Delay-Tolerant Networking: *Architecture & Applications*

Kevin Fall

Intel Research, Berkeley

kfall@intel.com

http://WWW.DTNRG.ORG

Nov 9, 2004 / Princeton, NJ

Intel Research Berkeley

Outline

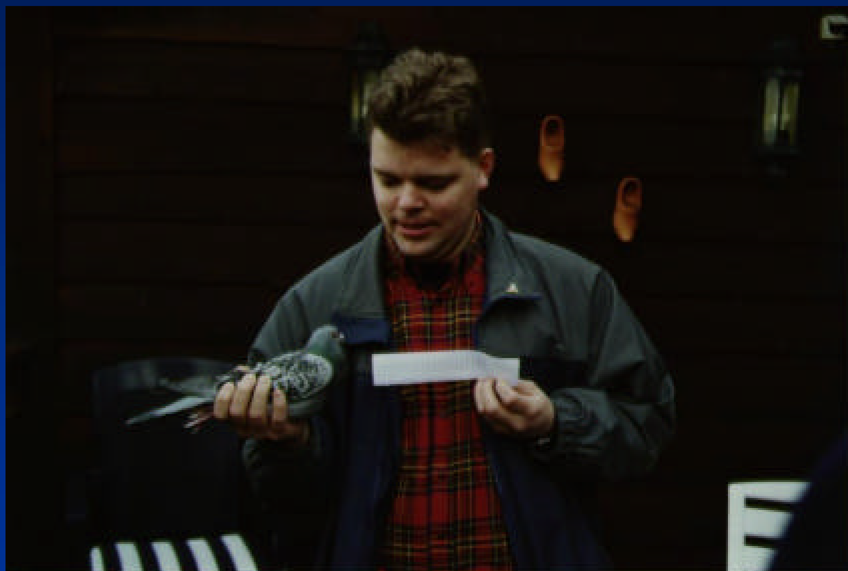
- *Why the Internet Architecture is not a ‘one-size-fits-all’ solution*
- *DTN Architecture Overview*
- *Applications & Recent Implementation Work*

RFC1149 : A Challenged Internet

- "…encapsulation of IP datagrams in avian carriers" (i.e. birds, esp carrier pigeons)
- Delivery of datagram:
  - Printed on scroll of paper in hexadecimal
  - Paper affixed to AC by duct tape
  - On receipt, process is reversed, paper is scanned in via OCR

Intel Research Berkeley

Implementation of RFC1149



CPIP: Carrier Pigeon Internet Protocol

- See http://www.blug.linux.no/rfc1149/

Ping Results

```
Script started on Sat Apr 28 11:24:09 2001
vegard@gyversalen:~$ /sbin/ifconfig tun0
tun0      Link encap:Point-to-Point Protocol
          inet addr:10.0.3.2  P-t-P:10.0.3.1  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:150  Metric:1
          RX packets:1 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0
          RX bytes:83 (83.0 b)  TX bytes:168 (168.0 b)

vegard@gyversalen:~$ [illegible] 10.0.3.1
PING 10.0.3.1 (10.0.3.1): 56 data bytes
64 bytes from 10.0.3.1: icmp_seq=0 ttl=255 time=6165731.1 ms
64 bytes from 10.0.3.1: icmp_seq=4 ttl=255 time=3211900.8 ms
64 bytes from 10.0.3.1: icmp_seq=2 ttl=255 time=5124922.8 ms
64 bytes from 10.0.3.1: icmp_seq=1 ttl=255 time=6388671.9 ms

--- 10.0.3.1 ping statistics ---
9 packets transmitted, 4 packets received, 55% packet loss
round-trip min/avg/max = 3211900.8/5222806.6/6388671.9 ms
vegard@gyversalen:~$ exit

Script done on Sat Apr 28 14:14:28 2001
```

Private Addresses

About 1.5 Hrs

High Loss

Unstated Internet Assumptions

- End-to-end RTT is not terribly large
  - A few seconds at the very most [typ < 500ms]
  - (window-based flow/congestion control works)
- Some path exists between endpoints
  - Routing usually finds single “best” existing route
    - [ECMP is an exception]
- E2E Reliability using ARQ works well
  - True for low loss rates (under 2% or so)
- Packet switching is the right abstraction
  - Internet/IP makes packet switching interoperable

Intel Research Berkeley

Non-Internet-Like Networks

- Stochastic and periodic mobility
  - Military/tactical networks
  - Mobile routers w/disconnection (e.g. ZebraNet)

  Spacecraft communications (LEO sats)
  - Busses, mail trucks, delivery trucks, etc. (InfoStations)
- "Exotic" links
  - Deep space [Mars: 40 min RTT; episodic connectivity]
  - Underwater [acoustics: low capacity, high error rates & latencies]
  - Sensor networks, mules

DTN challenges…

- Intermittent/Scheduled/Opportunistic Links
  – Scheduled transfers can save power and help congestion; scheduling common for esoteric links
- High Link Error Rates / Low Capacity
  – RF noise, light or acoustic interference, LPI/LPD concerns
- Very Large Delays
  – Natural prop delay could be seconds to minutes
  – If disconnected, may be (effectively) much longer
- Different Network Architectures
  – Many specialized networks won't/can't ever run IP

Intel Research Berkeley

What to Do?

