An Introduction to the Identifier-Locator Network Protocol (ILNP)

Presented by Joel Halpern

Material prepared by
Saleem N. Bhatti & Ran Atkinson

https://ilnp.cs.st-andrews.ac.uk/

Thanks!

- Joel Halpern: presenting today!
- Ran Atkinson: co-conspirator.
- Students at the University of St Andrews:
  - Dr Ditchaphong Phoomikiatissak (Linux)
  - Dr Bruce Simpson (FreeBSD, Cisco)
  - Khawar Shezhad (DNS/Linux, Verisign)
  - Ryo Yanagida (Linux, Time Warner)
  - … many other students on sub-projects …
- IRTF (Routing RG, now concluded)
- Discussions on various email lists.

Background to Identifier-Locator

"IP addresses considered harmful"

- **"IP addresses considered harmful"**
  Brian E. Carpenter
  ACM SIGCOMM CCR, vol. 44, issue 2, Apr 2014
  http://dl.acm.org/citation.cfm?id=2602215
  http://dx.doi.org/10.1145/2602204.2602215
- *Abstract*
  This note describes how the Internet has got itself into deep trouble by over-reliance on IP addresses and discusses some possible ways forward.

RFC2104 (I), IAB, Feb 1997

- RFC2104, “IPv4 Address Behaviour Today”.
- Ideal behaviour of Identifiers and Locators:

Identifiers should be assigned at birth, never change, and never be re-used.
Locators should describe the host's position in the network's topology, and should change whenever the topology changes.
Unfortunately neither of the these ideals are met by IPv4 addresses.

RFC6115 (I), Feb 2011

- RFC6115, "Recommendation for a Routing Architecture".
- Section 17.3, page 65:

We recommended ILNP because we find it to be a clean solution for the architecture. It separates location from identity in a clear, straightforward way that is consistent with the remainder of the Internet architecture and makes both first-class citizens. Unlike the many map-and-encap proposals, there are no complications due to tunneling, indirection, or semantics that shift over the lifetime of a packet's delivery.

Key architectural concepts of ILNP

with examples based on ILNPv6

ILNP

- Identifier-Locator Network Protocol (ILNP)
- ILNP is both:
  - a set of architectural concepts for naming in an Internet Protocol.
  - a set of protocol mechanisms and behaviours for realising the ILNP architectural concepts in the existing Internet Protocol.
- Different architectural concepts to the current Internet Protocol, but engineered for leveraging the currently deployed Internet Protocol.

IP addresses bound to interfaces

```
saleem@ilnp-test-07:~$ ifconfig eno1
eno1      Link encap:Ethernet  HWaddr fc:aa:14:0a:96:5f
          inet addr:138.251.30.207  Bcast:138.251.30.255  Mask:255.255.255.192
          inet6 addr: 2001:630:35::207/64 Scope:Global
          inet6 addr: fe80::feaa:14ff:fe0a:965f/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1262690 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1649118 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:458358209 (458.3 MB)  TX bytes:339948777 (339.9 MB)
          Interrupt:20 Memory:f7800000-f7820000
saleem@ilnp-test-07:~$
```

IP addresses tied to a single interface.

Naming Architecture: IP vs ILNP

Protocol Layer	IP	ILNP
Application	FQDN or IP address	FQDN (RFC1958)
Transport	IP address (+ port number)	(Node) Identifier (+ port number)
Network	IP address	Locator
(Interface)	IP address	(dynamic mapping)
	Entanglement ☹	**Separation ☺**

FQDN = fully qualified domain name

ILNP – Engineering Summary

- **ILNPv6 on-the-wire is the same as IPv6 on-the-wire, end-to-end.**
  - Existing IPv6 routers can handle IPv6 packets.
  - Existing IPv6 switches can handle ILNPv6 packets.
- **Existing IPv6 applications and binaries can work unchanged on ILNPv6:**
  - Examples from testbed (details later).
- End-to-end:
  - Transport protocol bindings change (details later).
  - End-to-end state invariance preserved.

ILNP – Locators and Identifiers

- **Locator, 64 bits, L64:**
  - **Topologically significant.**
  - Names a (sub)network
    (same as today's **network prefix**).
  - Used only for routing and forwarding.
- **Node Identifier, 64 bits, NID:**
  - **Is not topologically significant.**
  - Names a logical/virtual/physical node, does **not** name an interface.
- **Upper layer protocols bind only to Identifier.**

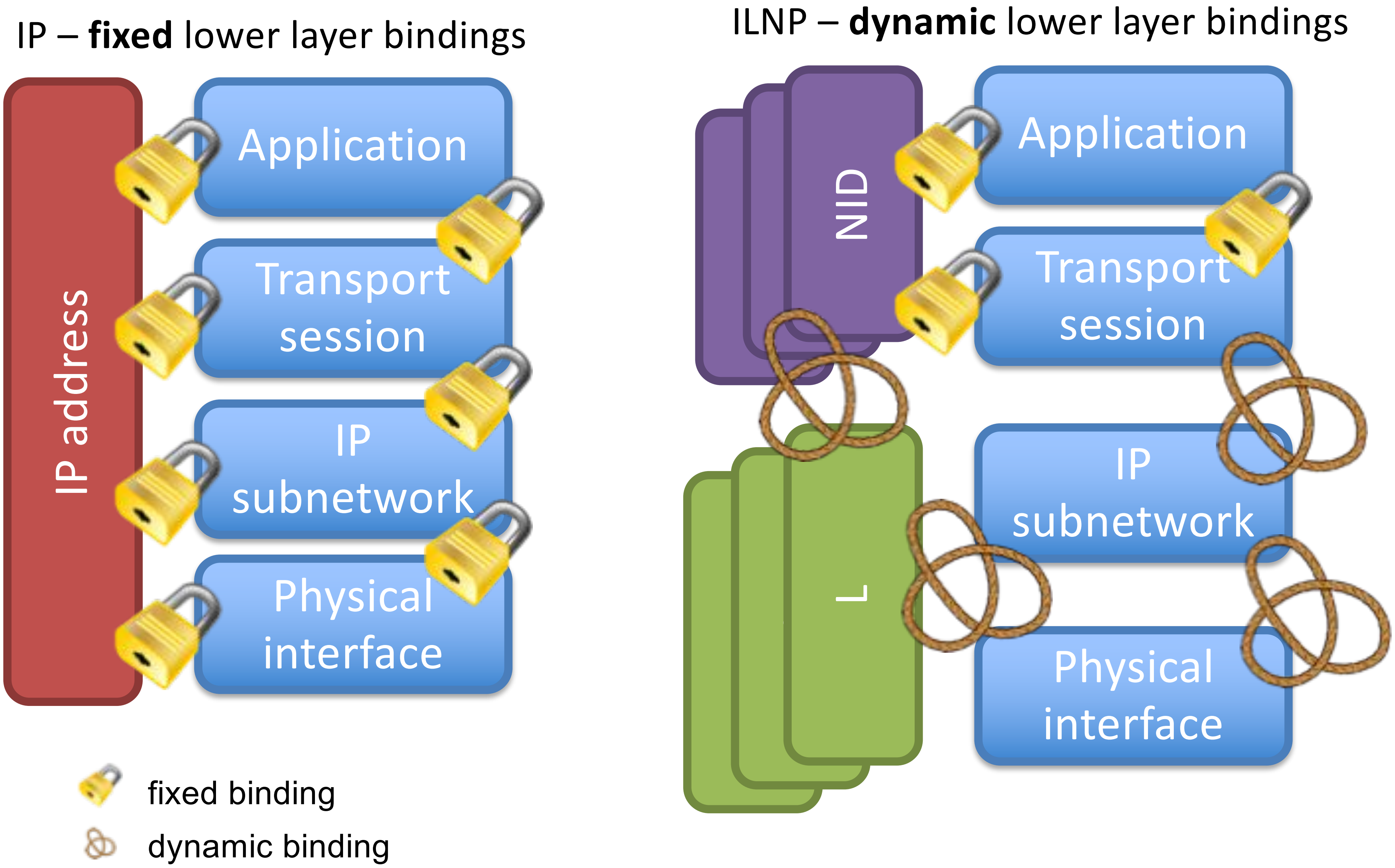
ILNP: L64 Properties

- L64 names an IP Subnetwork.
- L64 is equivalent to an IPv6 Routing Prefix.
- **Nodes can change their Locator values during the lifetime of an ILNP session:**
  - Enables mobility, multi-homing, NAT, end-to-end IPsec, site-controlled traffic engineering, etc.
- Multiple Locators can be used simultaneously:
  - Enables **network-level soft-handoff** for seamless mobility at the network level (example later).

ILNP: NID Properties

- NID names a **node**, *not* an **interface**.
- A host can use multiple NID values.
- NID values can be assigned statically:
  - administratively configured (e.g. /etc/hosts).
- NID values can be **generated dynamically**:
  - auto-configuration (e.g. ala SLAAC).
  - privacy (e.g. ephemeral values ala RFC8064).
  - assured identity (e.g. CGA ala RFC3972).
- NID must remain constant for a session:
  - E.g. TCP connection, UDP session, IPsec session.

