

UNUM: NETMOD Unified Model

Martin Swany
U. Delaware

Direction

- Chairs asked for: presentations on "first takes" and "outlines" on how such documents could/should look
- Charter calls for "...a set of core YANG data models"
 - Presumably models of network elements and characteristics that can be reused and extended
- draft-ietf-netmod-arch-10 says about YANG
 - Adding "features" and replacing "third normal form" with a natural data hierarchy should reduce complexity.

Brief Background

- perfSONAR is a web services based network measurement and monitoring system
 - Efforts in the Grid Forum
 - Widely used in R&E networks
- We developed a network topology schema
 - Initially to codify and reuse measurement “subjects”
 - Capture the relationship among those elements
 - Eventually extended to support dynamic virtual “circuit” allocation in Internet2, ESnet, GEANT, etc.
 - Now investigating it for resource description and allocation in GENI

Network Description Schema

- YANG seems to be a solution to many of the problems that we have had with Relax NG
- I started working with YANG recently to produce a revised version of the perfSONAR topology schema (called UNIS)

Schema Approach

- Define basic network elements: node, port, link, etc.
- Reuse element names, vary the namespace for layer-specific properties
 - Also serves as the basis for other extensions (vendor or domain-specific)
- Example: ethernet:interface and ip:interface have distinct attributes (and potentially distinct counters)
 - Multiple ip:interface atop single ethernet:interface
- Two ways to look at it:
 - Separate but related elements
 - “Outer join” to create single entity with attributes from both namespaces

