

Experiences Implementing the Node Identity Internetworking Architecture using HIP

Simon Schuetz
simon.schuetz@nw.neclab.eu

NEC Laboratories Europe

Ambient Networks-Daidalos Workshop on
Locator/Identifier/Identity based Networking



Why a Node Identity Internetworking Architecture?



Problems with IP?

- ❖ IP serves as host identifier and host's location in the network
 - hinders (host) mobility (change of IP = change of identity)
 - limited multihoming support (multiple IPs = multiple identities)
 - lacking support for (host) authentication
- ❖ NATs (and other middleboxes) filter based on upper layer information
 - encrypted communication e.g. using IPsec encryption usually breaks at NATs
 - new (unknown) protocols are not supported
- ❖ can route from private to public network, but not vice-versa
- ❖ cannot easily integrate different networking technologies, e.g. cannot route towards an IPv6 address in a IPv4 domain
- ❖ IP infrastructure does not provide sufficient support for administrative boundaries

What is the Node Identity Internetworking Architecture?



Goals for the NodeID Arch

- ❖ must integrate heterogeneous network domains
- ❖ require only minimal set of common pieces, e.g., avoid new global (structured) address spaces
- ❖ provide
 - mobility support
 - multihoming support
 - always-on security
 - DoS protection
 - support administrative boundaries/domains

