AADL Networking Annex

Status Report

Alexey Khoroshilov

SAE AS-2C AADL Spring Meeting 2018

Question 1. Max_Supported_VLs

Feedback - Steve Vestal (2)

```
Max_Supported_VLs:
     inherit AADLINTEGER => 1024
applies to (device,system,processor);
```

Feedback - Steve Vestal (2)*

```
Max_Supported_VLs:
     inherit AADLINTEGER => 1024
applies to (device,system,processor);
```

It was considered as a reasonable addition with the note that we should omit the default value in the annex.

As for the semantics of the property, the question arisen is: does Max_Supported_VLs mean a limit for

- outgoing VLs,
- incoming VLs,
- or a sum of incoming and outgoing VLs?

As far as AFDX end systems can have independent limits on maximum number of incoming and outgoing VLs as well as a combined one, we may want to support all of them.

For switches the situation is even worse, as they have VLs routing through the switch, but the switch can operates as an AFDX end system on some of its ports.

Feedback - Steve Vestal (2)*

So, it can have limits on a number of incoming/outgoing VLs (as end system) and a separate limit on a number of routed VLs. Also the limits can be a limit for the switch as whole or a limit per incoming/outgoing port.

The question is very close to your white paper regarding Capacity&Load properties and to the discussion of binding points of AADLv3.

A possible approach to cover most of the cases in AADLv2 could be to add four properties:

(1) Max_Supported_Outgoing_VLs or Outgoing_VLs_Capacity

(2) Max_Supported_Incoming_VLs or Incoming_VLs_Capacity

(3) Max_Supported_VLs or Total_VLs_Capacity

(4) Max_Supported_Routing_VLs or Routing_VLs_Capacity

Feedback - Steve Vestal (2)*

(1) Max_Supported_Outgoing_VLs or Outgoing_VLs_Capacity
(2) Max_Supported_Incoming_VLs or Incoming_VLs_Capacity
(3) Max_Supported_VLs or Total_VLs_Capacity
(4) Max_Supported_Routing_VLs or Routing_VLs_Capacity

with the following consistency rules:

1. if an **end system** has property (1), a number of outgoing VLs bound to the end system should be no more than its value;
2. if an **end system** has property (2), a number of incoming VLs bound to the end system should be no more than its value;
3. if an **end system** has property (3), a sum of incoming and outgoing VLs bound to the end system should be no more than its value;
4. if **bus access of a switch** has property (1), a number of outgoing VLs bound to the bus access should be no more than its value;
5. if **bus access of a switch** has property (2), a number of incoming VLs bound to the bus access should be no more than its value;
6. if a **switch** has property (4), a number of routing VLs bound to the switch should be no more than its value;
7. if a **switch** has property (1), a number of outgoing VLs bound to the switch (VLs handling in end system mode) should be no more than its value;
8. if a **switch** has property (2), a number of incoming VLs bound to the switch (VLs handling in end system mode) should be no more than its value;
9. if a **switch** has property (3), a sum of incoming, outgoing, and routing VLs bound to the switch should be no more than its value;

