SMIng Mappings to DIAMETER

Jürgen Schönwälder

Technical University of Braunschweig
Bültenweg 74/75
38106 Braunschweig, Germany

AAA WG Meeting, 49th IETF, San Diego, 2000-12-13

What is SMIng?

- SMIng is a new data definition language proposed to replace the SMIv2 and the SPPI:
  - object-oriented (classes, attributes, inheritance, containment)
  - protocol independent
  - instance naming independent
  - formal machine readable language (ABNF grammar)
  - compact and consistent syntax to increase human readability
  - controlled language extensibility

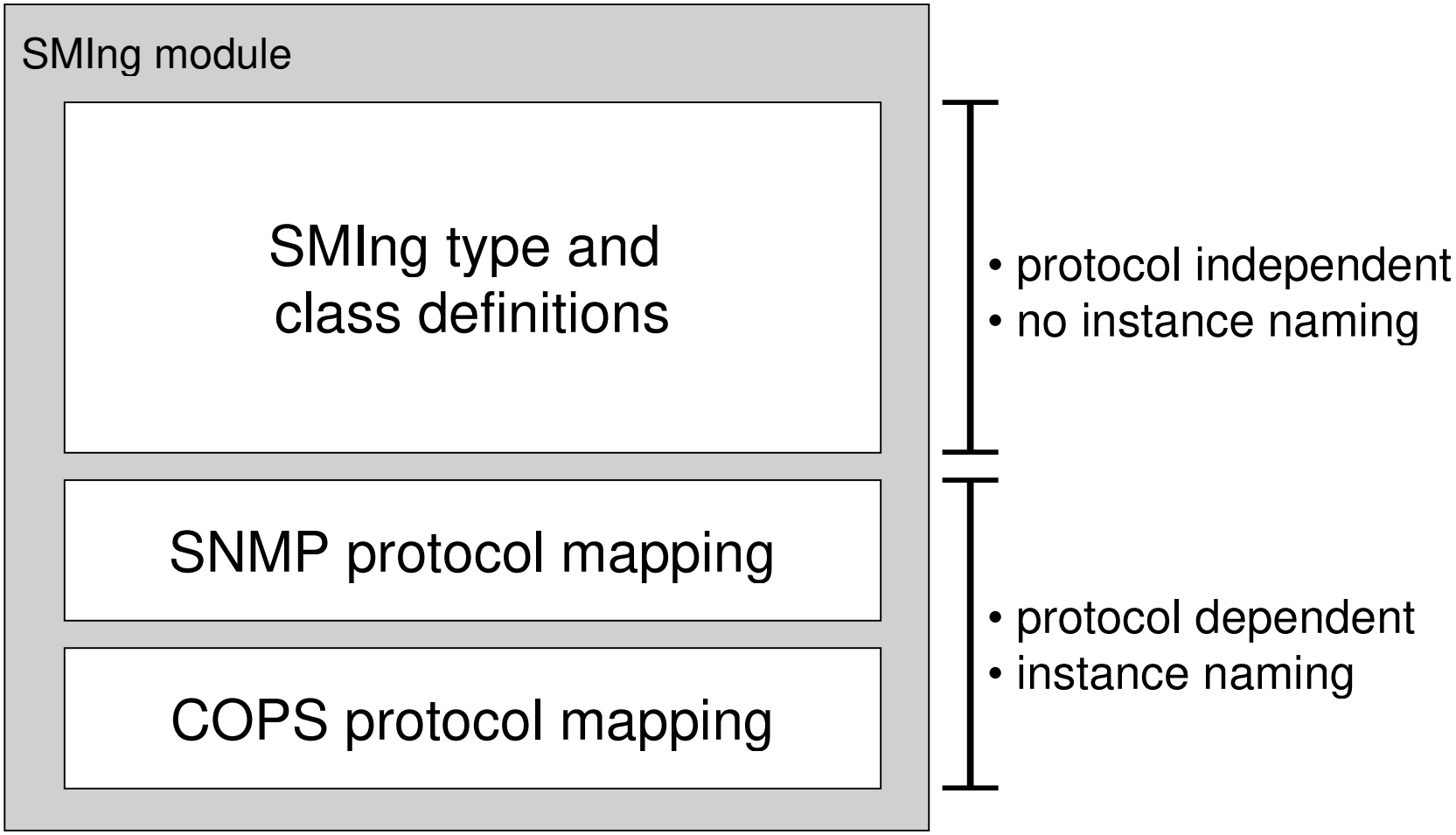
- Join the SMIng working group meeting for more details!

