Who Are You?

Identity and Location in IP

Geoff Huston

In this presentation:

- I’d like to explore the issues around identity and the structure of identity name spaces
- Look at what makes identity realms relevant and useful for a communications network
- The implications of using identity within the architecture of the Internet

Addresses and IP Architecture

Within the IP architecture addresses are:

- Endpoint identifiers
- Routing objects
- Key value for Forwarding Lookup

Architecturally, IP Addresses are:

- Drawn from a stable global space
- Intended to be used in a unique context

IP Addresses are:

A means of uniquely identifying a device interface that is attached to a network – the **WHO**

Endpoint identifier

A means of identifying where a device is located within a network – the **WHERE**

Location identifier

A lookup key into a forwarding table to make local switching decisions – the **HOW**

Forwarding identifier

Overloaded semantics?

This deliberate overload of sematic intent of IP addresses by mixing who, where and how into a single token has been a basic property of the IP architecture since its inception

- **Elegant simplicity?**

or

- **Fundamental weakness?**

Challenges to the IP Address Model

- Mobile endpoints – Home and Away
- Roaming endpoints - Nomadism
- Multi-homed endpoints and “session” resiliency
- Scoped address realms
- NATs and ALGs
- Anycast services
- VOIP
- Peer-to-Peer applications
- Session hijacking and general disruption

Why is identity a current topic?

The Internet appears to have worked just fine for the past few decades with this overloaded semantic of addresses

What is changing in the environment to make this topic current?

Where’s the pain?

Routing Complexity and Scaling

- Carrying highly dynamic and more specific prefixes in the routing system is inexorably killing the efficiency of routing
  - And possibly threatening the longer term viability of our routing systems
  - If we can’t push the requirement for more specific and timely information about individual device location out of the routing system then we may be doomed!

Where’s the pain?

Applications are no longer end-to-end

- In order to perform identity-based rendezvous applications need to sustain application-specific identity realms, application-specific mappings, and connectivity exploration with intermediaries and agents
  - Skype, ENUM, Stun, Torrents,…
- New applications are harder (if not impossible) to deploy unless they graft themselves onto the infrastructure of existing applications
  - IP over HTTPS is a classic example
- Applications are more expensive, and the network resists new applications

Where’s the pain?

We have no defence against address abuse

- Spoofed source addresses used to cause massive DDOS attacks through co-opted data amplifiers
- Disrupting location information in the routing system to cause breakdown of integrity of service behaviour

Wouldn’t it be good if…..

- Your identity was stable irrespective of your current location
- You could maintain sessions while being mobile (handover)
- You could maintain sessions across changes in local connectivity (failover)
- That locator use was a dynamic association while identity was long-term stable (mapping properties)

- In other words:
  - Anyone could reach you anytime, anywhere
  - You could reach anyone, anytime, anywhere

Wouldn’t if be good if…

- True identity concepts actually worked in IP

Wouldn’t if be good if…

IPv6 offered solutions in this space that allowed endpoint identity to be distinguished from location and forwarding functions

“Second-Comer” Syndrome:

This perspective can be phrased as: Unless IPv6 directly tackles some of the fundamental issues that have caused IPv4 to enter into highly complex solution spaces that stress various aspects of the deployed environment than I’m afraid that we’ve achieved very little in terms of actual progress in IPv6. Reproducing IPv4 with larger locator identifiers is not a major step forward – its just a small step sideways!

“We’ve Been Here Before” Warning:

Of course this burdens the IPv6 effort in attempting to find solutions to quite complex networking issues that have proved, over many years of collective effort, to be intractable in IPv4. If the problem was hard in an IPv4 context it does not get any easier in IPv6! That should not stop further exploration of the space, but it should add a touch of caution to evaluation of solutions in this space.

The Hard Lesson

Attempting to overload a single identification system with a diverse set of intended roles may look like an elegant and useful shortcut at the time

But it’s often a terrible mistake!

So what?

All this is rather abstract

How does this relate to the nature of an information infrastructure and the architecture of the Internet?

We've done a pretty lousy job so far!

