Networking Named Content

Van Jacobson, Diana K. Smetters, James D. Thornton, Michael Plass, Nick Briggs, and Rebecca Braynard

Presented by Ross Lagerwall

University of Cambridge

January 27, 2013


오빤 '강남스타일'
STYLE

Background

Networking is done host-to-host in a conversation fashion.

But our internet usage doesn't fit this pattern.

We care about content, not location of the data.

Previous Work

Research has tried to use content as a first class citizen rather than location.

Previous work has used flat opaque names to data.

These names are then mapped to user friendly ones.

TRIAD uses structured, user friendly names with a modified DNS server.

Solution

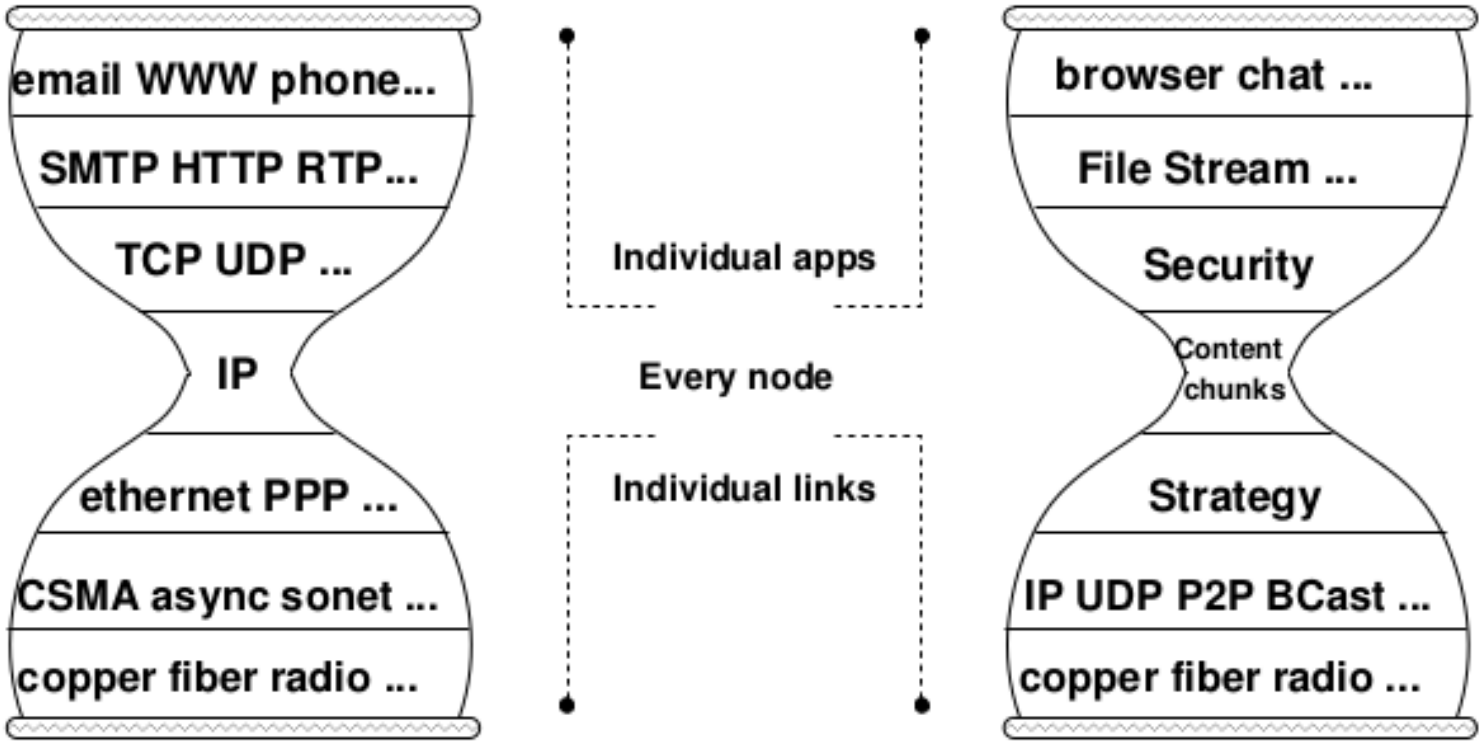
CCN tries to solve the problem by using user friendly names which describe content.

Routing is done on content rather than host location.