Question 2. Is_AFDX_XXX

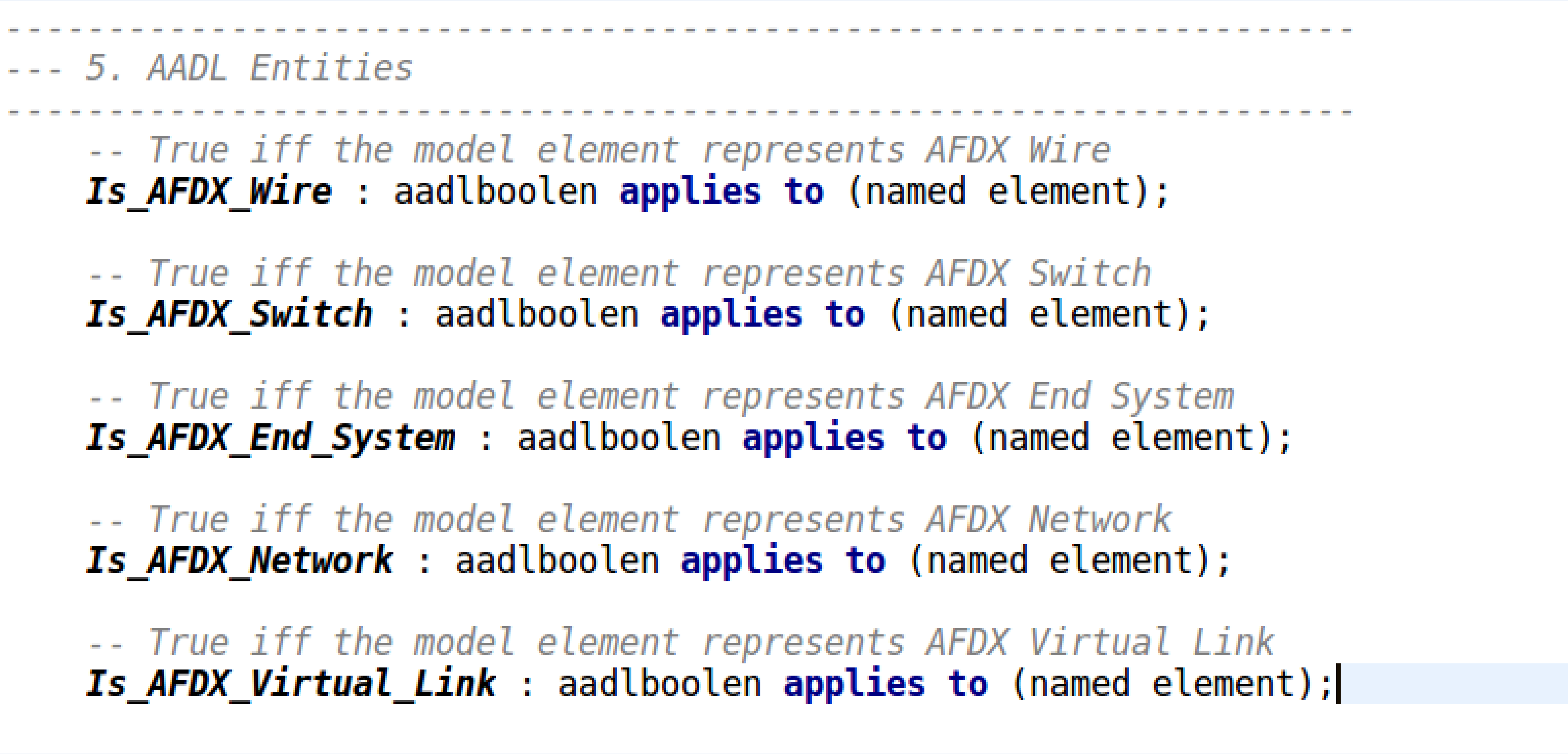
Feedback - Steve Vestal (3)

We did not necessarily require switches and end systems be modeled using a single specific category or pattern.

End systems in particular may be combinations of multiple software and hardware components that collectively provide the required interface functionality.

There will probably need to be some sort of **patterns or modeling conventions** defined in order to make tool development reasonably feasible (the ARINC 653 annex does this some, e.g. the pattern for a partition is a process hosted on a virtual processor).