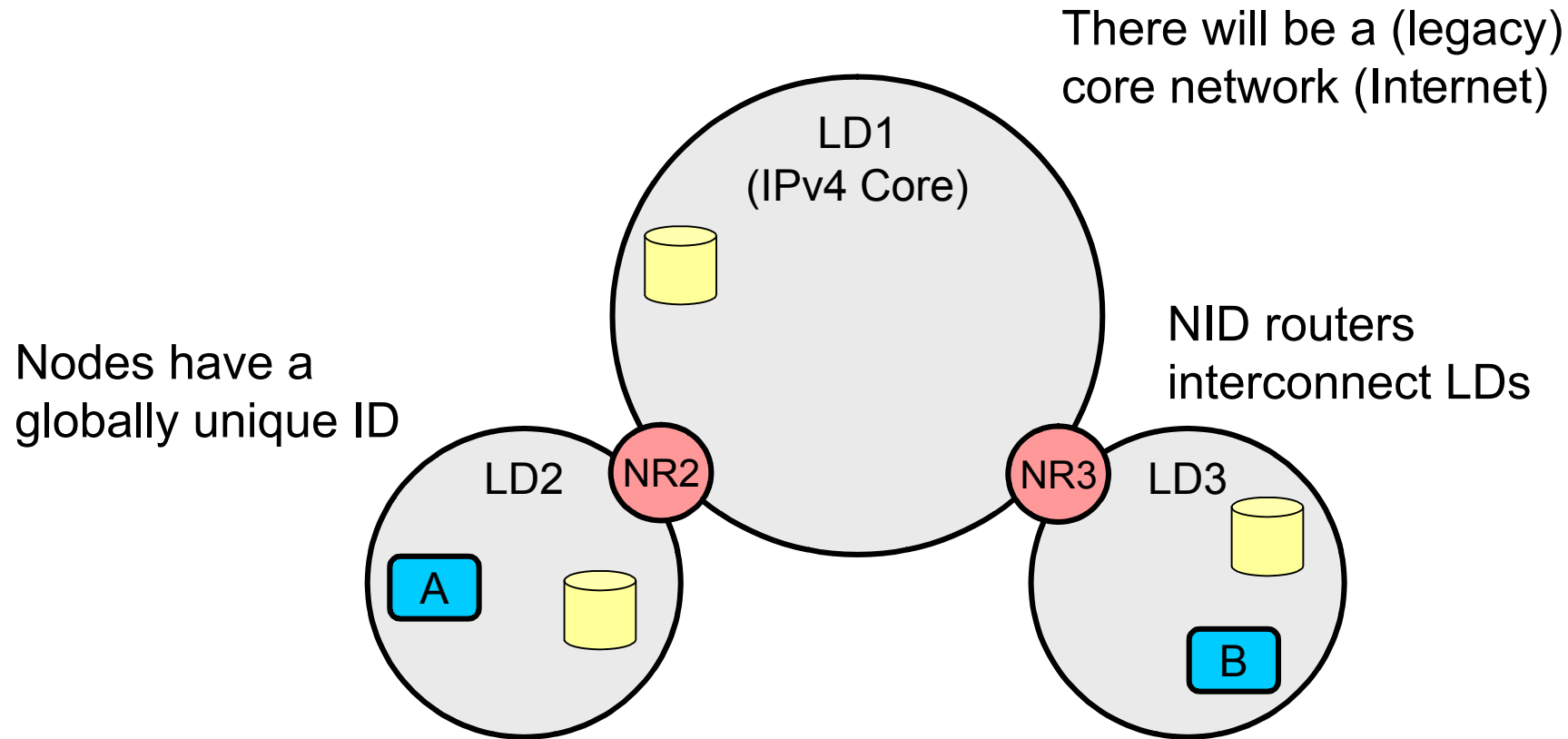
Assumptions / Observations

- ❖ nodes are grouped in **locator domains** (LDs)
- ❖ locator domains
 - consist of a **single networking technology** (IPv4, IPv6, etc.)
 - have a **consistent internal routing system**, i.e. do not rely on any external entities/services
- ❖ will have one or a few rather **static** LDs, the **core LDs**(e.g. current IPv4 backbone)
- ❖ LDs are arranged in tree-like structures hanging from the core
- ❖ **mobility** occurs more frequently **at the edges**

Fundamental Features

- ❖ separation of node identity and node location(s)
 - addresses are only used as locators
 - a node's **locators can change** over time
 - a node's locator **types can change** over time
- ❖ **cryptographic node identities**
 - public key represents node identity (NID)
 - NID hash used as forwarding token
- ❖ **intra**-LD routing relies on the specific network technology
- ❖ **inter**-LD routing based on NIDs

Node ID Architecture: Overview



Locator domains use their own internal addressing and routing scheme

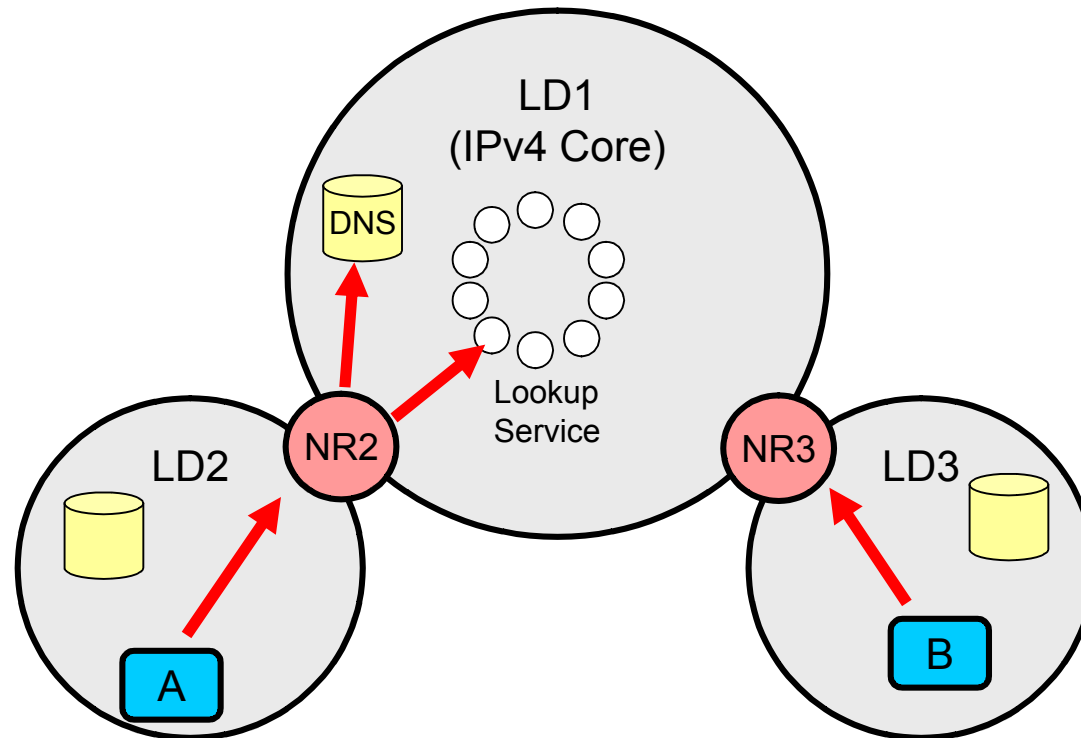
Node ID Architecture: Routing

- ❖ nodes register their NID with all NID routers (NRs) along a path towards the core
- ❖ registration path serves as a default route
- ❖ a home NR in the core serves as rendezvous point (similar to MIP home agent or HIP RVS)
- ❖ the home NR is used as a routing hint for a partial source route
- ❖ routing hint stored in a global naming system (e.g. DNS)

Example (Registration)

NID routers register their ID and NID ID mapping with the a
lookup service (e.g. DNS), such that NID can be used
as a routing hint