SMIng and DIAMETER

- Advantages of using SMIng for AAA data modeling:
  - reuse data definitions provided by other working groups
  - allow other working groups to reuse AAA data definitions
  - common data definitions across several management protocols
  - simplifies applications that have to interface with equipment supporting multiple protocols
  - reuse of tools to verify and transform data definitions
  - benefit from a module concept, an import mechanism, a type system, version handling, ...

- All we need is a DIAMETER protocol mapping!

SMIng Protocol Mappings



Proposed DIAMETER Changes

- The DIAMETER protocol should be modified to support the following primitive base types:
  - Integer32, Unsigned32,
  - Integer64, Unsigned64,
  - Float32, Float64, Float128,
  - OctetString
- In addition, the DIAMETER protocol should support a compound base type AVP which contains an ordered list of AVPs (Group AVP).
- Additional AAA specific types can be introduced by using the SMIng type system.

Data Type Mapping

- Mapping SMIng types to DIAMETER types:
  - OctetString ⇒ OctetString
  - Integer32 ⇒ Integer32
  - Integer64 ⇒ Integer64
  - Unsigned32 ⇒ Unsigned32
  - Unsigned64 ⇒ Unsigned64
  - Float32 ⇒ Float32
  - Float64 ⇒ Float64
  - Float128 ⇒ Float128
  - Enumeration ⇒ Integer32
  - Bits ⇒ OctetString
- The SMIng Pointer type is not mapped to a DIAMETER type.

Classes versus AVPs and Messages

- AVPs are protocol specific mappings of SMIng classes.
  - Classes with one attribute are encoded by a single AVP which uses the DIAMETER type that is consistent with the attribute´s SMIng base type.
  - Classes with multiple attributes are encoded by a Group AVP which contains AVPs for each attribute contained in the SMIng class.
- DIAMETER Messages are defined by listing the AVPs included in the message.
  - SMIng extensions could be provided to formally define multiplicity and whether AVPs are mandatory or optional.

Example

```
module SMING-DIAMETER {

  import IRTF-NMRG-SMING    (Utf8String);
  import SMING-DIAMETER-EXT (diameter);

  organization   "IETF AAA Working Group";
  contact        "J. Schoenwaelder";
  description    "DIAMETER core definitions.";
  revision {
      date         "2000-12-13";
      description "First revision, RFC XXXX";
  };
```

Example (cont.)

```
typedef NetworkAccessIdentifier {
    type         Utf8String;
    description "...";
    reference   "RFC 2486";
};

class HostName {
    attribute NetworkAccessIdentifier name {
        description "...";
    };
    description "...";
};
```

Example (cont.)

```
diameter {
    avp HostNameAVP {
        code 264;  implements HostName;
    };

    avp VendorNameAVP {
        code 266;  implements VendorName;
    };

    avp FirmwareRevisionAVP {
        code 267;  implements FirmwareRevision;
    };
```

Example (cont.)

```
  msg DeviceRebootInd {
      code 257;
      includes NonceAVP;
      includes HostNameAVP;
      includes HostIPAddressAVP;
      includes VendorNameAVP;
      includes ExtensionIdAVP;
      includes FirmwareRevisionAVP;
      includes IntegrityCheckAVP;
      description "...";
  };
};
```

Further Information

- <draft-schoenw-sming-diameter-00.txt>
  (currently only available from www.diameter.org)

- SMIng working group meeting (next meeting slot)

Questions?