Feedback - Steve Vestal (3)*

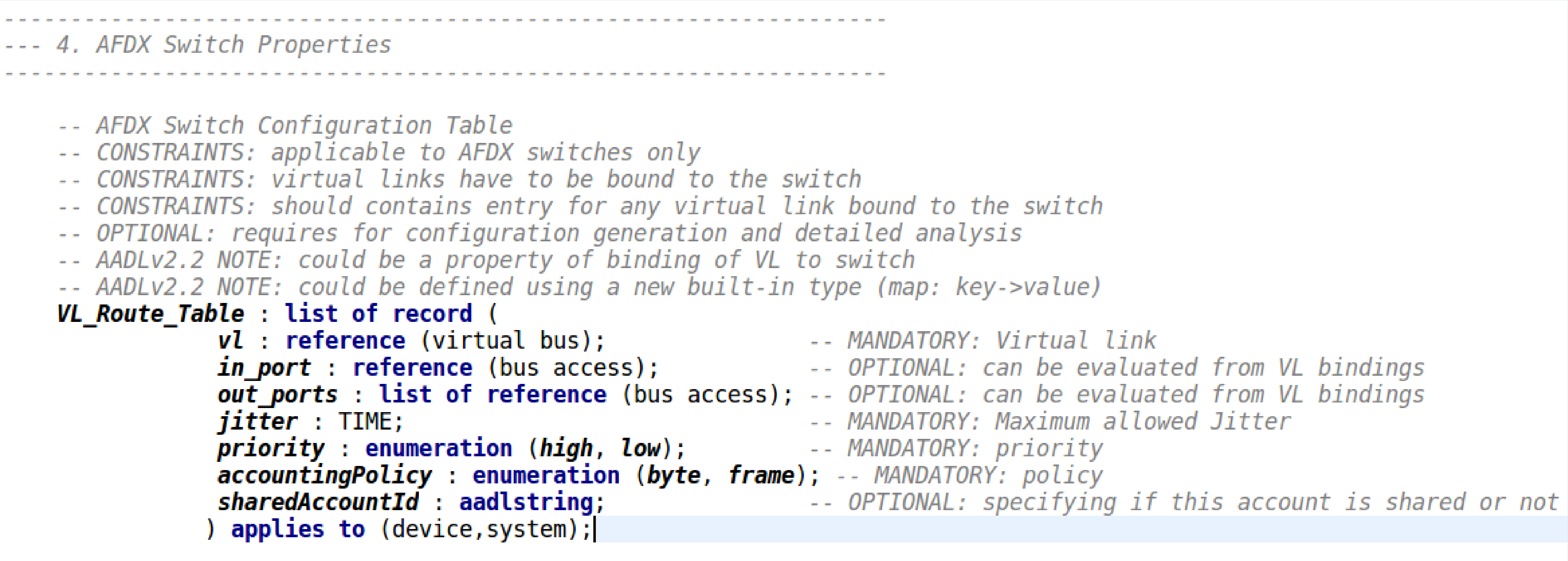


```
------------------------------------------------------------------
--- 5. AADL Entities
------------------------------------------------------------------
    -- True iff the model element represents AFDX Wire
    Is_AFDX_Wire : aadlboolen applies to (named element);

    -- True iff the model element represents AFDX Switch
    Is_AFDX_Switch : aadlboolen applies to (named element);

    -- True iff the model element represents AFDX End System
    Is_AFDX_End_System : aadlboolen applies to (named element);

    -- True iff the model element represents AFDX Network
    Is_AFDX_Network : aadlboolen applies to (named element);

    -- True iff the model element represents AFDX Virtual Link
    Is_AFDX_Virtual_Link : aadlboolen applies to (named element);
```

Question 3. VL_Route_Table

Feedback - Etienne (1)

> However, I don't see the rationale for having this property applied on devices and referencing the virtual bus. Specially since I am trying to specify the proprity and jitter for the virtual bus itself.



```
-----------------------------------------------------------------
--- 4. AFDX Switch Properties
-----------------------------------------------------------------

    -- AFDX Switch Configuration Table
    -- CONSTRAINTS: applicable to AFDX switches only
    -- CONSTRAINTS: virtual links have to be bound to the switch
    -- CONSTRAINTS: should contains entry for any virtual link bound to the switch
    -- OPTIONAL: requires for configuration generation and detailed analysis
    -- AADLv2.2 NOTE: could be a property of binding of VL to switch
    -- AADLv2.2 NOTE: could be defined using a new built-in type (map: key->value)
    VL_Route_Table : list of record (
                vl : reference (virtual bus);               -- MANDATORY: Virtual link
                in_port : reference (bus access);           -- OPTIONAL: can be evaluated from VL bindings
                out_ports : list of reference (bus access); -- OPTIONAL: can be evaluated from VL bindings
                jitter : TIME;                              -- MANDATORY: Maximum allowed Jitter
                priority : enumeration (high, low);         -- MANDATORY: priority
                accountingPolicy : enumeration (byte, frame); -- MANDATORY: policy
                sharedAccountId : aadlstring;               -- OPTIONAL: specifying if this account is shared or not
               ) applies to (device,system);
```