Namespaces and Bindings



Key engineering and systems considerations for ILNPv6

based on experience with prototype implementations in the Linux kernel and FreeBSD kernel

ILNP
Ongoing research and implementation

- 9 *Experimental* status RFC documents:
  - RFCs 6740-6748
- Open source prototypes of ILNPv6 being implemented at University of St Andrews, UK.
- Support today in commercial DNS servers.
- Recommended by IRTF Routing RG chairs (RFC6115).
- ~14 years of peer-reviewed research:
  - Testbed implementations and evaluations of ILNPv6.
  - Papers and talks available at ILNP project web site: https://ilnp.cs.st-andrews.ac.uk/

IPv6 addresses and ILNPv6 I-L vectors

Encoding of L64 and NID values into IPv6 packets

IPv6 address (as in RFC3587 + RFC4291):

```
| 3 |      45 bits          | 8/16 bits |       64 bits                 |
+---+-----------------------+-----------+-------------------------------+
|       Unicast Routing Prefix          |     Interface Identifier      |
+---+-----------------------+-----------+-------------------------------+
```

IPv6 routing (address) prefix

same syntax, different semantics

ILNPv6 I-L vector (as in RFC6741):

```
|                64 bits                |       64 bits                 |
+---+-----------------------+-----------+-------------------------------+
|                Locator                |    Node Identifier (NID)      |
+---+-----------------------+-----------+-------------------------------+
```

same syntax and semantics as
IPv6 routing (address) prefix,
so IPv6 core routers work as today

these bits only examined and
acted upon by end systems

Name resolution

- Mapping application-level names to IL-Vs:
  - DNS records for L64 and NID.
  - Supported by BIND, KnotDNS, and NSD.
  - New /etc/hosts entries for I-LV values.
- Modified packet-handling code path for IPv6.
- Well-behaved IPv6 applications work unmodified over socket(2) API (see later).
- Application-specific naming services possible!
  - Do not have to use DNS, but DNS available if needed.

Address resolution

- Mapping I-LVs to lower-level addresses:
  - I-LV is 128-bits, same size as an IPv6 address.
  - So, IPv6 Neighbour Discovery can be used directly.
- No updates needed for:
  - Existing ethernet switches that handle IPv6.
  - Existing routers that handle IPv6.
  - IPv6 Neighbour Discovery.

End-system OS kernel

- Updates required:
  - IPv6, ICMPv6, packet-handling paths, I-L bindings.
  - Transport level packet handling paths and PCB.
  - getaddrbyname(3) family code (libc).
- **Existing socket(2) API works for well-behaved IPv6 applications:**
  - **IPv6 binaries can be used directly (see later).**
- Future – API that knows about ILNP:
  - benefits of using L64 and NID values directly.
  - ILNP could be “hidden” in frameworks/libraries, as sockets(2) is today in many cases.

End-to-end protocol

- **No NATs needed.**
- **No tunnels needed.**
- **No proxies needed.**
- Harmonised functionality in the **end-system**:
  - mobility without agents or proxies.
  - multihoming without extra routing state.
  - mobility and multihoming together (duality).
  - end-to-end packet-level security.
  - support for wide-area VM-image mobility.

Example:
ILNP mobility

A performance comparison with
Mobile IPv6 on a Linux testbed.

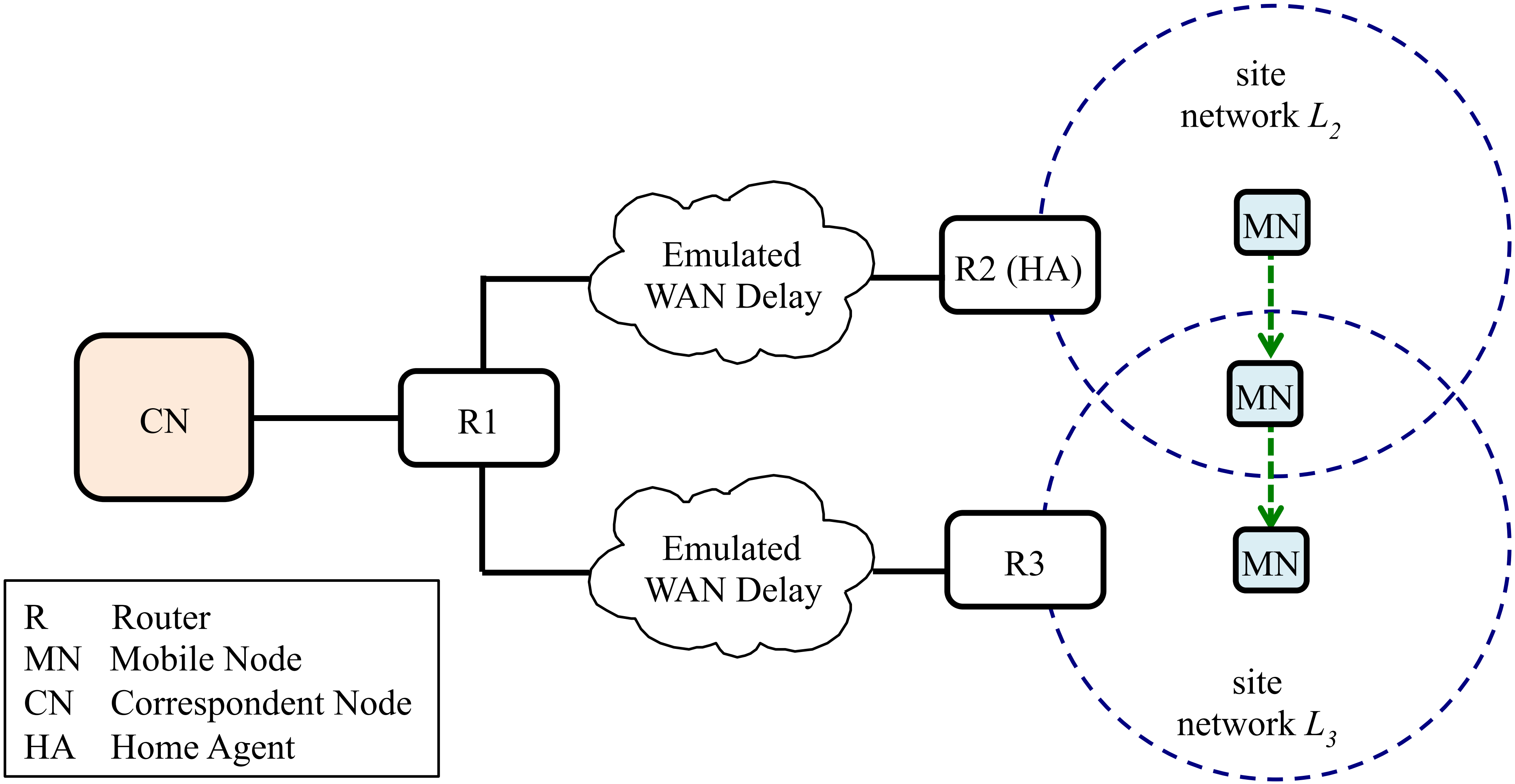
Performance evaluation

- **ILNP used with unmodified IPv6 binaries:**
  - without recompilation.
  - standard C sockets(2) API.
- User (data) plane performance with TCP:
  - hard-handoff: switch to "new" L64 immediately.
  - soft-handoff: use "old" and "new" L64s in cell overlap.
  - Comparison with Mobile IPv6 (w/ and w/o RO).
- "IP without IP addresses", AINTEC 2016
  http://dx.doi.org/10.1145/3012695.3012701
  ACM Digital Library, **Open Access**

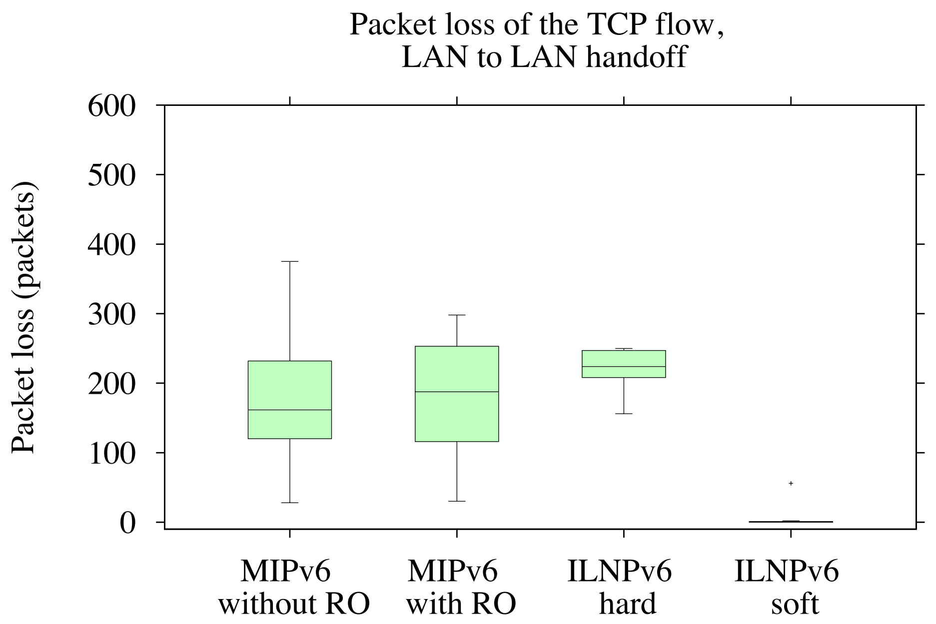
Testbed [1]