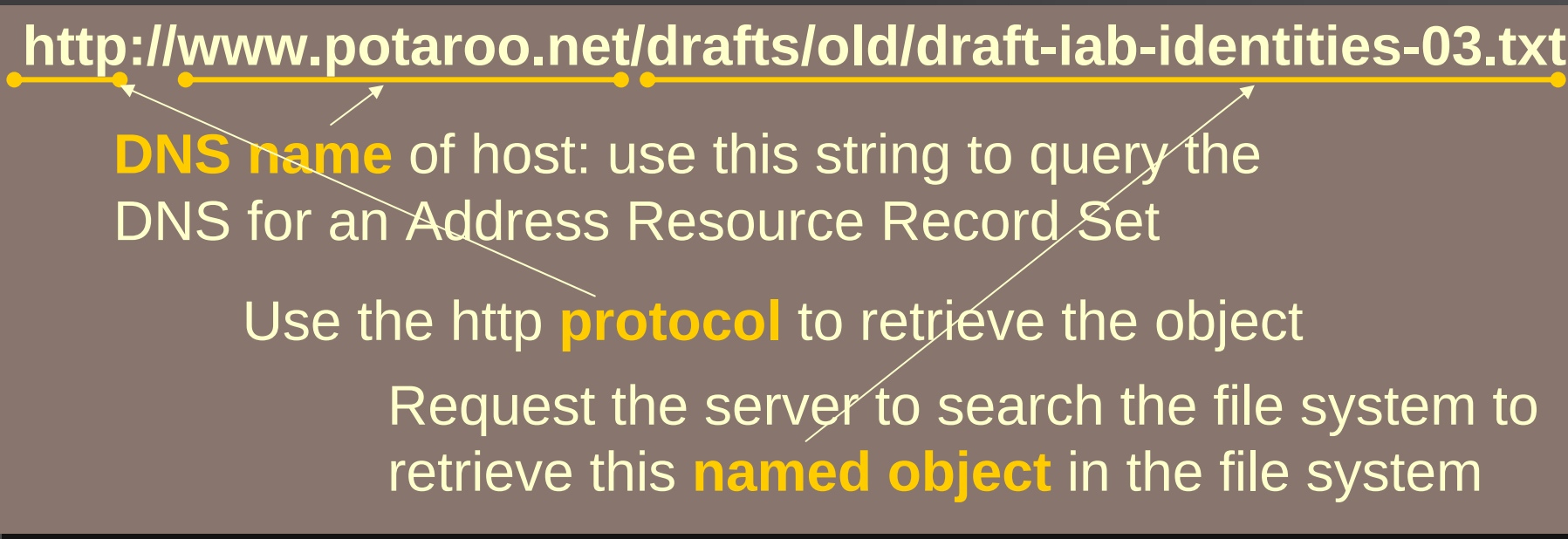
The information infrastructure has fallen into the same trap as IP addressing in its adoption of URLs as the underlying identity realm:

- **what** is synonymous with **where** in an object-oriented world
- **where** then becomes a viable non-clashing identifier scheme that also happens to dictate a resolution mechanism at the same time
- So all we need to a methodical approach to **where** and we're done!

Easy, simple and extremely inelastic!

What's so bad about URLs?

- URLs describe a retrieval algorithm for an object instance, not an object identifier
- Device and application selectors coupled with application-specific query string



A URL is not an “atomic” identity

A URL is a derived identity schema

- Protocol identifier
- DNS identifier
- Filesystem name

Uniqueness is a derived property of the hierarchical structure of the DNS and the relative uniqueness of names objects in a local filestore

Its insecure, vulnerable to all kinds of abuse and inappropriate to our conventional methods of utilizing information

What happens to a URL when:

- The site changes its name?
- The server changes its name?
- The filesystem changes?
- The access protocol changes?
- The document changes?
- The document is cloned?
- Your DNS Root is changed underneath you?
- Your DNS resolution is perverted?
- The name part no longer resolves?
- The protocol part is unrecognised?

What do we want from “Identity”?

Varying degrees of:

- Uniqueness
- Persistence
- Structure
- Clear Scope of Applicability
- Validity and Authenticity
- Clear line of derivation authority

Identity is not a unilateral assertion – it is better viewed as a recognition of derived uniqueness within a commonly understood context

What should we avoid in “Identity”?

Varying degrees of:

- Uncoordinated self-assertion
- Arbitrary token value collisions
- Ill-defined temporal validity
- No coherent structure
- Unclear applicability
- Semantic overload
- Structural overload and complexity of the token space
- Insecure and unclear authority
- Cost

Choices, Choices, Choices

Its possible to inject an identity object at almost any level of the protocol stack model

- **Application Identities** shared across transport sessions
- **Transport Identities** to allow agility of stack location
- **Host identities** to allow agility of location of all hosted sessions

In this context an “identity” is a token to allow multiple lower level “locators” to be recognised as belonging to a single communication state at both (or multiple) ends of the communication

Choices, Choices, Choices

Identity at the Application level

- Use a stable name space that is mapped to a locator (using the DNS)
  - DNS incremental updates
- Allow indirection and referral via DNS NAPTR and URI Resource Records
  - Generic identity ornamented with service-specific mappings
  - ENUM
- Use application agents to provide stable rendezvous points
  - For example: *sip:gih@sip.apnic.net*
- Issues:
  - Can the DNS support dynamic interaction at a suitable scale and speed?
  - Are a family of diverse application-specific identities desireable (cross-application referral and hand-over)
  - Can we stop application designers from creating NAT-agile locator-independent application-specific solutions that rely on an application-specific identity space?