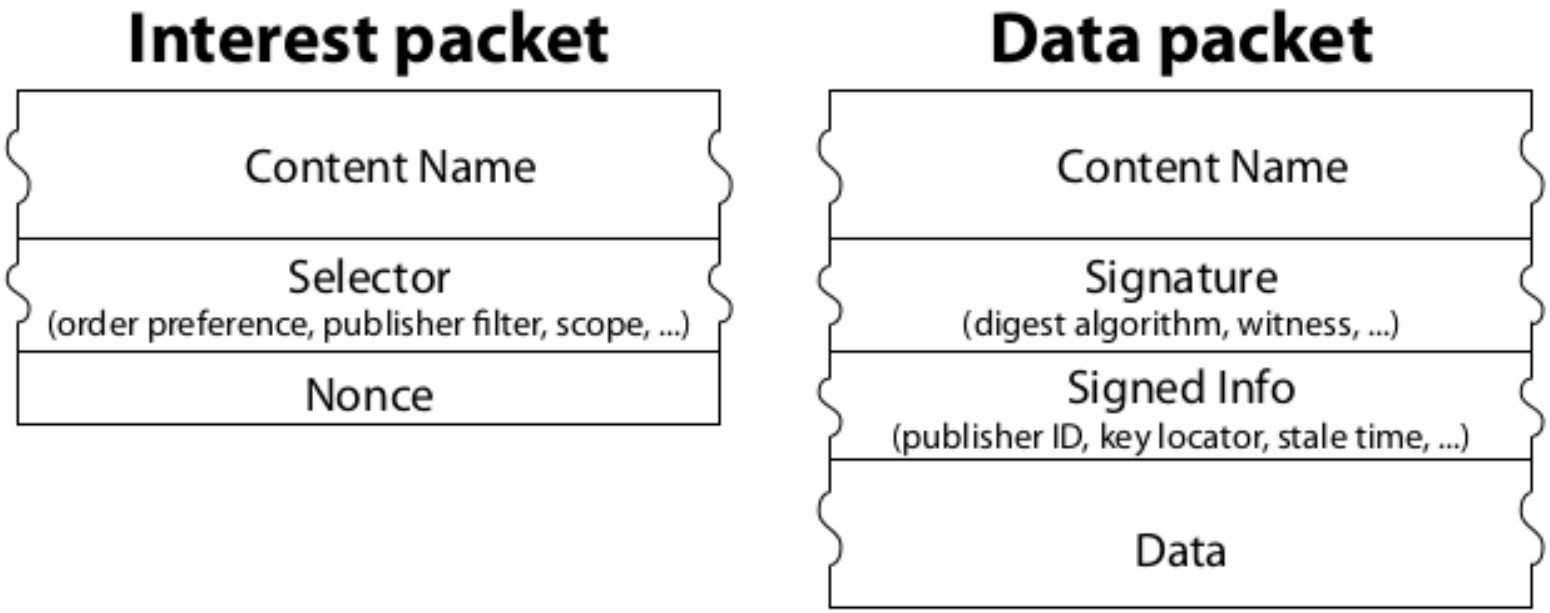
CCN defines Interest and Data packets.

To retrieve data, a node broadcasts an Interest.

Solution


email WWW phone...
SMTP HTTP RTP...
TCP UDP ...
IP
ethernet PPP ...
CSMA async sonet ...
copper fiber radio ...
Individual apps
Every node
Individual links
browser chat ...
File Stream ...
Security
Content chunks
Strategy
IP UDP P2P BCast ...
copper fiber radio ...

Solution


Interest packet
Content Name
Selector
(order preference, publisher filter, scope, ...)
Nonce
Data packet
Content Name
Signature
(digest algorithm, witness, ...)
Signed Info
(publisher ID, key locator, stale time, ...)
Data

Solution

On receiving an Interest:

- reply with data
- reply with cached data
- store a breadcrumb in the pending interest table and forward it

Solution

Data follows the breadcrumbs back.

Breadcrumbs are replaced with a cached copy of the packet.

Interests can be pipelined similar to TCP's window.

Sequencing is achieved by altering the structured name.

A limited form of querying is available with primitives such as RightMostChild.

Strategy

The strategy layer is responsible for reliability.

It uses timeouts to resend interests.

The sender of data is stateless.

The strategy layer can also use multiple faces with a policy such *UsageBasedCharging* or *PeakUseLimited*.

Sender will only deliver data once.

CCN can work with IP's routing scheme.

Security

Protection and trust are part of the content, not the connection.

Content packets are signed which binds the name, key and content giving integrity, pertinence and provenance.

Keys are retrieved as content which allows arbitrary trust models to be developed.