- Some problems surmountable using Internet/IP
  - ‘cover up’ the link problems using PEPs
  - Mostly used at “edges,” not so much for transit
- Performance Enhancing Proxies (PEPs):
  - Do “something” in the data stream causing endpoint (TCP/IP) systems to not notice there are problems
  - Lots of issues with transparency– security, operation with asymmetric routing, etc.
- Some environments *never* have an e2e path
  - Consider an approach tolerating disconnection, etc...

Delay-Tolerant Networking Architecture

- Goals
  - Support interoperability across ‘radically heterogeneous’ networks
  - Acceptable performance in high loss/delay/error/disconnected environments
  - Decent performance for low loss/delay/errors
- Components
  - Flexible naming scheme with *late binding*
  - Message overlay abstraction and API
  - Routing and link/contact scheduling w/CoS
  - Per-(overlay)-hop reliability and authentication

Naming

- Support ‘radical heterogeneity’ using *regions*:
  - Instance of an internet, not so radical inside a region
  - Common naming and protocol conventions
- Endpoint Name: ordered name pair **{R,L}**
  - **R**: routing region [globally valid]
  - **L**: region-specific, opaque outside region **R**
- **Late binding** of **L** permits naming flexibility:
  - L used only in destination region of interest R
  - Could be associative or location-oriented names [URN vs URL]
  - May encompass esoteric routing [e.g. diffusion]
  - Perhaps an Internet-style URI [see RFC2396]
- *To do*: make **R,L** compressible in transit networks

Intel Research Berkeley

Message Overlay Abstraction

- E2E Async Message Service: “Bundles”
  - “postal-like” message delivery over regional transports with coarse-grained CoS [4 classes]
  - *Options*: return receipt, “traceroute”-like function, alternative reply-to field, custody transfer
  - Supportable on nearly any type of network
- Applications send/receive <u>messages</u>
  - “Application data units” of possibly-large size
  - May require framing above some transport protocols
  - API supports response processing long after request was sent (application *re-animation*)