Choices, Choices, Choices

Identity at the Transport Level

- Can we provide a mechanism to allow identity / locator independence at the session level?
  - An application opens a session with a generated session identity token
  - The identity token is dynamically associated with locator pairs
  - Changes in locators do not change the session token
- Application of the layering approach
  - Allow applications to assume a framework of identity association
  - Perform identity / locator association at a lower level of the protocol stack
  - Use opportunistic identity values that have a limited context and role of supporting session integrity
  - Support legacy applications by providing a consistent API

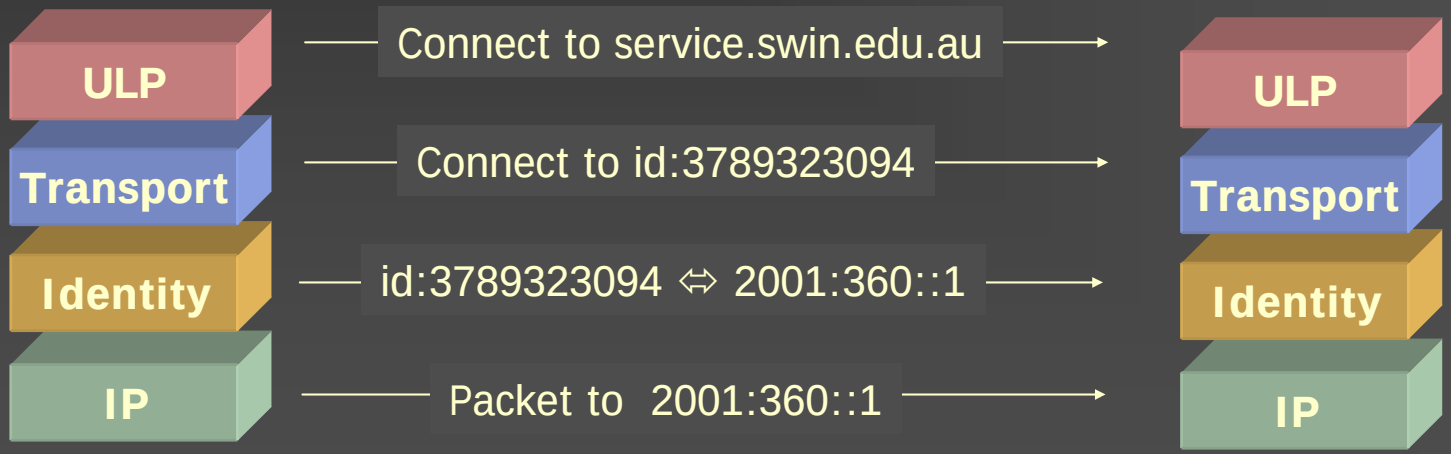
Choices, Choices, Choices

Identity at the IP level

- Can we provide an identity / locator association that is shared across multiple services and sessions?
- Reduce the overhead of identity locator mappings to allow all sessions to a common endpoint to share a mapping state
- Want to provide a more comprehensive support of identity to support both session-oriented transport protocols and (potentially) datagram transactions
- Reduce the complexity of applications and transport sessions and place the per-endpoint mapping state in the IP level