Feedback - Etienne (1)

> However, I don't see the rationale for having this property applied on devices and referencing the virtual bus. Specially since I am trying to specify the proprity and jitter for the virtual bus itself.

Actually the configuration is a property of a binding of a VL to a switch. In AADLv3 we will have an ability to move it there.

For now we have to choose between one of the ends of the binding: VL or device. Both options are not ideal.

Device was chosen because it is more natural from point of view of the physical world. It is the device who keeps the configuration tables.

Anyway the properties are defined in a containing system.

For me, the only practical difference is the grouping. Currently it is grouped by switch, otherwise it would be grouped by VL.

Question 4. Priority Type

Feedback - Etienne (2)

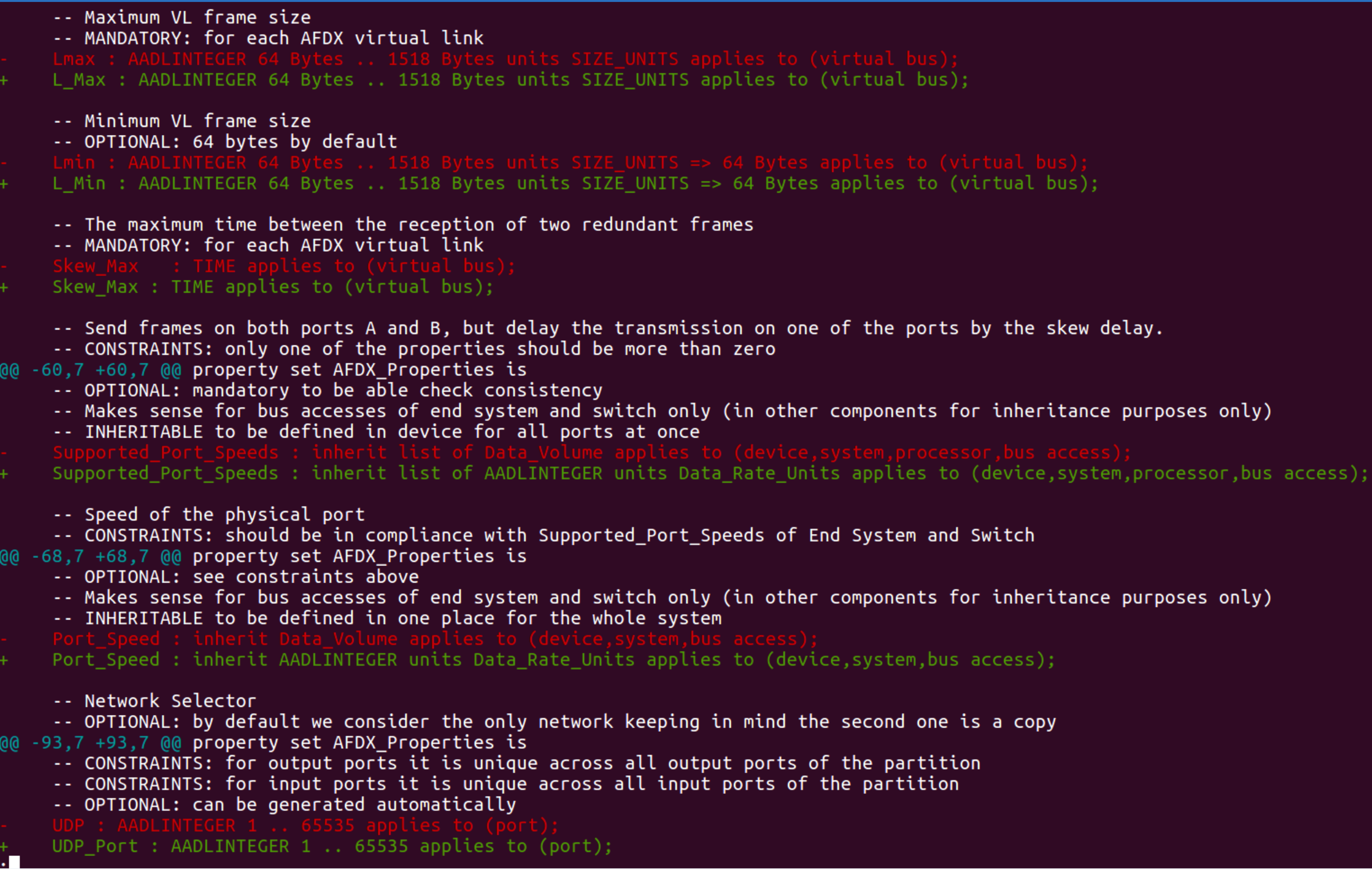
> Besides, I would have thought of the priority as an integer value.

