DARPA Discussions, 10/3/97

Jonathan M. Smith

University of Pennsylvania

Agenda Items:

- Protocol Boosters, relationship with A.N.
- Dissemination in Europe
- Secure Active Network Element (SANE)
- Technical Problem Areas for A.N.
- Program Management Challenges
- What's coming next from Penn

Protocol Boosters and Active Networks

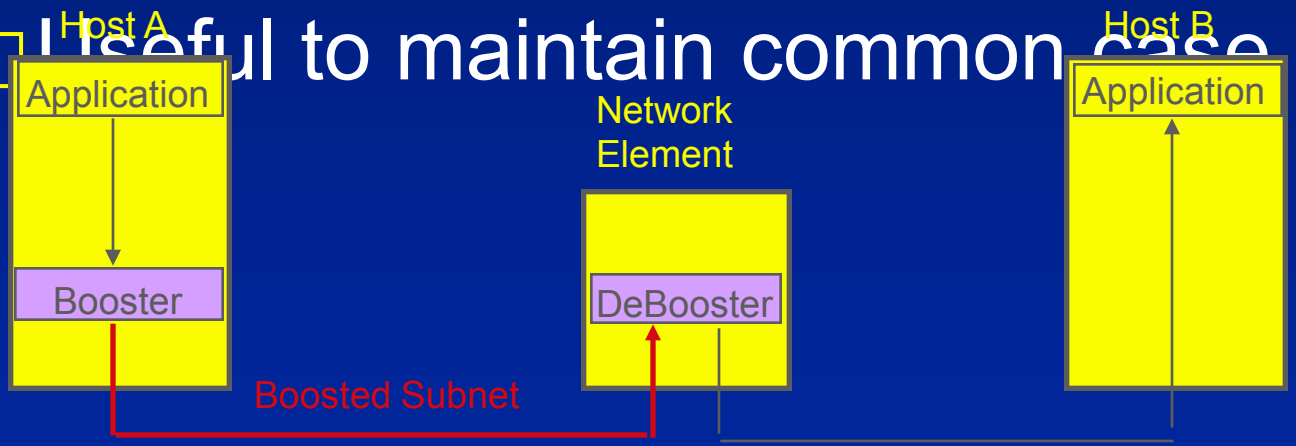
Design Methodology versus Infrastructure

Protocol Design: Current Methodology

- Pessimistic Design Style
  - » Assume worst-case
  - » Pare away functions to get “fast-path”
- Optimizations Fragile
  - » Environment Changes (WWW)
  - » Common Cases Change (delay, loss, …)
  - » Things can break BADLY! (try at home :-)