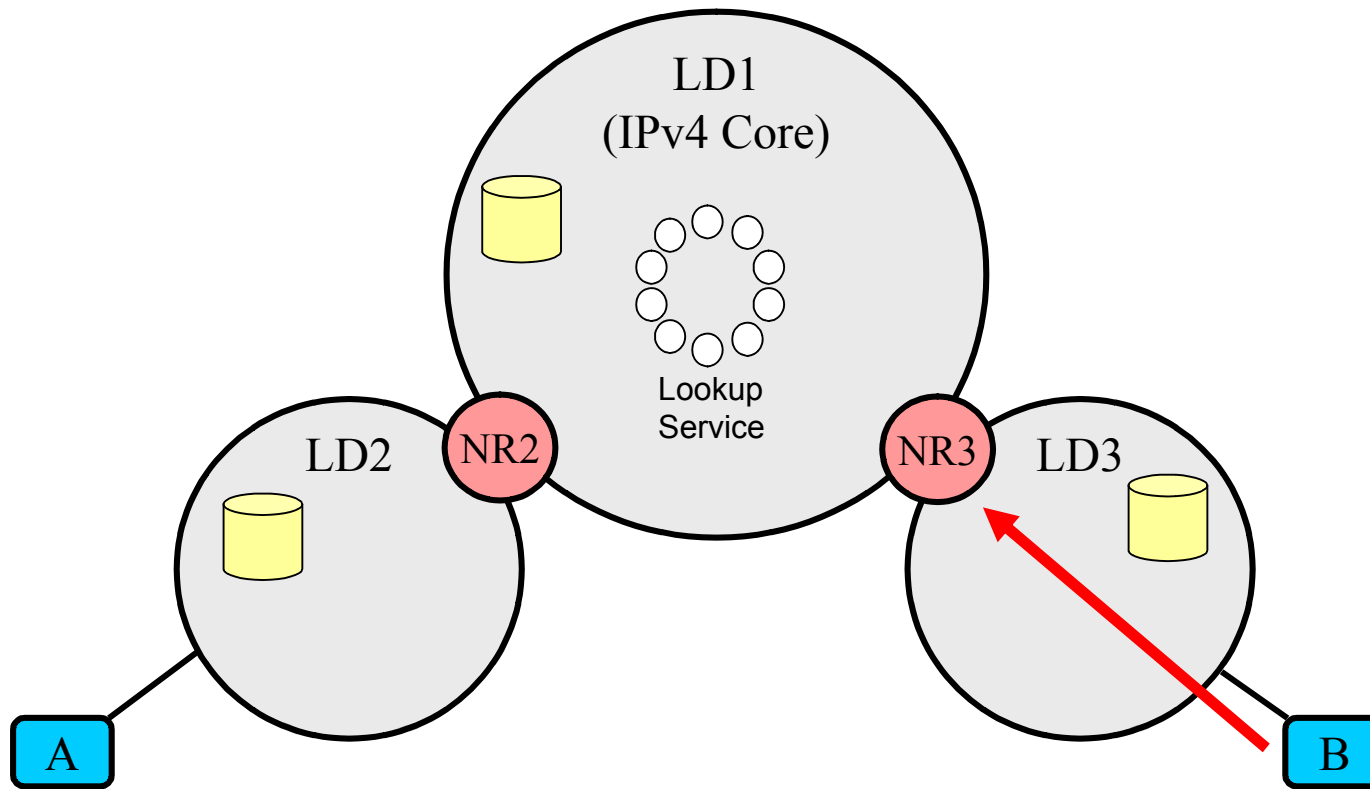
- NID NR 2
- NR NR 2



Nodes register locally with their NID router,
i.e. the NID router installs a mapping between LD internal IP and a
node's NID

Example

(Routing, inside or out of the edge)



