A Modest Approach to Active Networking

CANE5
Outline

Active Networking: What?

Active Networking: Why?

Active Networking: How?
Active networking is:

- Putting the ‘work’ in network.
- Providing a programmable meta-level interface to the network.
- The placement of user-controllable computing capabilities inside the communication subnetwork.
- Happening.

Active Networking: What
To speed the deployment of new/enhanced services.

- To gain the benefits of being able to put application knowledge and network knowledge together in the same place and time.

- It’s already happening in an ad hoc way. Why not provide an open, well-designed platform?

- Why: Active Networking
Application-Specific Congestion Control

There will always be applications that prefer to use best-effort service and dynamically adjust rate.

Sender adaptation has worked well in Internet.

Sender adaptation has well-known challenges:
- Difficulty of detecting congestion
- Time required to detect congestion and adjust rate
- Difficulty determining increase in available bandwidth

Observation: Application knows how to adapt to congestion, while network knows when to adapt. Move advice about adaptation into network.

Claims:
- Application-Specific Congestion Control
Based on triggers that indicate congestion control should take place, how state is examined for advice about how to reduce quantity of data. Based on triggers that indicate congestion control should take place, how state is examined for advice about how to reduce quantity of data.

Operation:

Evaluation: Active IP option, ATM links
Reduction methods: Compress, transform, discard
Example Result: Selective Frame Discard

GOP receives 148 out of 220 frames (12 out of 23 I frames)

PPD receives 168 out of 220 frames (22 out of 23 I frames)
What about the End-to-End Argument?

 applications using it.

 - The cost of providing a service is paid only by those
   applications using it.

 - The cost of the interface is a one-time cost (infrastructure).

 - It allows users more precise selection of services.

 - It provides a generic interface, available to all users.

 Argument: "Active networking is consistent with the end-to-end
 function."

 Layered system closer to the application that uses the
 "...provides a rationale for moving a function upward in a
 claim: Active networking is consistent with the end-to-end
 argument.

 #Claim Active networking is consistent with the end-to-end
 argument:

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.

 #5C/././.provides a rationale for moving a function upward in a
 layer.
Each approach can emulate the other.

- Packets indicate function(s), supply parameters
- Active node supports fixed set of active functions

Menu Approach:
- MIT Capsules, Penn-Bellcore Switchware
- User-network interface is programming language
- Packets carry programs; active node runs interpreter

Language Approach: How?
1. Generic packet-processing model

- analogue of process + stdin/stdout in UNIX
- defines basic forms of service composition

2. Higher-Level Services: packet forwarding, install state, ...
- offer various forms of customizability, from selectable policies to language interpreters
- analogue of system calls in UNIX

3. Primitive Elements: output queues, state store, etc.
- analogue of system calls in UNIX
Advantages of the Approach

- Migration path (backward compatibility).
- Optimize generic processing (e.g., cryptographic ops).
- Constrained functionality simplifies composition.
- Simple, stable network interface.
- Function id, parameters, security object, [body]