CANEs:
A Modest Approach to Active Networking

Ken Calvert Ellen Zegura James Sterbenz

College of Computing GTE Laboratories

Georgia Tech Waltham, MA

Contributors: S. Bhattacharjee, D. Haynes

`http://www.cc.gatech.edu/projects/canes`

IEEE Computer Communications Workshop

8–10 September 1997

Outline

- Active Networking: What?
- Active Networking: Why?
- Active Networking: How?

Active Networking: What

Active networking is...

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

Active Networking: Why?

- To speed the deployment of new/enhanced services.
- It's already happening in an ad hoc way. Why not provide an open, well-designed platform?
- To gain the benefits of being able to put application knowledge and network knowledge together in the same place and time.

Application-Specific Congestion Control

Claims:

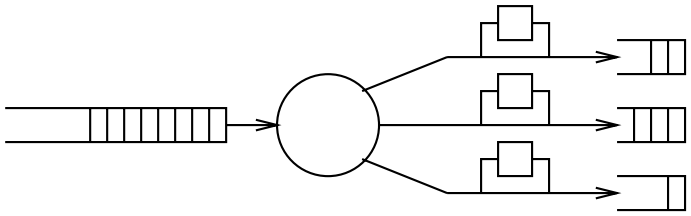
- There will always be applications that prefer to use best-effort service and dynamically adjust rate.
- Sender adaptation model has worked well in Internet.
- Sender adaptation has well known challenges:
  - difficulty of detecting congestion
  - time required to detect congestion and adjust rate
  - 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.

Generic Programmable Congestion Control

Operation:

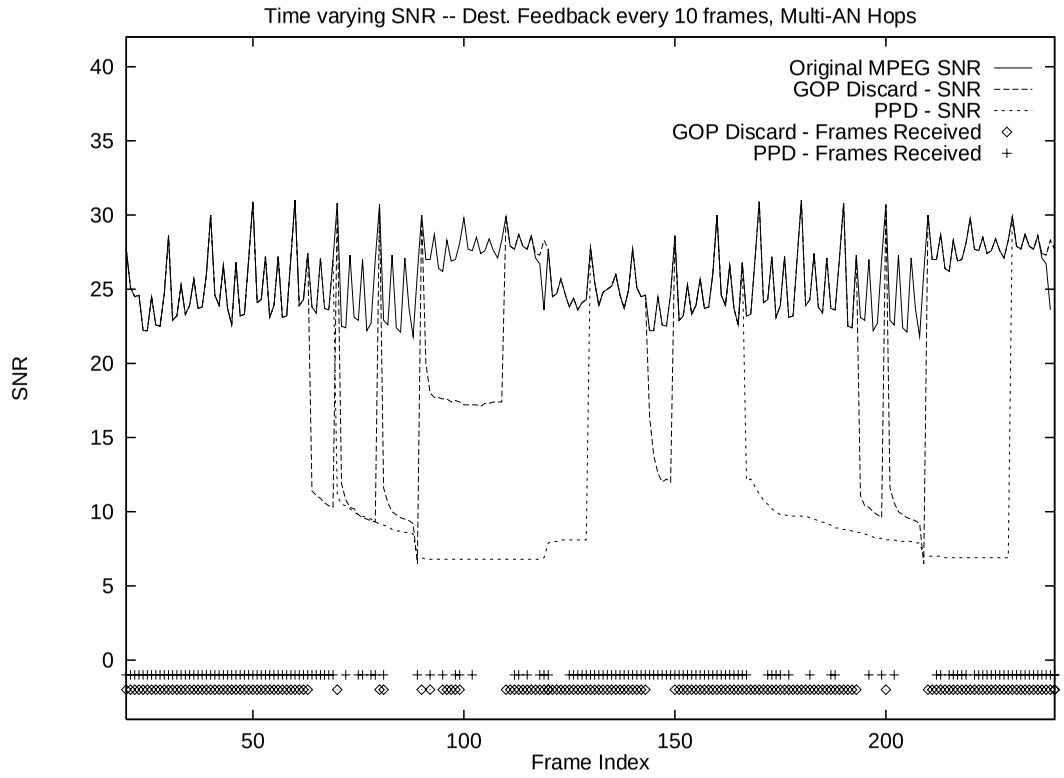
Based on triggers that indicate congestion control should take place, flow state is examined for advice about how to reduce quantity of data.