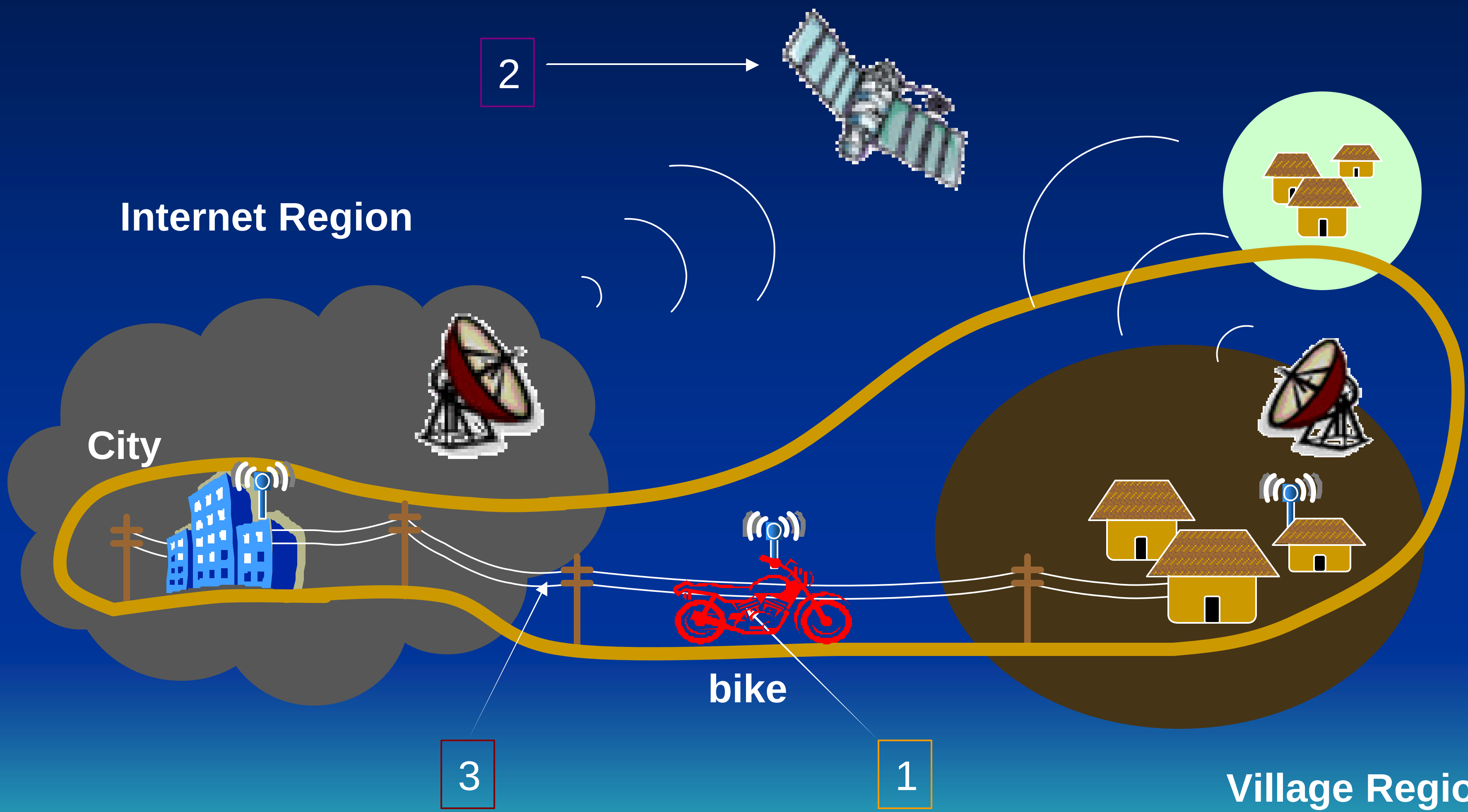
Intel Research Berkeley

So, is this just e-mail?

	naming/	routing	flow	multi-	security	reliable	priority
	late binding		contrl	app		delivery	
e-mail	Y	N	N(Y)	N(Y)	opt	Y	N(Y)
DTN	Y	Y	Y	Y	opt	opt	Y

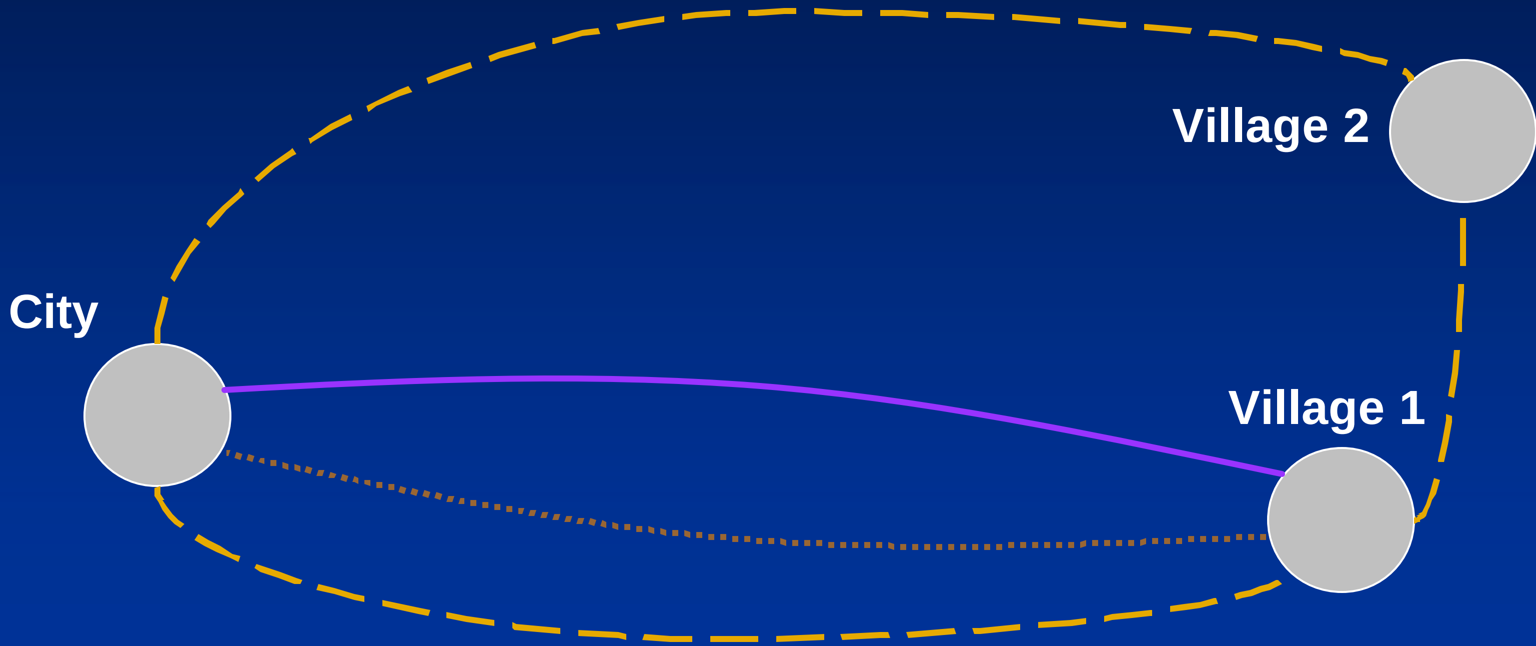
- Many similarities to (abstract) e-mail service
- Primary difference involves routing/restart and API
- E-mail depends on an underlying layer’s routing:
  - Cannot generally move messages closer to their destinations in a partitioned network
  - In the Internet (SMTP) case, not disconnection-tolerant or efficient for long RTTs due to “chattiness”
- E-mail security authenticates only user-to-user

