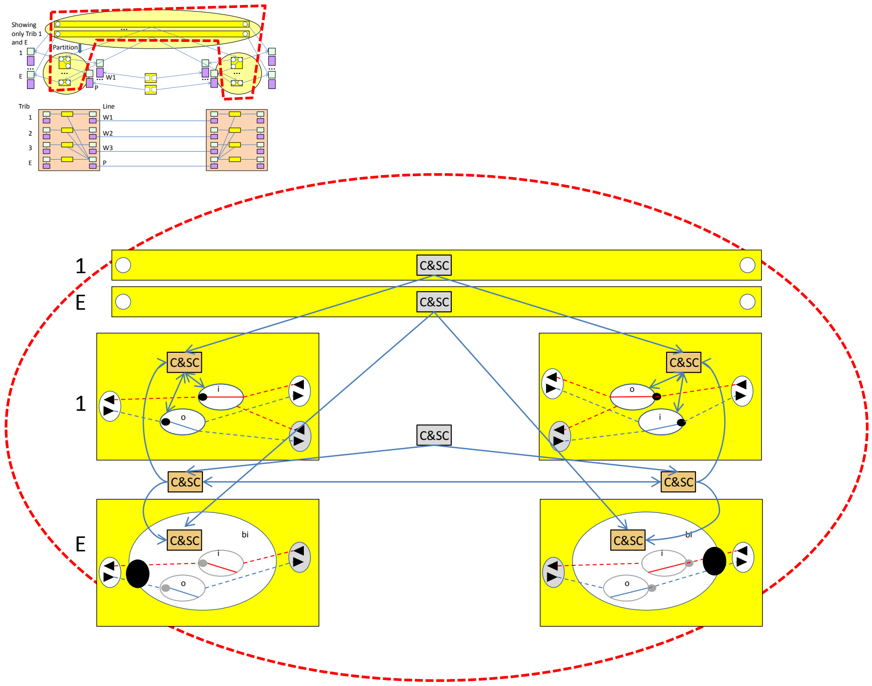


Figure 4-19 Showing an emergent abstract controller in a 1:N case

The figure above shows a case of 1:N independent switching (where the two directions of traffic are switched independently). The figure assumes that there is a distributed control solution (where the C&SCs in the FCs signal each other) and highlights the emergent C&SCs (not all controllers are relevant for control and control may be scattered across the controllers[6]).

The abstract C&SCs can be expressed to collect together parameters that should be set to the same value at both ends or to some other complementary values for competing switches at the same end. In the network the coordination occurs through peer signaling. Above the network the SDN controller may realize the coordination[7].

In the figure above the Extra Traffic has been switched off although only one direction of the Protection route is being used. It is assumed here that the Extra Traffic is bidirectional in nature and the loss one direction makes the signal useless (and hence both directions should be switched together.

[6] At a later point this will be clarified and the C&SCs that are relevant for control will be highlighted. The scheme spec will define which C&SCs are the target for commands etc

[7] This recognition of levels of control from the most basis local two state switch controller through the various levels shown here and on two ring controllers and the SDN controller peer-hierarchy is a manifestation of and a validation of the concept of the Management-Control Continuum. Representation of the Management-Control Continuum will be further explored in the next release.