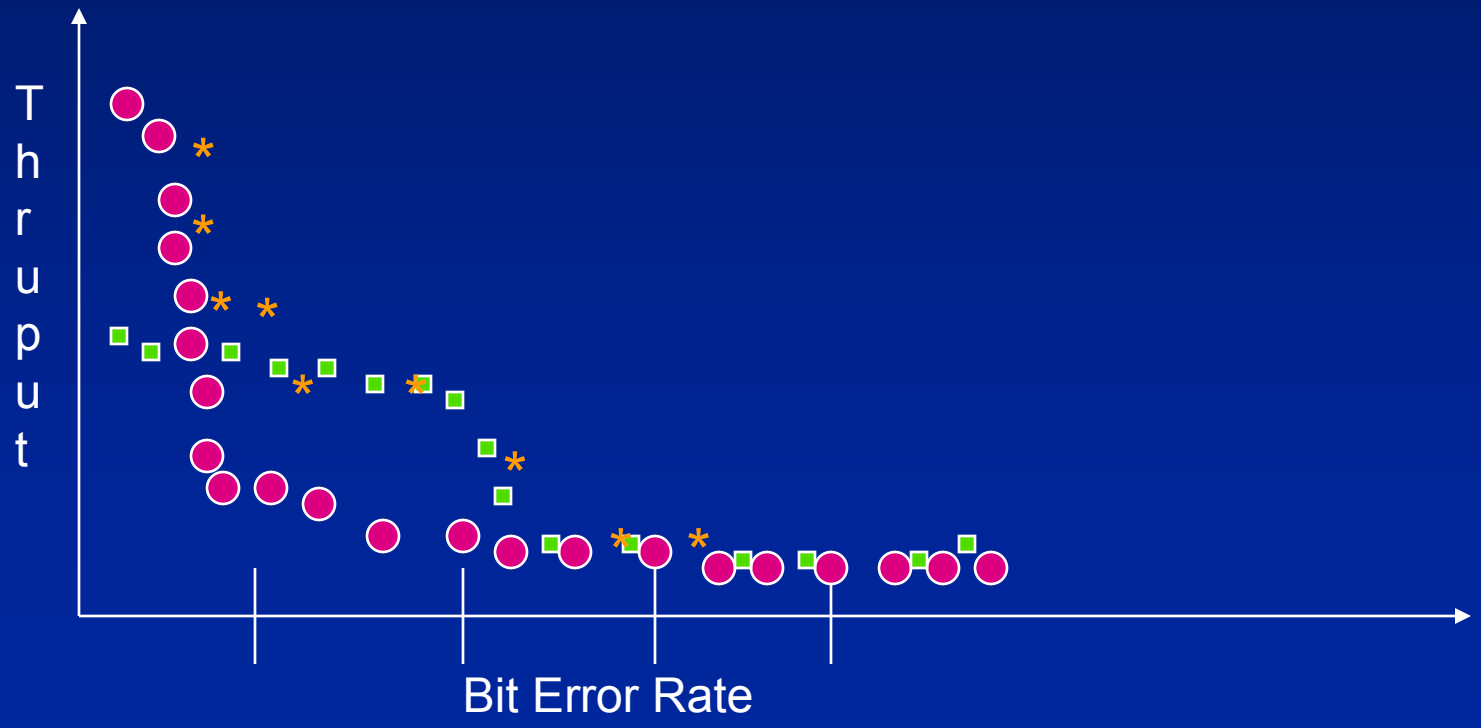
Protocol Boosters

- Protocol Elements added ‘‘as-needed’’
- Example of “optimistic” design method
- Useful to maintain common case



Performance Potential:

- Thruput: TCP, TCP/FEC, Hybrid



Examples (and leadin to A.N..)

- Implemented over IP on FreeBSD
  - » Encryption + Compression Boosters
- FEC Booster at Bellcore
- Hardware Support: The P4*
- Q: What's the network infrastructure needed to support this idea & others???

*see http://www.cis.upenn.edu/~boosters/boosters.html

European Dissemination

4/3/97-8/31/97

Strategic Goal: Enlist others to Solve Hard Problems

- Formal Methods: Talk to Milner's group
  - Pi-calculus to specify distributed behavior
  - Need for first-class time types
  - Integration with mobile work (e.g., Cardelli)
- Protocol Boosters and A.N.
  - HIPPARCH '97 Invited Speech
- SwitchWare and Network Evolution
  - U.C.L., Lancs, Sussex, Glasgow, BT Labs