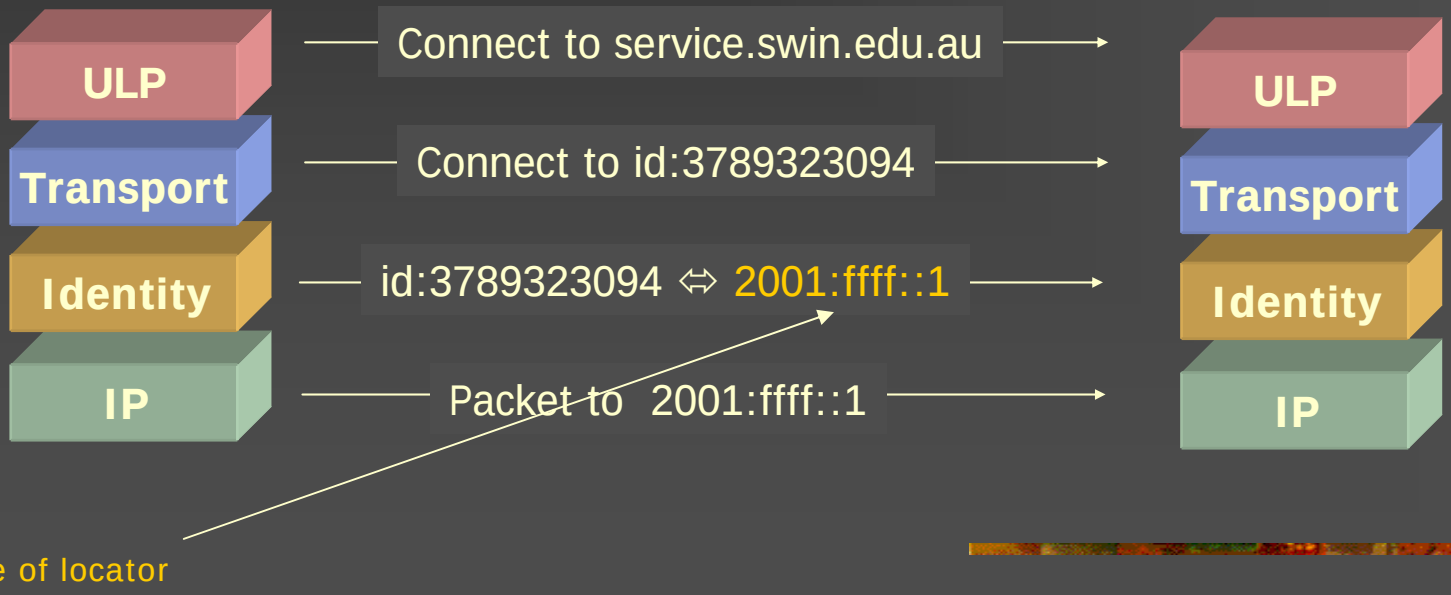
Identity Issues

How could an identity mapping function?



Identity Issues

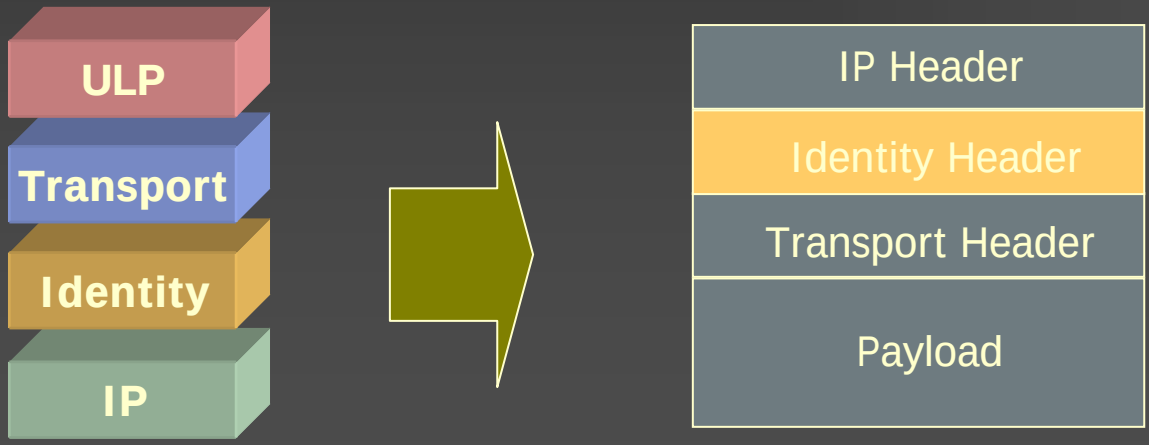
How could an identity mapping function?



Identity Implementations

"Conventional"

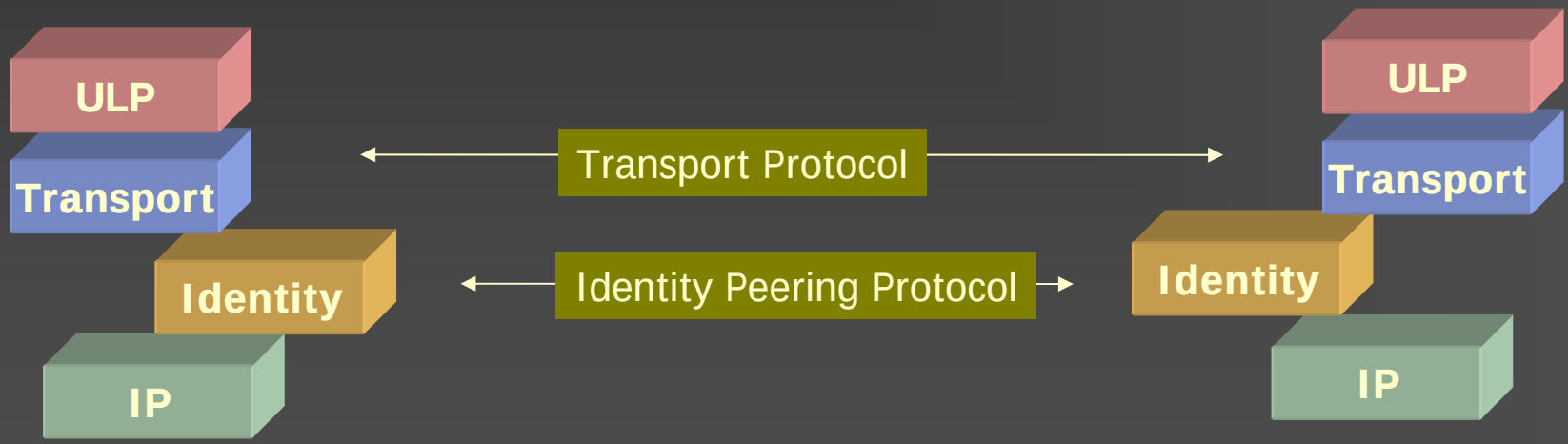
- Add a wrapper around the upper level protocol data unit and communicate with the peer element using this "in band" space