Example Routing Problem



Intel Research Berkeley

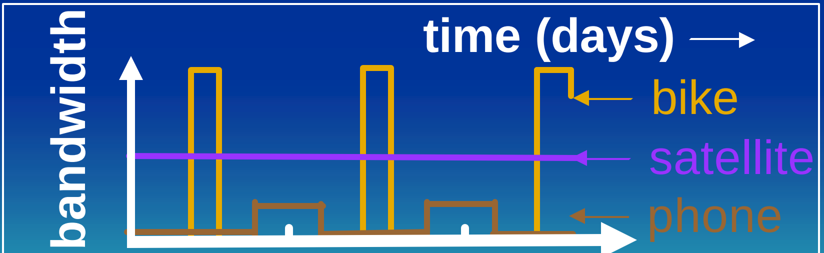
Example Graph Abstraction



bike (data mule)
intermittent high capacity

Geo satellite
medium/low capacity

dial-up link
low capacity



Connectivity: Village 1 – City

Routing on Dynamic Graphs

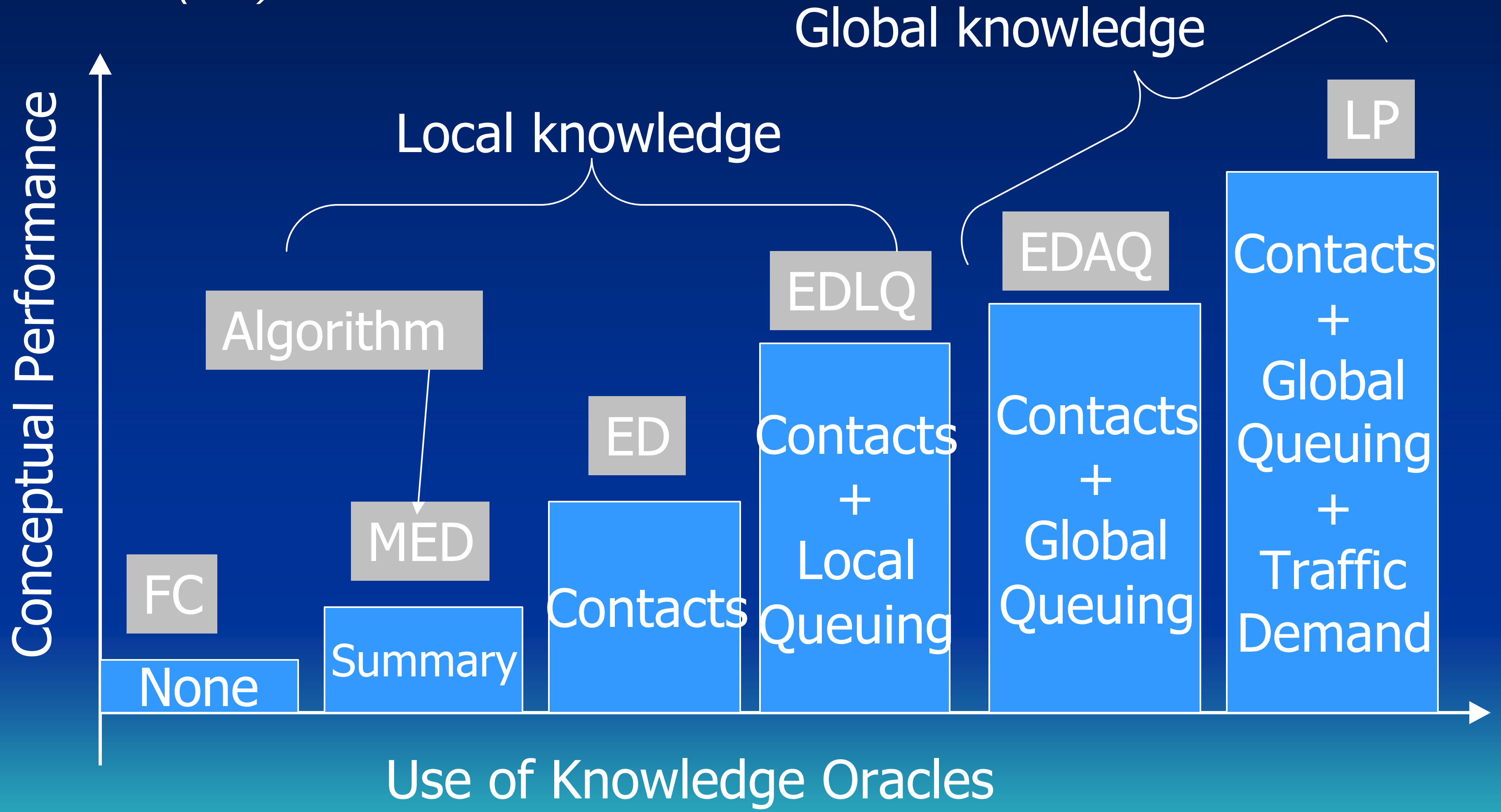
- DTN routing takes place on a time-varying topology
  - Links come and go, sometimes predictably
  - Use any/all links that can possibly help
- Scheduled, Predicted, or Unscheduled Links
  - May be direction specific [e.g. ISP dialup]
  - May learn from history to predict schedule
- Messages fragmented based on dynamics
  - Proactive fragmentation: optimize contact volume
  - Reactive fragmentation: resume where you failed
  - Both are important for high utilization of precious link resources