Possible Follow-ons

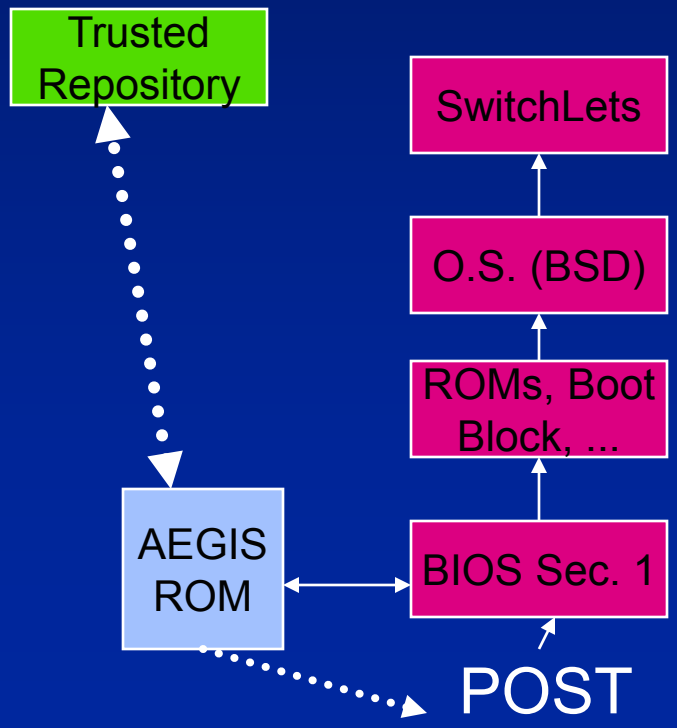
- Lancs and Sussex: EPSRC $$ for A.N.
- Cambridge DCAN project
  - » Restrict Programmability to Admin. Plane
- Cambridge Nemesis project
  - » Ideal for SwitchWare approach
  - » Investigating collaboration
    - – upcoming BAAs???

Secure Active Network Element (SANE)

From Bootstrap to Operation

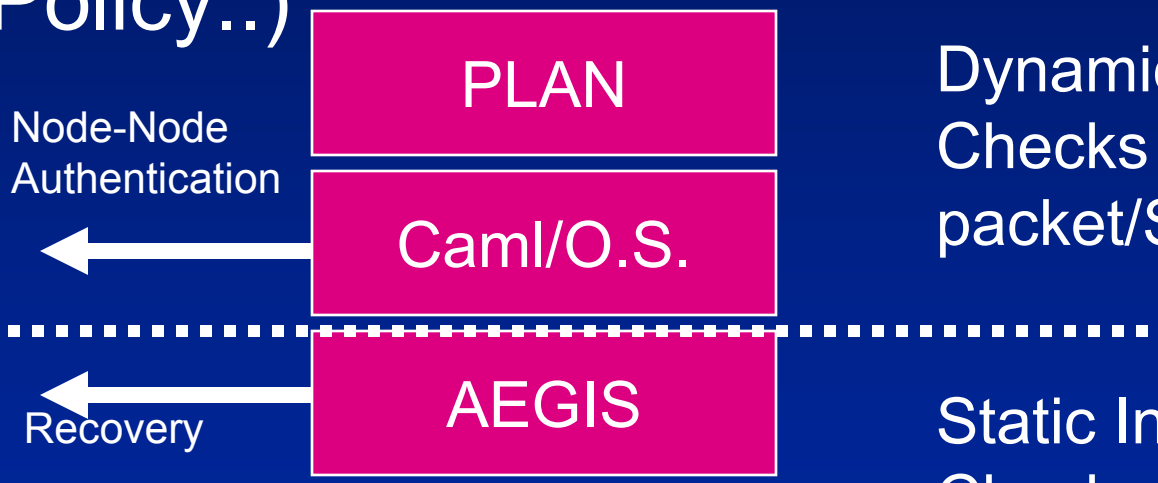
AEGIS Secure Bootstrap

- Integrity Guarantees for Dynamic Integrity Checking



Secure Active Network Element (SANE)

- "Trust, but Verify" (U.S. Nuclear Policy..)