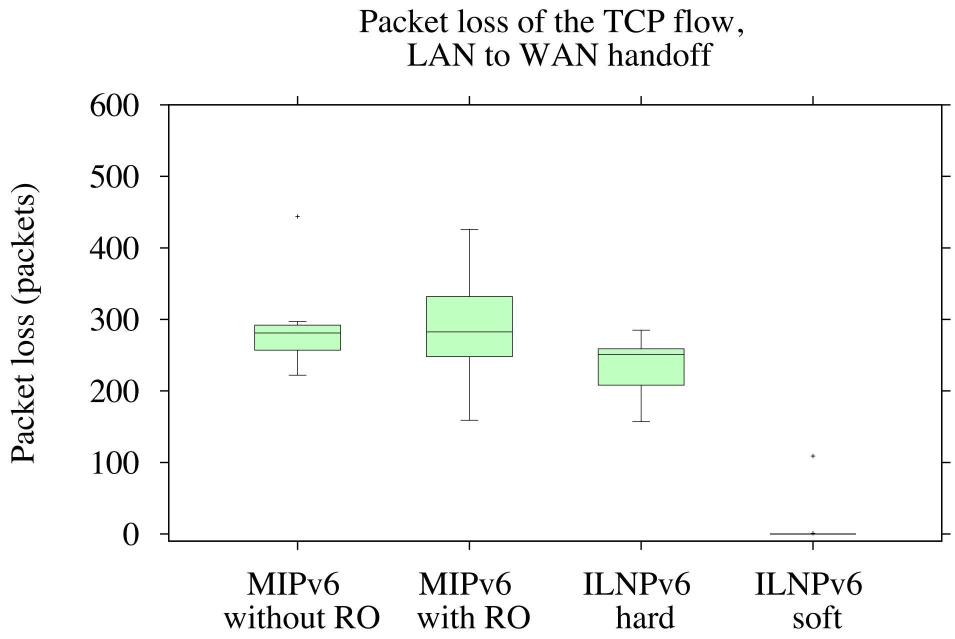
- **Use of unmodified *iperf2* binary for TCP flows.**
- Linux kernel v3.9:
  - Linux default TCP (CUBIC).
  - **Unmodified kernel used for IPv6 routers (R1, R2, R3).**
- In-kernel modifications for end-systems:
  - TCP state management (use NIDs).
  - IP-level changes for L64 / NID.
  - Locator Update (LU) processing.
  - Mobility / handoff processing.
- Emulation of WAN by adding delay:
  - use of *netem* software.

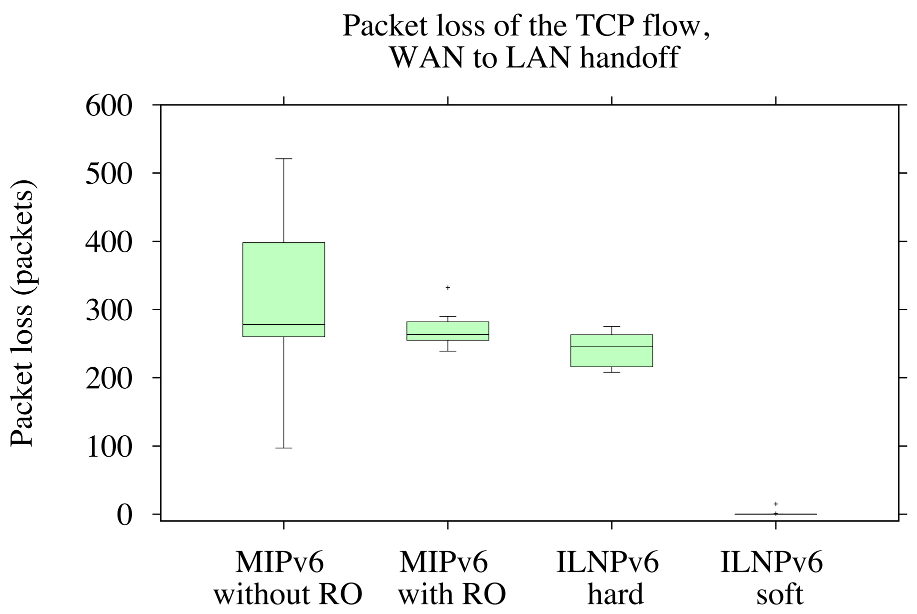
Testbed [2]

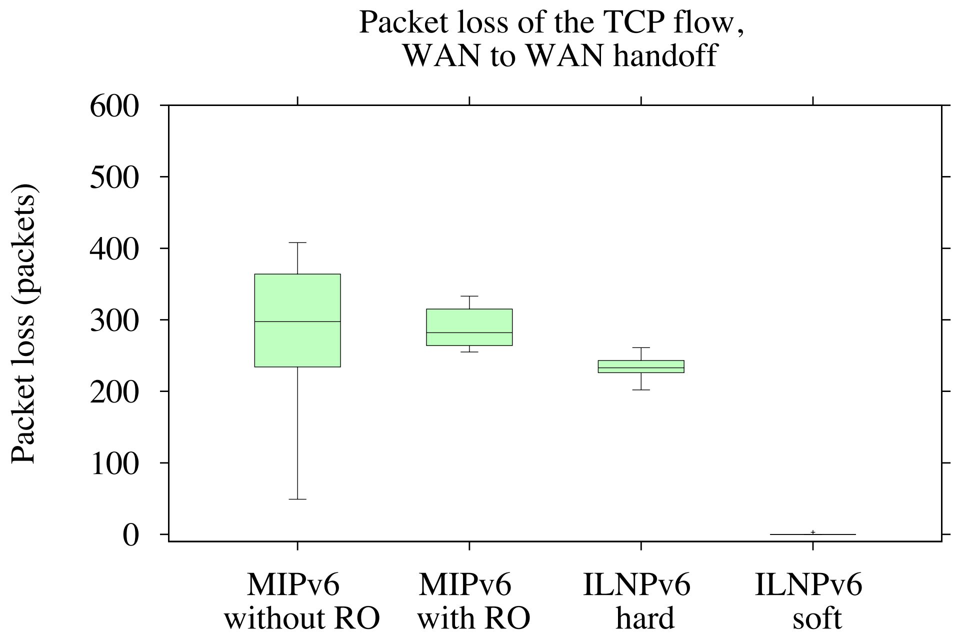

CN
R1
Emulated
WAN Delay
Emulated
WAN Delay
R2 (HA)
R3
site
network L_2
MN
MN
MN
site
network L_3
R Router
MN Mobile Node
CN Correspondent Node
HA Home Agent

Results – loss (due to handoff)