The DTN Routing Problem

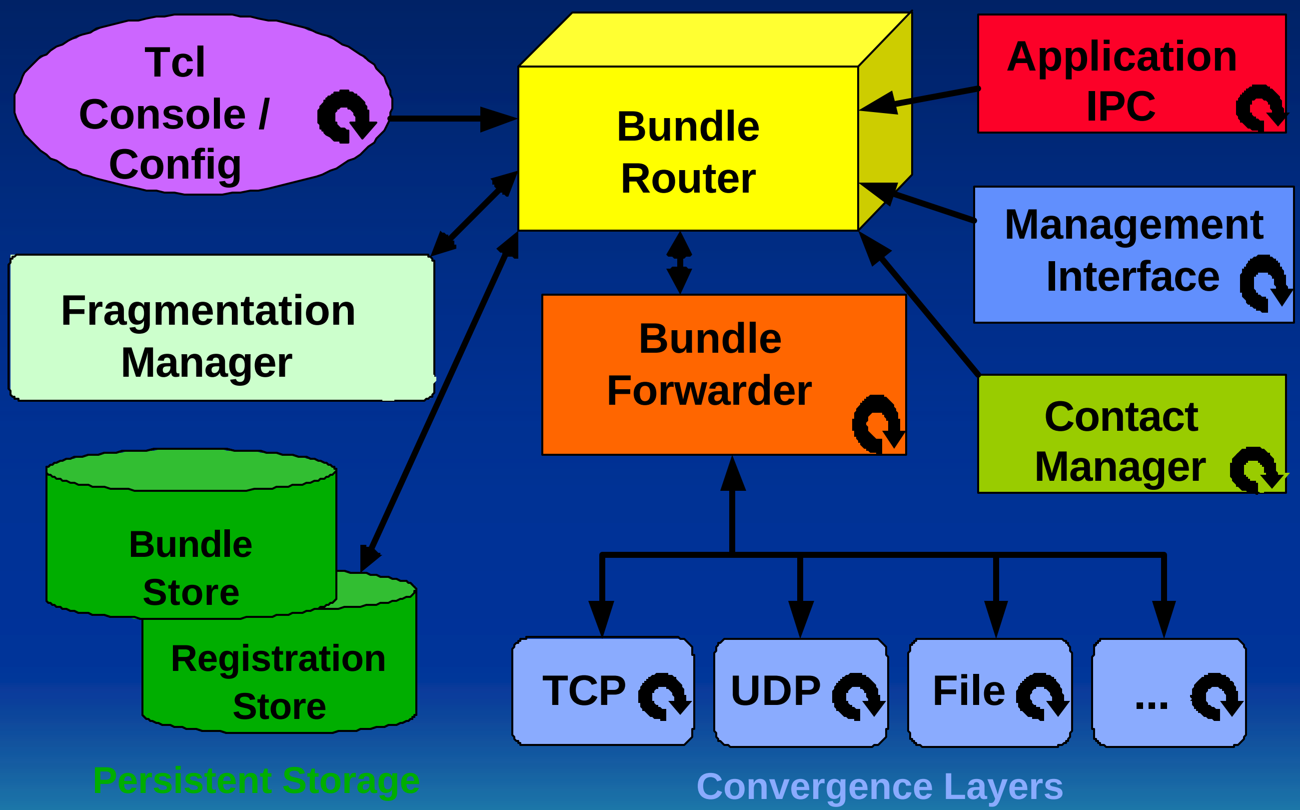
- *Inputs*: topology (multi)graph, vertex buffer limits, contact set, message demand matrix (w/priorities)
- An ***edge*** is a possible opportunity to communicate:
  - One-way: (S, D, c(t), d(t))
  - (S, D): source/destination ordered pair of contact
  - c(t): capacity (rate); d(t): delay
  - A ***Contact*** is when c(t) > 0 for some period $[i_k, i_{k+1}]$
- Vertices have buffer limits; edges in graph if ever in any contact, multigraph for multiple physical connections
- *Problem*: optimize some metric of delivery on this structure
  - Sub-question: what metric to optimize?