See http://www.cis.upenn.edu/~waa

Penn/Bellcore SwitchWare Project: A Language-Oriented Model

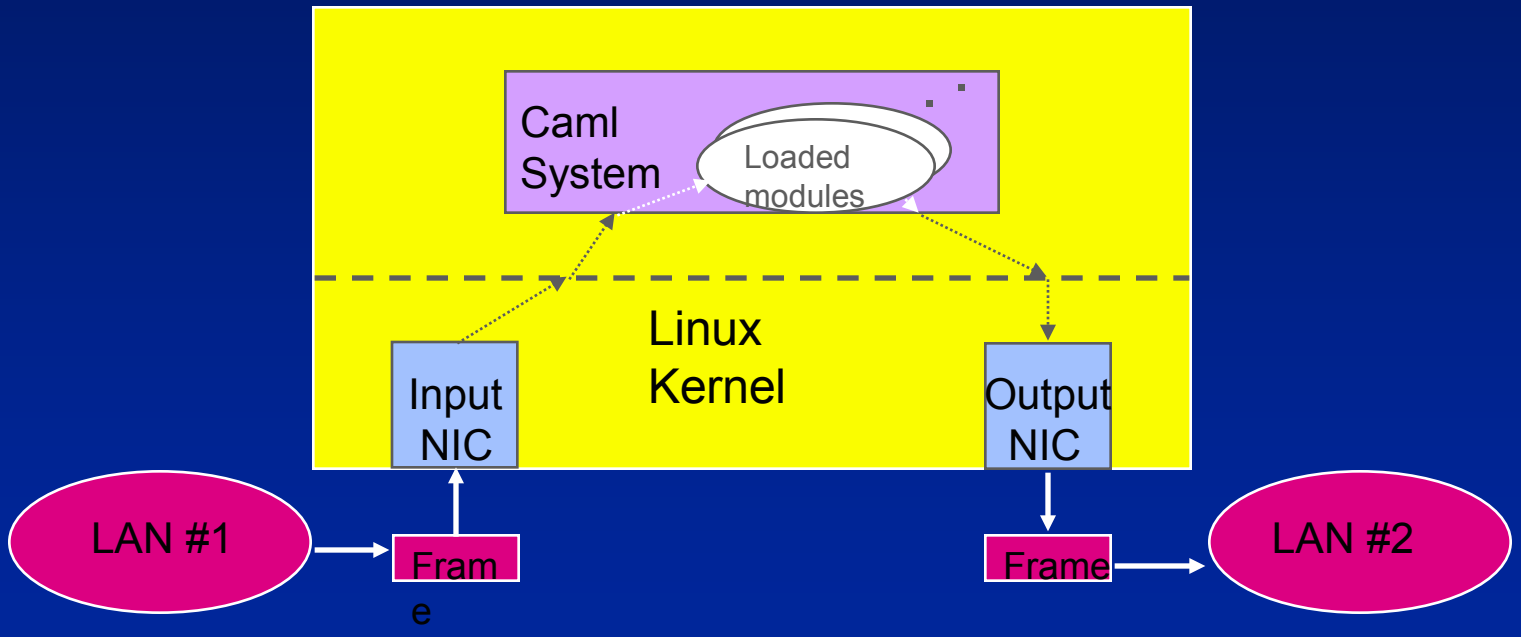
- Switchlet Language for users (SL)
  - » formal semantics restrict programs
  - » (Boosters make *fine* Switchlets :-)
  - » Prog. Language for Active Nets (PLAN)
- Wire Language for communicating (WL)
  - » formal semantics across boundaries
  - » Java or Caml bytecodes
- Infrastructure Language for Virtual Machine (IL)
  - » formal semantics supported on metal: run-time

SIGCOMM Recap

- Active Bridging Paper went over well
- A.N. Debate stimulated arguments
  - Not clearly won/lost
  - Considerable animosity about $$
  - Pointed out need for compelling applications