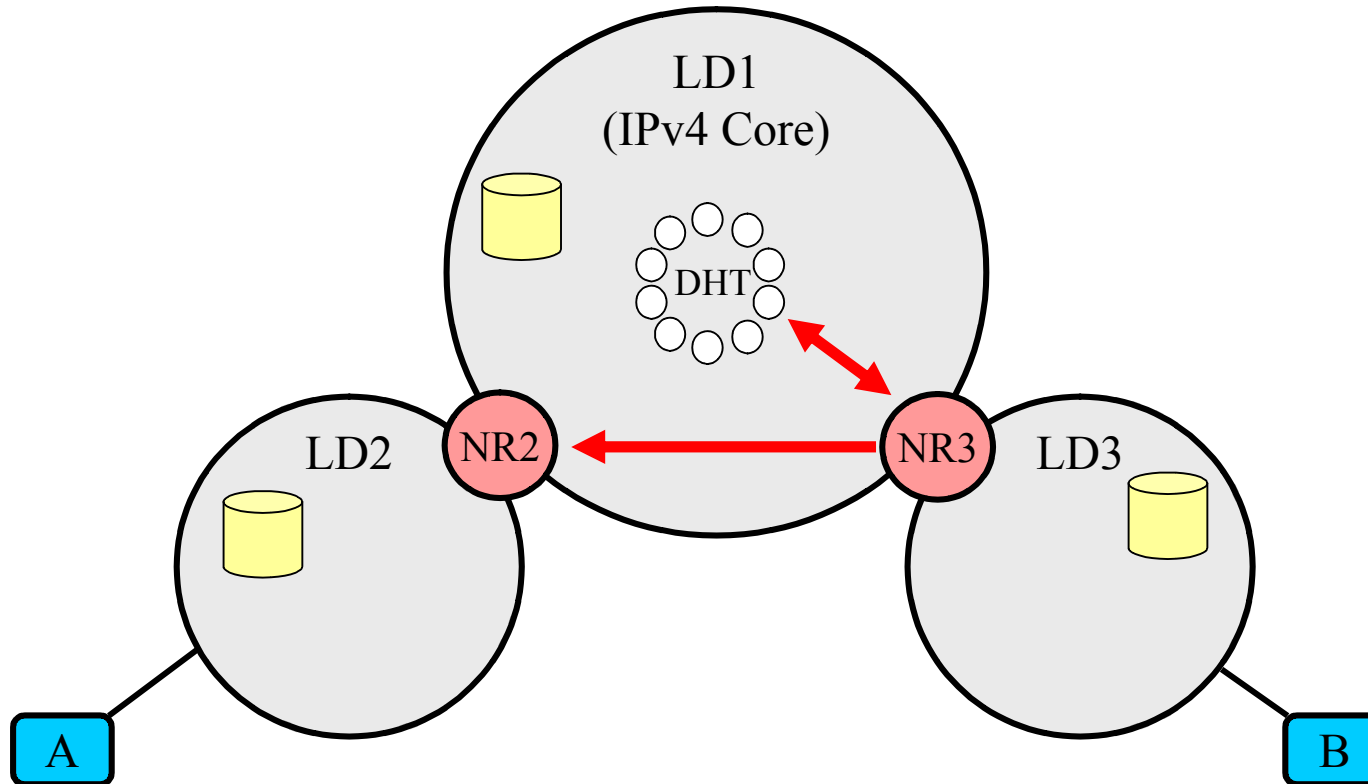
IPv4 Header

Node ID Header

ESP Payload

Destination = NR3	Destination NID = A Routing hint = NR2
-------------------	---	-----	-----

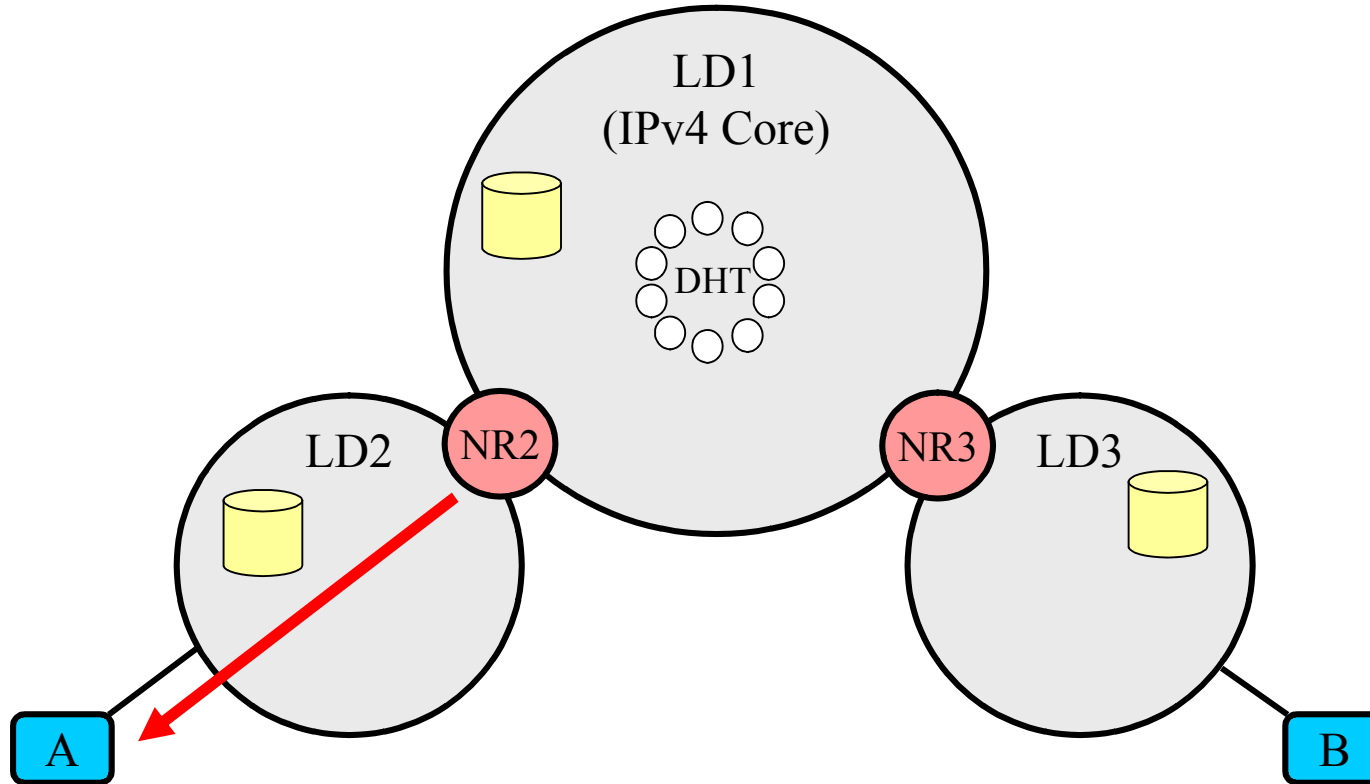
Example (Routing, traversing the core)



