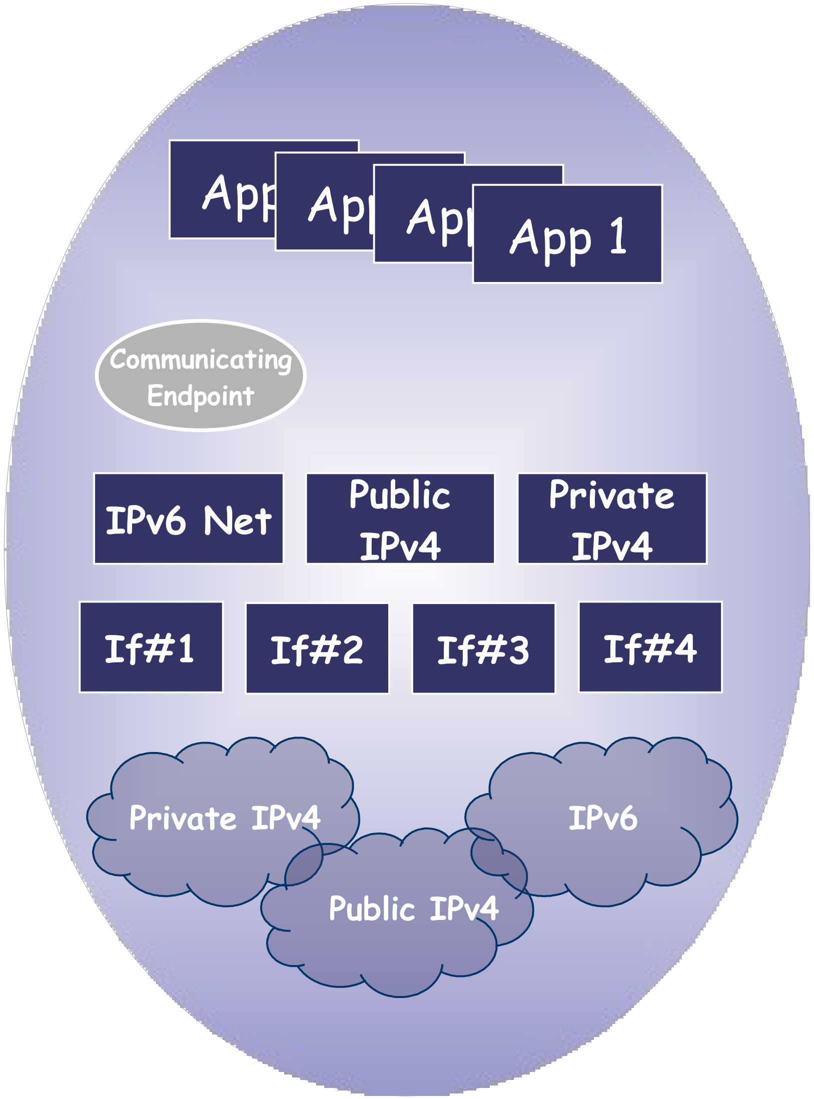
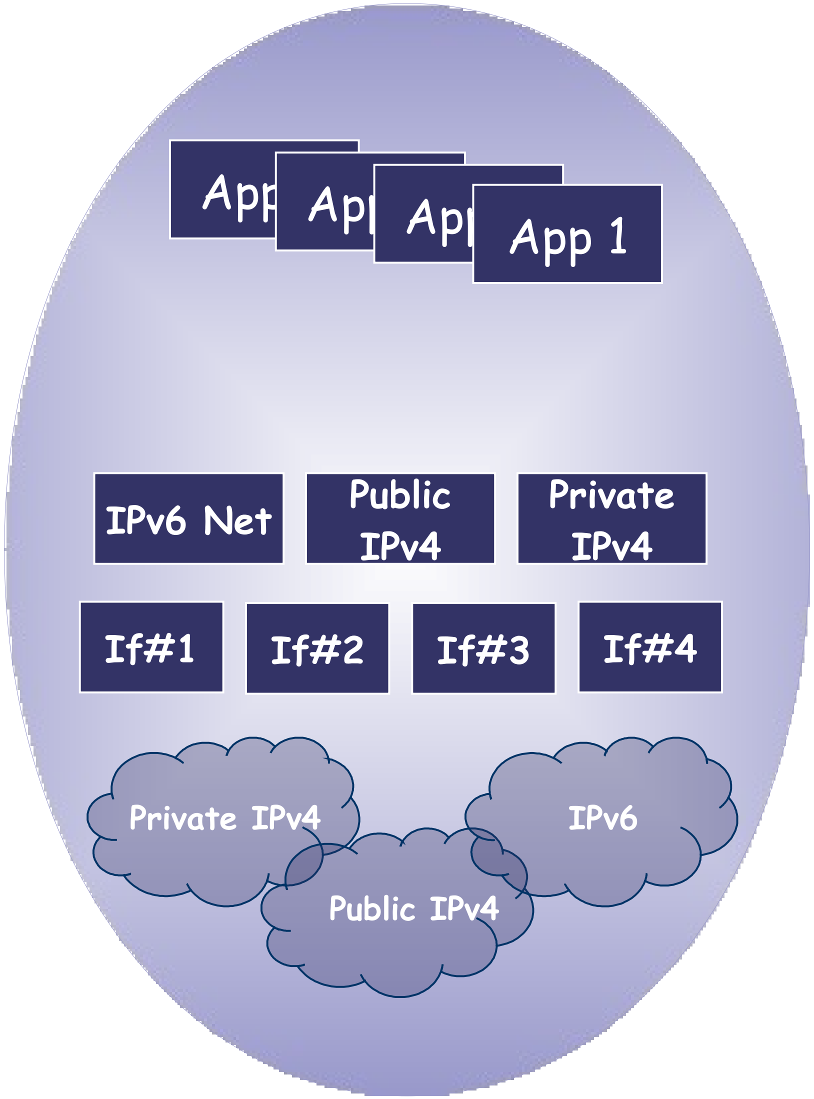
Locator - Identifier Split (LIS)

LIS: Introducing Notion of an Endpoint (Vision)





LIS: Endpoints (Vision Definition)

- Endpoint is defined as a logical object terminating communication and handing in payload to an application for application specific processing
- Endpoint is required to have capability of being attached to different access networks, IP networks and any virtual overlay networks operated above physical infrastructures
- End point is required to have capability of changing its attachment to any of the above mentioned networks while providing undisrupted communication on behalf of applications
- Mobility is viewed as the system capability to change binding between a movable object – endpoint and dynamically changing network specific ID, e.g. IP address

LIS: Endpoints (Back to the Reality)*

- The notion of "communicating endpoint" is pretty much about networked objects and their naming
- Bindings and name spaces
  - The association between a name and an object is called a "binding"; bindings may also map from one kind of name for an given object, to another name
  - It is important to realize that a single instance of an object (i.e. a member of an "object class") may have more than one name
- Structure and Representation of Names
  - Names may have multiple "representations", or ways of encoding the same individual name
  - A collection of bindings in which a system records and looks up the connections is called a "context"

* Based on J. Noel Chiappa "Endpoints and Endpoint Names: A Proposed Enhancement to the Internet Architecture".

LIS: Endpoints Examples in TCP/IP Stack

- IP Addresses
  - Used directly by the routers, as the data in the packet which the routers look at to forward (i.e. process) user data packets.
  - Used to name the place in the internetwork (i.e. the destination) to which the packet is to be directed to; i.e. the place in the internetwork where that host is connected; this is referred to as the "network attachment point", or "interface").
  - Used to name the host which is doing the end-end communication. They are the only information identifying the hosts on each end which appear in any TCP/IP packet headers. They are thus part of the identification of a TCP connection, together with a TCP port (which simply disambiguates among multiple TCP connections on a single host)
  - In fact, however, IP addresses are basically the **only** name used throughout the TCP/IP architecture

Naming and Addressing: What to Name?

Static/Persistent Name vs. Dynamic/Changeable

http:// www.computer.com /whatever/directory/ file.html

- http:// → Service Name → Static → Mapped to Port Number → Dynamic
- www.computer.com → Computer Name → Static → Mapped to IP Address → Dynamic
- /whatever/directory/ → Directory Path Name → Dynamic
- file.html → File Name → Dynamic

Think of two issues:

1. Naming and Addressing
2. Dynamic Bindings

LIS: Naming and Addressing References

- Danny Cohen, "On Names, Addresses and Routings", http://www.isi.edu/in-notes/ien/ien23.txt Internet Experiment Note 23, University of Southern California, Information Sciences Institute, Marina Del Rey, Calif., 1978.
- E.C. Rosen. Logical Addressing, IEN (Internet Experiment Note) 183, University of Southern California, Information Sciences Institute, May 1981.
- Hauzeur, Bernard M. A Model for Naming, Addressing, and Routing, ACM Transactions on Information Systems Vol 4, No. 4 (Oct. 1986), pp. 293-311
- Jerome H. Saltzer, "On The Naming and Binding of Network Destinations," in Local Computer Networks, edited by P. Ravasio et al., North Holland, Amsterdam, 1982, pp. 311-317. Also available as IETF RFC 1498, http://www.ietf.org/rfc/rfc1498.txt
- Sunshine, Carl A., "Addressing Problems in Multi-Network Systems", in Proc. IEEE INFOCOM 82, Las Vegas, Nevada, March 30-April 1, 1982
- Cerf V., Cain E. The DoD Internet Architecture Model, Computer Networks 7, October 1983, pp. 307-318.
- E. Lear, Name Space Research Group, IRTF, draft-irtf-nsrg-report-02.txt "What's In A Name: Report from the Name Space Research Group", Expires: August 14, 2002. http://search.ietf.org/internet-drafts/draft-irtf-nsrg-report-02.txt
- Endpoints and Endpoint Names: A Proposed Enhancement to the Internet Architecture. J. Noel Chiappa. Internet draft dated 1999 available at http://users.exis.net/~jnc/tech/endpoints.txt

LIS: Problems With the Existing IP Object Naming

- The fact is that one name (the IP address) is used to identify two completely different things (the host and the interface) [AKA Semantic Overloading]
- According to Saltzer (see References)
  - "One way or another, the permanent binding of attachment point name to [host] name has made some function harder to accomplish…."
- Problems with mobility arise in the first hand
- A new fundamental object - the Endpoint is needed
- An endpoint is a new concept, a fundamental object of networking, and requires an independent existence

LIS: What To Use As an Endpoint Name

- For example, Uniform Resource Names (URN)
- Uniform Resource Names (URN) are intended to serve as persistent, location-independent, resource identifiers and are designed to make it easy to map other name spaces (which share the properties of URN) into URN-space. [RFC 2141, "URN Syntax"]
- In addition to locating resolvers, the NAPTR provides for other naming systems to be grandfathered into the URN world, provides independence between the name assignment system and the resolution protocol system and allows multiple services (Name to Location, Name to Description, Name to Resource,…) to be offered [RFC 2168 "Resolution of URI using the Domain Name System"]

Naming, Naming, Naming, ...