Identity Implementations

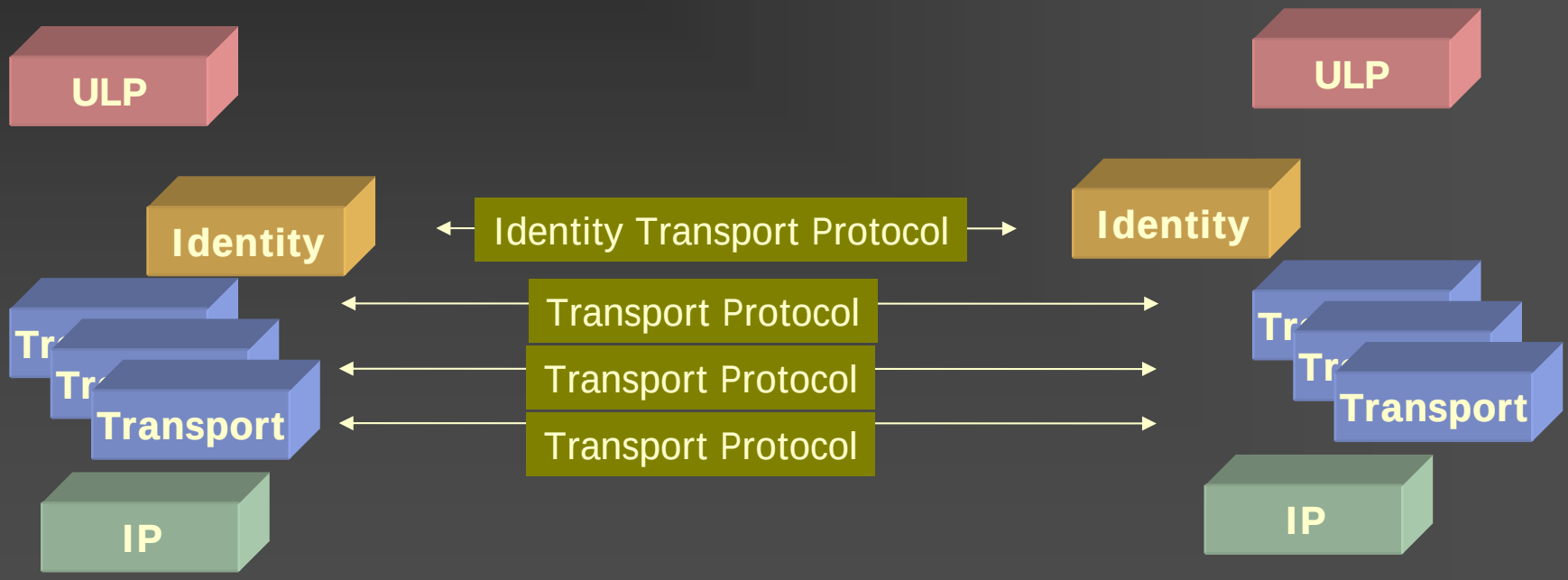
"Out of Band"

- Use distinct protocol to allow the protocols element to exchange information with its peer



