Babel Information Model
draft-stark-babel-information-model-00

babel WG

Barbara Stark (AT&T)

Modeling Philosophy for 00 Draft

- Sources:
  - Create parameters and objects for constants, variables, and tables mentioned in RFC 6126, except those that seem to ephemeral.
  - Placeholder for RFC 7298 (I ran out of time to analyze what was needed for that).
  - Did not include any other drafts – intention would be to include working group drafts when adopted.

- Notation:
  - Taken from draft-ietf-lmap-information-model-10

- State vs. Configuration:
  - Did not try to separate state reporting and configuration.
  - Very little configuration should be supported. Most of what could be included in a model would be state.
  - Some parameters MAY be configurable at discretion of implementation.

Object Hierarchy

babel-information

- constants object
  - interfaces object(s) [includes parameter for reference to data model interface object]
  - neighbors object(s)
  - csa object(s)
- sources object(s)
- routes object(s)

[did not include Table of Pending Requests, since that seems like it would generally be empty or populated with short-lived info]

Suggestions for Potentially (“MAY”) Configurable Parameters

UDP Port

Default hello interval for lossy interfaces

Default hello interval for lossless interfaces

Threshold for requesting acknowledgements on an interface

Whether or not an interface should be considered “lossy”

External input to cost of link (MUST be configurable if implemented, but optional to implement)

Questions or Comments?

Backup: Fitting into a YANG data model?

draft-ietf-netmod-routing-cfg-22 is “A YANG Data Model for Routing Management”. Section 5.3.2 describes what needs to be done to add a new control plane protocol, with Appendix C providing an example for RIP. The way the Babel information is organized in babel-information-model draft does not seem consistent with how routing-cfg wants it organized.

```
+--rw routing
   +--rw router-id?
   +--rw control-plane-protocols
   |  +--rw control-plane-protocol* [babel here?]
   |     +--rw type
   |     +--rw name
   |     +--rw description?
   |     +--rw static-routes
   |        +--rw v6ur:ipv6
   |        |     ...
   |        +--rw v4ur:ipv4
   |              ...
   +--rw ribs
     +--rw rib* [name]
        +--rw name
        +--rw address-family?
        +--rw description?
```

Figure 1: Configuration data hierarchy.

```
+--ro routing-state
   +--ro router-id?
   +--ro interfaces
   |  +--ro interface*
   +--ro control-plane-protocols
   |  +--ro control-plane-protocol* [babel here?]
   |     +--ro type
   |     +--ro name
   +--ro ribs
     +--ro rib* [name]
        +--ro name
        +--ro address-family
        +--ro default-rib?
        +--ro routes
           +--ro route*
                 ...
```

Figure 2: State data hierarchy.

Backup: Fitting into a TR-069 Data Model?

tr-181-2-10-0.html has the TR-069 data model.

For comparison: See tr-181-2-10-0.html#D.Device:2.Device.Routing.RIP. for the manner in which RIP is modeled. The RIP model is very small and just allows for very minimal configuration (enabling / disabling the protocol on an interface, and whether or not to send and/or receive RIP route advertisements). There is no state reporting.