AFDX specification explicitly defines two priority levels: The Switch should have a traffic prioritization mechanism based on MAC destination address with 2 traffic classes: High Priority and Low Priority. The priority level should be defined in the configuration table on a Virtual Link basis.

What are benefits of the integer? More number of priority levels? Comparison operator?

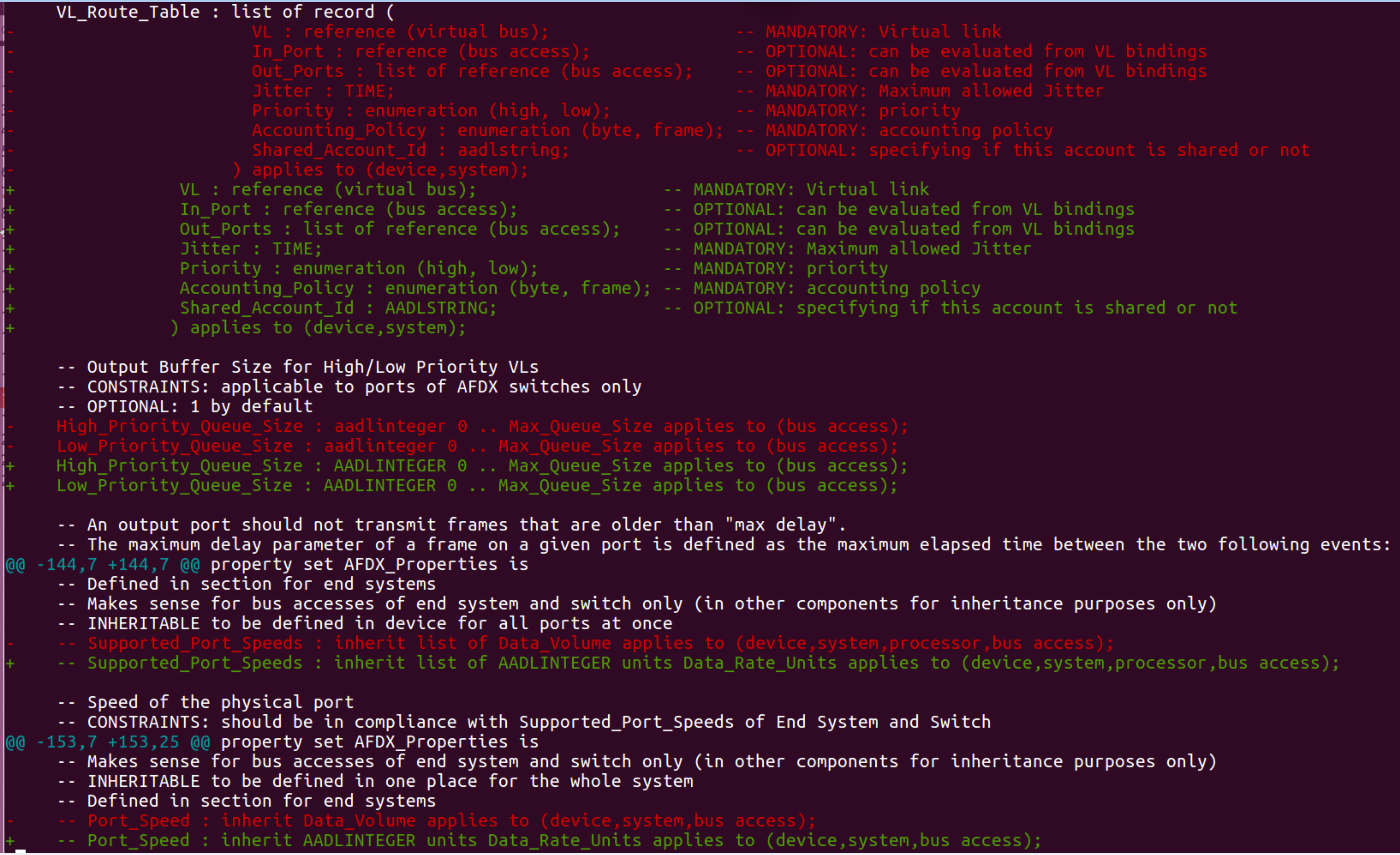
Question 5. Coding Style

Coding Style Fixes (1)



```
     -- Maximum VL frame size
     -- MANDATORY: for each AFDX virtual link
-    Lmax : AADLINTEGER 64 Bytes .. 1518 Bytes units SIZE_UNITS applies to (virtual bus);
+    L_Max : AADLINTEGER 64 Bytes .. 1518 Bytes units SIZE_UNITS applies to (virtual bus);
 
     -- Minimum VL frame size
     -- OPTIONAL: 64 bytes by default
-    Lmin : AADLINTEGER 64 Bytes .. 1518 Bytes units SIZE_UNITS => 64 Bytes applies to (virtual bus);
+    L_Min : AADLINTEGER 64 Bytes .. 1518 Bytes units SIZE_UNITS => 64 Bytes applies to (virtual bus);
 
     -- The maximum time between the reception of two redundant frames
     -- MANDATORY: for each AFDX virtual link
-    Skew_Max   : TIME applies to (virtual bus);
+    Skew_Max : TIME applies to (virtual bus);
 
     -- Send frames on both ports A and B, but delay the transmission on one of the ports by the skew delay.
     -- CONSTRAINTS: only one of the properties should be more than zero
@@ -60,7 +60,7 @@ property set AFDX_Properties is
     -- OPTIONAL: mandatory to be able check consistency
