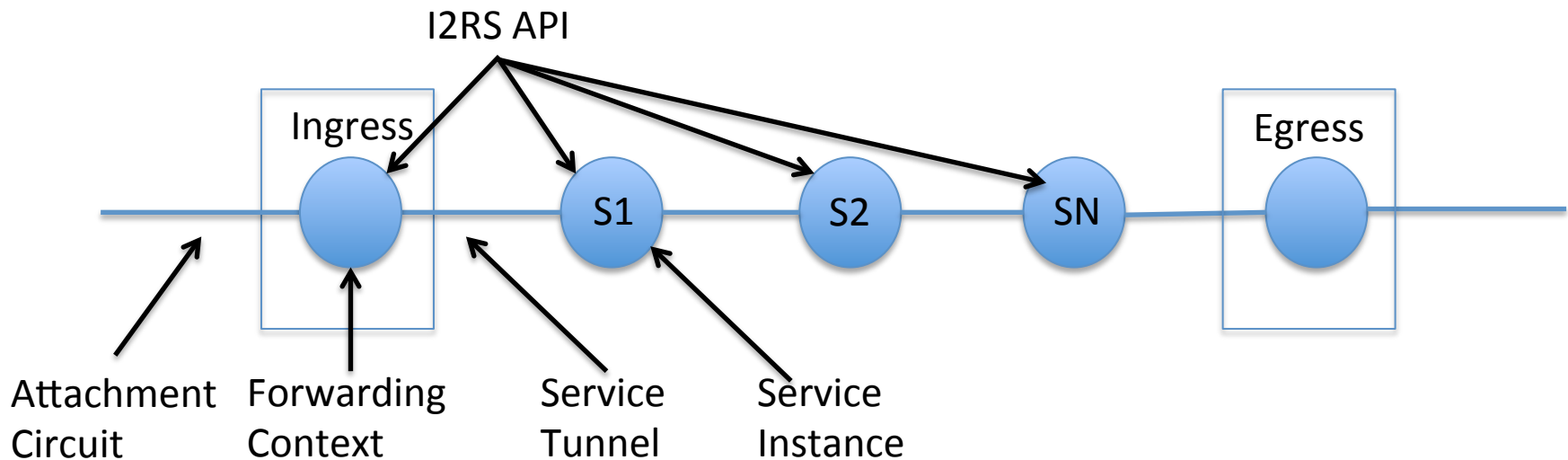


# I2RS Topology Use Cases

Cases 1, 6, 8.

# General Description

- Mechanism to learn and to provision:
  1. customer topology
  2. service topology
  3. service chaining



# Customer Topology

- Interfaces
- Network Service and Attributes

# Service Topology

- Service node per service
  - Liveliness
  - Performance (monitoring and stats)
- Service instantiation
  - Pre-built, spin up on demand, n:1 vs 1:1
- Service node capacity
  - CPU, B/W, policies, ACLs etc.

# Service Chaining

- Service chaining / forwarding
- Classification of packets
- Service profile
- 2 models:
  1. Create whole chain at ingress (requires metadata in packet header?)
  2. Recursive model where provision each link in the chain
- Need to support service multiplexing/demux
  - Mux example – single firewall for multiple users
  - Demux example - stateful load balance etc.

# Scope and Scale

- Applicable to SP edge network and to NFV in the data-center.
- Applicable to service-aware network elements (PE routers, NFV elements)
- 3 types of operations:
  - Discovery/publishing – when resources change. Can be learned dynamically or stored in centralized database.
  - Configuration – when setting up a service
  - Monitoring/statistics – very high frequency (should these be passed over I2RS API?)

# What's Different for I2RS

- Via CLI and/or AAA systems (RADIUS or DIAMETER)
- Data/information model standardization
- Standardized APIs with rich semantics
- I2RS solution requires:
  - Data Model / Information Model
  - APIs
  - Provisioning / De-provisioning mechanics