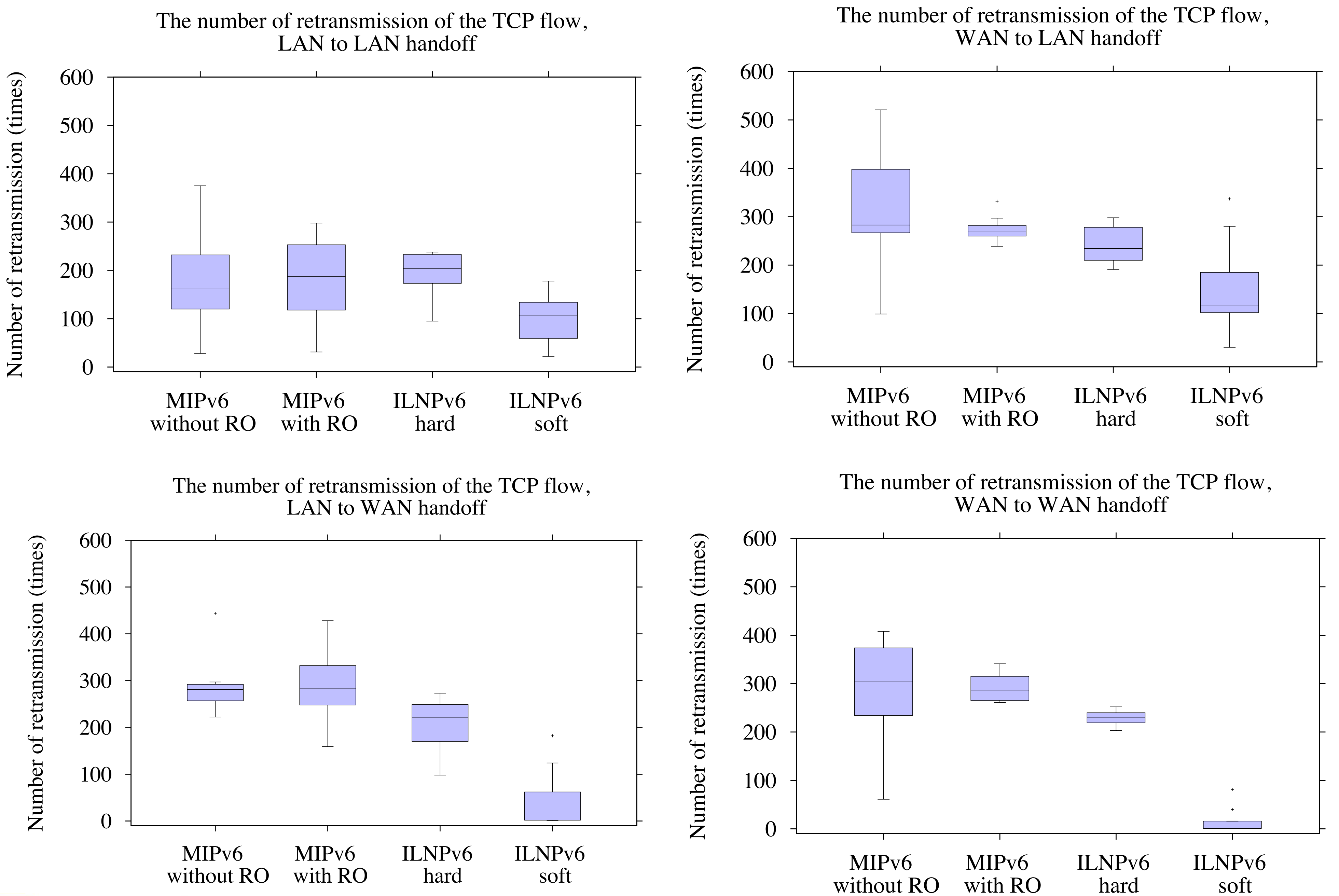








Results – re-tx (due to handoff)



University of St Andrews
Scotland's first university

Mobility experiment summary

- ILNP implementation in Linux kernel v3.9:
  - internal testbed at St Andrews.
  - LAN links and emulated WAN links.
- **ILNP used unmodified IPv6 *iperf2* binary.**
- Compared ILNP with MIPv6:
  - ILNP hard-handoff and soft-handoff.
  - MIPv6 w/ and w/o RO.
- **ILNP has better performance than MIPv6 in terms of loss and retransmission.**

ILNP – summary

ILNP

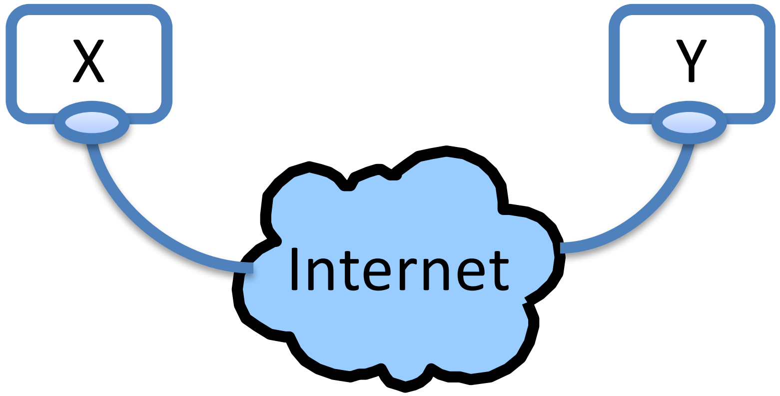
- Addressing without addresses ☺
- Identifier-Locator architecture split gives cleaner naming across layered system, enabling:
  - dynamic, flexible bindings for end-systems.
  - scalable host mobility without tunnels or agents.
  - scalable multihoming without additional routing state.
- Radical architectural approach realised with careful engineering – backwards compatible:
  - works today on IPv6 networks.
  - well-behaved IPv6 binaries can be used directly.
- No NATs, tunnels, or network upgrades needed.

Backup Slides

Early history

- Potential issues with TCP/IP addressing identified at least as far back as 1977 (IEN-1).
- Identifier/Locator Split proposed for IPv6 in the early 1990s:
  - Bob Smart, then Dave Clark, then Mike O'Dell.
  - I/L split was not adopted by IETF IPv6 WG.
- IAB Network-Layer Workshop, late 1990s.
- NIMROD, RFC1992 (I) (Aug 1996)
- IRTF Name Space Research Group (NSRG).
- IRTF Routing RG (RRG).

ILNP: transport layer state example



A = IP address
P = port number

L = Locator
I = (Node) Identifier
P = port number

At X:
<TCP: A_X, P_X, A_Y, P_Y> <IP: A_X, A_Y>

At Y:
<TCP: A_Y, P_Y, A_X, P_X> <IP: A_Y, A_X>

At X:
<TCP: I_X, P_X, I_Y, P_Y> <IP: L_X, L_Y>

At Y:
<TCP: I_Y, P_Y, I_X, P_X> <IP: L_Y, L_X>

IPv6 packet header – router view

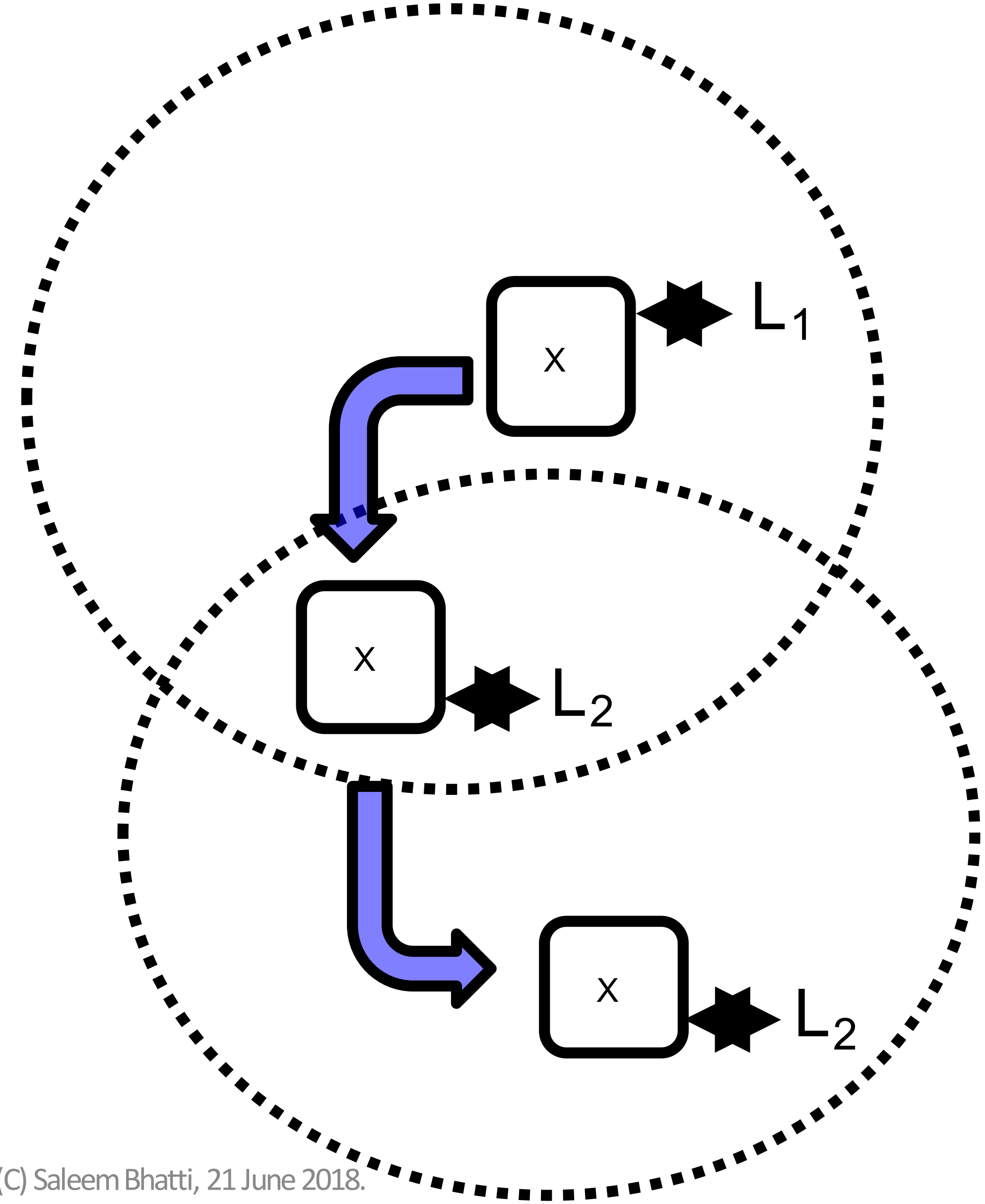
```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version| Traffic Class |           Flow Label                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Payload Length        |  Next Hdr     |   Hop Limit   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                         Source IPv6 Address                   +
|                                                               |
+-                                                             -+
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                      Destination IPv6 Address                 +
|                                                               |
+-                                                             -+
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

ILNPv6 packet header – host view

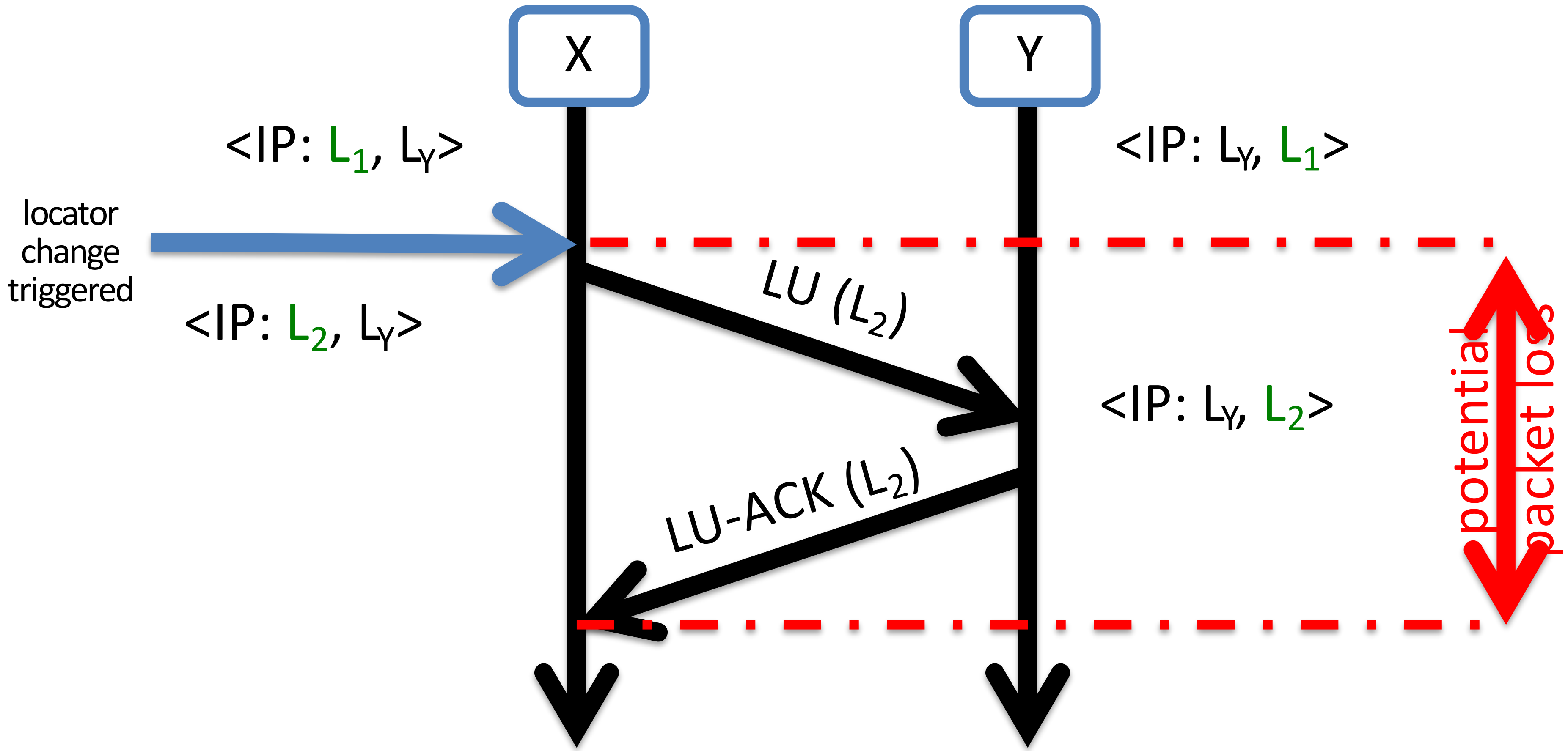
```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version| Traffic Class |           Flow Label                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Payload Length        |  Next Hdr     |   Hop Limit   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                       Source Locator                          +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                     Source Identifier                         +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                    Destination Locator                        +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                  Destination Identifier                       +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Hard handoff