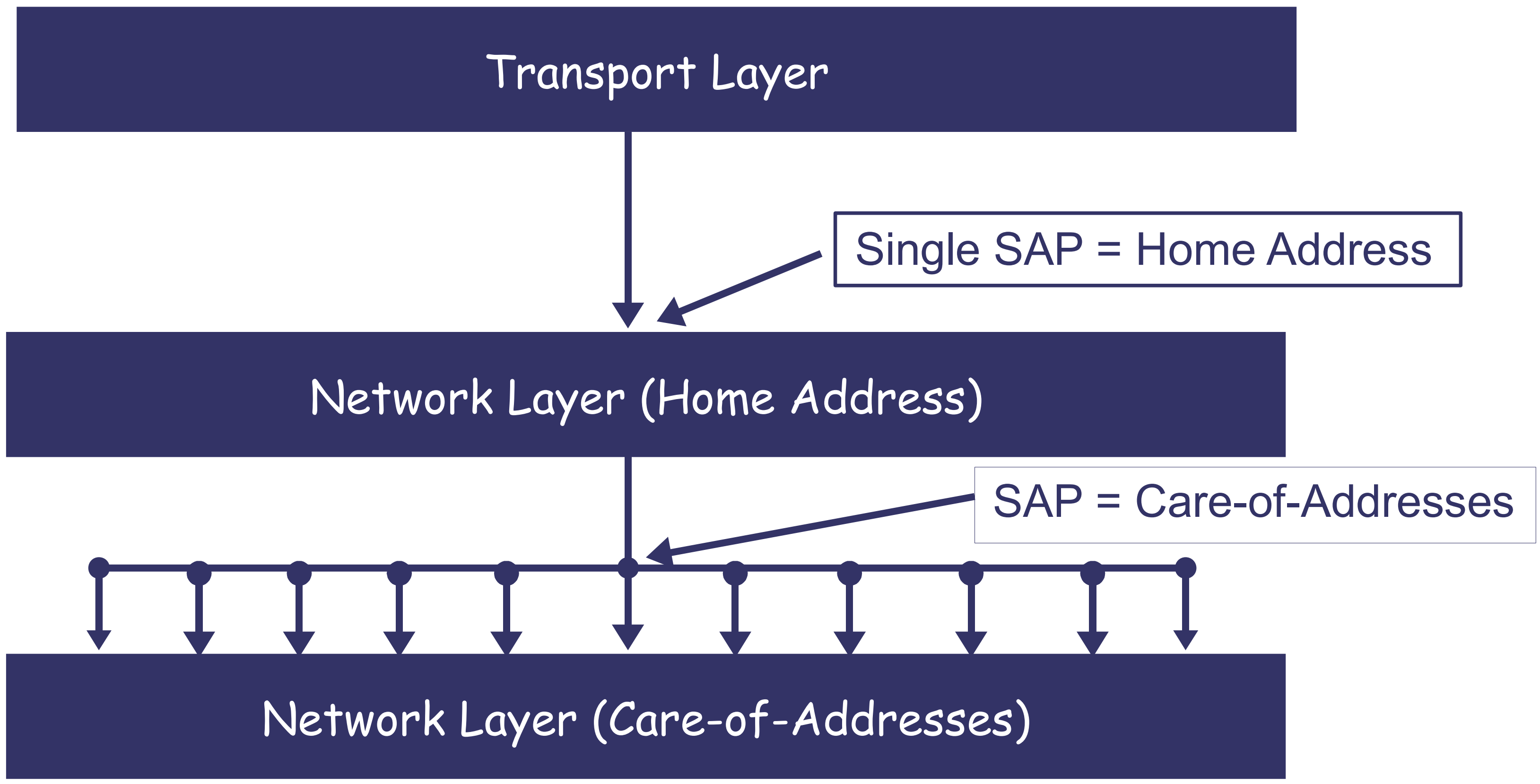
- How would a stack name improve the overall functionality of the Internet?
- What does a stack name look like?
- What is its lifetime?
- Where does it live in the stack?
- How is it used on the end-points?
- What administrative infrastructure is needed to support it?
- What would the resolution mechanisms be, or what characteristics of a resolution mechanisms would be required?

Lear, E. and R. Droms, "What's In A Name: Thoughts from the NSRG", Work in Progress, September 2003

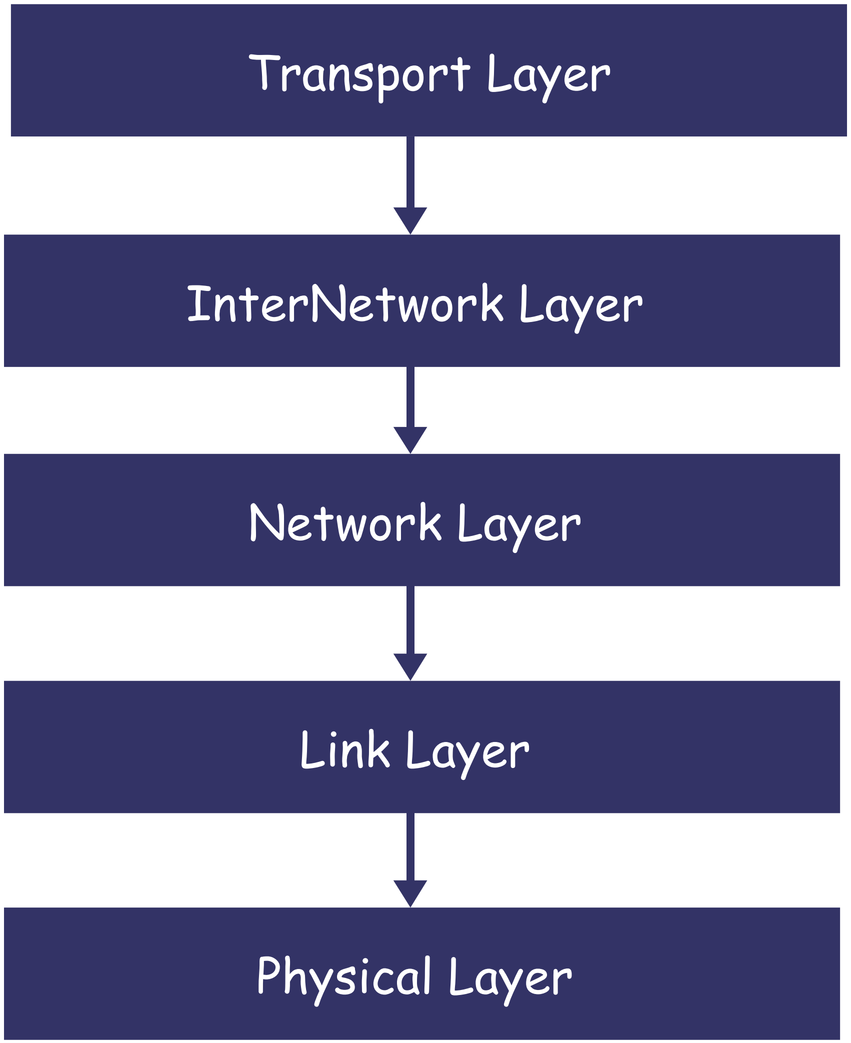
LIS: Exercise

- What is used as an endpoint name in MIPv4/6?
- How does the endpoint for MIPv4/6 look like?
- Draw the picture thinking of endpoints, names, dynamic/static bindings, state which has to be preserved and anything else you would like

Semantic Overloading of IP Addresses

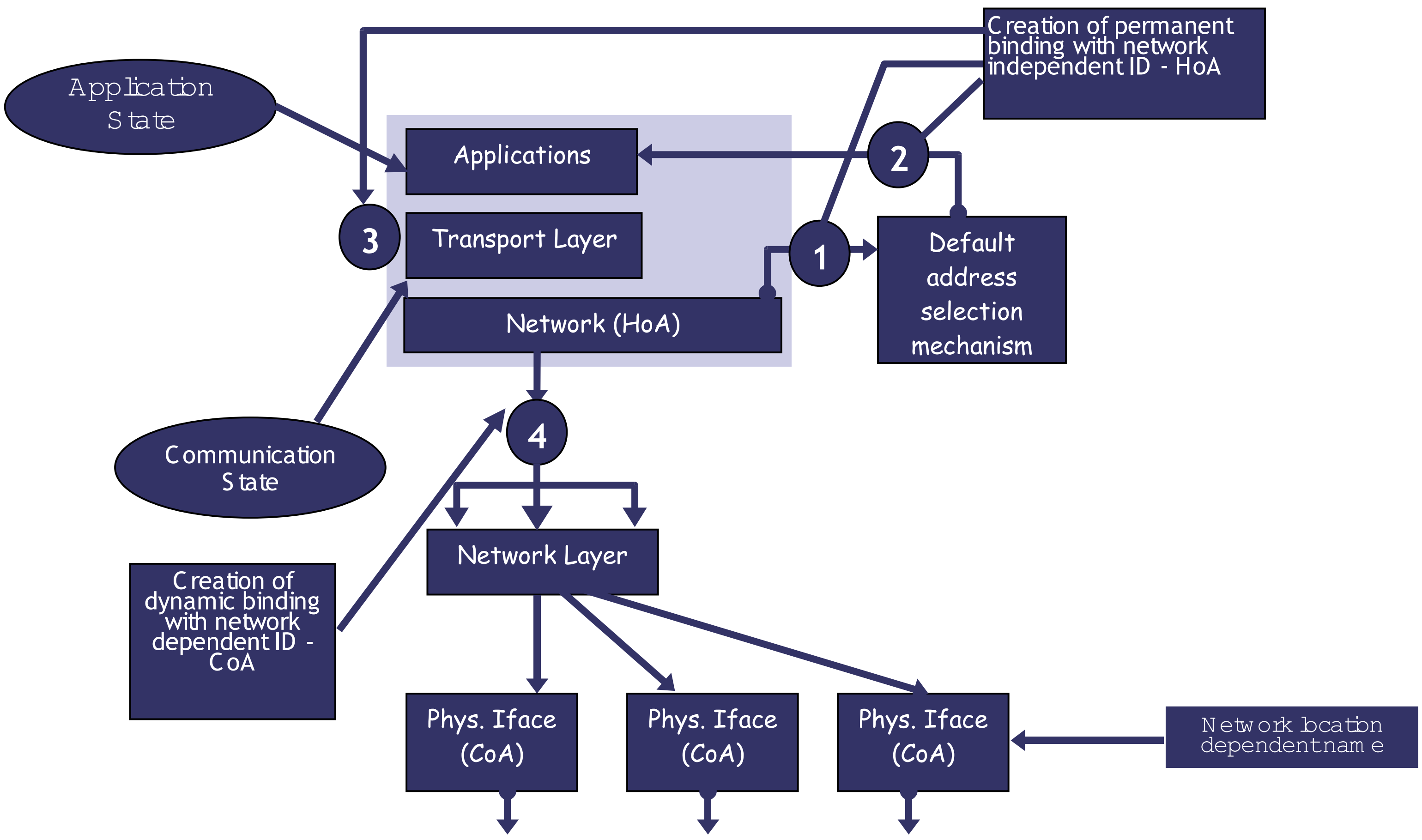


Forgotten InterNetworking Layer



- Cerf V., Cain E. "The DoD Internet Architecture Model," Computer Networks 7, October 1983, pp. 307–318.