     -- Makes sense for bus accesses of end system and switch only (in other components for inheritance purposes only)
     -- INHERITABLE to be defined in device for all ports at once
-    Supported_Port_Speeds : inherit list of Data_Volume applies to (device,system,processor,bus access);
+    Supported_Port_Speeds : inherit list of AADLINTEGER units Data_Rate_Units applies to (device,system,processor,bus access);
 
     -- Speed of the physical port
     -- CONSTRAINTS: should be in compliance with Supported_Port_Speeds of End System and Switch
@@ -68,7 +68,7 @@ property set AFDX_Properties is
     -- OPTIONAL: see constraints above
     -- Makes sense for bus accesses of end system and switch only (in other components for inheritance purposes only)
     -- INHERITABLE to be defined in one place for the whole system
-    Port_Speed : inherit Data_Volume applies to (device,system,bus access);
+    Port_Speed : inherit AADLINTEGER units Data_Rate_Units applies to (device,system,bus access);
 
     -- Network Selector
     -- OPTIONAL: by default we consider the only network keeping in mind the second one is a copy
@@ -93,7 +93,7 @@ property set AFDX_Properties is
     -- CONSTRAINTS: for output ports it is unique across all output ports of the partition
     -- CONSTRAINTS: for input ports it is unique across all input ports of the partition
     -- OPTIONAL: can be generated automatically
-    UDP : AADLINTEGER 1 .. 65535 applies to (port);
+    UDP_Port : AADLINTEGER 1 .. 65535 applies to (port);
:
```