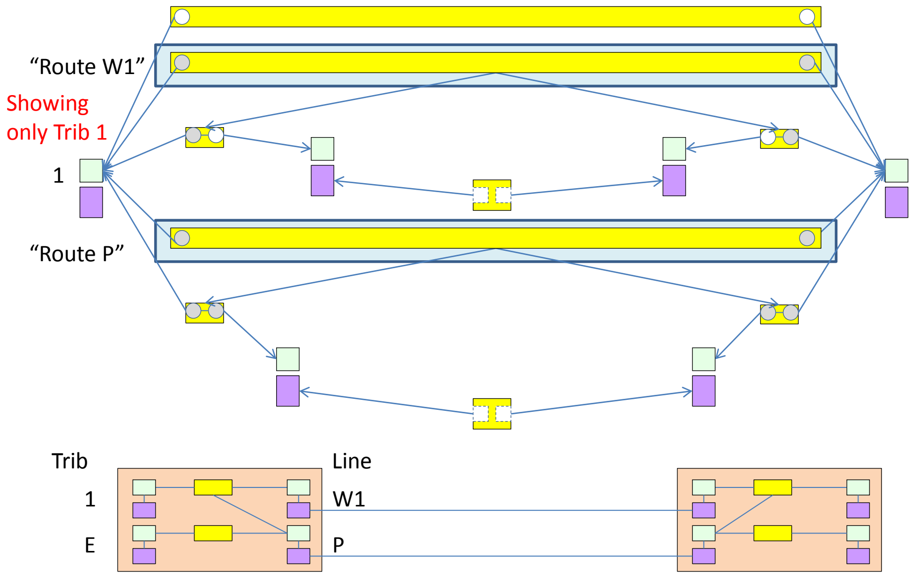


Figure 4-20 Showing route based representation of the 1:N protection scheme

The figure above shows a fragment of the route based representation. The figure detail only shows one traffic signal to avoid clutter. All traffic signals and the Extra Traffic are modeled with the same essential form. Extra Traffic is shown in the figure below.

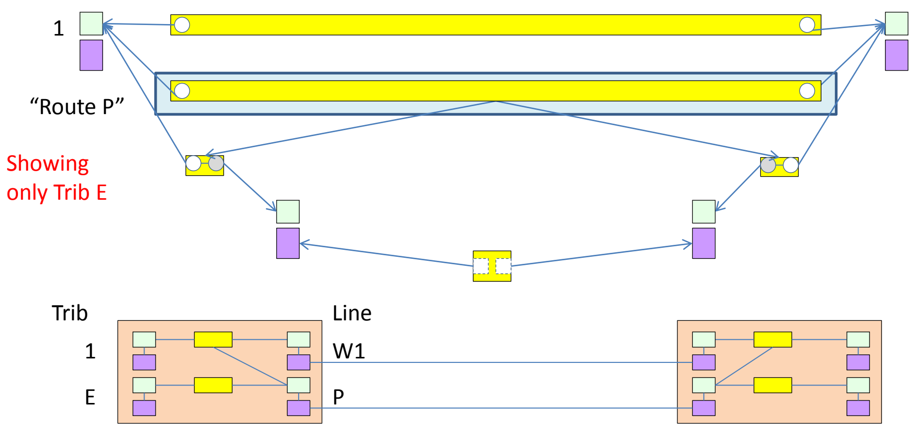


Figure 4-21 Showing Extra Traffic in a route based representation of the 1:N protection scheme

The figure below shows detail of C&SCs and switches for W1 in the 1:N scheme.