LIS : MIPv4/6 Endpoint



LIS: Solution Examples

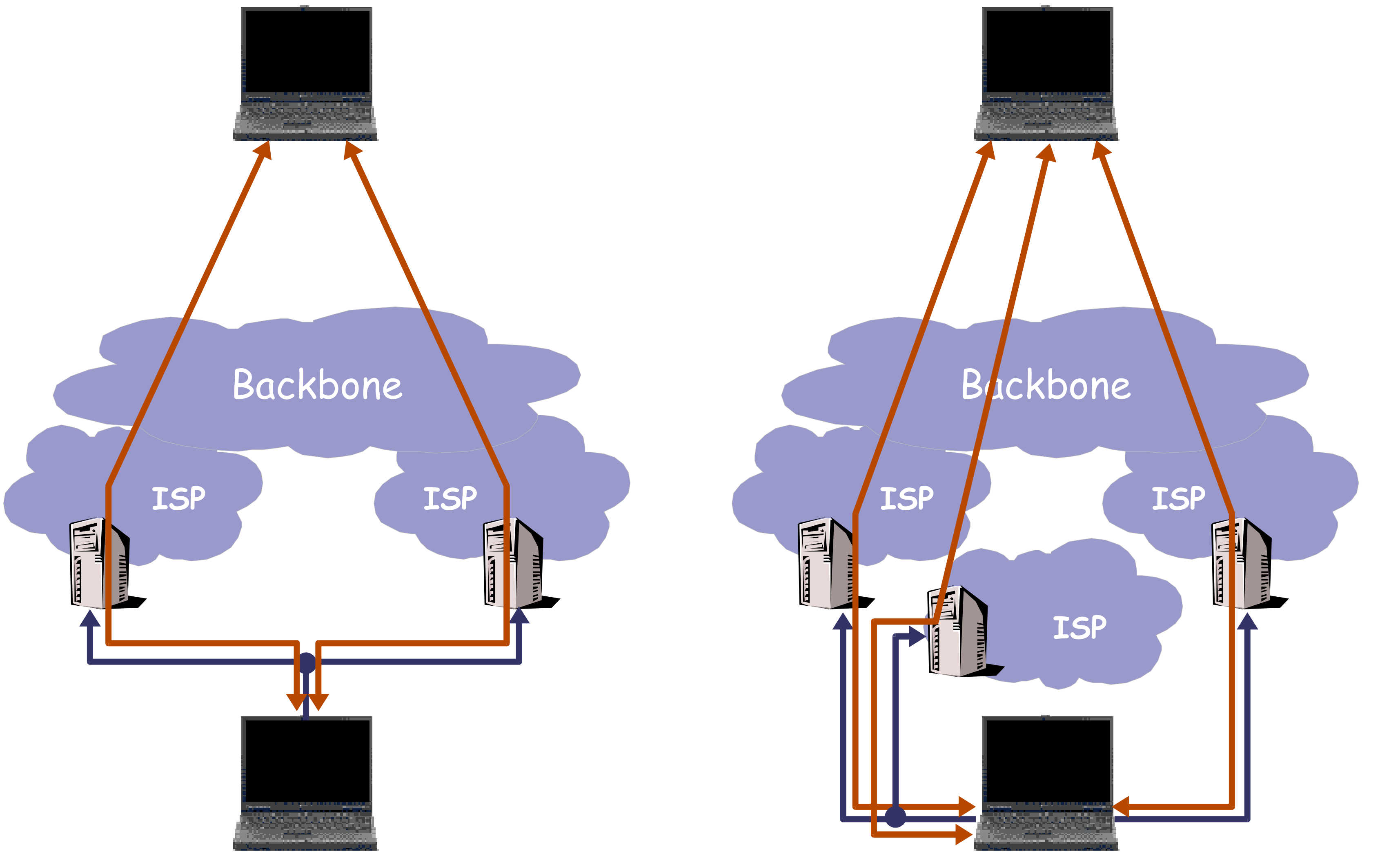
- Walk through two major solutions
  - Level 3 Multi-homing Shim Protocol for IPv6 (SHIM6)
    - Step aside to multi-access and IETF MEXT activities
  - Host Identity Protocol (HIP)

Level 3 Multi-homing Shim Protocol for Ipv6 (SHIM6)

SHIM6: Overview

- The draft "Shim6 Protocol" defines the Shim6 protocol, a layer 3 shim for providing locator agility below the transport protocols, so that multi-homing can be provided for IPv6 with fail-over and load sharing properties
- The Shim6 protocol is a site multi-homing solution in the sense that it allows existing communication to continue when a site that has multiple connections to the internet experiences an outage on a subset of these connections or further upstream. However, Shim6 processing is performed in individual hosts rather than through site-wide mechanisms.