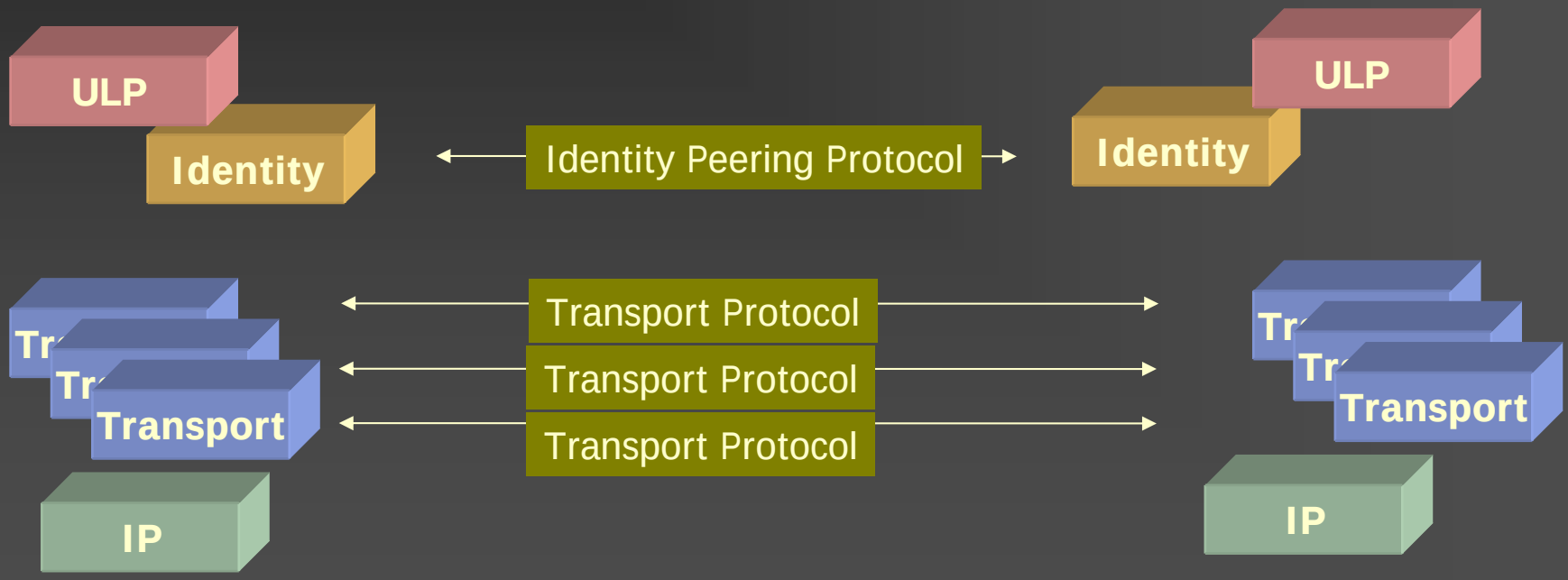
Identity Implementations

Transport: Below the Session



Identity Implementations

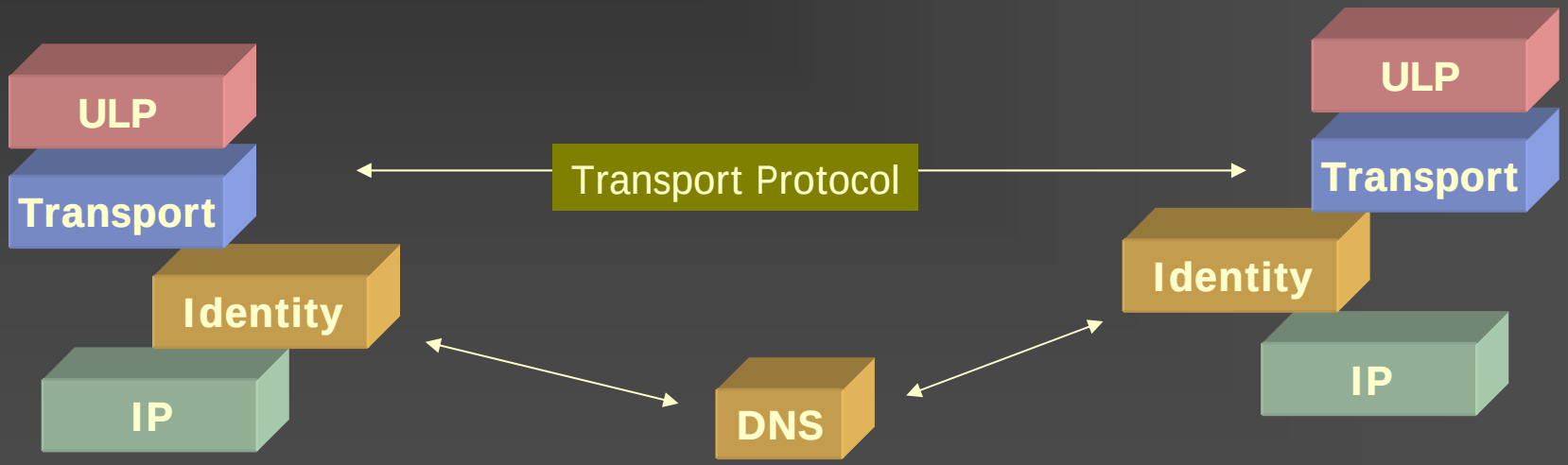
Application Identity: Above the Session



Identity Implementations

"Referential"