Coding Style Fixes (2)



```
     VL_Route_Table : list of record (
-                       VL : reference (virtual bus);                  -- MANDATORY: Virtual link
-                       In_Port : reference (bus access);              -- OPTIONAL: can be evaluated from VL bindings
-                       Out_Ports : list of reference (bus access);    -- OPTIONAL: can be evaluated from VL bindings
-                       Jitter : TIME;                                 -- MANDATORY: Maximum allowed Jitter
-                       Priority : enumeration (high, low);            -- MANDATORY: priority
-                       Accounting_Policy : enumeration (byte, frame); -- MANDATORY: accounting policy
-                       Shared_Account_Id : aadlstring;                -- OPTIONAL: specifying if this account is shared or not
-                     ) applies to (device,system);
+                VL : reference (virtual bus);                  -- MANDATORY: Virtual link
+                In_Port : reference (bus access);              -- OPTIONAL: can be evaluated from VL bindings
+                Out_Ports : list of reference (bus access);    -- OPTIONAL: can be evaluated from VL bindings
+                Jitter : TIME;                                 -- MANDATORY: Maximum allowed Jitter
+                Priority : enumeration (high, low);            -- MANDATORY: priority
+                Accounting_Policy : enumeration (byte, frame); -- MANDATORY: accounting policy
+                Shared_Account_Id : AADLSTRING;                -- OPTIONAL: specifying if this account is shared or not
+               ) applies to (device,system);

     -- Output Buffer Size for High/Low Priority VLs
     -- CONSTRAINTS: applicable to ports of AFDX switches only
     -- OPTIONAL: 1 by default
-    High_Priority_Queue_Size : aadlinteger 0 .. Max_Queue_Size applies to (bus access);
-    Low_Priority_Queue_Size : aadlinteger 0 .. Max_Queue_Size applies to (bus access);
+    High_Priority_Queue_Size : AADLINTEGER 0 .. Max_Queue_Size applies to (bus access);
+    Low_Priority_Queue_Size : AADLINTEGER 0 .. Max_Queue_Size applies to (bus access);

     -- An output port should not transmit frames that are older than "max delay".
     -- The maximum delay parameter of a frame on a given port is defined as the maximum elapsed time between the two following events:
@@ -144,7 +144,7 @@ property set AFDX_Properties is
     -- Defined in section for end systems
     -- Makes sense for bus accesses of end system and switch only (in other components for inheritance purposes only)
     -- INHERITABLE to be defined in device for all ports at once
-    -- Supported_Port_Speeds : inherit list of Data_Volume applies to (device,system,processor,bus access);
+    -- Supported_Port_Speeds : inherit list of AADLINTEGER units Data_Rate_Units applies to (device,system,processor,bus access);

     -- Speed of the physical port
     -- CONSTRAINTS: should be in compliance with Supported_Port_Speeds of End System and Switch
@@ -153,7 +153,25 @@ property set AFDX_Properties is
     -- Makes sense for bus accesses of end system and switch only (in other components for inheritance purposes only)
     -- INHERITABLE to be defined in one place for the whole system
     -- Defined in section for end systems
-    -- Port_Speed : inherit Data_Volume applies to (device,system,bus access);
+    -- Port_Speed : inherit AADLINTEGER units Data_Rate_Units applies to (device,system,bus access);
```

Summary

- Q1: Max_Supported_VLs
- Q2: Is_AFDX_XXX
- Q3: VL_Route_Table
- Q4: Priority Type
- Q5: Coding Style

Recent Version

The most recent version available here:

https://gitlab.com/sae_as2c/networking-annex/wikis/home

Backup Slides

Alexey Khoroshilov
khoroshilov@ispras.ru

http://masiw.ispras.ru