- Hard handoff model used by MIP
- ILNP supports hard handoff also:
  - move from one cell to another
  - drop locator (prefix) L_1, use locator (prefix) L_2



ILNP Locator Update (LU) [1]



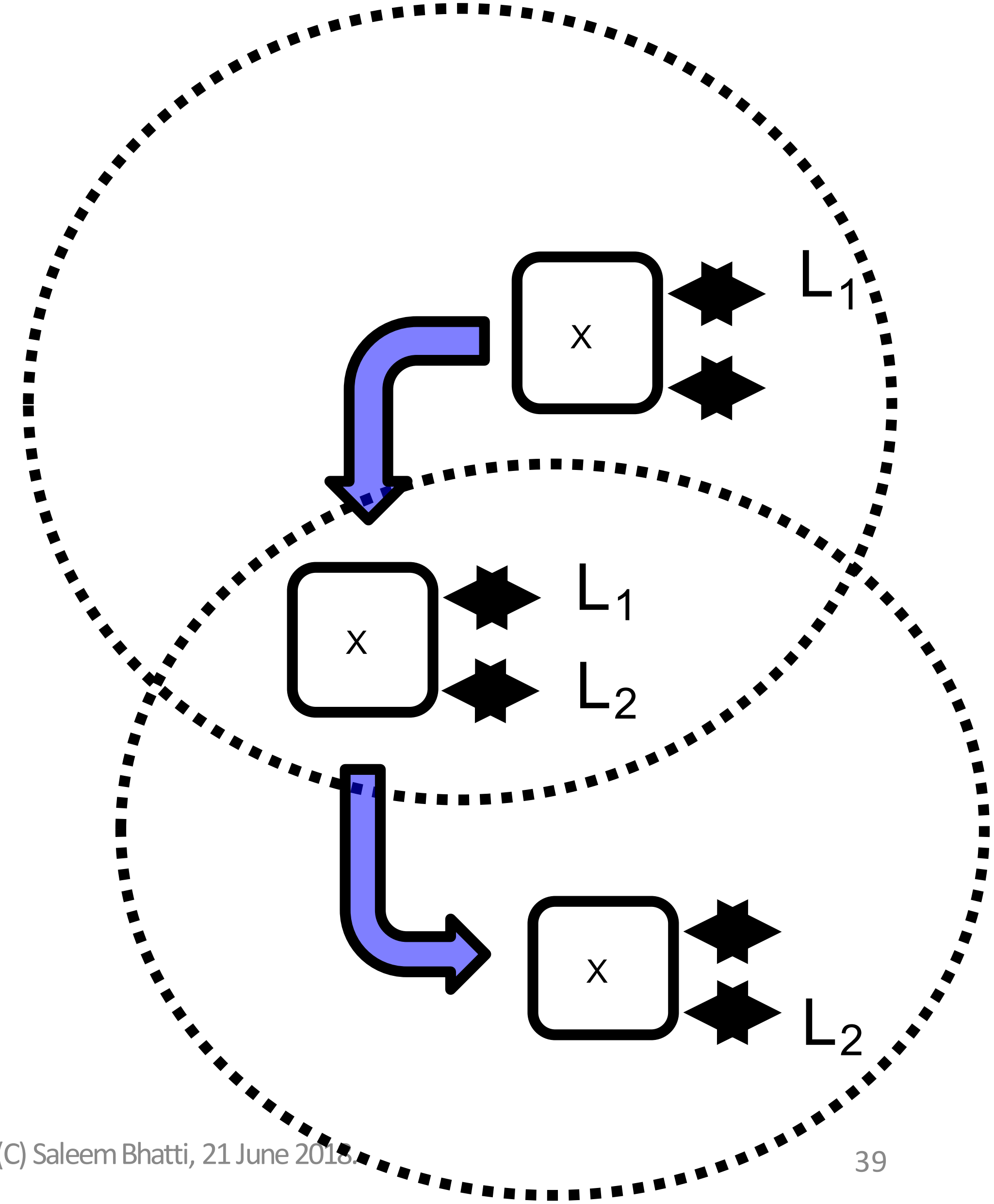
Hard handoff

(similar to Binding Update for Mobile IPv6)

