Translation tables / rules for IA-stations

Requirements for Plug & Produce

03/22 v01

Make network configured traffic types and PTP instances available for middleware / application alignment

Thanks to Lukas Wuesteney and Josef Dorr for their support!

Günter Steindl
Siemens AG
Splitting network configuration from middleware/application configuration

Plug & Produce requires the splitting of responsibilities between middleware/application and network provisioning.

This make it impossible for the middleware/application configuration tool to assign PCP/VID values to middleware defined traffic types.

Another topic is the alignment of PTP instance IDs.

Thus, a translation

• between middleware defined traffic types and network provisioned traffic types is needed.
• between middleware defined ClockTargets and/or ClockSources, and network provisioned PTP instance IDs is needed.
Decoupling between middleware and communication

Splitting of responsibilities between middleware and network provisioning requires a standardized way for the middleware to access the information.

The local YANG database contains all network provisioned information.

Thus, a local access for the middleware to this information is required to implement the decoupling.
Traffic Type Translation
Middleware translates its Traffic Types into the network provided traffic types!

Provide this information to the different middlewares to support the translation / use of the network configured values.

This values are defined by the CNC and provided by Network Provisioning (NPE).
Decoupling is nothing new for IEEE 802.1Q – its already defined for MAC and VLAN interfaces.

**Principle:**
End station component MAC interface hosts multiple VLAN interfaces
VLAN Interfaces in the IETF Interface-list

How could a translation based on the IETF interface entries look like?

Content of name and/or description string shall be profiled by the TSN-IA profile.

Example:
Name := vlan-id-100
Description :=
- List of allowed traffic types and their PCP (optional and their DSCP) value
- Optional vlan-id of the VLAN used for redundancy in case of end station FRER
- [Version 2] Optional assigned CUC in case of multiple CUC at one host

<table>
<thead>
<tr>
<th>module: ietf-interfaces</th>
</tr>
</thead>
<tbody>
<tr>
<td>+--rw interfaces</td>
</tr>
<tr>
<td>+--rw interface* [name]</td>
</tr>
<tr>
<td>+--rw name</td>
</tr>
<tr>
<td>+--rw description?</td>
</tr>
<tr>
<td>+--rw type</td>
</tr>
<tr>
<td>+--rw enabled?</td>
</tr>
<tr>
<td>+--rw link-up-down-trap-enable?</td>
</tr>
<tr>
<td>+--ro admin-status</td>
</tr>
<tr>
<td>+--ro oper-status</td>
</tr>
<tr>
<td>+--ro last-change?</td>
</tr>
<tr>
<td>+--ro if-index</td>
</tr>
<tr>
<td>+--ro phys-address?</td>
</tr>
<tr>
<td>+--ro higher-layer-if*</td>
</tr>
<tr>
<td>+--ro lower-layer-if*</td>
</tr>
<tr>
<td>+--ro speed?</td>
</tr>
<tr>
<td>+--ro statistics</td>
</tr>
</tbody>
</table>

string
string
identityref
boolean
enumeration {if-mib}?
enumeration {if-mib}?
enumeration
yang:date-and-time
int32 {if-mib}?
yang:phys-address
interface-ref
interface-ref
yang:gauge64
Sanity check

How could a translation based on the IETF interface entries for the IA-station example look like?

Example:
IETF Interface-list contains at least twelve entries:
End station component 1:
- MAC interface
- Three different VLAN interfaces
End station component 2:
- MAC interface
- Two different VLAN interfaces
End station component 3:
- MAC interface
- Four different VLAN interfaces
Conclusion - Traffic Type Translation

A translation table which allows late binding of the middleware to the network resources already exists.

Profiling the content of name and description would allow the required translation for TC, VID, PCP and if needed DSCP.

CNC/NPE configured entries IETF Interface-list should solve the problem.
PTP Instance ID translation
Provide this information to the different middlewares to support the translation / use of the network configured values.

This values are defined by the CNC and provided by Network Provisioning (NPE).
Decoupling for the to be used PTP instance could be done following the same principle.

Problem:
IETF interface-list doesn’t have an defined ifType for gPTP.

Temporary solution:
Content of name and/or description string (descriptionDS.userDescription) shall be profiled by the TSN-IA profile.

Example:
userDescription :=
- List of clock type e.g. WorkingClock and/or GlobalTime
Sanity check

How could a translation based on the descriptionDS.userDescription entry for the IA-station example look like?

Example:
PTP instance list contains at least four entries:
End station component 1:
- Two PTP instances, one for WorkingClock and one for GlobalTime
End station component 2:
- One PTP instance for GlobalTime
End station component 3:
- One PTP instances for WorkingClock and GlobalTime
Conclusion - PTP Instance ID translation

Profiling named references instead of instanceID in the 60802 profile gives automation system providers more freedom.

If no ifType for the IETF interface-list will be defined by 1588e/.1ASdn, then the string “descriptionDS.userDescription” could be used for the profiling.

Open:
Binding between ClockSources or ClockTargets and PTP instances seems at the moment not covered by the 1588e/.1ASdn drafts.
Conclusion

Splitting network configuration from middleware configuration could be configured by existing/in definition YANG modules.

Late binding even for PTP instances could be done.

CNE/NPE decides which PTP instance/ PTP instanceID shall be used for what at the IA-stations.
Questions?