SHIM6: Multi-Homing vs Multi-Access



Multi-Prefixed

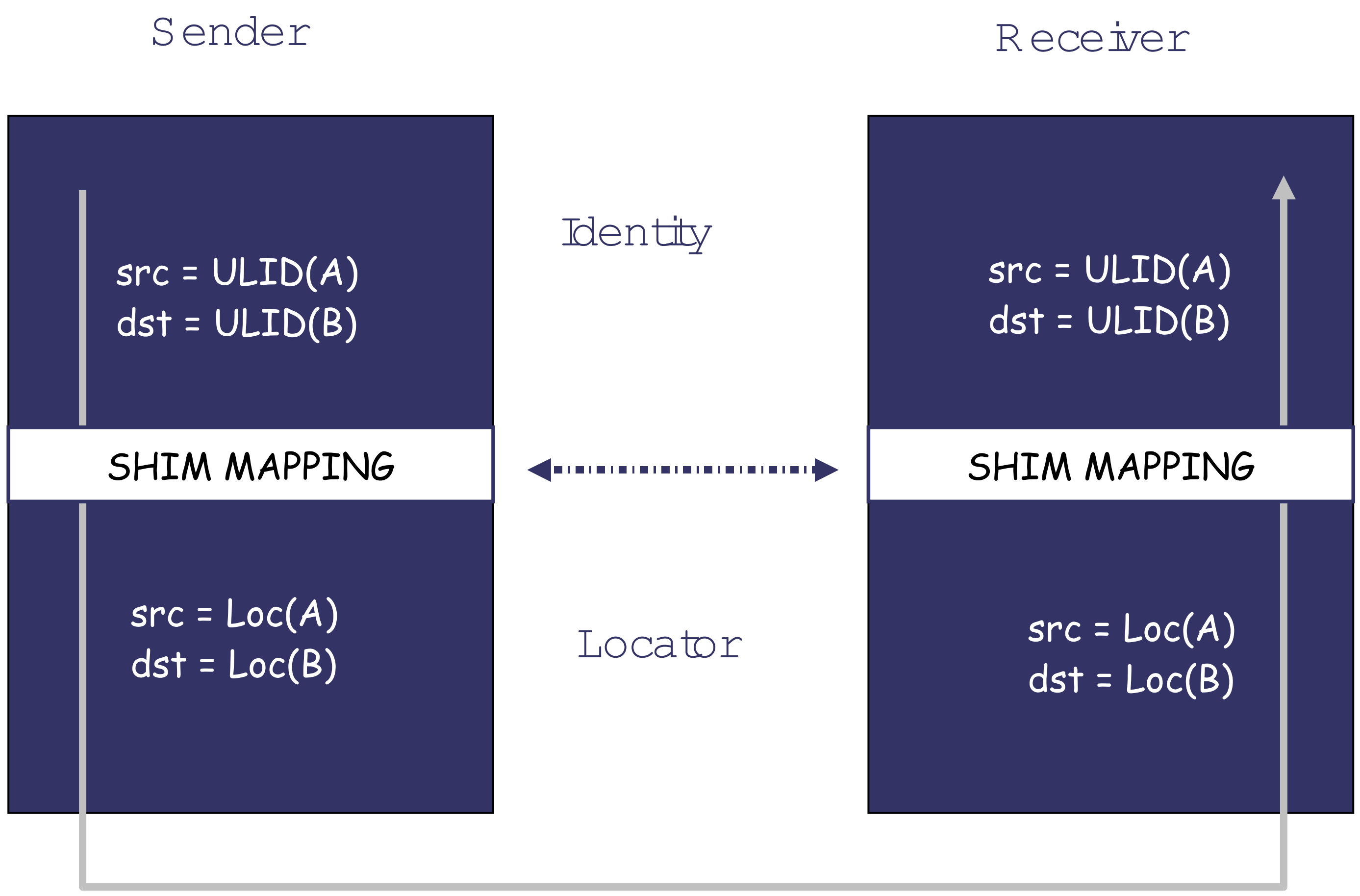
Multi-Interfaced

Combination of both

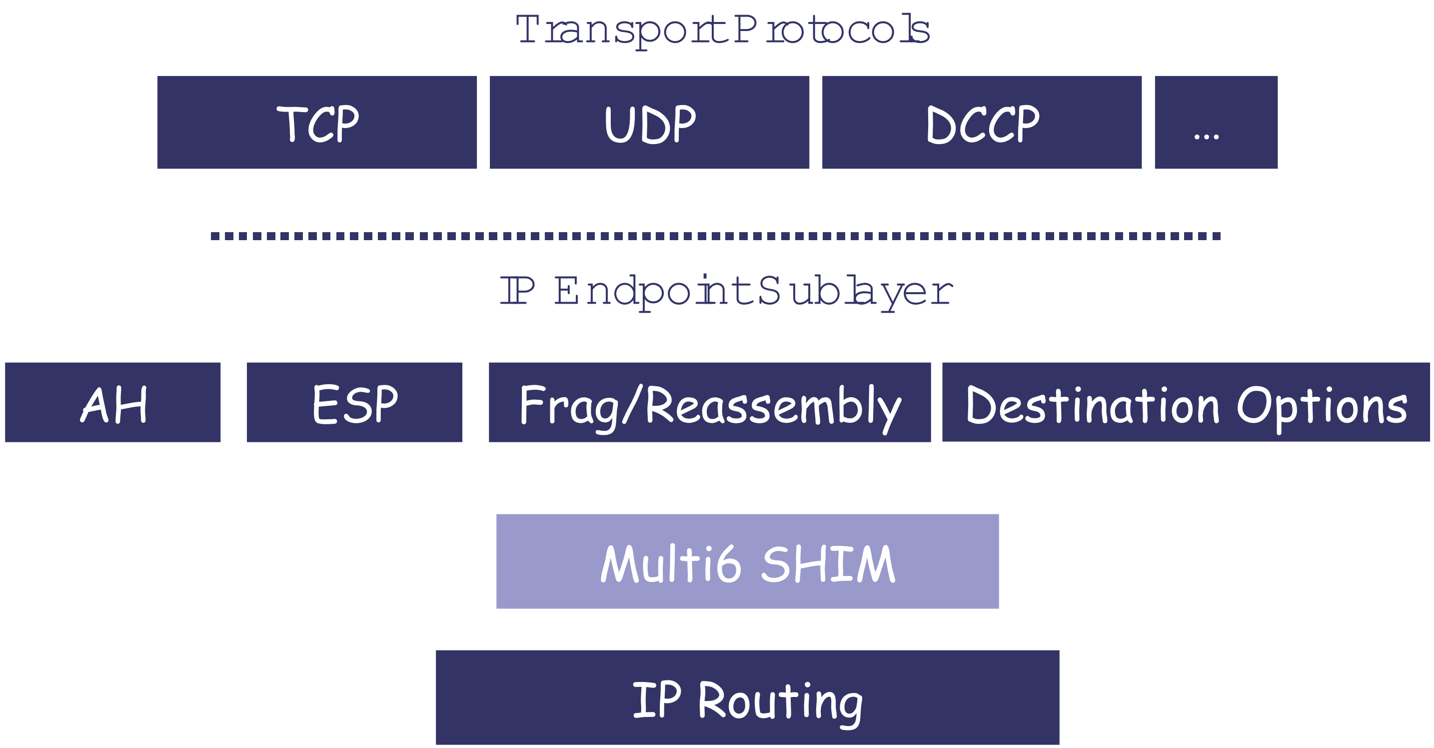
SHIM6: Overview

- Locators as Upper-layer Identifiers (ULID)
  - The approach does not introduce a new identifier name space but instead uses the locator that is selected in the initial contact with the remote peer as the preserved Upper-Layer Identifier (ULID)
  - The ULID selection is performed as today's default address selection as specified in RFC 3484
  - Using one of the locators as the ULID has certain benefits for applications which have long-lived session state

SHIM6: ID-Locator Split Approach

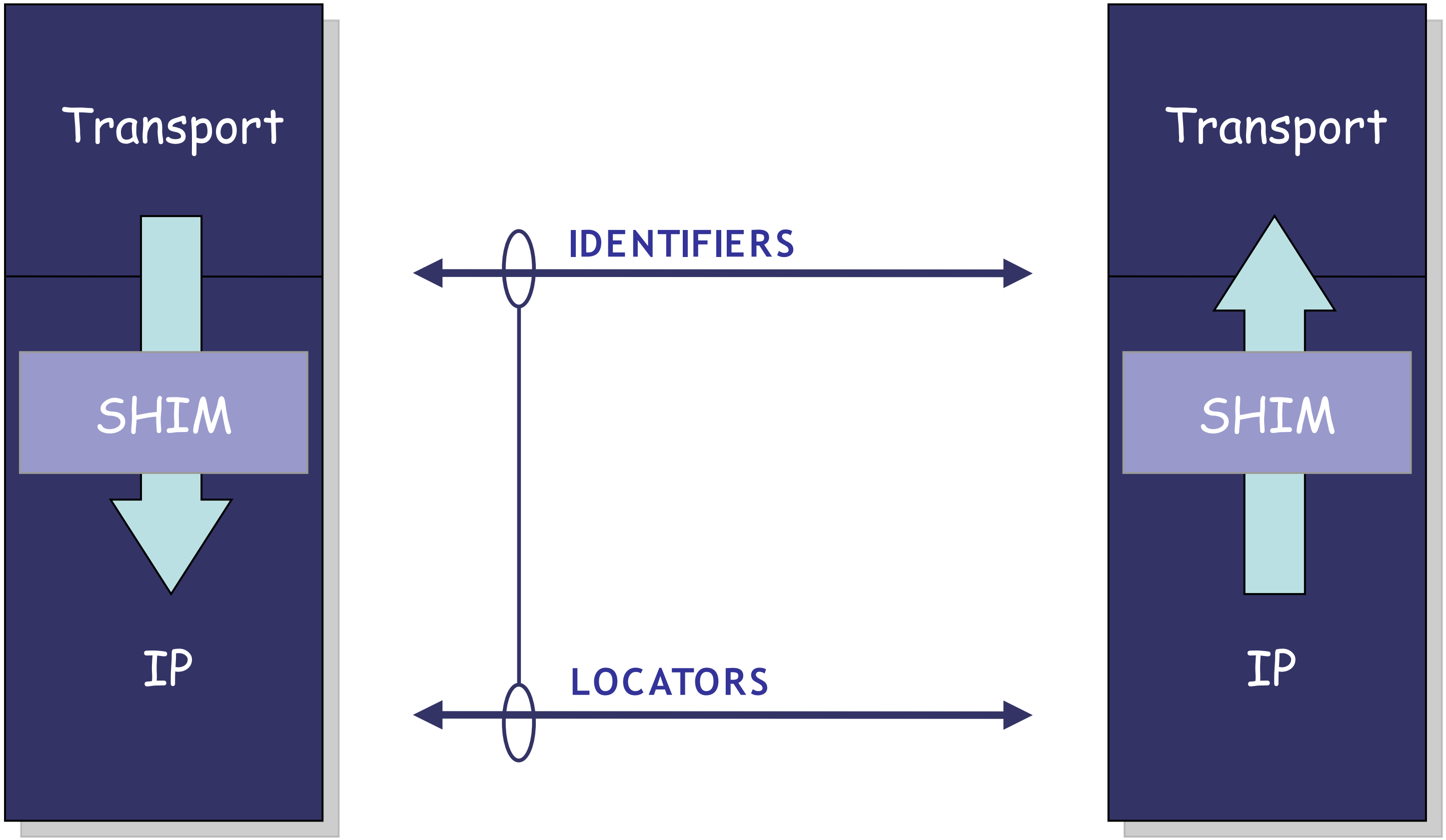


SHIM6: Sub-Layer Placement



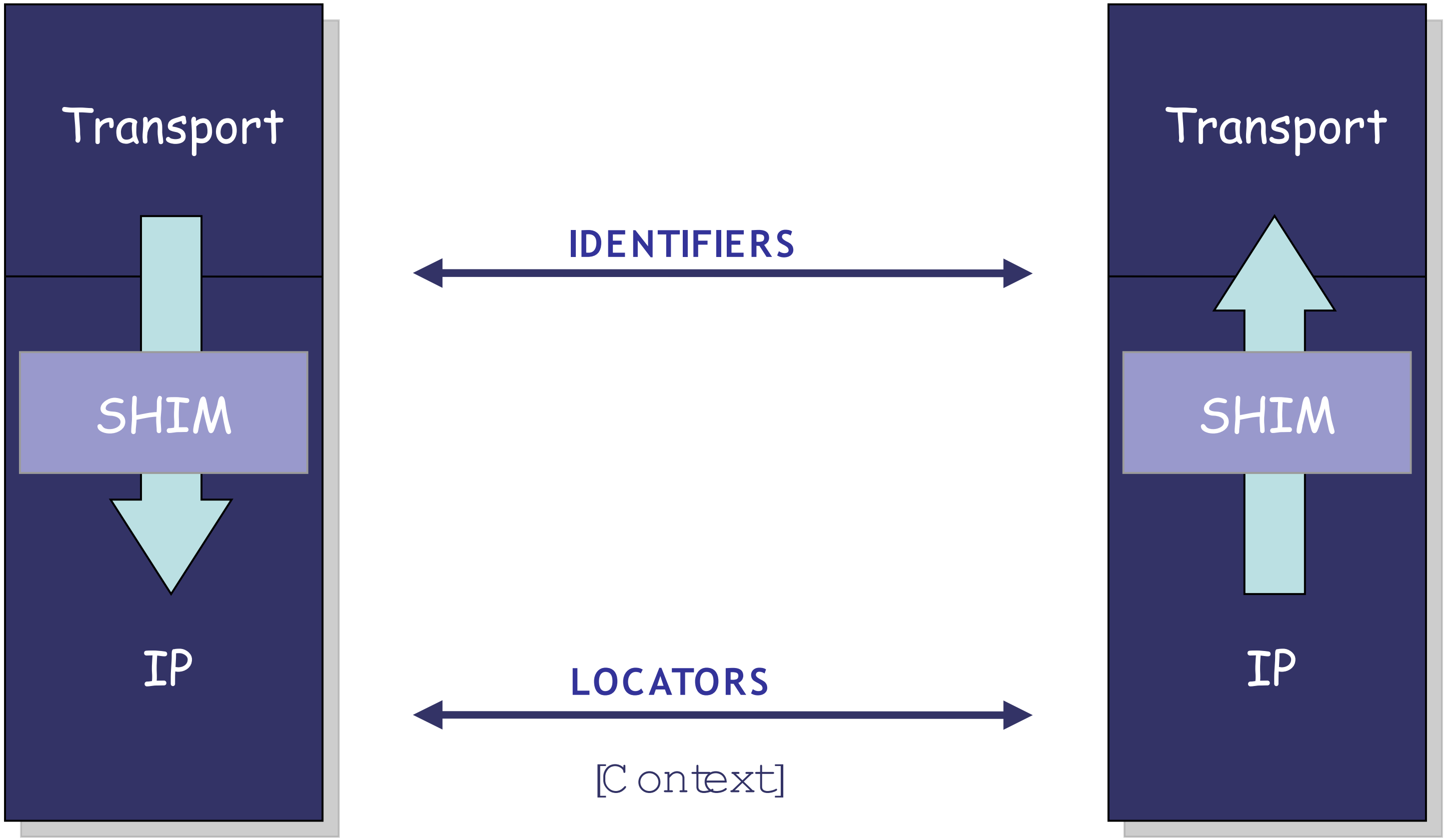
SHIM6: Architecture

- Initial Contact
  - No SHIM state active
  - Locator Selection using RFC 3484
  - Locators and Identifiers are equivalent