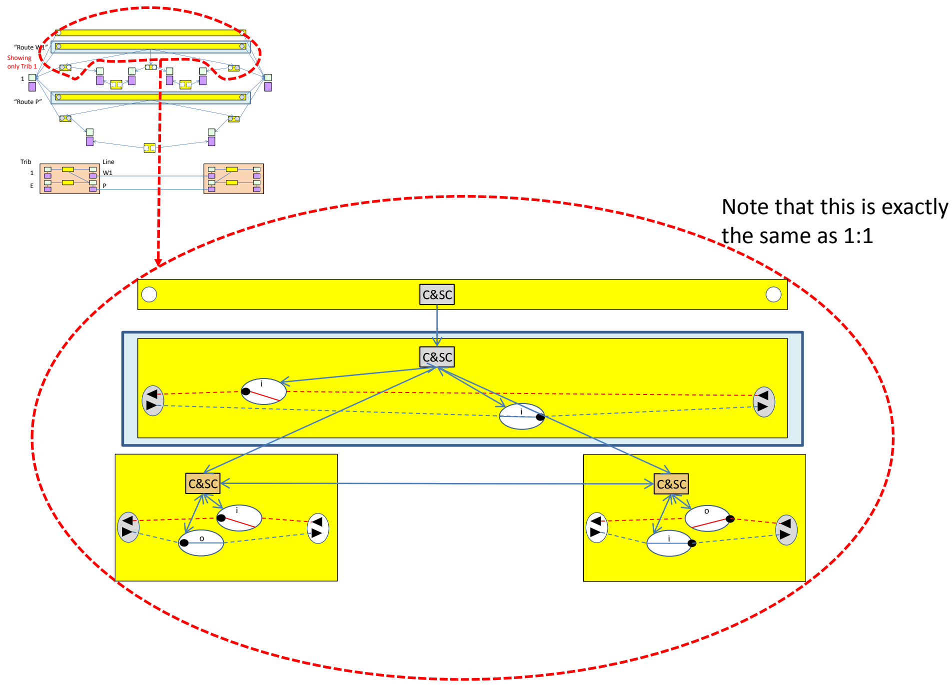


Figure 4-22 Showing independent two ended view of W1 route detail in a 1:N protection scheme

4.5 N:1 Cases

This section deals with N:1 cases focusing on timing synchronization as an application of this form of protection.

The figure below shows a fragment of an N:1 network wide scheme. It is assumed in the figure that I1 is an input from a network external to the one depicted and hence I1 is represented in the upper FC.

The scheme depicted is related to distribution of timing (unidirectional from left to right in the figure). There is assumed to be a single timing network that has essentially one vast FC representing the points in the network where timing signals are originated and are terminated and used etc. The depiction of the upper FC is intentionally vague as the focus here is not on the representation of timing termination but solely on the protection scheme. Work is underway on the representation of timing (see TR-512.1 ONF Core IM - Overview for a section on future Core Model work).

Considering the protection scheme:

- There may be many inputs carrying the same signal (or an equivalent)
- There may be many outputs for this signal to be propagated to other places
- There may be monitoring or use of the signal at any switching point
- In the case of synchronization there is some processing of the signal on transit that needs to be represented

The figure below only shows one node in detail. The small FC taking the inputs I1 to In feeds to a signal processing element represented by an LTP (grey) that then feeds O1 to Om via a single unidirectional multi-cast FC.

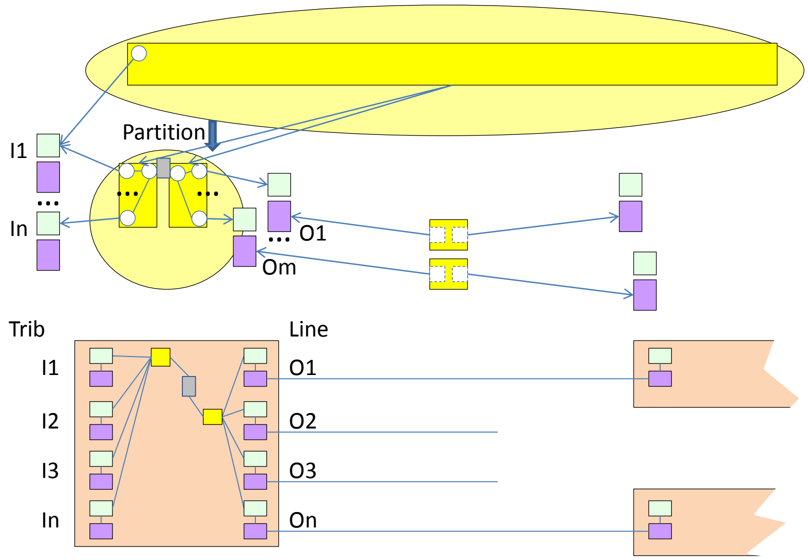


Figure 4-23 Showing a unidirectional N:1 scheme fragment

Further information is provided on potential timing distribution models in 6.5 Resilience and timing distribution on page 43.

