Traffic Engineering Service

- Manipulation of traffic to reflect network intent (mission requirements)
  - Traffic prioritization
  - Traffic shaping
  - QoS signaling
- Primary focus of a number of protocols such as OSPF-TE

QoS
- Latency
- Jitter
- Bandwidth

Traffic Engineering Objectives

Reliability
- Path stability
- Packet loss

Security
- C-I-A
- Assurance
- Compliance
Elements of Traffic Engineering

- Load balancing
- Load distribution
- Different objective functions
- Broadcast, multicast, and unicast
- Estimation and measurements
- Priority signaling
Elements of Traffic Engineering

- Load balancing
- **Load distribution**
- Different objective functions
- Broadcast, multicast, and unicast
- Estimation and measurements
- Priority signaling
Elements of Traffic Engineering

• Load balancing
• Load distribution
• **Different objective functions**
• Broadcast, multicast, and unicast
• Estimation and measurements
• Priority signaling
Elements of Traffic Engineering

- Load balancing
- Load distribution
- Different objective functions
- **Broadcast, multicast, and unicast**
- Estimation and measurements
- Priority signaling
Elements of Traffic Engineering

- Load balancing
- Load distribution
- Different objective functions
- Broadcast, multicast, and unicast
- Estimation and measurements
- Priority signaling

Diagram:
- Link reliability
- Link latency
- Link data rate

Traffic demand:
- Measurement
- Estimation
- Request

Node S:
- 250K, 1ms

Node D:
- 500K, 10ms
Elements of Traffic Engineering

- Load balancing
- Load distribution
- Different objective functions
- Broadcast, multicast, and unicast
- Estimation and measurements
- Priority signaling

Differentiated Service Code Point (DSCP)

- 500K, 10ms
- 250K, 1ms

Drop Probability
- 3 Bits
- DSCP
Who signals traffic prioritization?

<table>
<thead>
<tr>
<th>Application does it</th>
<th>• Tragedy of the commons</th>
</tr>
</thead>
<tbody>
<tr>
<td>In-line device that prioritizes traffic</td>
<td>• Loaded with a static (non-adaptive) asynchronous traffic prioritization policy</td>
</tr>
<tr>
<td></td>
<td>• Some of them perform caching functions</td>
</tr>
</tbody>
</table>
Today’s model

- Today’s traffic engineering model is anchored off of end-point semantics and not content
  - Anchored off of TCP/IP header information
  - Per-flow granularity

![Diagram of traffic engineering model with DSCP marked packets, Traffic Engineering, and Computed Paths]

- 500K, 10ms
- 250K, 1ms
Why is this approach limiting?

- QoS must reflect content not end-point semantics
  - Different chunks of data may carry information with different QoS requirements
  - Per-chunk granularity as opposed to per flow granularity
- The dynamics and challenges of tactical networks mandate an adaptive approach

Example application: CoT

- Track info with different content

```
<?xml version="1.0" standalone="yes"?>
<event how="m-r-p" opex="e-JCIDENT" qos="0-6-c" stale="2016-04-12T16:47:42.722Z" start="2016-04-12T16:45:42.722Z" time="2016-04-12T16:45:42.722Z" type="a-f-G-I-r-h" uuid="Debug.032" version="2.0"/>
```

- Ground hostile target
- Ground friend target
- Reliability
- Low latency

Why is this approach limiting?

- Ground hostile target
- Ground friend target
- Reliability
NDN functions can provide the foundation for strong Traffic Engineering

- Producer or consumer driven QoS signaling
- Caching policy can retain latency-constrained data longer
- Forwarding strategy can select forwarding paths based on QoS req.
- NDN Sync can be used to synchronize policies
QoS signaling: producer driven

• Explicit QoS signaling (embedded in the name)
  • /ndn/ucla/tutorials/milcom2019/QoS/latency-max=2
  • Same CoT data can be serviced at different QoS

• Implicit QoS signaling (inferred from the name)
  • /mission-a/CoT/ground/hostile → reliability and low latency
  • /mission-a/CoT/ground/friendly → reliability
  • CoT data chunks can carry different QoS signals

• Forwarding strategy will attempt to honor both QoS signaling approaches
QoS signaling: driven by consumer

- QoS signaling can be part of the ApplicationParameters of an Interest packet
  - /mission-a/CoT/ground/friendly
  - Forwarders will attempt to honor the embedded QoS signal

```
Interest = INTEREST-TYPE TLV-LENGTH
  Name
  [CanBePrefix]
  [MustBeFresh]
  [ForwardingHint]
  [Nonce]
  [InterestLifetime]
  [HopLimit]
  {ApplicationParameters [InterestSignature]}
```
• Forwarding strategies are equipped to provide very strong TE mechanisms
  • Can derive link (or path) reliability through the Interest ←→ Data packet feedback loop.
  • Can derive link (or path) latency through the Interest ←→ Data packet feedback loop.
  • NDN-LP provides reliability and latency information between NDN nodes

• Can also measure traffic demand from Interest rate/class of service
• Forwarding decision can be based on QoS signaling from consumer and/or producer

• Forwarding strategy can continue to leverage the in-band Interest/Data feedback loop to measure demand as well as link (or path) characteristics and adjust FIB accordingly
• Load balancing is typically not implemented because of the impact of putting TCP segments from the same flow on two different links (different latencies can cause out-of-order delivery)

Forwarding strategies can drive load balancing
• An NDN forwarding strategy can be devised to accomplish load balancing (and/or load distribution)
• Forwarding strategies can easily service unicast, multicast, as well as anycast (anyone has this data type) request

Examples:
• /mission-a/CoT/ground/hostile
  • Broadcast since this information is important to deliver to all nodes fast
  • Other nodes can enable opportunistic caching for this prefix

• /mission-a/CoT/ground/friendly/latency=x
  • Send on link with latency less than x/2.

• /mission-a/ftp/file-111219
  • Send on highest capacity link
QoS-aware Caching strategy

• Insertion and eviction policies can factor in QoS
  • Retaining chunks that are latency-sensitive in the network (e.g. hostile tracks)
  • Factor in the value of the information over time
  • Factor in reliability of links where the data is typically serviced

Example policies:
• Retain only latest sequence of chunks where older sequence numbers do not matter
  • PLI information
• Retain chunks longest when the only path to retrieve them is through an unreliable link
  • Denied, Disrupted, Intermittent, and Limited bandwidth (DDIL) links
Policy synchronization

• QoS policy dataset can be retained and synchronized across the network using Sync protocols

• Appropriate security measures can be provided
  • Access control (who is authorized to a read/write policies)
  • Authentication (tie specific policies to specific administrative domains)
  • Confidentiality
  • Integrity (built-in by default)
  • Policy override (domain policy overrides local policy)