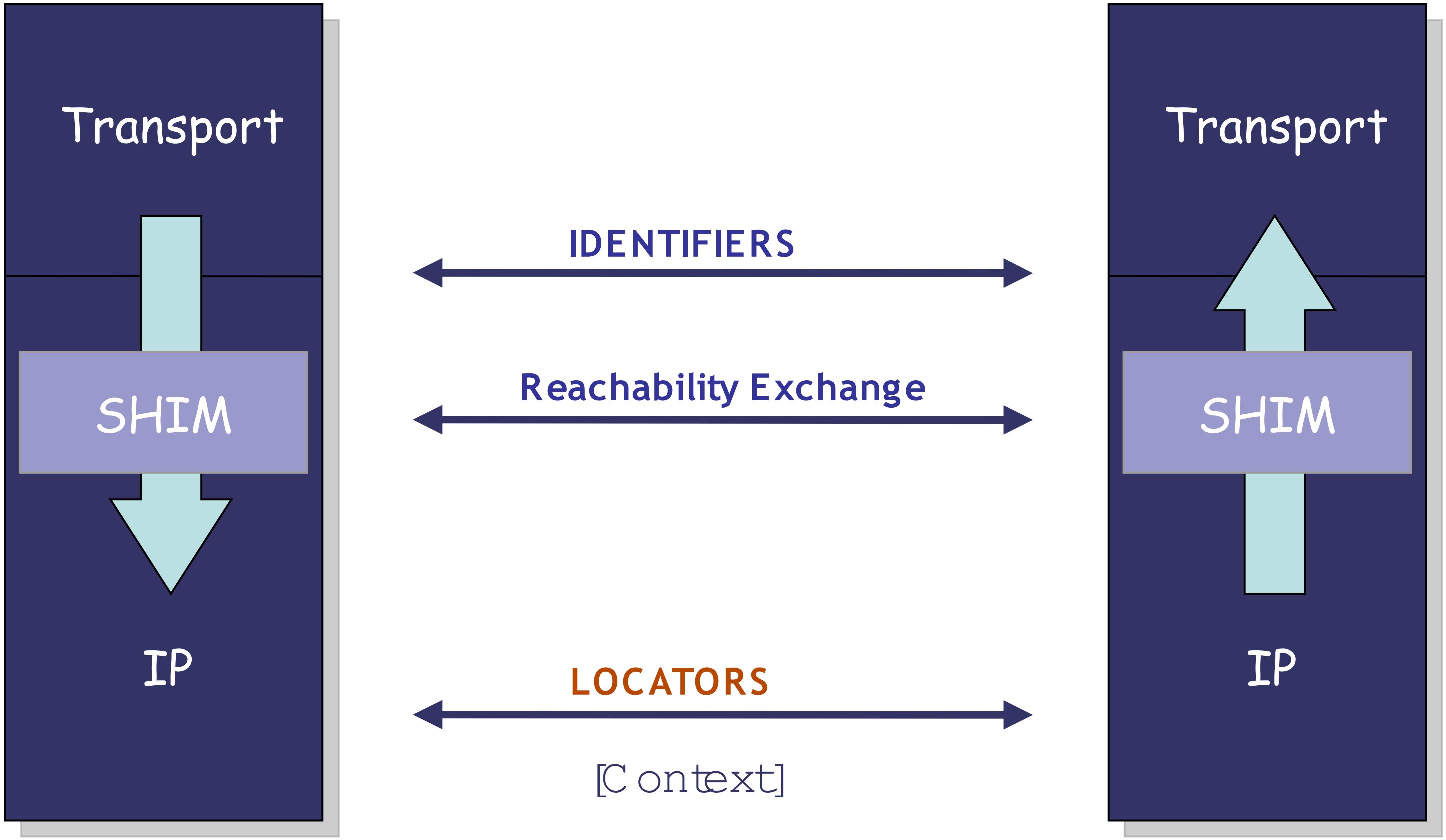
SHIM6: Architecture

- SHIM6 Activation
  - SHIM6 active
  - Current Locators Set exchanged
  - Locators and identifiers are equivalent



SHIM6: Architecture

- SHIM6 Locator Failure and Recovery
  - Detect Locator Failure
  - Explore for functioning locator pair
  - Use new locator pair, preserve identifier pair



SHIM6: Control Elements

- Initial handshake (4-way) and locator set exchange
- Locator list updates
- Explicit locator switch request
- Keep-alive
- Reachability probe exchange
- No-Context error exchange

Shim6: State Maintenance

- Detecting network failure
  (How does a host know that its time to use a different source and/or destination locator?)

  - Single per-endpoint state vs per session state
  - Heartbeat within the session
  - Shim heartbeat
  - Modified transport protocol to trigger locator change
  - Host/Router interaction to trigger locator change
  - Application time-frame vs network time-frame ???
  - Failure during session startup and failure following session establishment

SHIM6: Not Only Network Failure

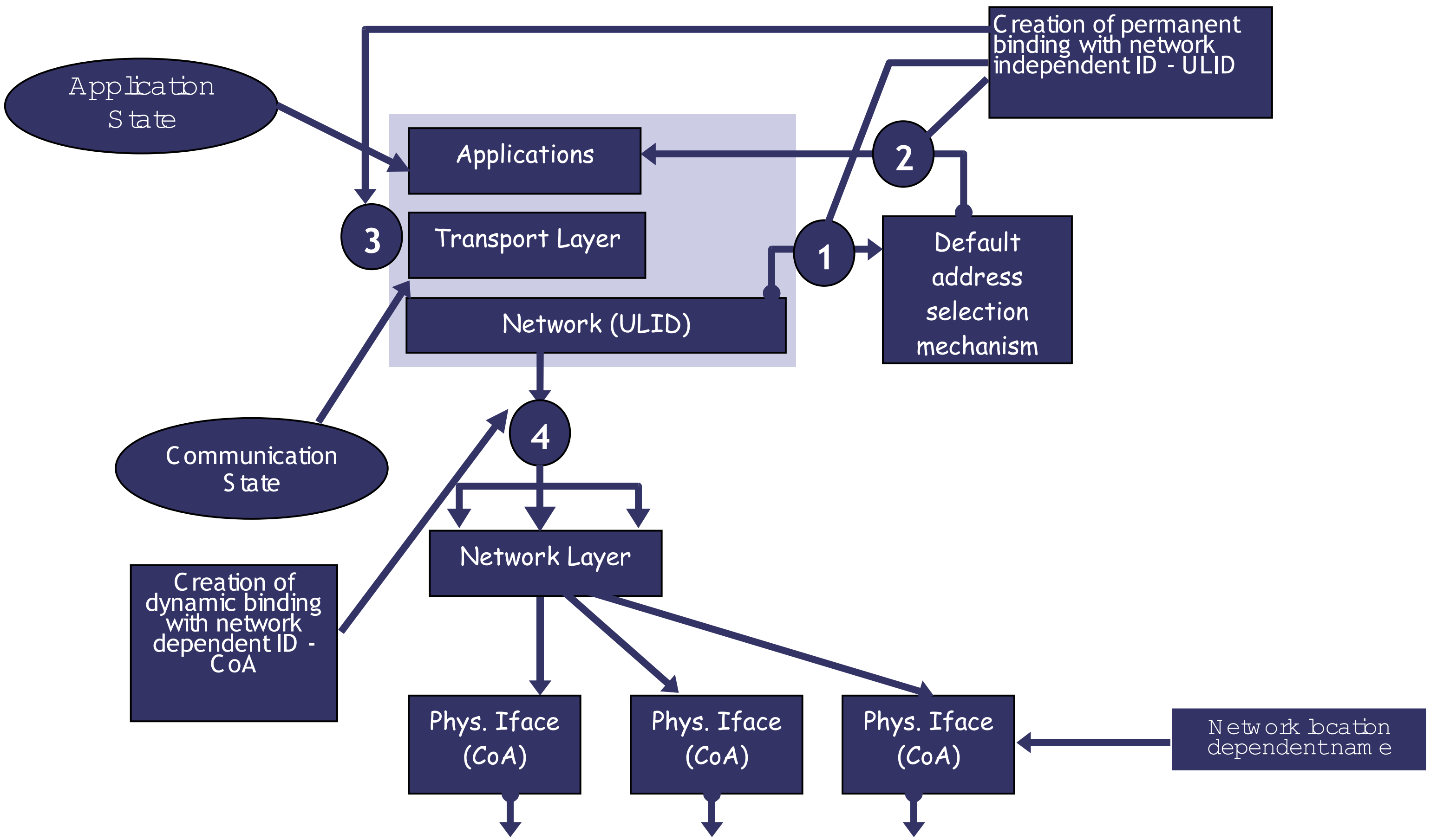
- Why an interface would be changed
  - New interface arrival
  - Interface removal
  - Change of interface configuration (multi-access vs multi-homing)
  - Change of interface characteristics
  - Wireless LAN signal strength goes below threshold

SHIM6: State Maintenance

- Locator Failure Triggers
  - Possible triggers include failure of upper level keepalive signal to the SHIM layer, explicit trigger from upper level, ICMP error, explicit SHIM level reachability failure
    - Any or defined subset?
  - Re-Homing may involve exhaustive pair exploration to establish a new viable locator pair
    - Reactive or Continuous Probe?
  - Signal upper level protocol of path state change
    - "Active" end state change procedure
    - "Passive" end state change procedure

SHIM6: Endpoint

- Would it be different as compared to MIPv6 endpoint? If yes then why?



Multi-Access Introduction

- Devices with multiple interfaces
- Interfaces can be configured and active at the same time
- Two natural features requiring support for multi-access devices
  - Possibility to move flows (e.g. TCP connections) between interfaces (change of characteristics, wireless coverage outages, etc.)
  - Possibility to send traffic simultaneously through multiple interfaces (Per flow basis? Per packet basis?)
- Multi-access is the use case for mobility
- Each interface can be mobile independently on each other

Multi-Access: Goals

- Ubiquitous Access *
- Flow Redirection *
- Reliability *
- Load Sharing *
- Load Balancing *
- Preference Settings *
- Aggregate Bandwidth *
- **Security (DoS prevention)**

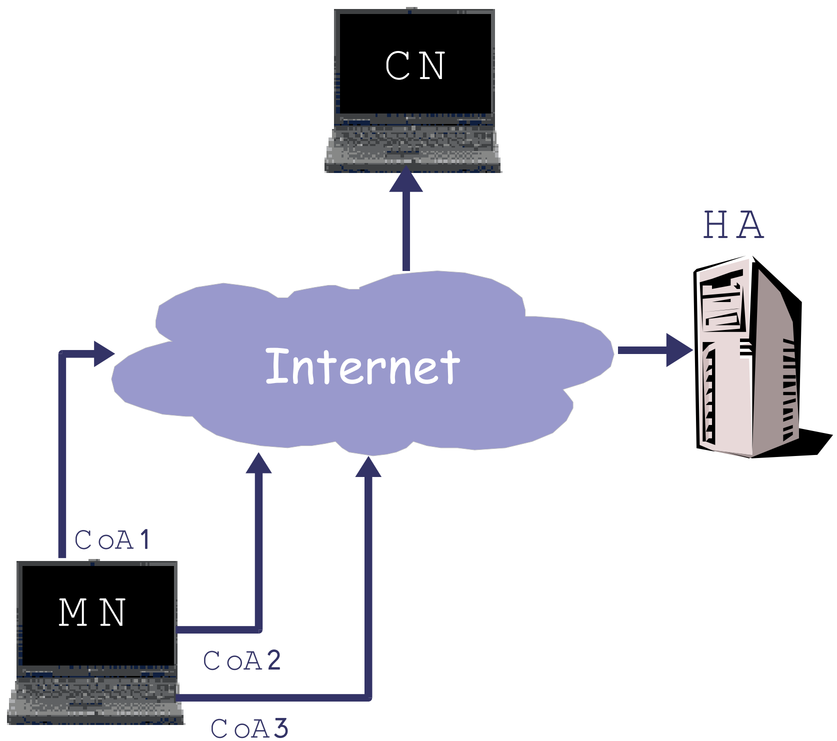
* T. Ernst, N. Montavont, R. Wakikawa, C. Ng, K. Kuladinithi "Motivations and Scenarios for Using Multiple Interfaces and Global Addresses". draft-ietf-monami6-multihoming-motivation-scenario-03.txt, May 3, 2008