5 Protection of other functions of physical things

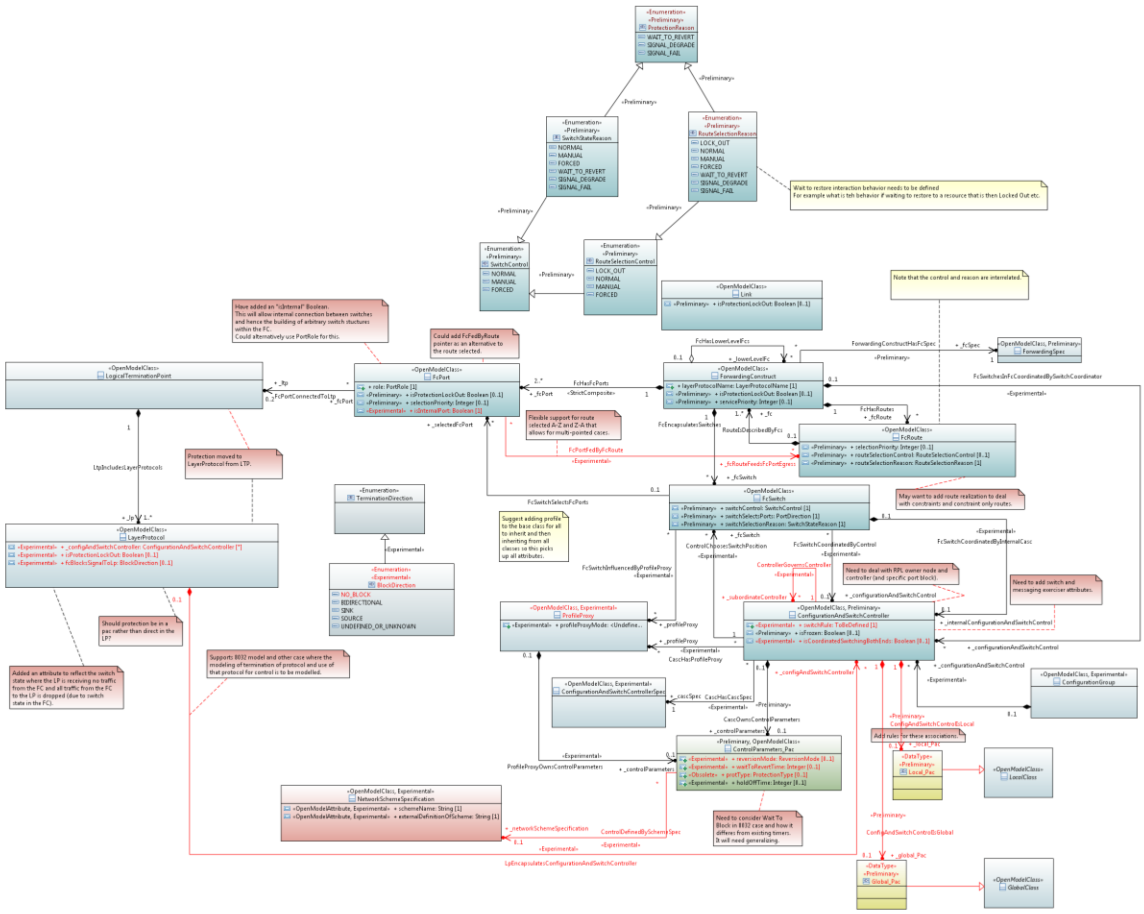
The Physical model covered by [TR-512.6 ONF Core IM - Physical](). This document focuses on the modeling of Equipment. The Equipment is considered to be purely physical. The document also provides some modeling of functions that are emergent from a physical assembly when powered. Clearly, all functions including those encapsulated by LTPs and FCs are only realized in by a powered physical assembly.

The functions being supported by the equipment can be protected. This type of protection often goes under the name "Equipment Protection". This name has not been used as it blurs the intentional constraint that Equipment is purely physical (where a physical thing can be measured with a ruler). Physical things are not protected,the functions that they support are protected; it is functions supported by additional physical things that give rise to resilient/protected functions.

The document provides a sketch of how functional resilience could be represented. This aspect is for further work in the next release. The intention is to use a switch/controller based pattern to represent functional resilience/protection.