NR3 might not know how to get to NR 2 (routing hint). The DHT resolves the hint to an IP of NR 2 that knows how to route towards A. For consecutive packets soft state can/should be installed.

Example

(Routing, down an edge)



A is known. A's ID is resolved to the LD internal IP/address and delivered.

Implementing the NodeID Internetworking Architecture using HIP



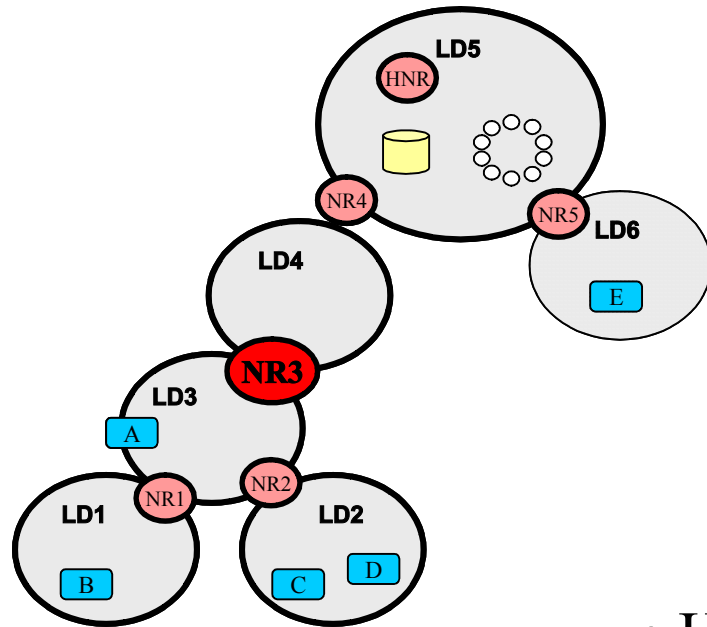
HIP ...

- ❖ ... already provides:
 - separation of identity and locator
 - host authentication
 - encrypted communication
 - support for mobility and multihoming
- ❖ ... does not provide:
 - easy bridging between networking technologies (“can only go where IP goes”)
 - support for administrative domains
 - cannot easily traverse NATs (though there is draft-ietf-hip-nat-traversal)

HIP extensions

- ❖ prototype based on HIP4inter.net implementation (formerly HIP4BSD)
- ❖ added new HIP control packet type NID_UPDATE
 - lowered implementation complexity, but
 - functionality could theoretically also be implemented in HIP_UPDATE
- ❖ added new HIP parameters (carried in NID_UPDATE)
 - NIDreg_req: NID registration request
 - NIDreg_res: NID registration response
- ❖ NID registration is recursively forwarded towards the home NR
- ❖ modified HIP SPINAT to implement NID routing
 - need to use other addresses on IP level
 - packets get addressed to next “NID-hop”, not to destination (except for last hop)