Multi-Access: Multiple CoA Registration

- According to the current Mobile IPv6 specification, a mobile node may have several care-of addresses, but only one, called the primary care-of address, that can be registered with its home agent and the correspondent nodes
- Binding Identification number (BID)
  - The BID is an identification number used to distinguish multiple bindings registered by the mobile node on behalf of a single Home Address
  - It is a new naming component for an ongoing session

R. Wakikawa (Ed), V. Devarapalli (Ed), T. Ernst, K. Nagami "Multiple Care-of Addresses Registration". draft-ietf-monami6-multiplecoa-10.txt, November 4, 2008

Multi-Access: Multiple-CoA Registration



Home Address: 2001:db8::EUI

Home Agent Bindings

Binding [2001:db8::EUI CoA BID1]
Binding [2001:db8::EUI CoA BID2]
Binding [2001:db8::EUI CoA BID3]

Correspondent Node Bindings

Binding [2001:db8::EUI CoA BID1]
Binding [2001:db8::EUI CoA BID2]
Binding [2001:db8::EUI CoA BID3]

Multi-Access: Packets on Different Interfaces

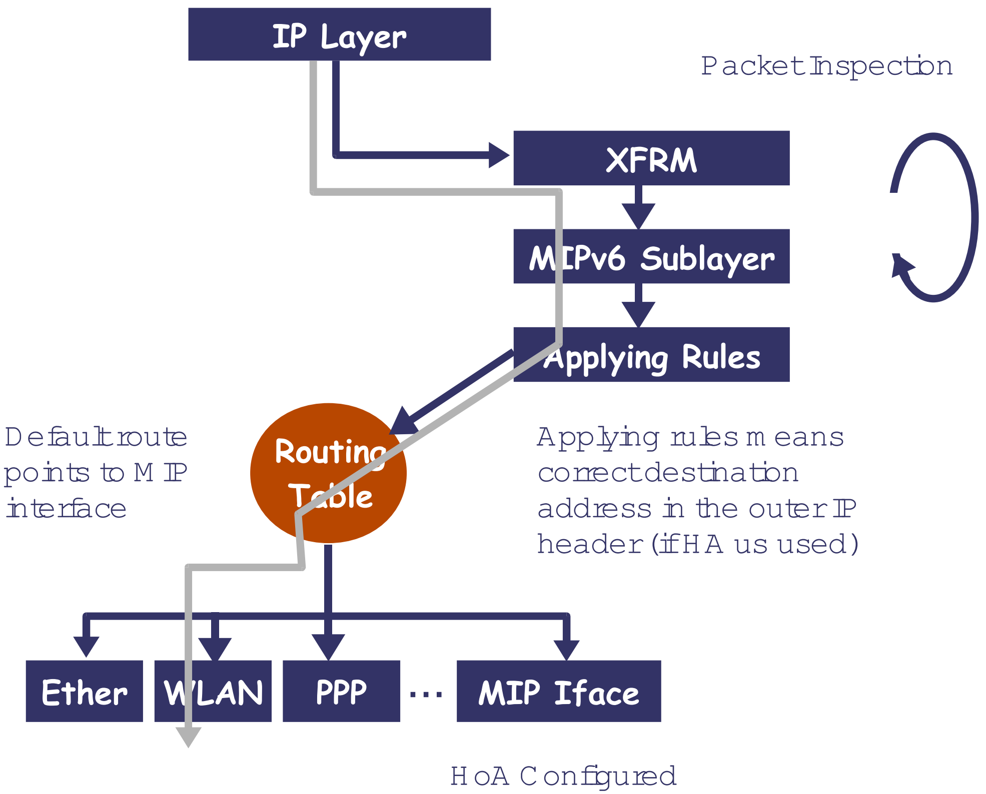
- We need to inspect every packet
  - Recall XFRM framework. Choice and ordering of the functions to be applied to a packet satisfying some criteria
- We need criteria matching the packet
  - Document "Flow Distribution Rule Language for Multi-Access Nodes" defines a language for flow distribution based on multiple criteria
  - Example of rules, which can be constructed using the language
    - tcp peerport 80 on 13 (Send HTTP traffic to peer using path 13)
    - udp localport 49724 peer "IPAddr" port 56512 on 800
  - Path Number is equivalent to BID and defines which interface the traffic should be sent through
- We need binding to an interface dependently on the criteria fulfillment
  - These are path/binding ID distributed among involved nodes

Multi-access: Rules and Bindings

- Rules and bindings have to be exchanged between all involved nodes (MN, CN, HA)
- Why do we need them both?
- Rules and bindings may be changed independently on each other
  - Assume we have a single active interface carrying all traffic. Rules and bindings are synchronized between MN and HA. Yet another interface becomes active at a time, however, we want to keep all traffic as it was previously. Result: Bindings have to be updated but not rules.
  - Assume we have two active interfaces and traffic from different applications is split between them. Due to some reason there is a need to move traffic from one application between active interfaces. Result: Rules have to be updated but not bindings

Multi-Access: MIPv6 Extension

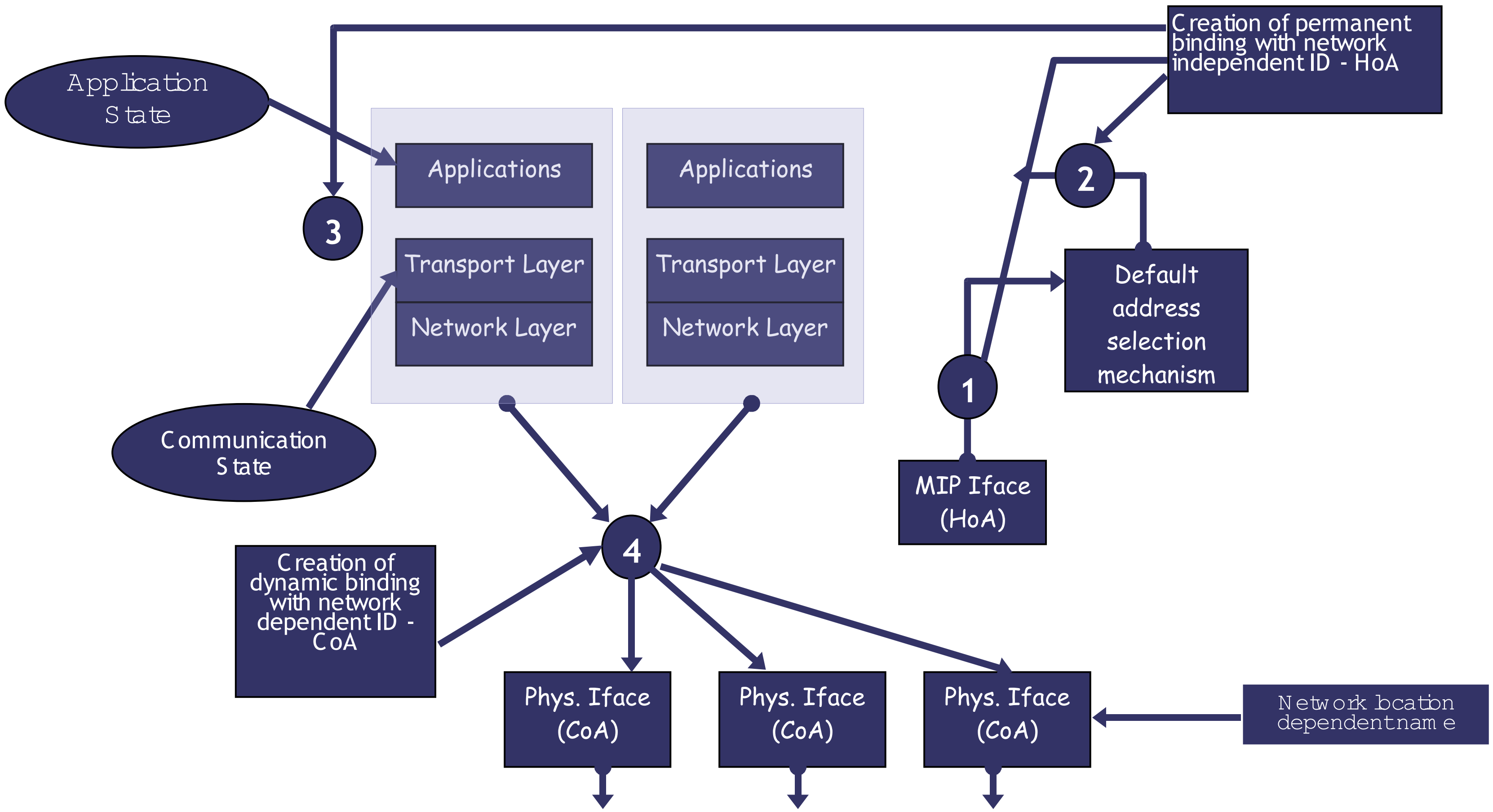
- Will this work? Is there something missing?
- Hint:
  - Can we make it with a single routing table?
- It looks like that a separate routing table required per interface