6 Work in progress related to resilience

The figure below highlights, using UML comments, some details of various areas of work in progress.



CoreModel diagram: Resilience-EnhancedSketchOfSwitchCoordinatorAndProfile

Figure 6-1 Some comments on on-going work

The ConfigurationAndSwitchController can provide the control parameter to the FcSwitch (directly or via a ProfileProxy) or the FcSwitch can reference a profile (also available to the ConfigurationAndSwitchController).

Not covered at this point:

- This controllers need to be controlled/managed
- The controllers need a capability spec

6.1 Signaling information flow

As noted in earlier sections the intention at this point is not to model signaling explicitly. However, it is clear that signaling is simply the conveying of information and the conveying of information in general is represented by FCs and LTPs. Hence it is clear that if there was an intention to model the signaling network it should simply be more of the same. This is also true for management messaging (which is simply signaling by another name!).

There are two distinct cases to consider:

- Closed: where the signaling/messaging is solely within the visible/controlled network
- Open: where the signaling/messaging emerges from the visible/controlled network

The open case occurs where, for example, there is an admin boundary the cuts a protection scheme and where the administrative entities have agreed to enable their management/control systems to exchange messages/signals to achieve inter-administration automation. This applies to B2B exchanged and E-NNI exchanges[8]

6.1.1 Closed case

- Current assumption is that a controller that uses signalling is identified in the appropriate spec
    - The model uses ControllerGovernsController in both directions to indicate a peer.
        - Attributes could be added to indicate whether the controller is signalling to a peer or not and that the signalling grouping is determined from the spec and switch orientation
- It would be possible to show
    - Signalling flow through the network by associating the C&SC with an LTP via a new association that indicates that signalling information is sent through the adapter of the LTP
        - The LTP spec would explain the adaptation and hence association with another C&SC could be derived
    - A direct peer association between C&SC with no view of the underlying mechanism
    - A full forwarding model for the signalling information flow
        - This could be in a referenced pattern that is the summarized rigorously in one of the above forms
        - The resilience scheme spec would explain the signalling flow alterntaives
- Note that a full forwarding model would appear to make sense when the signalling flow routing is not coincident with the traffic flow routing
    - An attribute could be added to indicate that signalling is co-routed with the traffic being controlled

[8] The intention in the long term is to unify these two currently distinct considerations under one single architecture.

6.1.2 Open case

- This case has an open signalling path so there needs to be an expression of the signalling where it will emerge explaining what it is etc.. Signalling information is exposed at the edge of the network
  - Again current assumption is that a controller that uses signalling is identified in the appropriate spec
    - Also with the attribute to indicate whether the controller is signalling to a peer or not and that the signalling grouping is determined from the spec and switch orientation
  - The ControllerGovernsController cannot name peer as it is not within the view so an off-net form of foreign pointer would be necessary (or there could be a dummy controller with a few parameters (perhaps discoverable, perhaps manually entered) as well as the name)
- Potentially more relevantly in this case we could show
  - Signalling flow through the network by associating the C&SC with an LTP via a new association that indicates that signalling information is sent through the adapter of the LTP
    - The LTP spec would explain the adaptation and hence association with another C&SC could be derived
  - A direct peer association between C&SC with no view of the underlying mechanism
  - A full forwarding model for the signalling information flow
    - This could be in a referenced pattern that is the summarized rigorously in one of the above forms
    - Note that a full forwarding model would appear to make sense when the signalling flow routing is not coincident with the traffic flow routing
      - An attribute could be added to indicate that signalling is co-routed with the traffic being controlled