Current Software

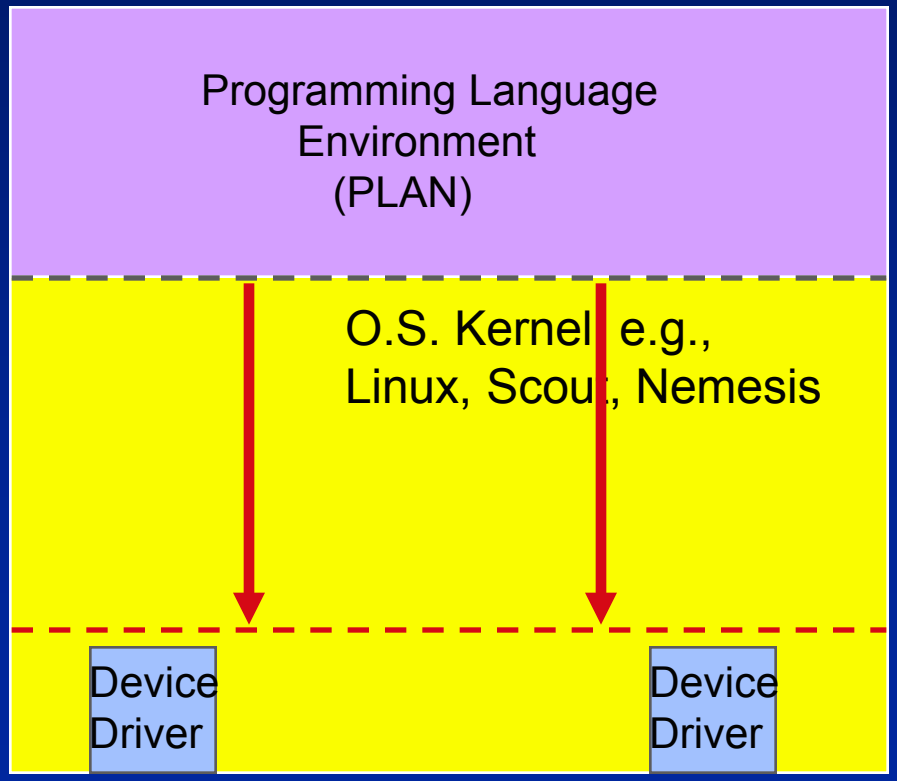
- Active Bridging



See http://oilhead.cis.upenn.edu/~salex

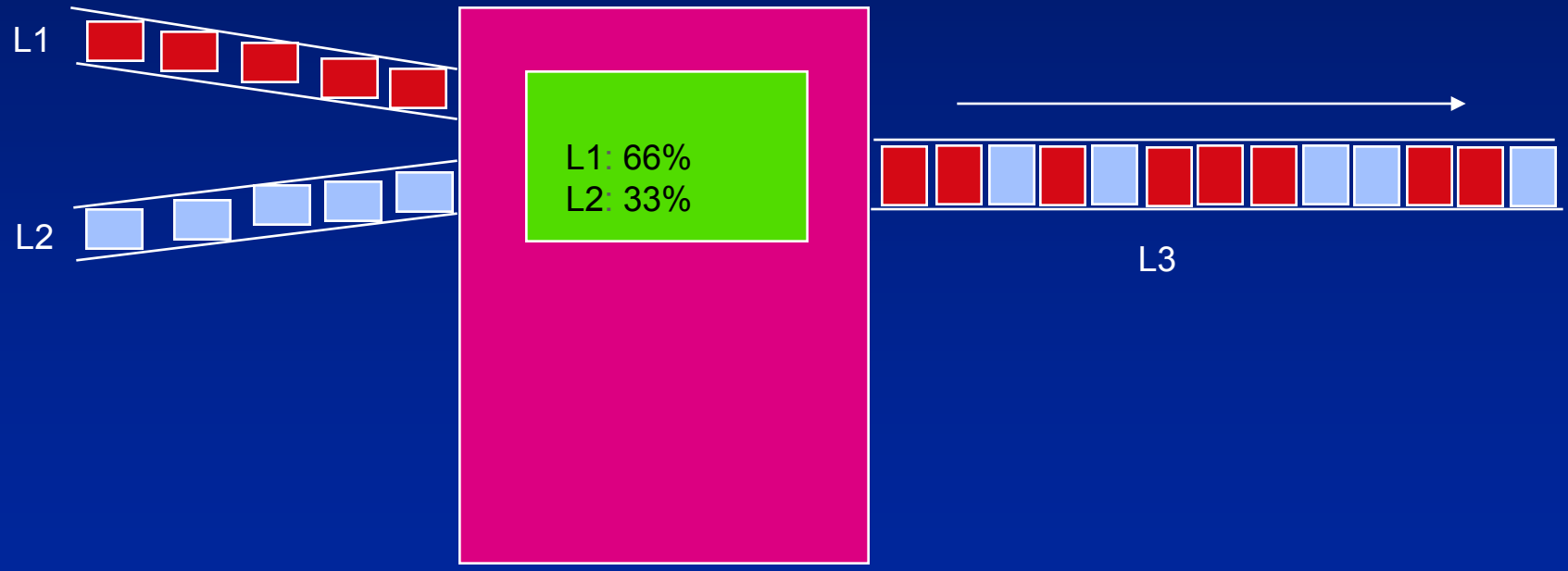
Protection of Resources?