Multi-Access: SHIM6 vs MIPv6 + MEXT

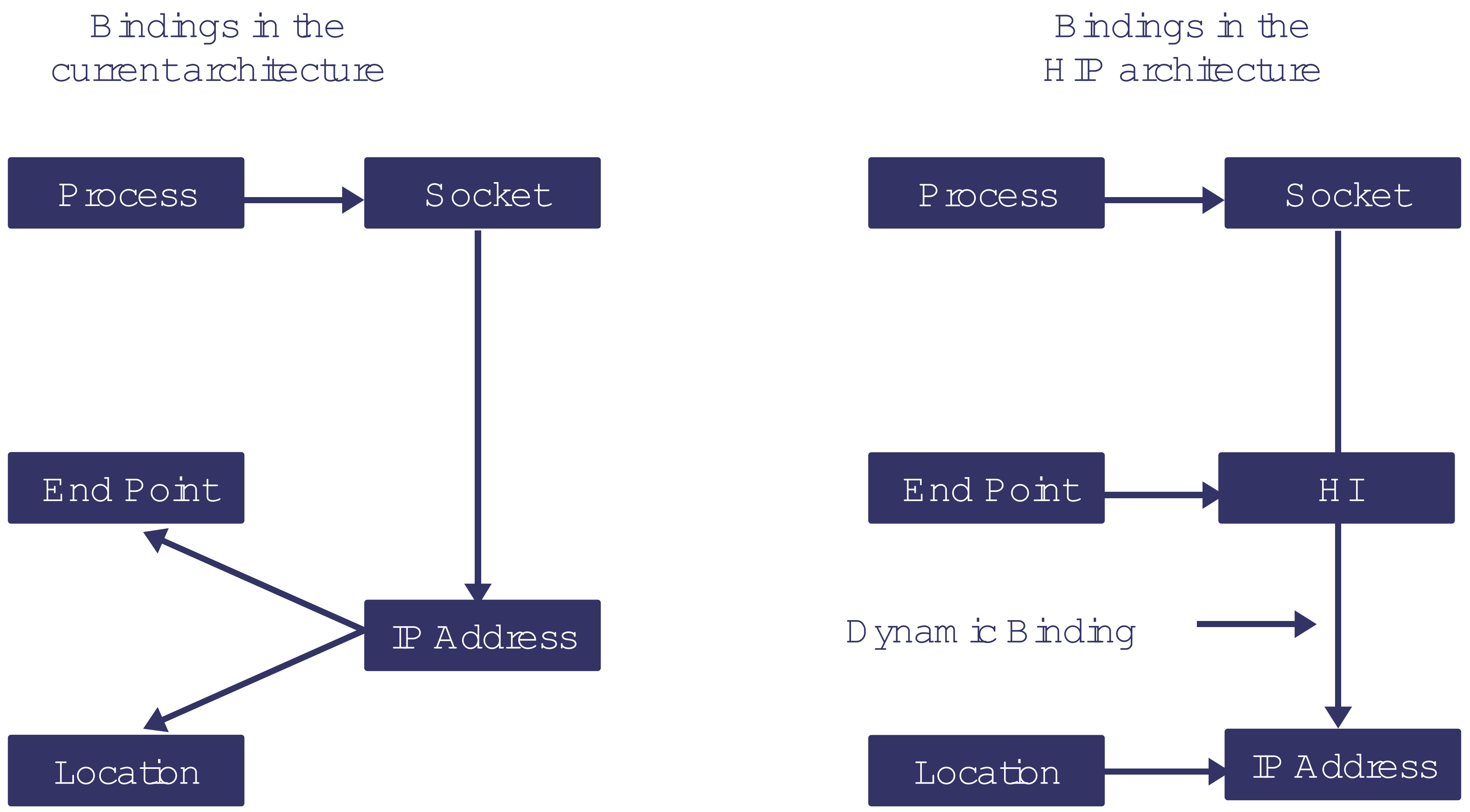
- What is the conceptual difference between site multi-homing and host multi-homing?
- What can we do with SHIM6 and cannot do with MIPv6 + MEXT?
- And finally, the favorite question: How does the MIPv6 + MEXT endpoint look like?

Multi-Access: MIPv6+MEXT Endpoint



Host Identity Protocol (HIP)

HIP: Bindings



HIP: Basics

- An independent name space for computing platforms (synonym of end-points) could be used in end-to-end operations independent of the evolution of the internetworking layer and across the many internetworking layers
- Such a name space (for computing platforms) and the names in it should have the following characteristics:

HIP: End-Point Names Characteristics

- The name space should be applied to the IP 'kernel'. The IP kernel is the 'component' between applications and the packet transport infrastructure
- The name space should fully decouple the internetworking layer from the higher layers. The names should replace all occurrences of IP addresses within applications (like in the Transport Control Block, TCB). This may require changes to the current APIs. In the long run, it is probable that some new APIs are needed

HIP: End-Point Names Characteristics

- The introduction of the namespace should not mandate any administrative infrastructure. Deployment must come from the bottom up, in a pairwise deployment
- The names should have a fixed-length representation, for easy inclusion in datagram headers and existing programming interfaces (e.g., the TCB)
- Name collisions should be avoided as much as possible
- The names should have a localized abstraction that can be used in existing protocols and APIs
- It must be possible to create names locally. This can provide anonymity at the cost of making resolvability very difficult

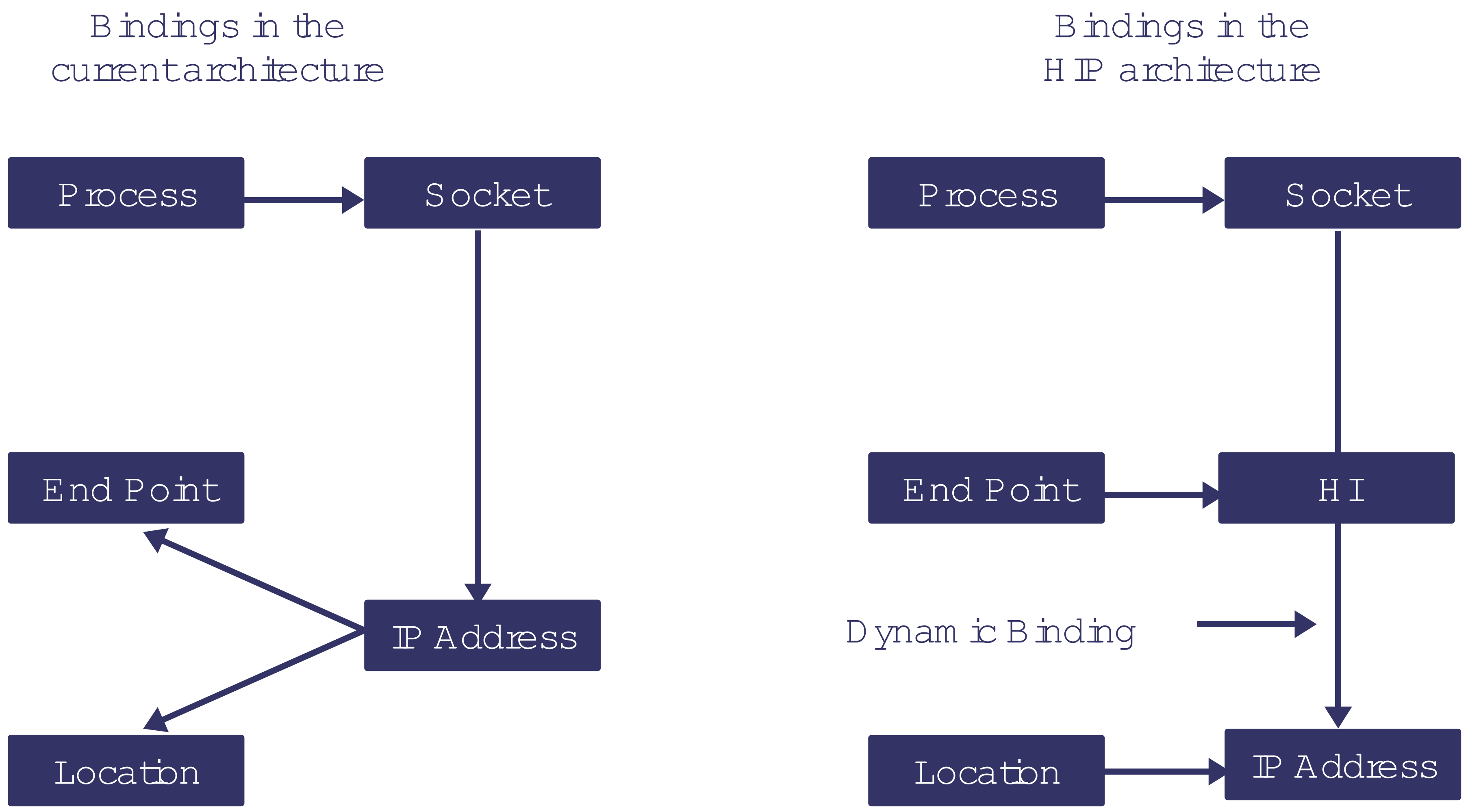
HIP: End-Point Names Characteristics

- The name space should provide authentication services
- The names should be long-lived, but replaceable at any time
- The only completely defined structure of the Host Identity is that of a public/private key pair. In this case, the Host Identity is referred to by its public component, the public key

