YANG extension for Common Augmentations

draft-bertz-netmod-commonaugment-01

IETF 98, Chicago
Lyle Bertz
Common Augments

• A common augment is one where multiple locations of the Schema Tree is augmented in the same way

• These can become sticky when
  • A single, common augment is made in many locations
  • Generations of augments are added
Classic RPC Example

... basemodule {
  prefix b;
  ...
  grouping base_element {
    ...
  }
}

rpc my_rpc {
  input {
    uses b:base_element;
  }
  output {
    uses b:base_element;
  }
}

Not uncommon to have same structure in requests and response (bold)
Challenges

• Sometimes adding a structure to each location is intended in two different ways:
  • This is the same entity @ locations A, B, C, ...
  • At location A & B the entity is interpreted as ‘foo’ but at C… it is interpreted as ‘bar’
  • (Fairly common for low level structures)
  • How do we determine intent?
IETF DMM FPC

- Describes information model and protocol concepts
- YANG model supported but optional
  - Core model
  - 3GPP model
  - PMIP model
  - Extensions
- Issues discovered with common augments in open source YANG code generators (ODL's yangtools and ONOS Yangforge)
  - Both needed better hints on common structures to generate better code
  - treated each augment as a different class with only copying of common on arguments on the constructor and direct copying (no merging) to accidentally prevent merging two entities and creating an invalid, new one
    - However, we wanted the merging cause it was the same augment - we as the module creators KNEW it was safe!
Solution

module old_way {
  import base { prefix base_module; }  
  import newstuff { prefix new_module; }
  augment "/base_module:my_rpc/base_element/input" {
    uses new_module:new_things;
  }
  augment "/base_module:my_rpc/base_element/output" {
    uses new_module:new_things;
  }
  augment "/base_module:my_bestrpc/base_element/input" {
    uses new_module:new_things;
  }
}

module new_way {
  import base { prefix base_module; }  
  import newstuff { prefix new_module; }
  import commonaugment { prefix c; }
  augment "/base_module:my_rpc/base_element/input" {
    uses new_module:new_things;
    c:also-augments "/base_module:my_rpc/base_element/output";
    c:also-augments "/base_module:my_bestrpc/base_element/input";
  }
}

Backwards compatible (if you like) or less text (w/o backwards compatibility)
Next Steps

• Would like more discussion on the mailing list on
  • feedback
  • related issue(s) - if any
  • best way to proceed