LIANE—Composition for Active Networks

Samrat Bhattacharjee
Ken Calvert
Ellen Zegura

Networking and Telecommunications Group
College of Computing
Georgia Institute of Technology
Atlanta, Georgia, USA

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

Sponsors: DARPA, NSF
Problem Statement

Dynamically create per-user network services whose behavior can be analyzed with respect to safety of the active network and progress of the composite computation.

• Analyze safety by formal methods
• Create dynamic services by composition and

Tension between network safety and programmability

• Network is protected
• Users can create dynamic services and

active network and progress of the composite computation behavior can be analyzed with respect to safety of the

Dynamically create per-user network services whose
Approaches towards Composition

- Turing-complete language
- Object-oriented composition
- Software bus — POLYLITH
- Event-driven framework — *micro-protocols* in *x*-kernel

![Diagram of composition mechanisms](#)

Figure 1: Composition Mechanisms
None are Perfect!

- Application structure is fixed — Object-oriented, POLYLITH
- Variable function signature — POLYLITH, micro-protocols
- Formal properties lay entirely on provable properties of code modules — in all cases

Need active software interconnect
- implements run-time composition framework
- can be used to assert provable node properties
LIANE — A formal composition framework

- Underlying program — supplied by node provider
- Processing slots — localizes user code
- Injected programs — selected, provided by users

Restrictions and obligations of injected code
Formal transformation technique to form composite program
Figure 2: Example Underlying Program with Processing Slots

(parse packet, obtain source, destination P)

select route table, caching

trace route, caching

select alternate interface

send route back

if error messages to source

else

if i is congested then

discard

else

Outputlist = ()

if Outputlist = () then

if i is congested then

select route table, caching

trace route, caching

parse packet, obtain source, destination P

if Outputlist = () then

Outputlist = (p)

route table R

lookup(p)
Used to implement AN-SIM

AN-SIM nodes simulate composition using LIANE

Ongoing work on UNITY model