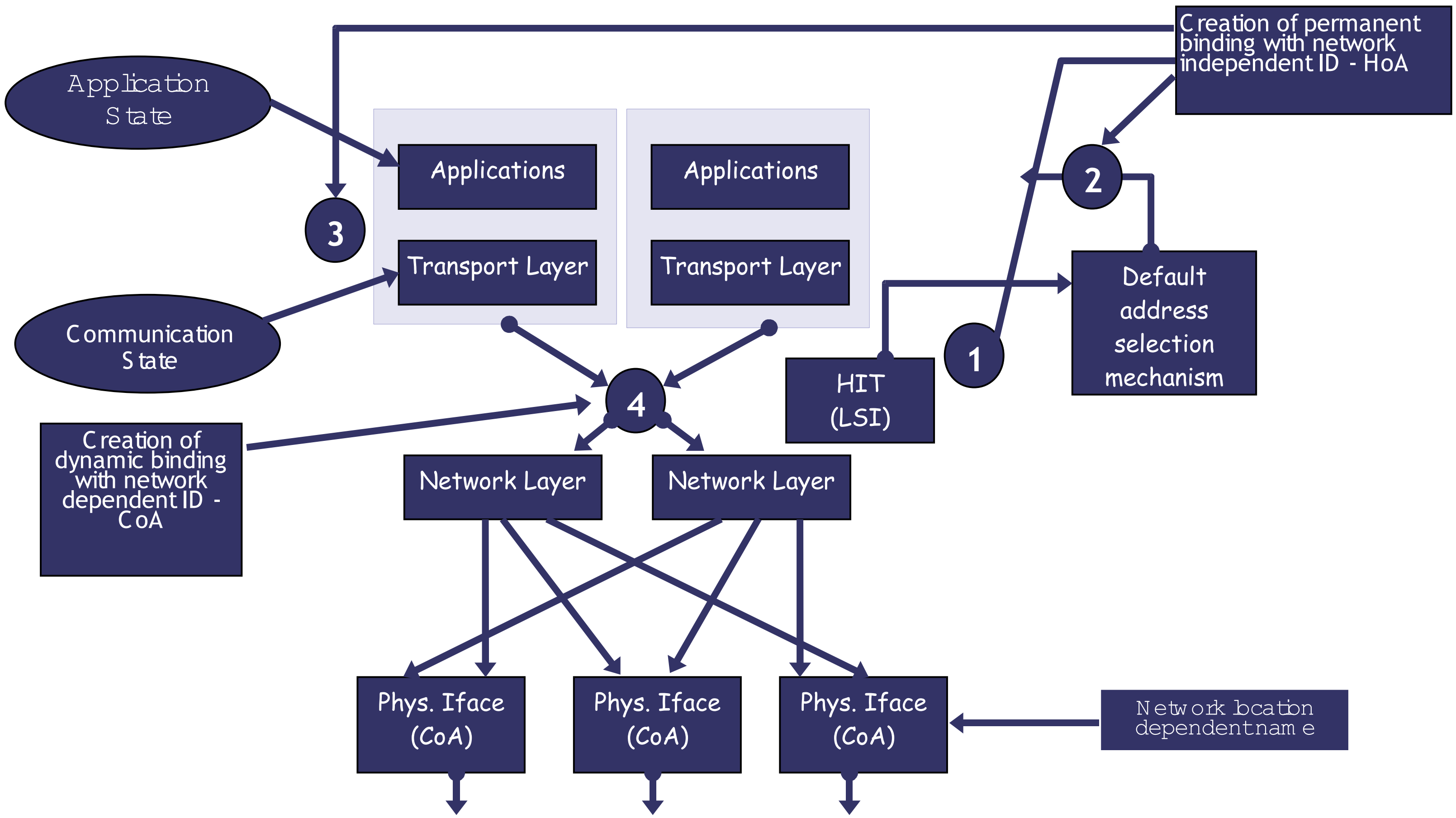
HIP: Naming Architecture

- Host Identifiers
  - The only completely defined structure of the Host Identity is that of a public/private key pair. In this case, the Host Identity is referred to by its public component, the public key
- Storing Host Identifiers in DNS
  - The (public) HI is stored in a new Resource Record (RR) type, to be defined
- Host Identity Tag (HIT)
  - A Host Identity Tag is a 128-bit representation for a Host Identity. It is created by taking a cryptographic hash over the corresponding Host Identifier
- Local Scope Identifier (LSI)
  - A Local Scope Identifier (LSI) is a 32-bit localized representation for a Host Identity. The purpose of an LSI is to facilitate using Host Identities in existing protocols and APIs. LSI's advantage over HIT is its size; its disadvantage is its local scope

HIP: Architecture



HIP: End-Point



LIS : References

- E. Nordmark, M. Bagnulo. "Shim6: Level 3 Multihoming Shim Protocol for IPv6", draft-ietf-shim6-proto-10.txt
- Sébastien Barré, Olivier Bonaventure. "Implementing SHIM6 using the Linux XFRM framework". Routing In Next Generation Workshop, 2007
- R. Wakikawa (Ed), V. Devarapalli (Ed), T. Ernst, K. Nagami "Multiple Care-of Addresses Registration". draft-ietf-monami6-multiplecoa-10.txt, November 4, 2008
- C. Larsson, M. Eriksson, K. Mitsuya, K. Tasaka, R. Kuntz "Flow Distribution Rule Language for Multi-Access Nodes", draft-larsson-mext-flow-distribution-rules-01, July 14, 2008
- T. Ernst, N. Montavont, R. Wakikawa, C. Ng, K. Kuladinithi "Motivations and Scenarios for Using Multiple Interfaces and Global Addresses". draft-ietf-monami6-multihoming-motivation-scenario-03.txt, May 3, 2008
- M. Eriksson, C. Larsson, R. Kuntz "Mobile IPv6 Flow Routing over Multiple Care-of Addresses". draft-eriksson-mext-mipv6-routing-rules-00, June 19, 2008
- R. Moskowitz, P. Nikander "Host Identity Protocol (HIP) Architecture" IETF RFC 4423, May 2006
- R. Moskowitz, P. Nikander, P. Jokela, (Ed) T. Henderson. "Host Identity Protocol", IETF RFC 5201, April 2008
- P. Nikander, T. Henderson, (Ed) C. Vogt, J. Arkko "End-Host Mobility and Multihoming with the Host Identity Protocol", IETF RFC 5206, April 2008

Questions and Discussions

Transport Layer Mobility Solutions (TLMS)

Migrate Approach

- Mobility rises five fundamental problems*
  - Locating the mobile host or service
  - Preserving communications
  - Disconnecting gracefully
  - Hibernating efficiently
  - Reconnecting quickly

* Alex C. Snoeren, Hari Balakrishnan, M. Frans Kaashoek. “Reconsidering Internet Mobility”, Proc. 8th Workshop on Hot Topics in Operating Systems (HotOS-VIII)

Migrate Approach

- Preserving Communications
  - Once a session has been established between end points (typically applications), communication should be robust across changes in the network location of the end points
- Hibernating Efficiently
  - If a communicating host is unavailable for a significant period of time, the system should suspend communications, and appropriately reallocate resources

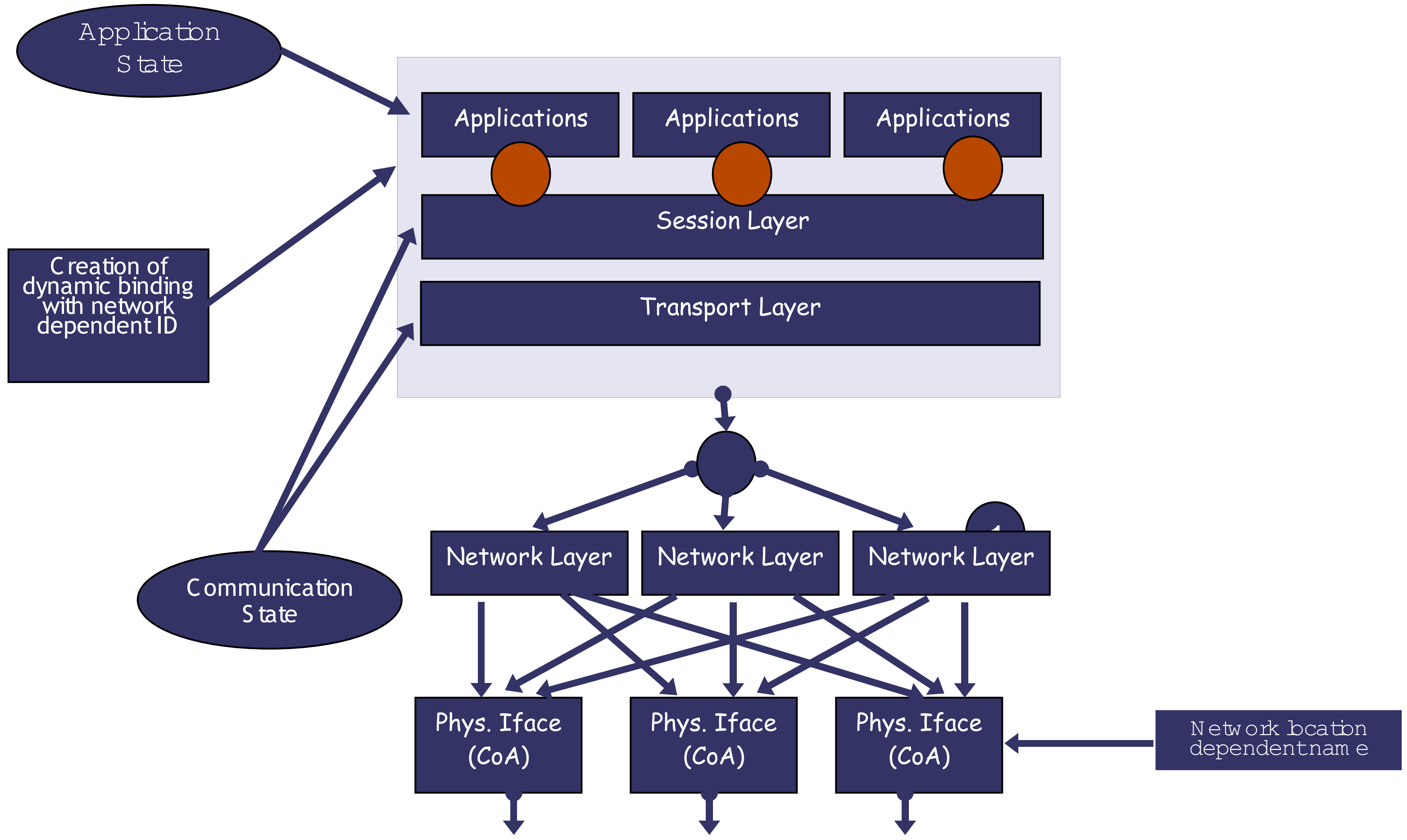
Migrate Approach: Design Guidelines

- Eliminate Lower-layer dependence from higher layers
- Do not restrict the choice of naming techniques
- Handle unexpected disconnections gracefully
- Provide support at the end hosts

Migrate Approach

- Propose to implement session layer between communicating applications and transport
- Main toolkit is a TCP migration design proposal
  - Introduces new state "Migrate Wait" and new TCP option "Migrate option" which enables TCP to move into this state.
  - When network is available again TCP can reestablish communication.
- If network is unavailable for a long period of time, session layer is capable of preserving the the state and release unused resources to the kernel

Migrate Approach: Endpoint



Transport Layer Mobility Solutions

Questions and Discussions

Stream Control Transmission Protocol (SCTP)

Stream Control Transmission Protocol

- Not directly designed for mobility
- Motivation is limitations of TCP for some applications
  - TCP provides both reliable data transfer and strict order-of-transmission delivery of data. Some applications need reliable transfer without sequence maintenance, while others would be satisfied with partial ordering of the data. In both of these cases, the head-of-line blocking offered by TCP causes unnecessary delay
  - The stream-oriented nature of TCP is often an inconvenience. Applications must add their own record marking to delineate their messages, and must make explicit use of the push facility to ensure that a complete message is transferred in a reasonable time
  - The limited scope of TCP sockets complicates the task of providing highly-available data transfer capability using multi-homed hosts
  - TCP is relatively vulnerable to denial-of-service attacks, such as SYN attacks

SCTP: End-Point Description

- SCTP endpoint: The logical sender/receiver of SCTP packets. On a multi-homed host, an SCTP endpoint is represented to its peers as a combination of a set of eligible destination transport addresses to which SCTP packets can be sent and a set of eligible source transport addresses from which SCTP packets can be received. All transport addresses used by an SCTP endpoint must use the same port number, but can use multiple IP addresses. A transport address used by an SCTP endpoint must not be used by another SCTP endpoint. In other words, a transport address is unique to an SCTP endpoint

SCTP: Functional View

Association Startup and Taking Down

- Sequenced Delivery Within Streams
- User Data Fragmentation
- Acknowledgment and Congestions Avoidance
- Chunk Bundling
- Packet Validation
- Path Management

SCTP: Mobility Related Protocol Features

- SCTP association creation allows exchange of valid IP addresses for this association
- Upper layer protocols (applications) may specify addresses to use
- Protocol supports failover from an inactive destination address
- However, the protocol does not support change of IP associated addresses during ongoing session
  - Some proposals exist, e.g. Wei Xing, Holger Karl, Adam Wolisz, Harald Muller "M-SCTP: Design and Prototypical implementation of an End-To-End Mobility Concept", Proc. of 5th Intl. Workshop The Internet Challenge: Technology and Applications, Berlin, Germany, Oct. 2002