XIA: Efficient Support for Evolvable Internetworking

Dongsu Han    Ashok Anand    Fahad Dogar
Boyan Li     **Hyeontaek Lim**    Michel Machado
Arvind Mukundan    Srinivasan Seshan
Aditya Akella    David G. Andersen    John W. Byers
Peter Steenkiste
IP: Narrow Waist of the Internet

Innovation both above and below IP

But what about IP?

Applications

Technology

Hyeontaek Lim / NSDI '12
Proposed Centric Networking

- Service: Serval (This NSDI!)
- Content: Named Data Networking
- Mobility: MobilityFirst
- Cloud: Nebula

Problem: Focusing on one communication type may hinder using other communication types, as occurred to IP

Can we support heterogeneous communication types on a single Internet architecture?
Future Centric Networking

• Service, content, mobility, and cloud did not receive much attention before as now

• Yet more networking styles may be useful in the future
  – E.g., DTN, wide-area multicast, ...?

Problem: Introducing additional communication types to the existing network can be very challenging

Can we support future communication types without redesigning the Internet architecture?
Legacy Router May Prevent Innovation

“I got a computer with Awesome-Networking announced in NSDI 2022! Can I use it right now?”

“Iough, we just replaced all of our routers built in 2012. Can you wait for another 10 years for new routers?”

Problem: Using a new communication type may require every legacy router in the network to be upgraded.

Can we allow using a new communication type even when the network is yet to natively support it?
XIA’s Goals and Design Pillars

“Principal types”
- Support multiple communication types (heterogeneity)

“Fallbacks”
- Support future communication types (evolution)
- Allow using new communication types at any point (incremental deployment)
Principal Types

Define your own communication model
## Principals

<table>
<thead>
<tr>
<th>Current Internet</th>
<th>XIA</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP address</td>
<td>Principal type</td>
</tr>
<tr>
<td>128.2.10.162</td>
<td>Host</td>
</tr>
<tr>
<td></td>
<td>Service</td>
</tr>
<tr>
<td></td>
<td>Content</td>
</tr>
<tr>
<td></td>
<td>Future</td>
</tr>
</tbody>
</table>

- Hash of host’s public key
- Hash of service’s public key
- Hash of content
Principal Type-Specific Semantics

Contact a host

Use a service

Retrieve content
Principal Type-Specific Processing

- Type-specific processing examples
  - Service: load balancing or service migration
  - Content: content caching

XIA router

Input

Common processing

Output

Host-specific processing

Service-specific processing

Content-specific processing

...
Routers with Different Capabilities

- Routers are **not** required to support every principal type
  - The only requirement: Host-based communication
Using Principal Types that are Not Understood by Legacy Routers?

Content-enabled router

Legacy router without content support

Content-enabled router

Want to communicate using content principals

Hyeontaek Lim / NSDI '12
Fallbacks

Tomorrow’s communication types... today!
Fallbacks: Alternative Ways for Routers to Fulfill Intent of Packet

Intent: Retrieve Content

**Fallback:** Contact Host, who understands Content request

What the network does:

- With content-enabled routers, use Content for routing
- Otherwise, use Host for routing (always succeeds)
DAG-Based Address

Your address is more than a number
DAG (Direct Acyclic Graph)-Based Addressing Enables Fallbacks

1. Packet sender
2. Routing choice
3. Intent
4. Content
5. Host
6. Another routing choice (with lower priority)
7. This host knows how to handle content request
8. Fallback

Hyeontaek Lim / NSDI '12
## DAG Addresses in Packet Header

<table>
<thead>
<tr>
<th>Common fields (Size, Hop limit, ...)</th>
<th>Including in-packet state for forwarding</th>
</tr>
</thead>
<tbody>
<tr>
<td>Destination address</td>
<td>For forward path</td>
</tr>
<tr>
<td>Source address</td>
<td>For return path</td>
</tr>
<tr>
<td>(Extension headers)</td>
<td></td>
</tr>
<tr>
<td>Payload</td>
<td>DAG encoding in paper</td>
</tr>
</tbody>
</table>

Hyeontaek Lim / NSDI '12
Scoping Using DAG

More specific intent & Better scalability

Tier-1 ISP → Tier-2 ISP → My laptop → Echo
Service Binding with DAG

Initial contact to a service

When a particular host should serve subsequent service requests

“Late binding”
DAG Allows Nested Fallbacks

Strong support for evolvable internetworking
Can We Forward DAGs Rapidly?

Expressive ≠ Expensive
XIA Software Router’s High Forwarding Throughput

Click-based implementation on commodity hardware
351K table entries based on a Route Views snapshot

≤26% slowdown for small packets with 3 fallbacks
XIA: eXpressive Internet Architecture

• Support for evolvable internetworking
  – Heterogeneous communication types
  – Future communication types
  – Incremental deployment of new communication types

• Principal types & fallbacks

• DAG-based addressing
XIA: Enabling Evolution by eXpression

- Prototype: [github.com/XIA-Project/xia-core](https://github.com/XIA-Project/xia-core)
  - Router, socket, content cache, etc.
  - Supports LAN, XIA-over-IP, GENI