From CURLING to COMIT: a View on Evolutionary ICN Architectures

Prof. George Pavlou
http://www.ee.ucl.ac.uk/~gpavlou/
Communications and Information Systems Group
Dept of Electronic & Electrical Engineering
University College London, UK
The History Behind CURLING

• At around 2008 we were thinking of Content-Aware Networks: being able to direct requests to the right copy but not at the access / CDN

• Accidentally bumped onto the DONA paper and saw a radical approach for a new content-oriented architecture that vindicated our thinking

• Also became aware of the FP7 PSIRP project and of Van Jacobson’s Google talk

• Started working on an FP7 proposal with our key partners codenamed CDN++ (later became COMET)
Why Evolutionary ICN Architectures

• IP represents too much investment and cannot be easily changed, especially with a radical content-oriented protocol!
  – Even IPv6 penetration has been quite small until now
• No significant changes since BGP policy routing in 1991 and CIDR in 1993
  – MPLS has been a compatible addition rather than replacement
• Please note: by evolutionary we do not mean overlay approaches which are limited to what they can achieve
  – We mean approaches that retain IP as the thin “hourglass waste”
  – They should also retain BGP-based inter-domain routing
• We cannot see radical approaches been deployed but can see gradual changes towards this direction
Key Requirements

• Ability to direct requests to replicated content in servers “a la CDN” and also to direct content to the closest cached copy in edge Content-Aware Routers (CARs)

• Ability to support multicast for real-time content and also “time-shifted multicast” for non real-time content

• Scale to the expected massive size of the future content ecosystem

• Be compatible with IP and allow deployment in incremental stages
  – Some domains may deploy it and some not
The CMP finds the right content copy (content mediation) and possibly prepares the network for contentID-based transmission (network mediation for popular content only)
Key Features

• Content is publicised following the inter-domain “provider route” forwarding rule
  – Like DONA but only uphill and not P2P – better scalability

• Content requests are resolved uphill to tier-1 domains and then downhill to domain where content lies

• Provider route rule has a “tromboning” effect, so route optimisation follows with shortcut paths learned from BGP
  – Prefer customer over peering and peering over provider routes

• CMSs install states in edge CARs so that content transmission is based on contentID and IP tunnels

• Network mediation is only for popular content which is also cached in CARs based on any orthogonal caching policy
Content Publication

Content Publication Dissemination

Content Registration Stage

S1::Register(X1)

Publish(A.A.A, X1)

Publish(A.A, X1)

X1->S1

A

A.A

A.A.A

A.B

A.B.B

From CURLING to COMIT - 7
Content Resolution

Downhill: follows content repository entry

Uphill: provider route forwarding rule

From CURLING to COMIT - 8
Incremental Deployment Stages

1. The CMP is deployed as a DNS++, in a pure “content-resolution-only” fashion for “advanced CDN operation”
   - No network changes at all, only changes to end systems

2. CARs are deployed *at the domain edges* only to support edge-network caching for popular content only (for scalability) and contentID-based forwarding
   - Routers use IP/BGP and intra-domain edge-to-edge tunneling

3. CARs are also deployed *within the network* to enable native in-network content caching, resolution & delivery
   - Support for native information-centric networking functionality, possibly with a protocol like CCN
   - The mediation plane effectively “collapses” onto the network
Evolution: the COMIT Approach

- CURLING intended not to touch the protocol stack
- With COMIT we take the liberty to go a step further and introduce a 3.5 content-oriented layer between IP and the transport layer
- The latter also needs to change to account for the closer to native content-based operation of layer 3.5
- But we still envisage a model of incremental deployment in parallel with the current TCP/IP-based Internet
Business Model Issues for ICN Migration

• Content Providers (CPs) want full control over the content they publish and ICN-based transparent in-network caching does not help
  – Content access logging, pay-per-click business models
Content Provider Assisted Name Resolution

• To deal with the scalability of name resolution and content location issue and also of the control over content desired by CPs, we “outsource” part of the name resolution to the CPs themselves

• Content provider-assisted name resolution
  – Content Providers control content names and provide these names to clients

• What does this achieve?
  – No need for extra name-resolution machinery
  – Fully backwards compatible with IP, DNS
  – Reduces significantly network state
Location-Independent Routing Layer (LIRA)

- **Ephemeral Names**: CPs have full control of the “up-to-date” names. *Fully* transparent caching is not possible
  - Content caching still possible!
- **An extra “3.5 layer”** placed between IP layer and transport layer – LIRA layer
- **A Content-FIB** to keep track of the state of neighbour caches
Content Provider-Assisted Name Resolution and Content Delivery

![Diagram showing the process of content delivery from A to CP/CDN via DNS and IP routing table.](image-url)
Content Provider-Assisted Name Resolution and Content Delivery

Diagram:
- A: www.url.com
- B: DNS
- C: 192.128.97.1
- D: Content FIB
- E: some/cID
- F: CP/CDN

Connections:
- A to B: www.url.com
- B to D: DNS
- D to C: 192.128.97.1
Key Features and Deployment

- LIRA layer operates a recently-served name table and forwarding information for the delivery of those contents
  - Provides support for in-network caching and multicast
- LIRA layer checks contentIDs against the recently-served name table but if nothing is found, request is forwarded based on the server IP address
- Not all nodes are required to support LIRA, allowing for incremental deployment of LIRA-capable nodes
- Changes are required in applications and transport layer although existing end-systems can continue to operate normally as they are doing today
Summary

- With **CURLING** we tried to come up with an *evolutionary DONA-style scalable approach*, in which names are assigned by the infrastructure (CMP/CMSs)
- Since then our thinking is that transparent caching is NOT desirable and we moved to **COMIT** which uses ephemeral names assigned by the CPs
  - Work in progress, more “intrusive” than CURLING
- For relevant papers look at relevant project web pages:
  - COMET: [http://www.comet-project.org/](http://www.comet-project.org/)
  - COMIT: [http://www.ee.ucl.ac.uk/comit-project/](http://www.ee.ucl.ac.uk/comit-project/)