Reduction methods: Compress, transform, discard

Evaluation: Active IP option, ATM links

Example Result: Selective Frame Discard



- GOP receives 168 out of 220 frames (22 out of 23 I frames)
- PPD receives 148 out of 220 frames (12 out of 23 I frames)

What about the End-to-End Argument?

"...provides a rationale for moving a function upward in a layered system closer to the application that uses the function."

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

- It provides a generic interface, available to all users.
- It allows users more precise selection of services.
- The cost of the interface is a one-time cost (infrastructure).
- The cost of providing a service is paid only by those applications using it.

Active Networking: How?

Language Approach:

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

Menu Approach:

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

Each approach can emulate the other.

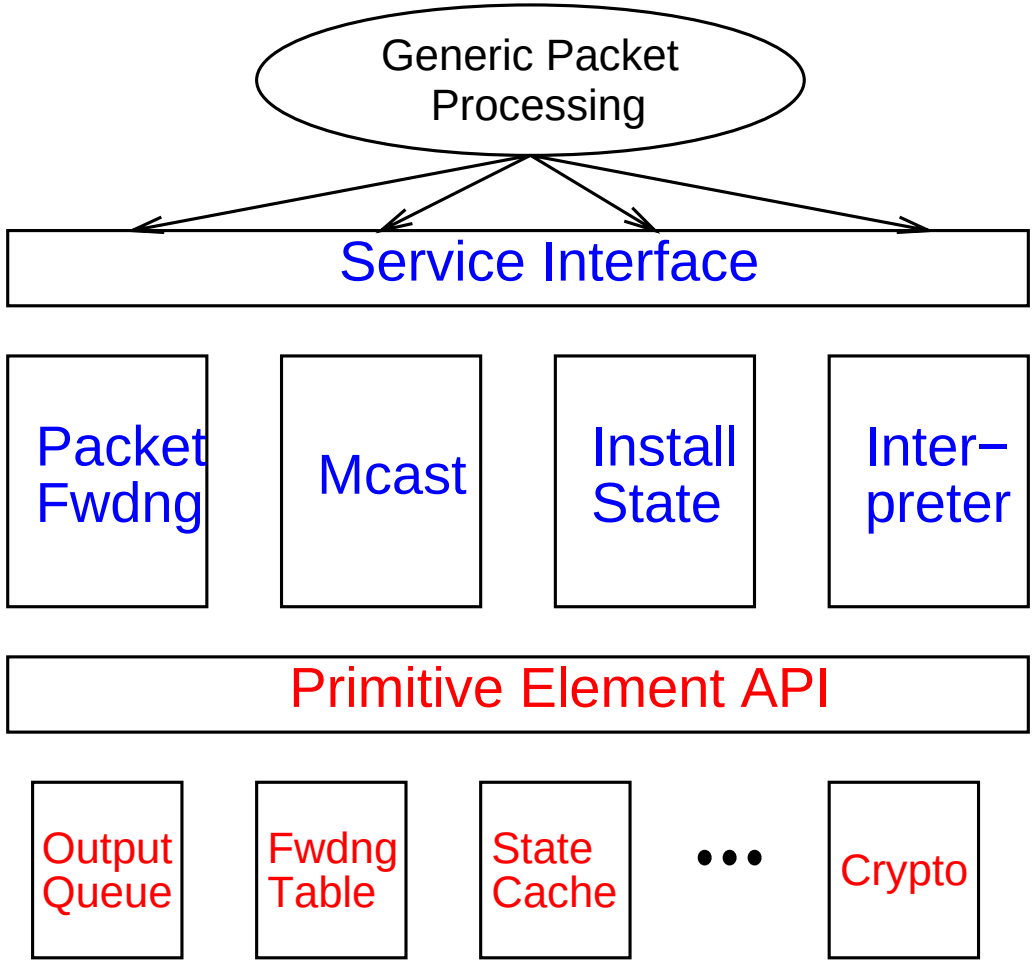
Composable Active Network Elements

1. **Generic packet-processing model**
   - defines basic forms of service composition
   - analogue of process + stdin/stdout in UNIX

2. **Higher-Level Services:** packet forwarding, install state, ...
   - offer various forms of customizability, from selectable policies to language interpreters

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

Service Architecture



Advantages of the Approach

- Optimize generic processing (e.g. cryptographic ops).
- Constrained functionality simplifies composition.
- Simple, stable network interface:

  function id, parameters, security object, [body]

- Migration path (backward compatibility).