- Dynamic versus Static Restrictions?



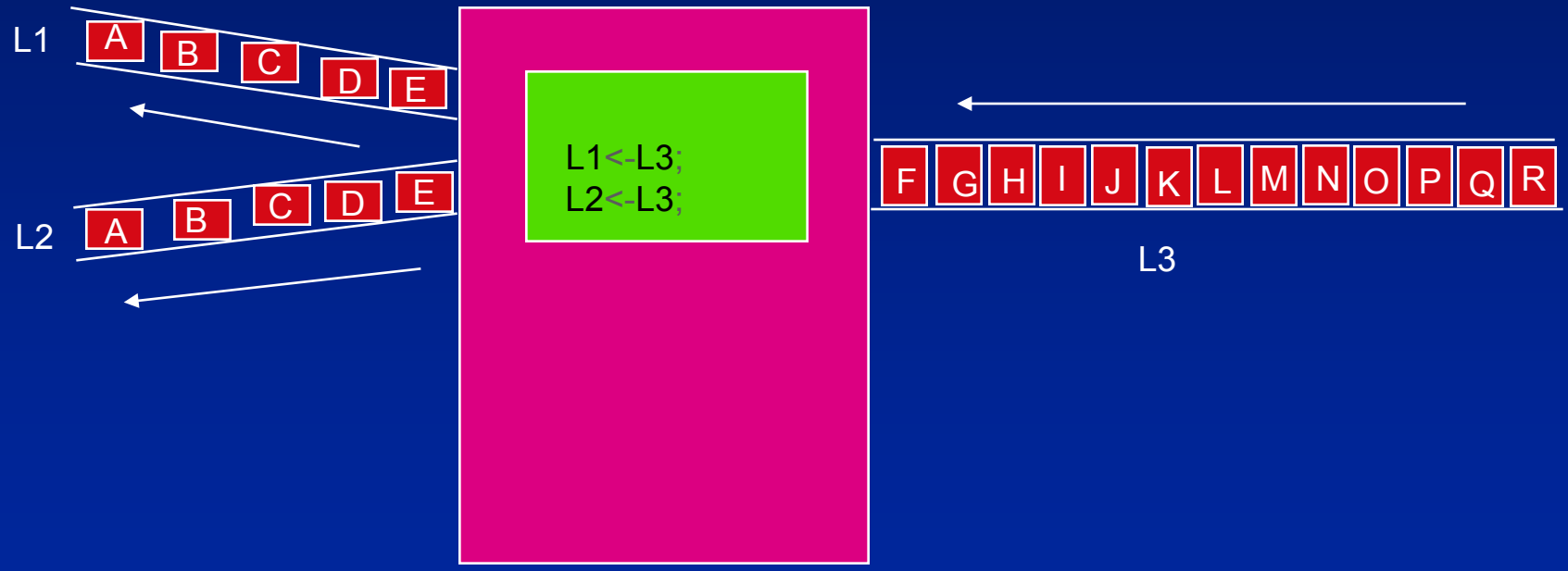
Example Problem #1: MUX

- Want to assign L3 bandwidth 66%/33%



Example #2: Multicast

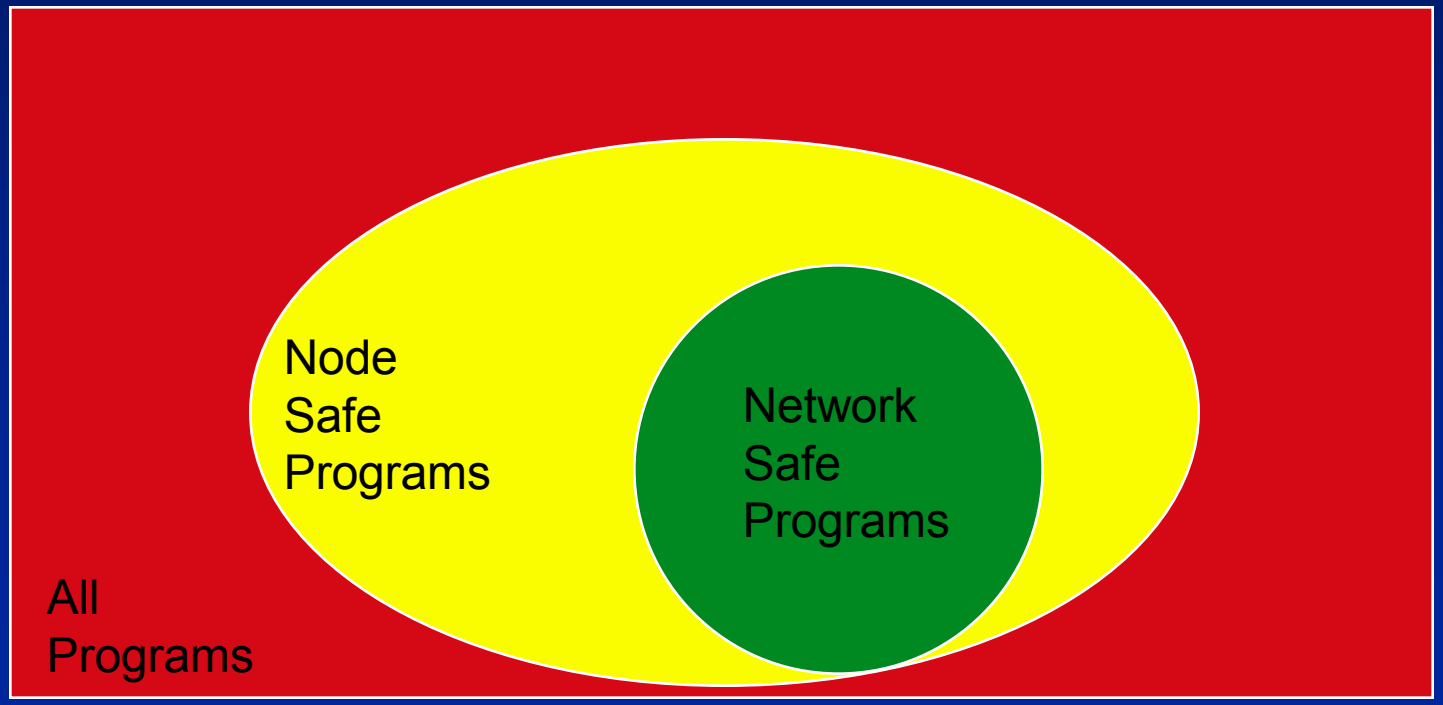
- Program copies L3 (in) to L1, L2 (out)



- Is this Program “safe”?

Restricting Programs

- Node safe versus network safe



Model->Modules->Actions

- Syntax, Semantics, Node vs. Network
- Example: Securing a Network