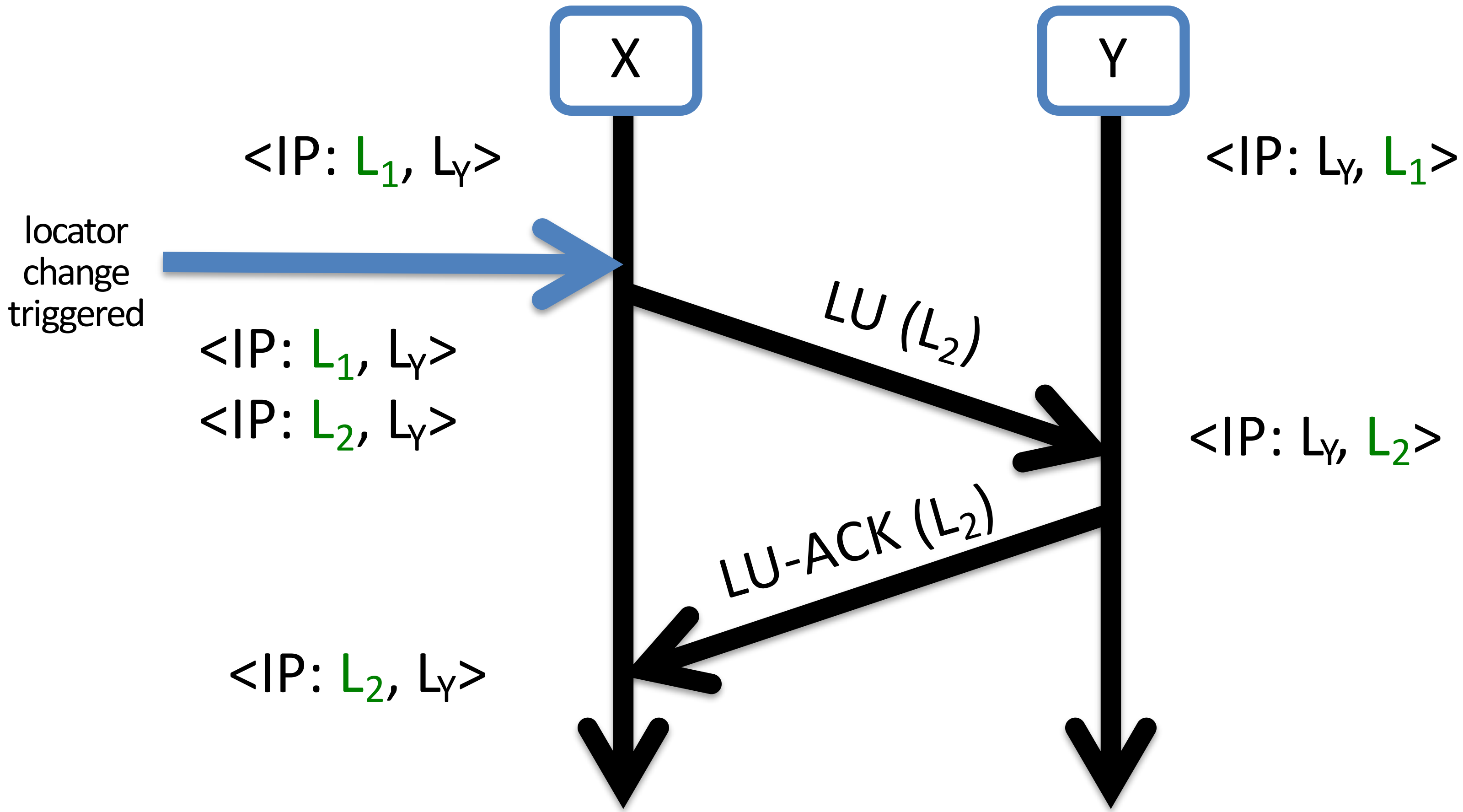
(new L values can be learned from IPv6 router advertisements)

Soft handoff

- ILNP support soft handoff (similar concept to CDMA)
- Both old locator (L_1) and new locator (L_2) used in overlap region
- Mobile host is multihomed during handoff



ILNP Locator Update (LU) [2]



Soft handoff

(new L values can be learned from IPv6 router advertisements)

Multi-path transport sessions [1]

- ILNP allows a NID to be bound simultaneously to multiple L64s, i.e. multipath is supported natively.
- ILNP manages this at the IP layer:
  - common L64 handling.
  - can be used for any transport/layer-4 protocol.
  - e.g. multihoming for hosts, mobile soft-handoff.
- Locator Update (LU) signalling:
  - simple end-to-end control of L64 values.
- Good IP-level security and privacy (RFC6740).
- Works with ILNP NAT-like functions (RFC6748).

Multi-path transport sessions [2]

- A multi-path TCP / ILNP would need MP-TCP's congestion control (ala RFC6356).
- Potential ILNP advantages for multipath:
  - works for TCP and UDP (as well as any layer 4).
  - "address" handling (multiple-L64 native to ILNP).
  - combined/dual multihoming and mobility.
  - can move **all** sessions between a pair of hosts in 1 RTT.
  - native security and privacy features.
  - native operation with ILNP NAT-like functions.

ILNP with non-ILNP nodes

- ILNP always sends a ***Nonce* Destination Option** in the first packet of a session.
- If recipient – an end-system – is also ILNP, then the ***Nonce*** is returned.
- If recipient is not ILNP, then it discards the ILNP packet, responds with ICMP message:
  - this is normal IPv6 behaviour – no code changes.
- ***Nonce*** is a Destination Option, so no adverse impact to switches or routers.

ILNP avoids Flag-Day transitions

- All ILNPv6 implementations fully support existing IPv6.
- ILNPv6 packets are IPv6 packets on-the-wire:
  - Some ILNPv6 packets have the *Nonce* Destination Option.
  - ILNPv6 adds to the set of the existing ICMPv6 messages.
- If an ILNPv6-capable host tries to talk with an IPv6-only host, the IPv6 host will drop the ILNPv6 packet (due to the unrecognized *Nonce* Destination Option) and (per existing IPv6 specs) send an ICMPv6 message back.
- The ILNPv6-capable host will then use existing IPv6 to communicate with that IPv6-only host.

ILNP deployments with /64 per host

- Many IPv6 deployments allocate /64 to a host.
  - Often this is done for operational security reasons.
  - Other times it is done for other operational reasons.
- RFC-7934/BCP-204 recommends that each host is allocated multiple IPv6 addresses and explains why this is important/valuable.
- ILNPv6 **supports** allocating a /64 to each host, but **does not require** allocating a /64 to each host:
  - a network operator may choose to have multiple hosts on a single /64.

Node Identifier (NID) Considerations

- **ANY method that can be used to generate an IPv6 Interface ID also MAY be used to generate an ILNP NID.**
- **ILNPv6 hosts MAY use many NID values at the same time:**
  - can be generated dynamically, when needed.
- **Host's valid ILNP NID values MAY change over time, as required (e.g. for privacy).**
- Many ILNP deployments will want to use existing (& future) IPv6 privacy algorithms and mechanisms. Please also see:
  - "Security and Privacy Considerations for IPv6 Address Generation Mechanisms", RFC-7721, March 2016.
  - "Privacy Considerations for IPv6 Adaptation-Layer Mechanisms", RFC-8065, February 2017

ILNPv6: Point-to-Point Router interfaces

- Using ILNPv6 does not preclude the use of /127 prefixes between routers.
- This can be handled as a special-case within ILNPv6, as for IPv6.
- Some operators configure their router-to-router, point-to-point interfaces as “unnumbered”, which also is fine for ILNP.
- Use of loopback for applications also fine, even when /127 is used.

Example: VM mobility

More details in:

S. N. Bhatti and R. Atkinson,
"Secure & Agile Wide-Area Virtual Machine Mobility",
IEEE MILCOM 2012, Oct 2012

http://dx.doi.org/10.1109/MILCOM.2012.6415716

Why use VM mobility (migration)?

- Datacentre(s):
  - single-site, multi-site, multiple datacentres
- Load balancing / load distribution
- Quality of service:
  - Latency
  - Throughput
  - Availability
- **Resource management:**
  - **CPU / disk / network / energy**
- OPEX / accounting / billing
- … etc …

Current approach to VM mobility

- **We see VM migration as VM mobility.**
- A TCP/IP session is bound to particular network interfaces on particular IP subnets:
  - IETF Mobile IP standards not widely implemented and are hard to deploy, so not the answer here.
- Widely used systems use large-scale bridged (V)LANs – including wide-area Layer-2 tunnels:
  - Enables multiple datacentres to share the same IP subnetwork/routing prefix.

Issues of today's solutions

- Requires additional layer-2 protocol and management complexity.
- Scaling issues for large-scale, multiple datacentres.
- Reduces flexibility and adaptability.
- Proprietary solutions – vendor lock-in:
  - interworking of solutions may not be possible
- Security exposure possible:
  - reliance upon third parties – trust

Increased CAPEX + OPEX

- Network design options greatly reduced.
- Must deploy more expensive switches that have much larger MAC address tables:
  - May need to replace existing switches which currently have smaller MAC address tables.
- Must deploy more expensive routers with more sophisticated Layer-2 VPN features.
- Network is much more difficult to configure, manage and troubleshoot – more expensive.

Current solution examples

- VMware and partners (inc Cisco) – VxLAN:
  - RFC7348 (I) https://tools.ietf.org/html/rfc7348
- Microsoft and partners (inc Cisco) – NVGRE:
  - RFC7637 (I) https://tools.ietf.org/html/rfc7637
- OpenFlow – Open Network Foundation:
  - https://www.opennetworking.org/
  - Originally from Stanford U, now many vendors, including Cisco, Extreme, Force10, Juniper, HP ...
- Juniper Networks – QFabric:
  - http://www.juniper.net/Qfabric/
- (... probably others ...)

VM Mobility using ILNP

Today, VM migration is a **network service**.
ILNP changes it to a **end-host function**.

ILNP goals for VM mobility

1. Enable datacentre operators to maintain essential services **without requiring specialised networking features** (e.g. no need for large flat networks).
2. **Enable scalable wide-area VM mobility** (e.g. between continents) across different routed IP networks, in addition to enabling local-area VM mobility (e.g. within a datacentre).
3. **Avoid interruptions to datacentre services** for critical applications, services, and other capabilities.
4. **Avoid dependence on any specific network design**, in order to enable adaptive datacentre network designs that maximise resilience, fault-tolerance, and scalability.