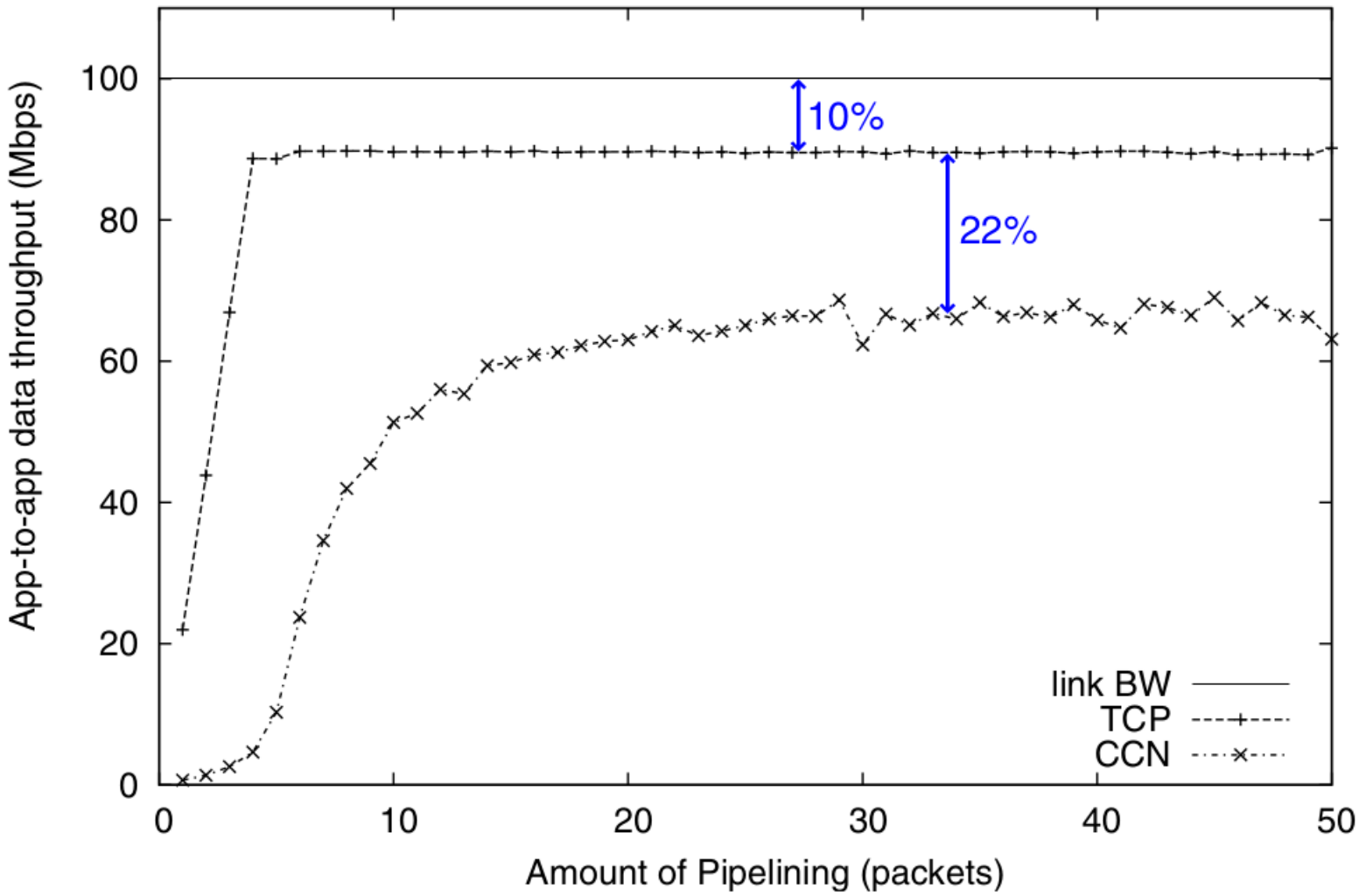
Example: hierarchical or peer-to-peer.