Knowledge vs Performance

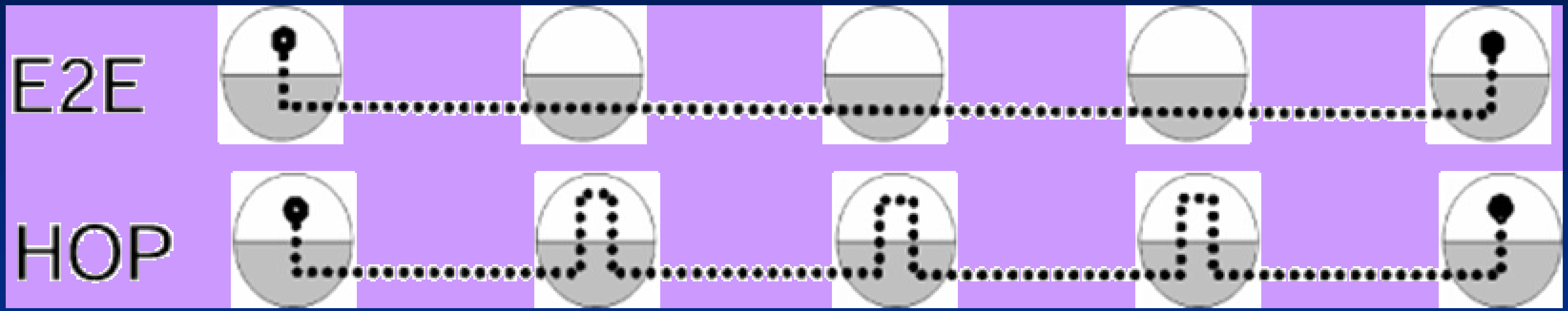
S. Jain (UW): SIGCOMM 2004



‘DTN2’ Implementation

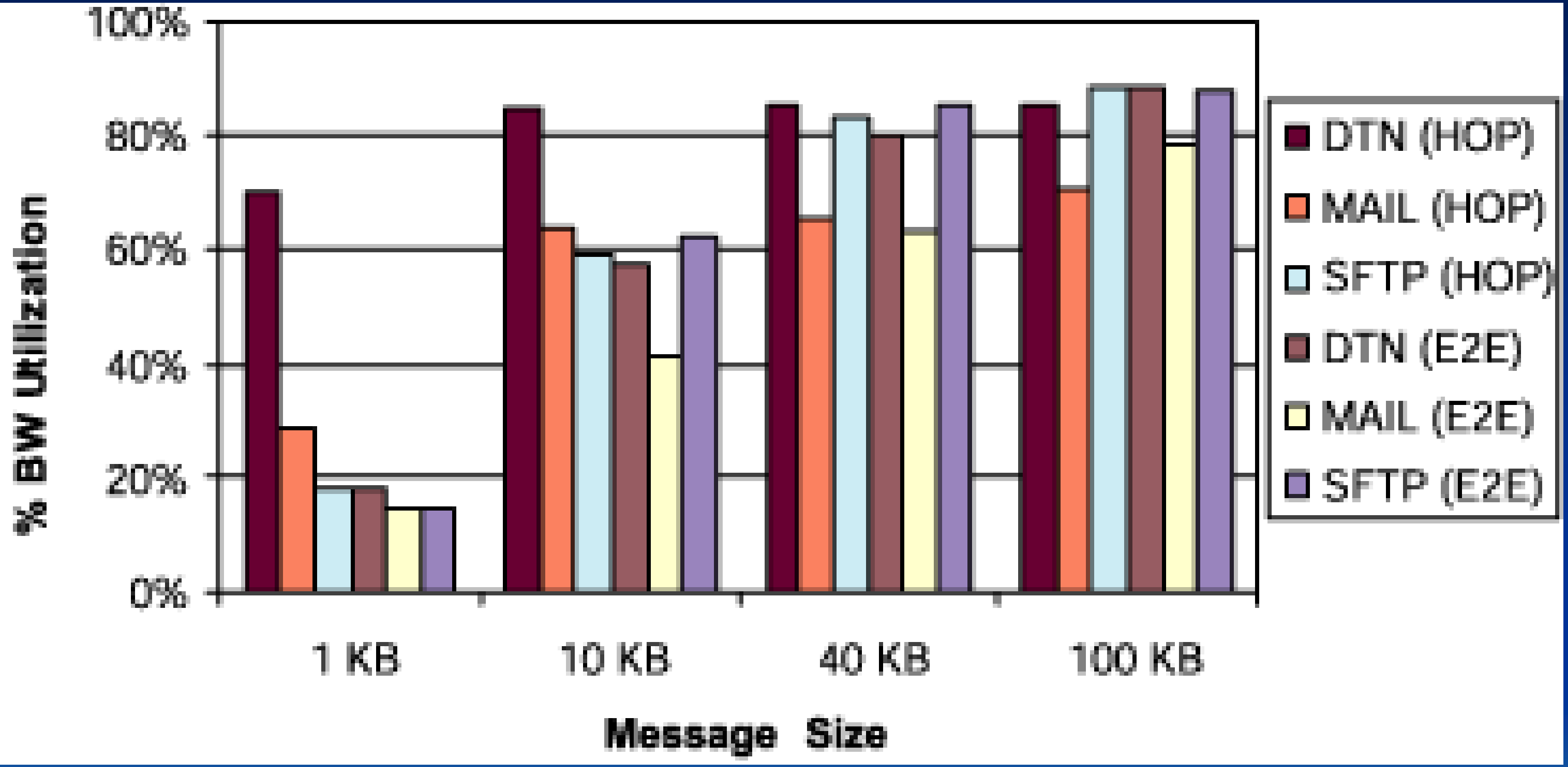


Experiment Setup



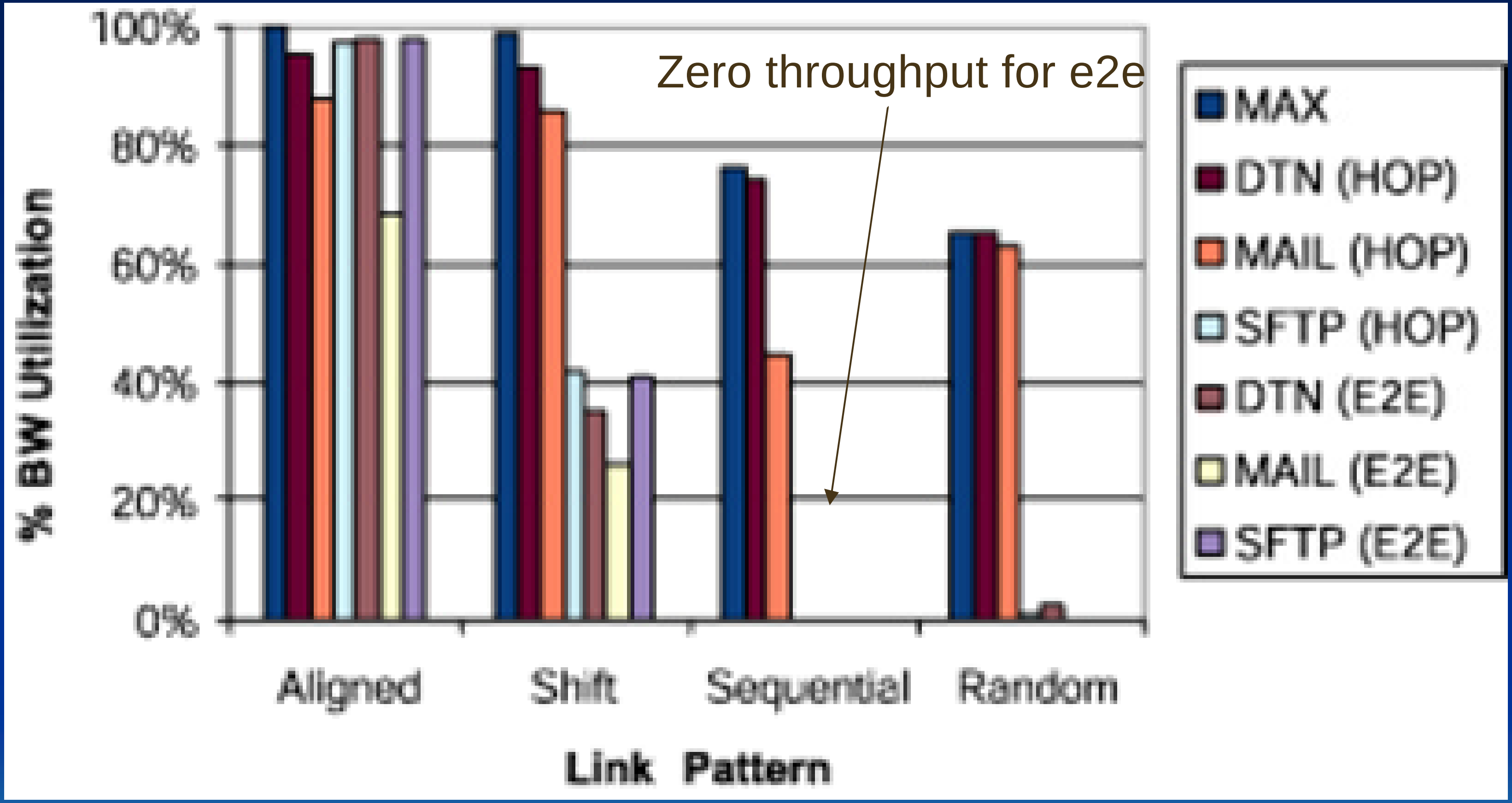
- Compare robustness to interruption / link errors
- Approaches compared
  - End-to-end TCP (kernel routing)
  - Proxied (TCP ‘plug proxies’)
  - Store-and-forward (Sendmail, no ckpoint/restart)
  - DTN (store-and-forward with restart)
- Link up/down patterns: aligned, shifted, sequential, random