IPsec with ILNP Mobility

- **IPsec with ILNP binds only to NID:**
  - NID is used in transport layer state
  - NID is not topologically significant
  - preserves end-to-end semantics
- **NID-L binding change does not impact VM:**
  - dynamic update of NID-L binding similar to that for Mobile IPv6
  - change in NID-L binding does not impact transport layer session, so IPsec can be used end-to-end during migration of a VM

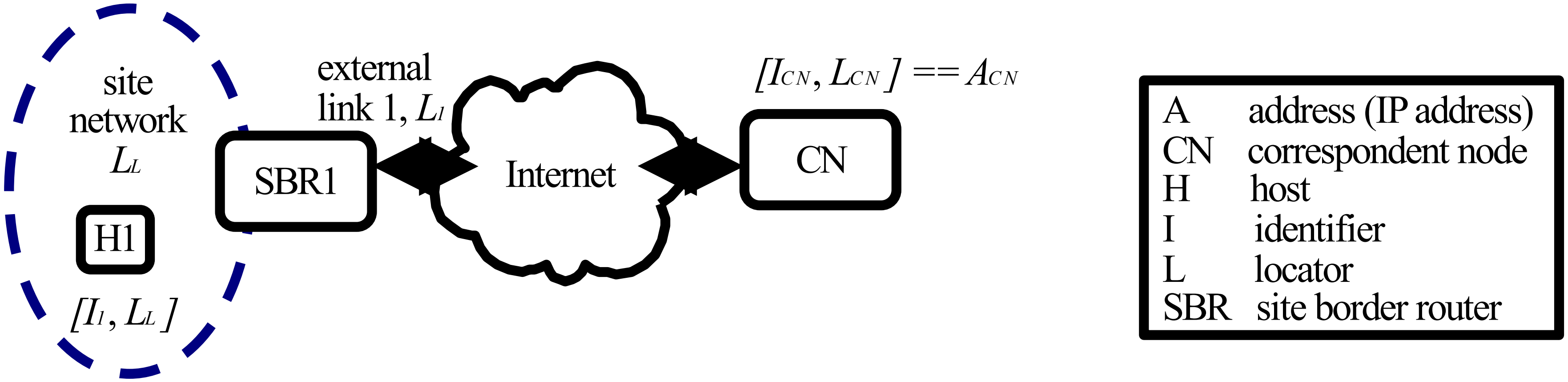
ILNP: Limitations

- End systems need to be upgraded for ILNP
  - VM platforms can do this, hiding changes from the guest VM instances (e.g. via ILNP-aware NAT/NAPT).
- Native ILNP, without any VM platform support, should work for all ***well-behaved*** applications
- Examples of well-behaved applications:
  - Application works fine with an existing IP NAT/NAPT
  - Application does NOT embed IP address inside application-layer protocol

ILNP scenarios summary

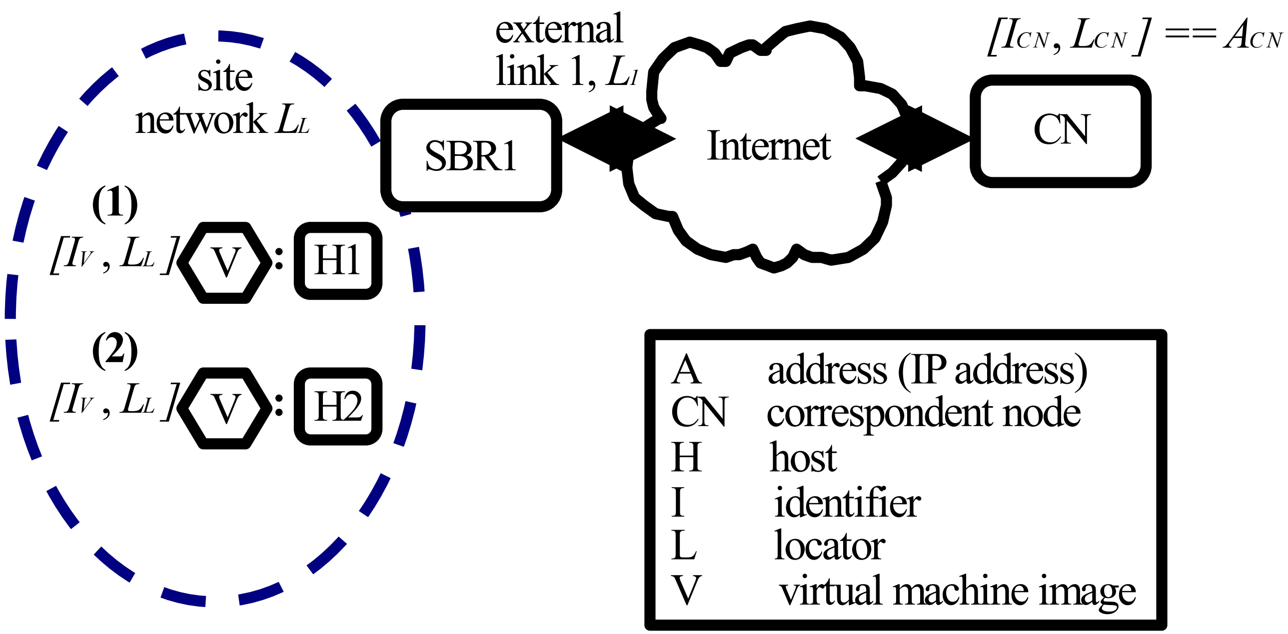
- **Insight: VM Mobility is a IP host mobility problem.**
- 3 different deployment scenarios described:
  - RFC6748 (I)
- 2 scenarios are invisible to remote clients:
  - mobility within datacentre
  - mobility across distributed datacentre
- 1 scenario provides high service resilience:
  - distributed application mobility across datacentres
- **None of these require any special network support:**
  - Existing switches & routers can be used as-is
  - Lowers OPEX and CAPEX for vendors and users

Intra-Datacentre Move: Invisible [1]

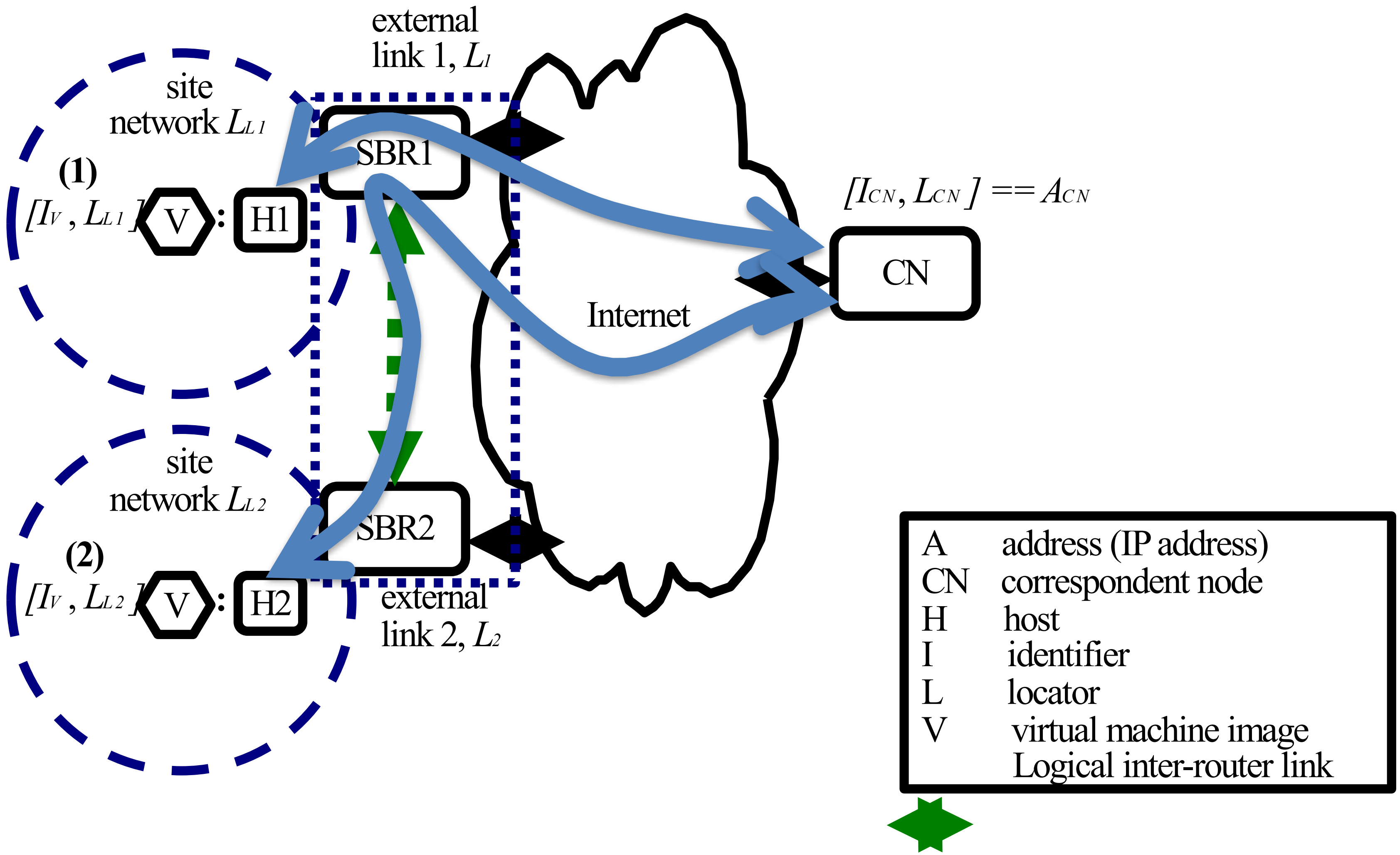


- Site network uses private addressing internally (L_L).
- SBR1 has global address on exterior interface (L_1).
- SBR1 re-writes Locator values in packets to/from Internet, which hides changes to Locator values within the site network from the CN.