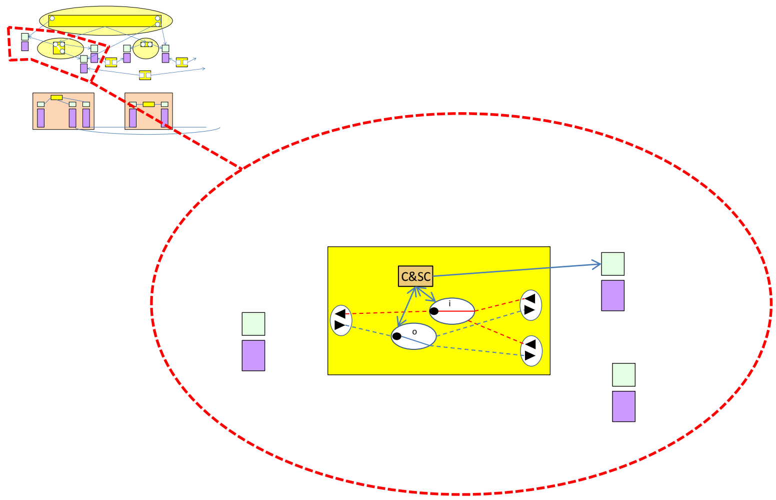


Figure 6-2 1:1 independent two ended view considering explicit signalling

6.1.3 Signaling control

- Need to identify parameters related to signalling and control that are independent of switching or only partly dependent on switching
  - Can timers be adjusted?
  - Can signalling be disabled?
  - Can aspects of signalling be disabled?
  - Can control be adjusted?

6.2 Additional considerations for FcRoute

An FcRoute may:

- Be provisioned in the network but turned off
- Have resources reserved in the network
- Have resources that are reserved but shared with one or more other routes (either in the same FC or a different FC)
- Have specified but not reserved resources
- Have partially specified resources
- Have no resources specified and hence no subordinate FC detail

This implies the need to add properties on LifeCycleState (reserved, provisioned etc for the route) and to support a route in terms of constraints

An FcRoute may have encapsulated protection or other complex nesting of resilience schemes. Whilst the model supports this it has not been exercised with any cases. The figure below has a sketch of two alternative routes both of which have internal protection

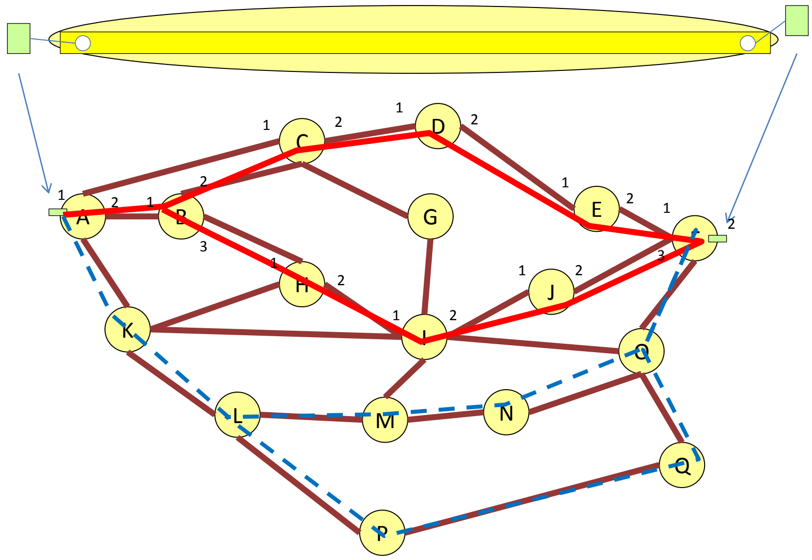


Figure 6-3 Sketch of two routes with internal resilience

6.3 Representation alternatives – Partition or Route

Consider the figure below of a simple network with relatively sophisticated switching nodesk with a single FC spanning from A1 to C2.

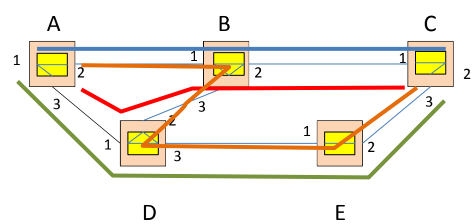


Figure 6-4 FcRoute in a complex network

A1-C2 has four routes each of which has one FC

- Blue: A1-A2, B1-B2, C1-C2
- Red: A1-A3, D1-D2, B3-B2, C1-C2
- Green: A1-A3, D1- D3, E1-E2, C3-C2
- Brown: A1-A3, B1- B3, D2-D3, E1-E2, C3-C2

In more complex cases there could be many potential routes for a sophisticated switch configuration where there are only a few well defined switches.