Evaluation

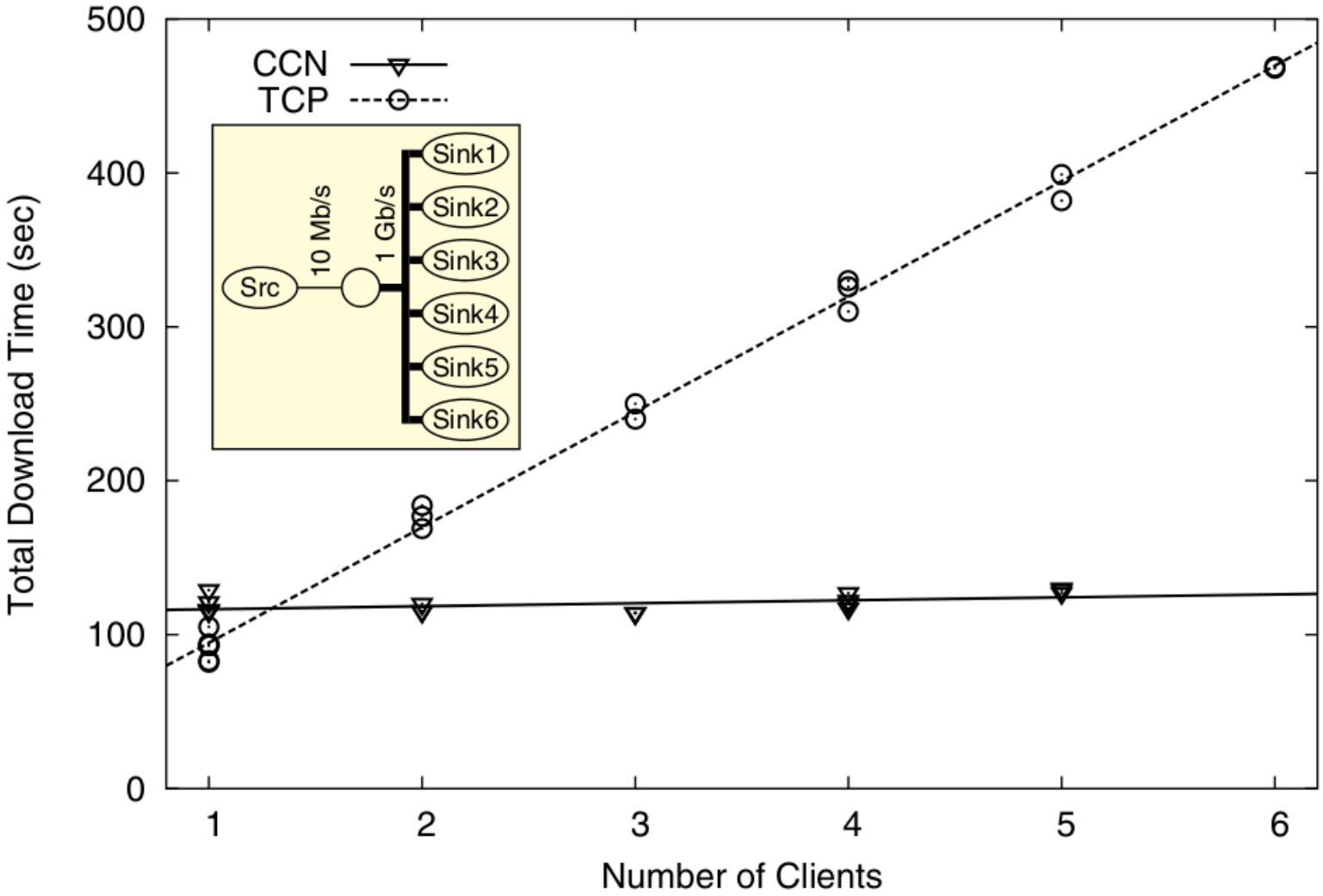
Implemented as userspace daemon and client library.

For testing, CCN packets are encapsulated in UDP.

Evaluation


App-to-app data throughput (Mbps)
100
80
60
40
20
0
0
10
20
30
40
50
Amount of Pipelining (packets)
10%
22%
link BW
TCP
CCN

Evaluation


CCN
TCP
Src
10 Mb/s
1 Gb/s
Sink1
Sink2
Sink3
Sink4
Sink5
Sink6
Total Download Time (sec)
Number of Clients
0
100
200
300
400
500
1
2
3
4
5
6

Opinion

The paper is overly ambitious.

It tries to replace:

- IP
- TCP and UDP
- HTTPS and other application layer protocols
- PKI

Opinion

Provides a good solution to the original problem.

It seems to match up with HTTP GET requests for static data.

It does not seem like a complete replacement for the TCP/IP network stack.

What about SSH, VPNs, pushing data with online backup, gaming, etc?

Opinion

What about unwanted caching?

What happens to all the application layer protocols?

What about broadcasting in a large busy LAN?

How does application demultiplexing work?

How does caching work with dynamically generated content?