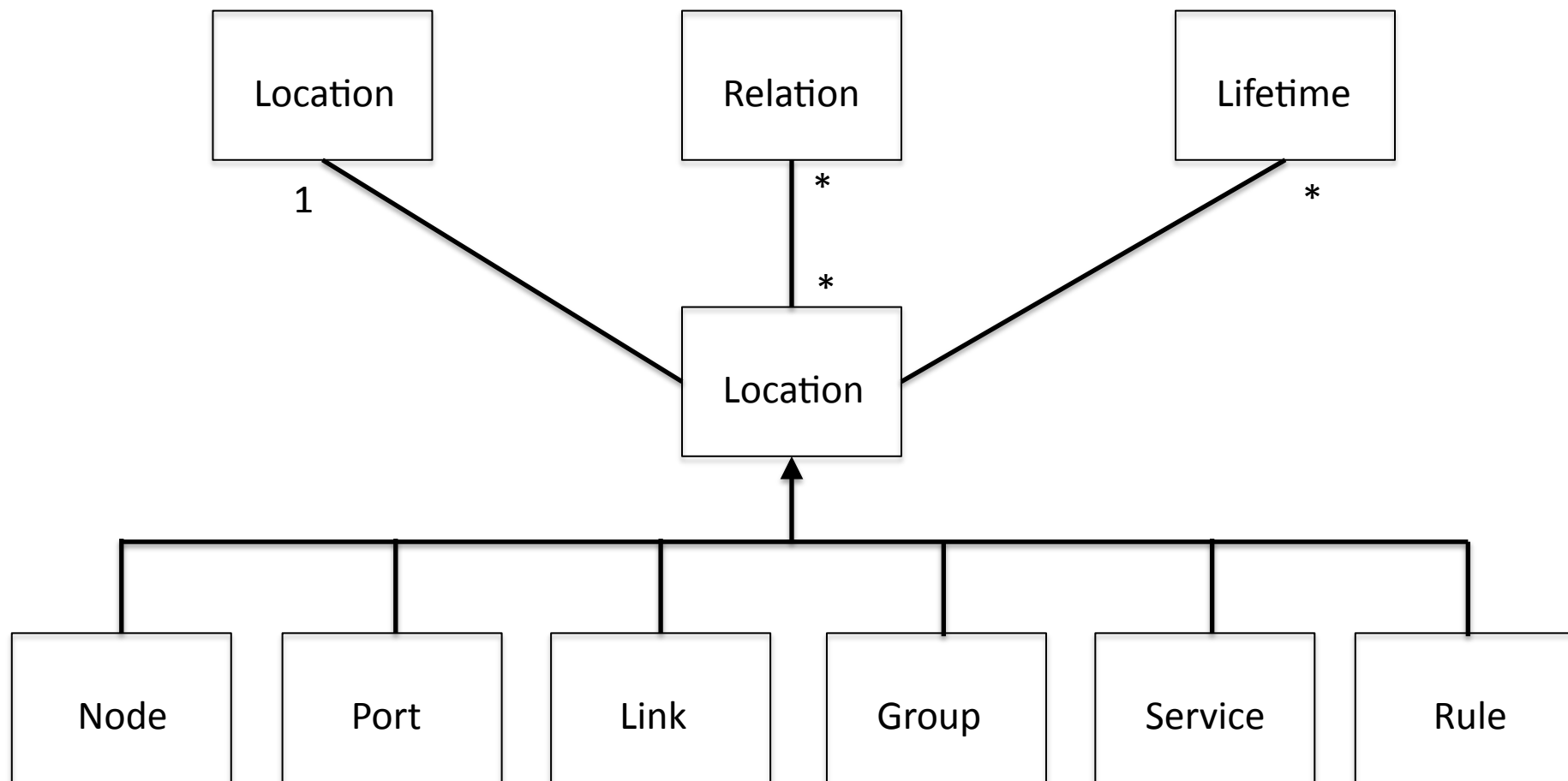
Existing Core Model

- Network Object (abstract base class)
 - Has Identifier, Lifetime, is the target of a Relation
- Node
- Port (Interface)
- Link
- Service
- Groups: Network, Path, Domain

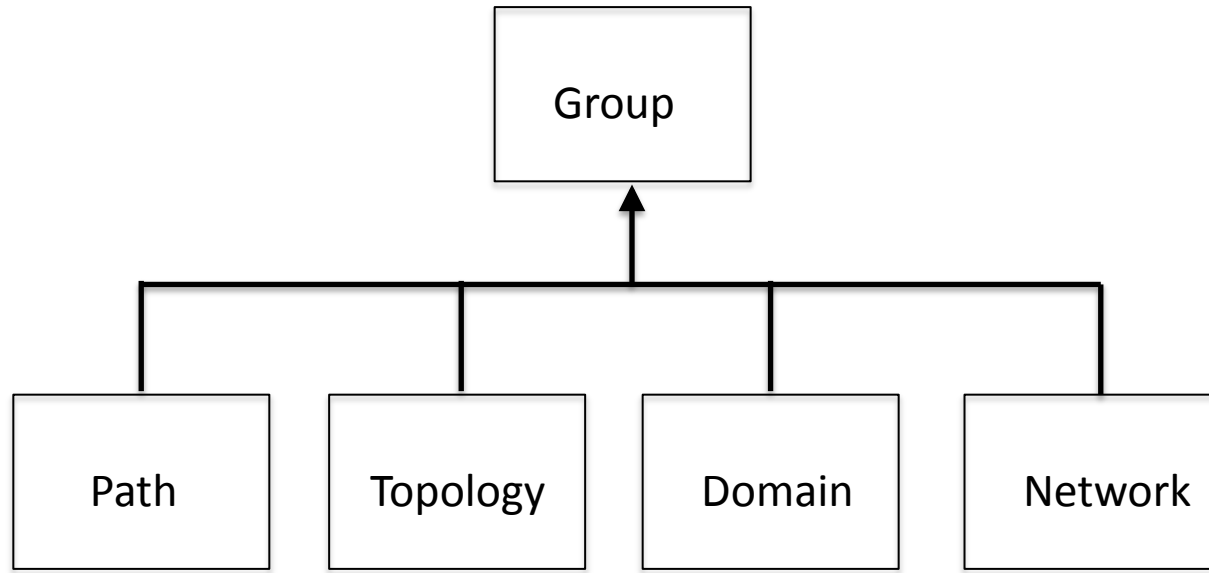
Additions to the Model

- Rule
 - Models routes, but other policy as well
 - N-tuple of all matchable header fields: (src/dst IP, src/dst MAC addr, etc.) followed by an action (send to output port, IP next hop, increment counter, drop, etc.)
- Very illustrative of how we need to reduce things to their most basic components, but preserve the ability to refer to them in shorter, more familiar forms

Model Relationships



Groups



Issues

- This model is broader than the current NETMOD mandate
 - If we can apply the broader model without impacting the current NETMOD needs, then good
 - We have to balance between over-specification, which slows and complicates this effort, and overly-narrow design decisions, which can make expansion more difficult later
- Can this work serve as a starting point for the NETMOD general models?

Issues - Modeling

- How to join basic elements, create views
- How best to manage inheritance (if at all?)