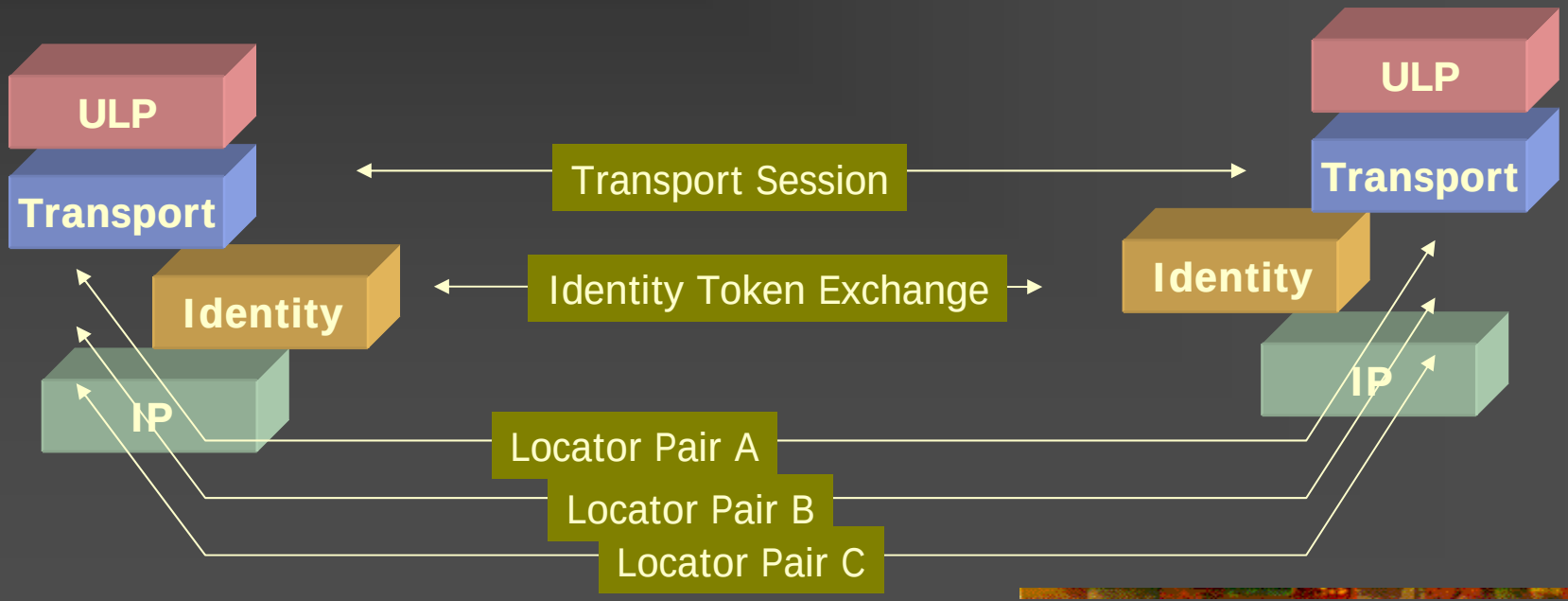
- Use a reference to a third party point as a means of peering (e.g. DNS Identifier)



Identity Implementations

Self-Referential

- Use an opportunistic identity as an equivalence token for a collection of locators



Identity Types

Use identity tokens lifted from a protocol's "address space"

- DNS, Appns, Transport manipulate a "distinguished address"
- IP functions on "locators"
- Stack Protocol element performs mapping

FQDN as the identity token

- Is this creating a circular dependency?
- Does this impose unreasonable demands on the properties of the DNS?

Structured token

- What would be the unique attribute of a new token space that distinguishes it from the above?

Unstructured token

- Allows for self-allocation of identity tokens that may not globally assuredly unique (opportunistic tokens)
- How to map from identity tokens to locators using a lookup service? Or how to avoid undertaking such a mapping function

Some Identity Suggestions

- IPv4 Address
- IPv6 Address
- Centrally Assigned IPv6 Unique Local Addresses
- A crypto hash of your public key
- A crypto hash of a set of locator values
- The IPv6 address used to initiate the communication
- MAC-48 address
- MAC-64 address
- DNS names
- URIs
- Telephone numbers

Identity Issues

- Identity / Locator Binding domain
  - Session or host?
  - Dynamic or static?
  - Configured or negotiated?
- Scope of identity role
  - Locator independent identity
  - Equivalence binding for multiple locators
- Locator Selection
- Application visibility of identity capability
- Scoped identities
- Identity Referrals and hand-overs
- Third party locator rewriting
- Security of the binding
- Context of use determining semantic interpretation

Upper Level Issues of Identity Realms

- The significant effort and cost of supporting a new global unique token distribution system as an endpoint identity system
- The side-effects of reusing some other existing token set as an identity set
- The issue of support of dynamic identity to locator binding
- The protocol overhead of identity handshake for datagram transactions
- The security issues in maintaining integrity of identity

IPv6 and Identity

- Is the 64bit Interface Identifier a rich location for carrying opportunistic identity?
- Can the Flow-Id field be exploited?
- Are header extensions and options useful?
- Is packet inflation necessary?
- Is IPv6 the only protocol for consideration of IP level identity approaches?
  - Is there any leverage for transport session approaches?
  - Can such approaches be IP version agnostic?