Storing NID Routing Entries



NID routing table for NR 3

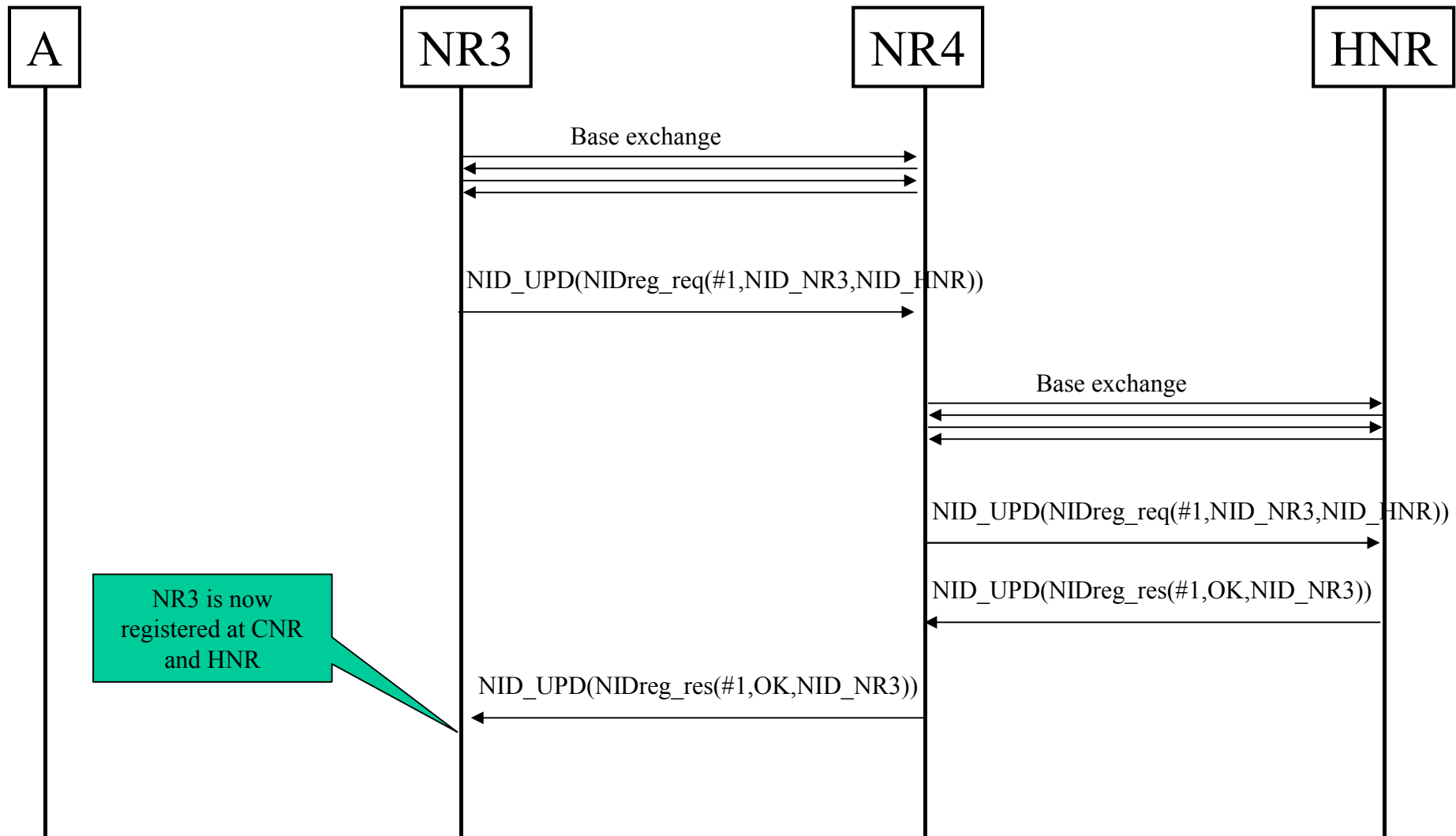
destination HIT	next-hop HIT
HIT(B)	HIT(NR1)
HIT(C)	HIT(NR2)
HIT(D)	HIT(NR2)
HIT(A)	HIT(A)
HIT(NR1)	HIT(NR1)
HIT(NR2)	HIT(NR2)
default	HIT(NR4)