Adding two more nodes and two more switches would double the number of routes. Adding more ends would further multiply the number of routes.

For complex layouts the route approach is not an efficient way of expressing the layout and instead the FC partition should be used.

6.4 Relationship to the ProtectionGroup approach

The brief figure below sketches the relationship between a Protection Group approach and the FcSwitch. Further work is required to formalize the relationships.

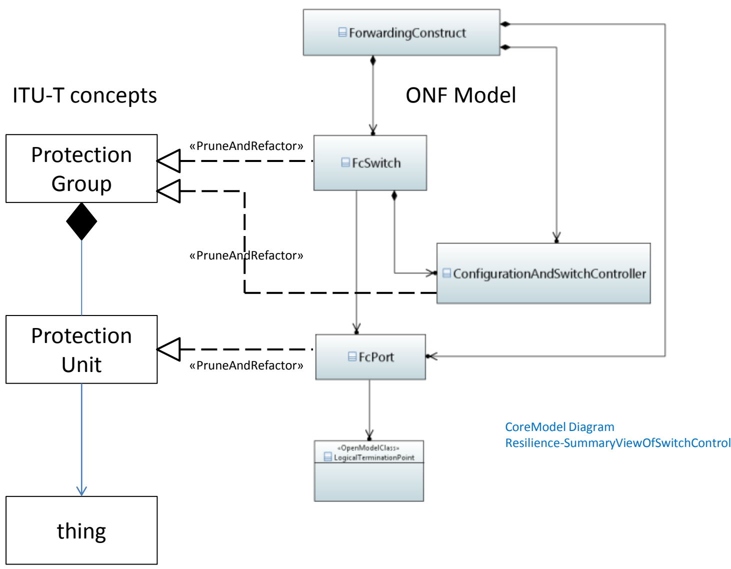


Figure 6-5 Relationship between FcSwitch approach and ProtectionGroup approach

6.5 Resilience and timing distribution

The figure below shows some stylized detail of a somewhat speculative model of timing processing showing several different timing signals being switched independently and regenerated in a common set of interrelated LTP-like functions. The particular view also shows a regeneration bypass LTP.

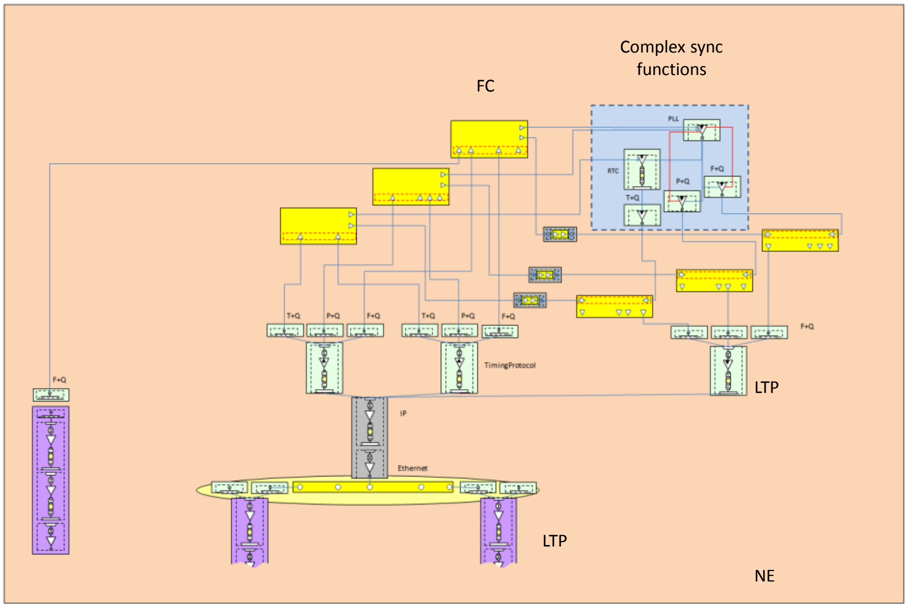


Figure 6-6 Speculative view of timing model

The figure below shows a simplified view that extracts the regeneration element as a single LTP.