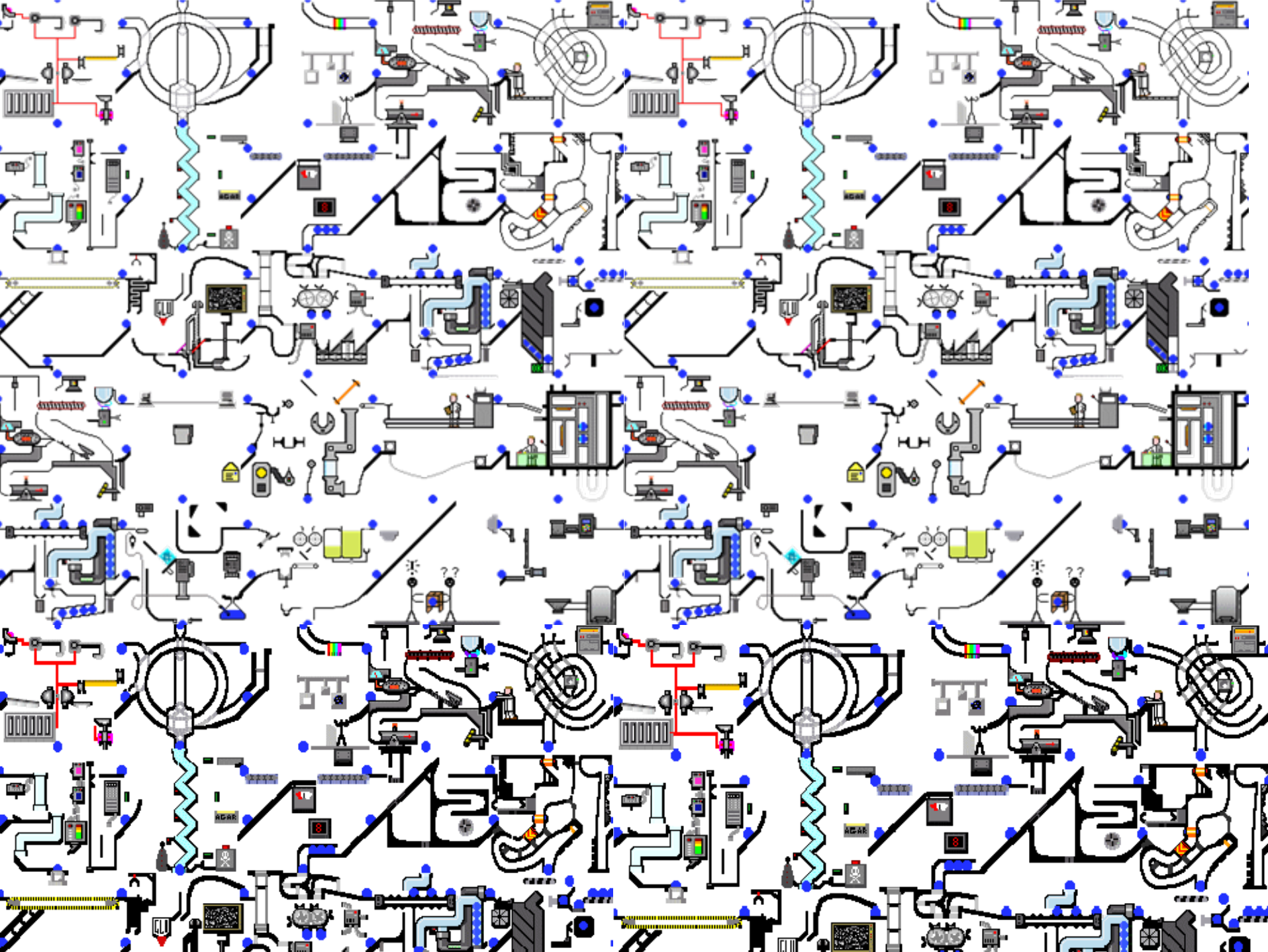
百花齊放，百家爭鳴 *

Our current direction appears to be developing solutions in **all** of these spaces simultaneously:

- Multi-Party Applications
- Application Agents
- Rendezvous protocols
- DNS Incremental Updates and DNSSEC
- DNS Indirection and Referral
- SCTP, HIP at the transport-layer
- Shim6
- Mobile IPv6
- Mobile IPv4
- MPLS
- Hierarchical Routing
- And probably many more!

* *Let a hundred flowers bloom: let a hundred schools of thought contend*
Mao Zedong, 1956

Is this getting all too complex?



一花独放，一家主鸣 ？ *

- At all levels of the protocol stack the disambiguation of the identity of the ‘other’ side and the means to maintain an information flow are distinct problems when you wish to include concepts of replication, equivalence, mobility, robustness
- When deconstructing the “address” into its structural components there is no ‘single’ solution
  - packet forwarding and destination identification
  - session agility across location change
  - application service point identification
  - information infrastructure identification and URIs

** Let one flower bloom: let one school of thought prevail*

Thank You