HIT-IP table for NR3

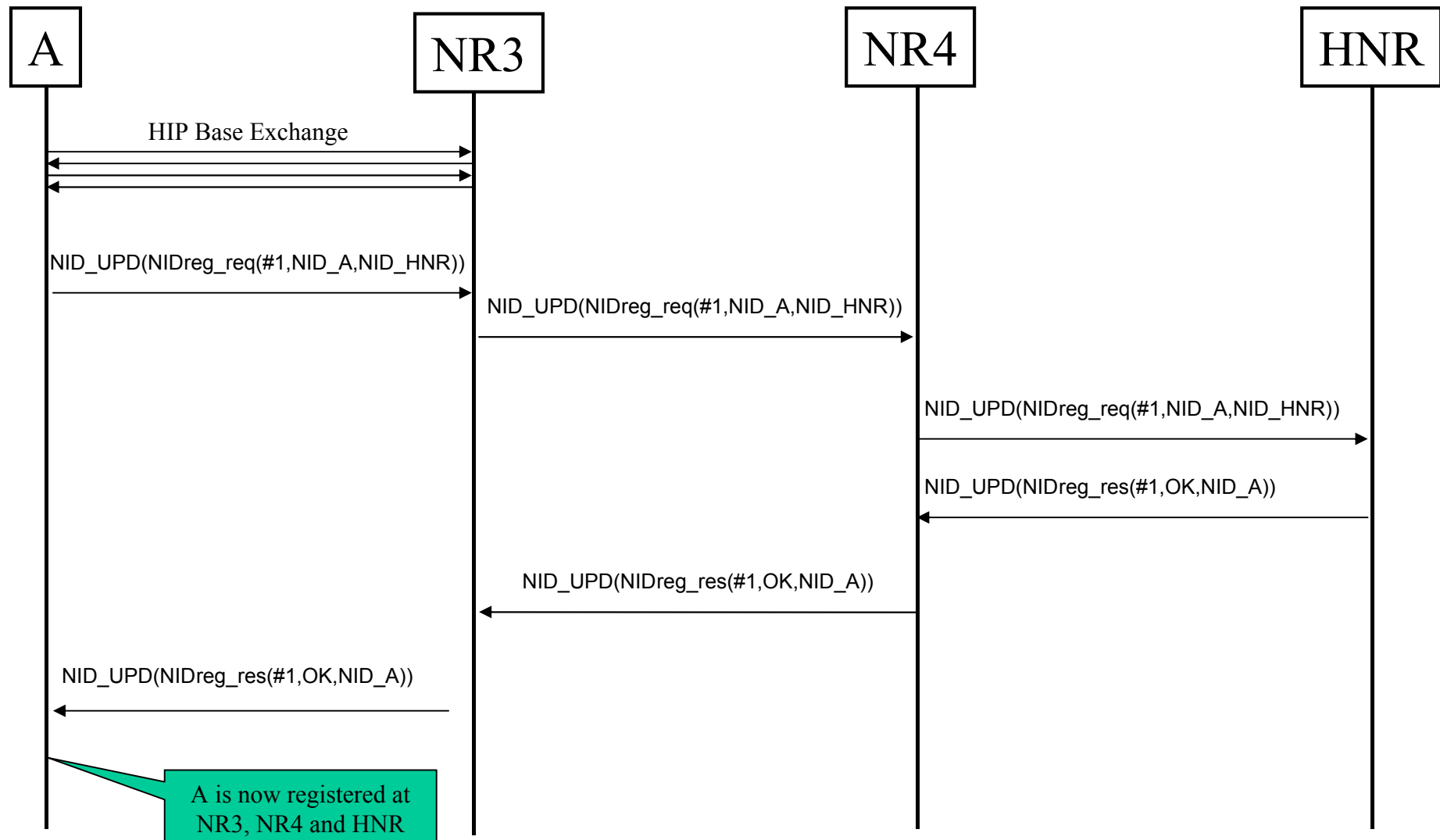
next-hop HIT	Locator
HIT(NR2)	IP_LD3(NR2)
HIT(NR1)	IP_LD3(NR1)
HIT(A)	IP_LD3(A)
HIT(NR4)	IP_LD4(NR4)

- HIP holds a HIT-IP mapping table
- NID also needs a table for mapping
dstHIT → next-hop HIT
- NID routing table filled through registration

Detailed MSC for NID Registration 1/2



Detailed MSC for NID Registration 2/2



Current Prototype features

- ❖ Recursive NID registration
- ❖ NID routing table setup based on NID registration
- ❖ HIP base exchange across multiple locator domains
- ❖ Bridging across heterogenous networking technologies (IPv4, IPv6, local and global address spaces)

Problems during Implementation

- ❖ Node ID Architecture allows overlapping address spaces
 - Need an LD identifier to distinguish where a packet came from/goes to
 - Local interface name/id would be sufficient for a prototype
 - BUT: within HIP daemon, we don't know about the interface being used, only about the IPs
 - Need multiple IP routing tables, one for each attached LD
 - Not implemented yet
- ❖ HIP is an end-to-end protocol
 - HIP mobility updates are sent end-to-end to each peer
 - In NID case, mobility updates need to be handled differently
 - E.g. for intra-LD mobility, only nodes within same LD need to know about changed IP
 - Implementation ongoing