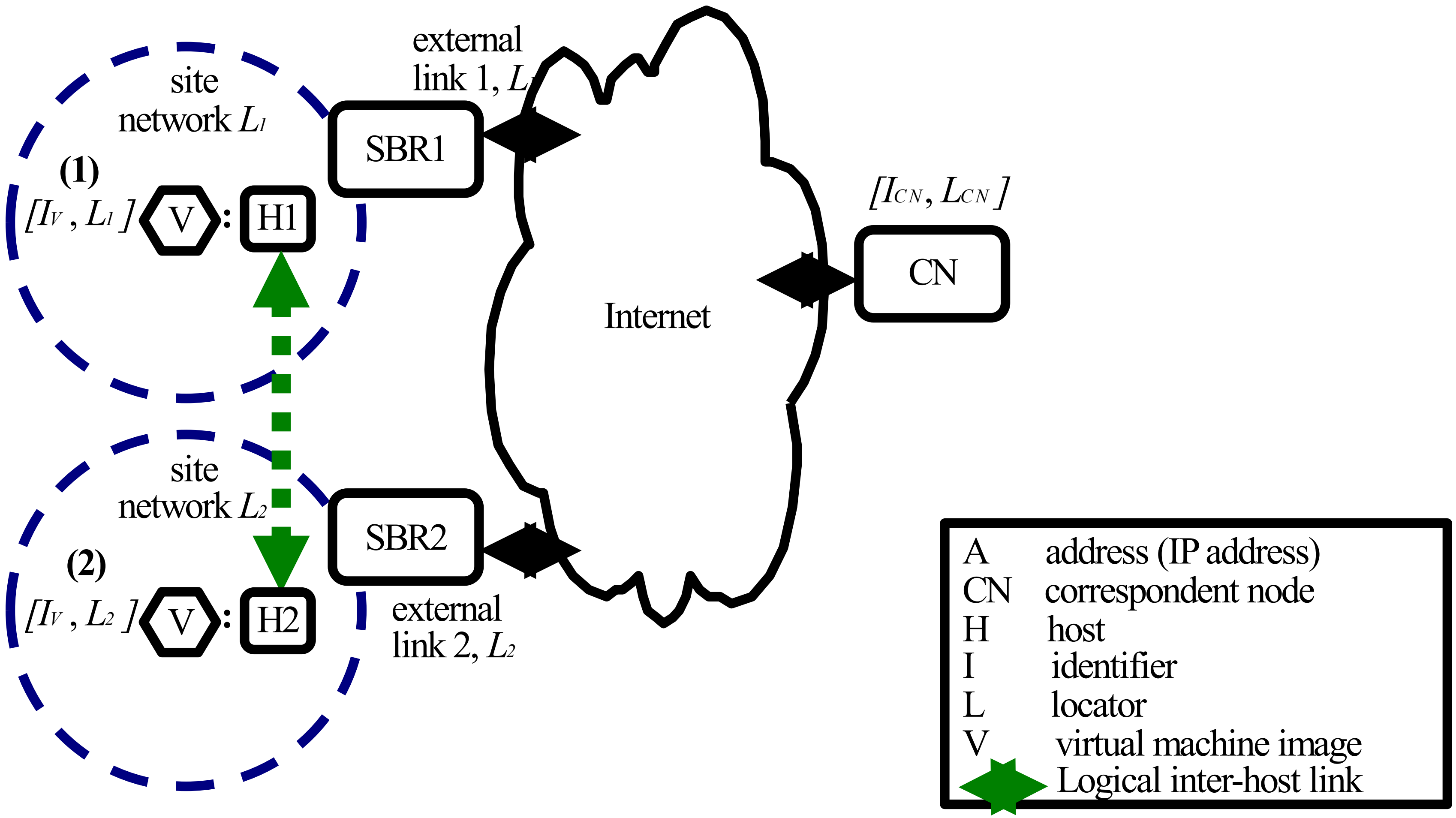
Intra-Datacentre Move: Invisible [2]



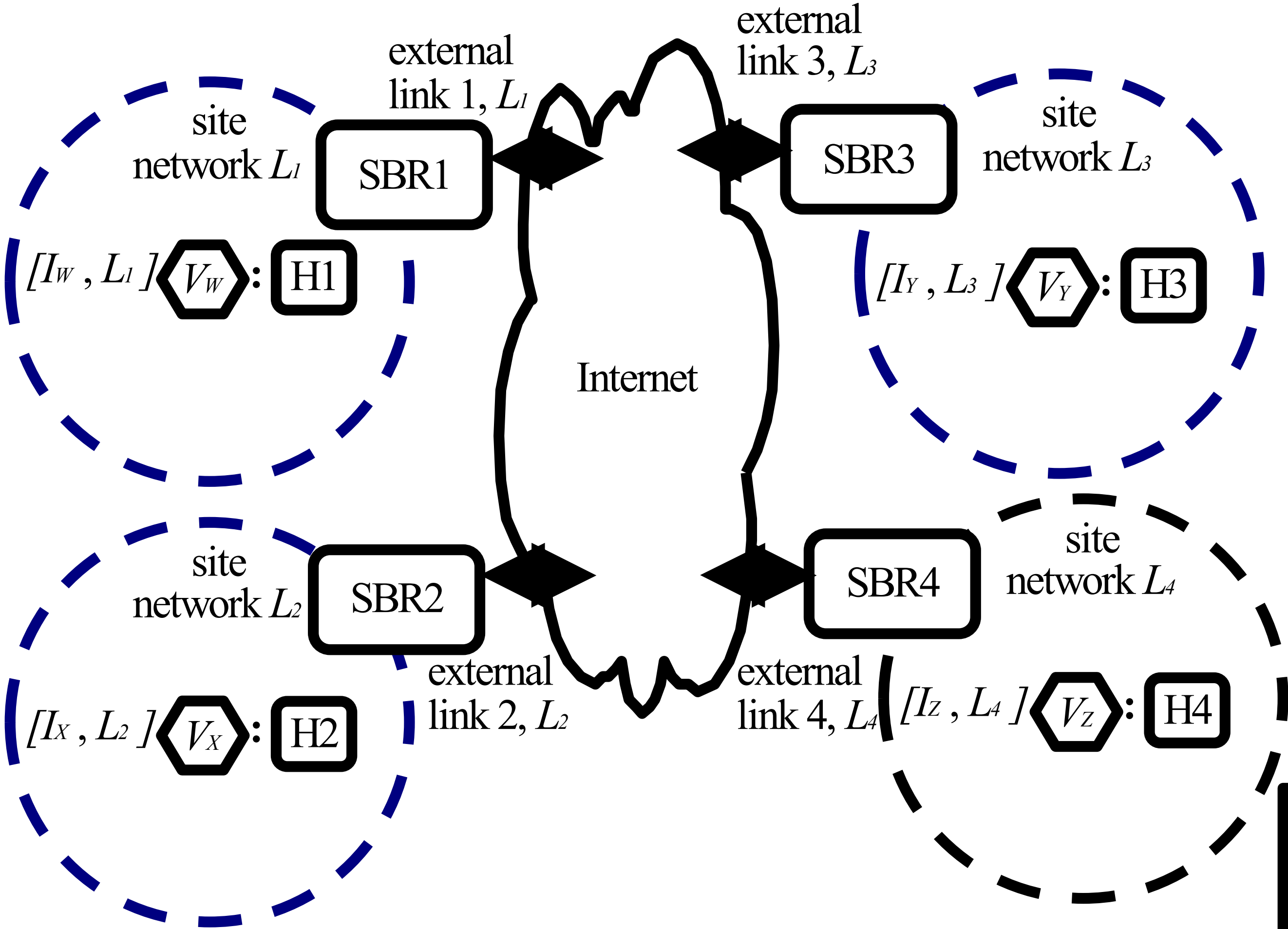
Inter-Datacentre Move: Invisible



Inter-Datacentre Move: Visible



Distributed Applications



A	address (IP address)
CN	correspondent node
H	host
I	identifier
L	locator
V	virtual machine image

ILNP Benefits

- Wide-area VM mobility without special network support
  - **Will operate over existing IP connectivity**
  - **No need for large Layer-2 (V)LANs**
- IPsec with ILNP works end-to-end:
  - no additional complexity
- Invisible to client systems (3 scenarios).
- Incremental deployment possible:
  - operates over current IPv6 backbone
- Mixed environments possible:
  - IPv4, IPv6, ILNPv6 all at the same datacentre

Advantages for global datacentres

- ILNP backwards-compatible with IP infrastructure
- Users have reduced network CAPEX and network OPEX, increasing VM platform value
- Simpler network designs/deployments likely have higher reliability & availability
- VM platform works well with any network design – increases market opportunity
- VM platform captures larger percentage of the total value chain

More information on ILNP

https://ilnp.cs.st-andrews.ac.uk/