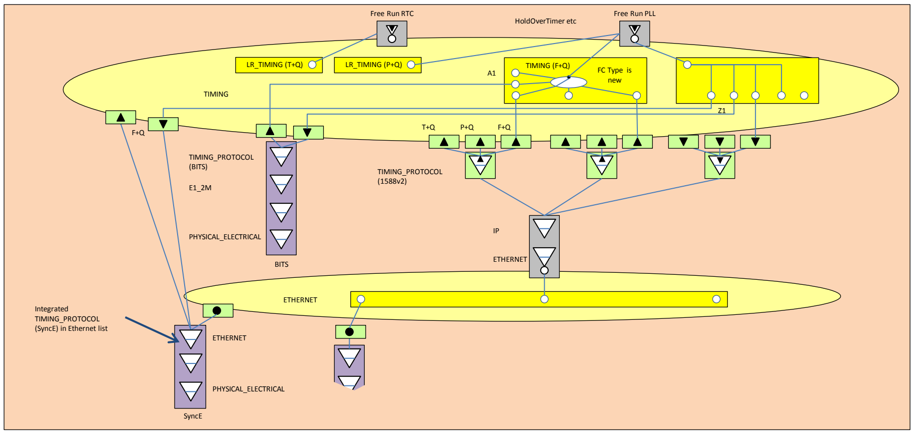


Figure 6-7 Speculative simplified view of timing model

6.6 Which route is feeding a port?

The figure below shows an association from the FcPort to the route. The association indicates which route is carrying the signal to the FcPort.

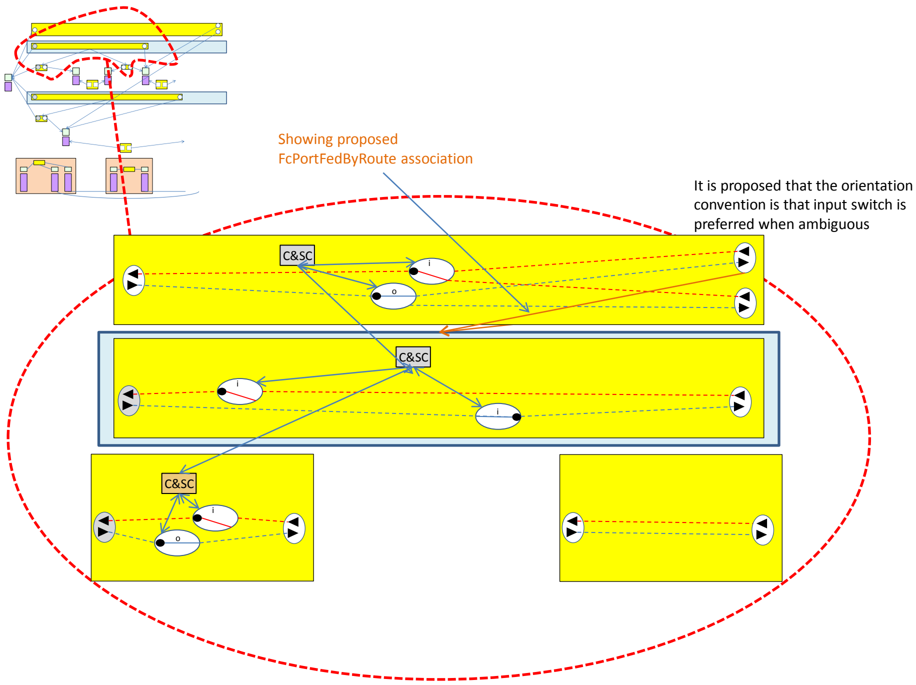


Figure 6-8 1:1 independent two ended view showing FcPortFedByRoute

End of document