Conclusion

- ❖ HIP solves some problems of the current/future networks
- ❖ Node ID builds on a similar concept
 - locator/identifier split
 - cryptographic identifiers
 - ➔ HIP used as basis for prototype
- ❖ Node ID aids integrating
 - administrative domains (LD concept)
 - heterogeneous network domains
 - migration towards new technologies (LDs hide interior technology)
- ❖ Is Node ID == HIP++ ??
 - HIP++ is one way to implement Node ID
 - stateful variant of Node ID, i.e. installing state along the path at connection setup
 - HIP provides good basis for rather static Node ID scenarios
 - in case of mobility, HIP UPDATE processing has to be changed quite extensively to meet Node ID requirements
 - rather stateless approach could be better suited for more dynamic scenarios
- ❖ Current prototype implements some features of Node ID, but more work required

Pointers

- ❖ *Node Identity Internetworking Architecture*. S. Schuetz, R. Winter, L. Burness, P. Eardley, B. Ahlgren. draft-schuetz-nid-arch-00 (work in progress), Sept. 2007
- ❖ *A Node Identity Internetworking Architecture*. Bengt Ahlgren, Jari Arkko, Lars Eggert and Jarno Rajahalme. 9th IEEE Global Internet Symposium, Barcelona, Spain, April 28-29, 2006.
- ❖ HIP4Inter.net project: <http://www.hip4inter.net>
- ❖ Ambient Networks project: <http://www.ambient-networks.org>

Thank you!

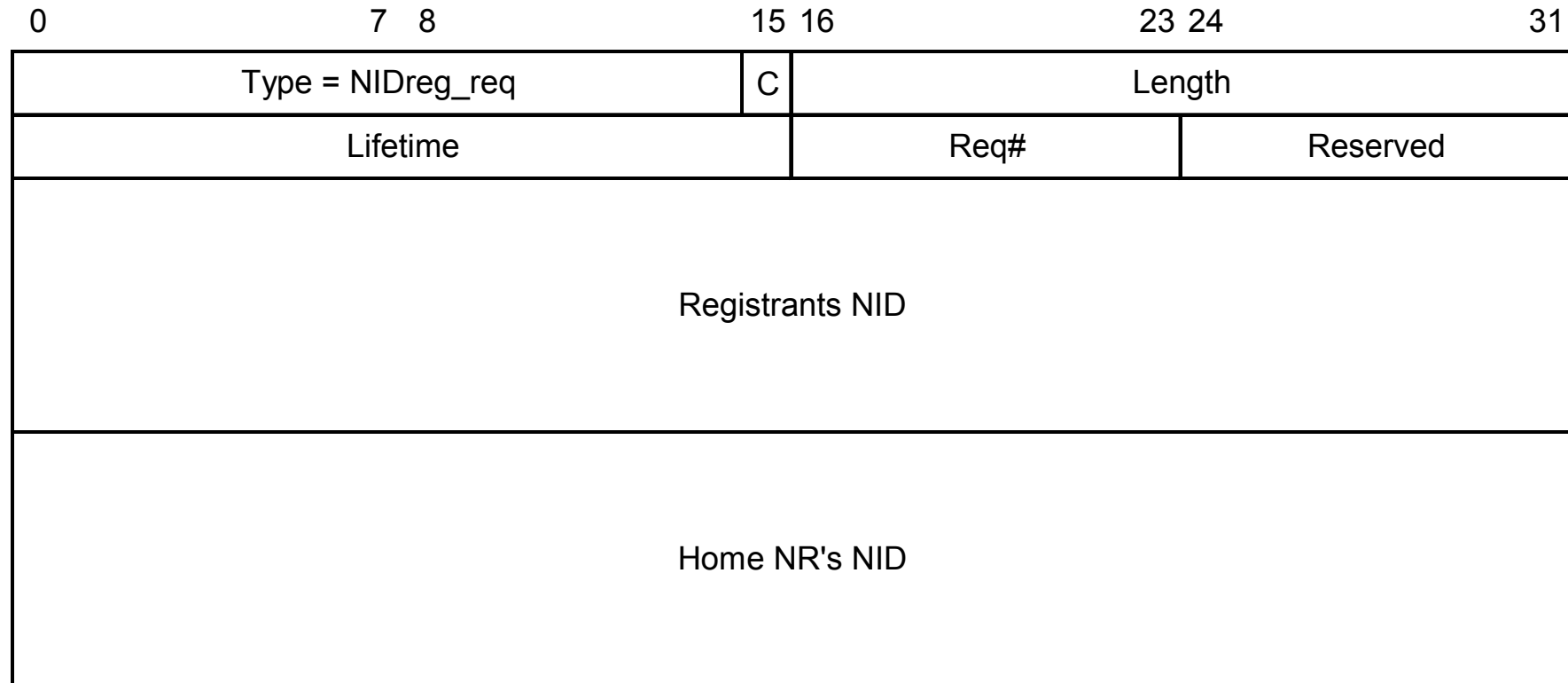
Question?



Backup slides



NIDreg_req



NIDreg_res