BW Efficiency



No disruptions: DTN does well for small msgs, modest overhead overall

Interruption Tolerance



Up/down 1m/3min; 40kb messages; shift: 10s

Conclusions

- DTN foundational concepts appear to have wide applicability
- DTN Routing is a rich and challenging problem
- Reference implementation can be tricky
- Early performance results suggest our approach to disruption tolerance is effective

Status

- IETF/IRTF DTNRG formed end of 2002
  – See http://www.dtnrg.org
- DTN1 Agent Source code released 3/2003
- SIGCOMM Papers: 2003 [arch], 2004 [routing]
- Several other documents (currently ID's):
  – DTNRG Architecture document
  – Bundle specification
  – Application of DTN in the IPN
- Basis for new DARPA DTN program
- Part of NSF 'ICT4B' Project (with UCB)

On to an application…

ICT for Billions (ICT4B)

- Information and Communication Technologies for Developing Regions of the World
- Networking focus: *intermittent networking*
  - *Mission-specific architecture and API*
  - *Multiple layers of network intermittency*

ICT4B Application Areas

- E-Government
  - Forms, status updates, certifications
- Health
  - Early screening
- Trade
  - Price dissemination, market making
- Education
  - Various topics: health, agriculture, microfinance, etc.
- Alerts / News / Weather
- General communication

Intel Research Berkeley

ICT4B Technology Areas

- Task-Specific Devices
  - Hand-held with speech recognition
  - Local wireless
  - Sensors
  - Uses: Medical, data entry, information, etc.
- Intermittent Networking
  - DTN forms the underlying networking technology
  - Capable of supporting async messaging over most any comms technology
- Distributed System Architecture
  - Back-end services in data center (databases, trading system, etc.)
  - Village-level kiosks (cache, I/O capability with devices, printer)
- Speech Recognition
  - Speaker-independent small-vocabulary approach
  - (currently taking samples in Tamil)
- Very Low Cost Displays
  - Using ink-jet printing approach

Some of The Team…[7/2004]

Intel Research Berkeley

MSSRF (Villianur)…[7/2004]

MSSRF (Kizhur?)...[7/2004]

Intel Research Berkeley

MSSRF (Veerampattinam)...[7/2004]

ICT4B Project Status

- ICT4B NSF ITR funded 10/2003 (5yr)
- DTN forwarding layer and early apps being tested (code released 3/2003)
- Joint UCB/Intel attendance at ‘ICT for Sustainable Development’ conference Jan 2004/Bangalore; ‘Bridging the Divide’ conference Mar 2004/Berkeley; ‘Digital Rally’ Apr 2004/San Jose; PolicyMaker’s Workshop July 2004/Delhi
- Fellow travelers: HP Labs India, IIT Bombay/Kanpur/Madras, Univ. of Washington, MITRE, DARPA, NSF, CMU, UCLA, JPL, U Waterloo, MCI

Intel Research Berkeley

Acknowledgements

- DTN/ICT4B People:
  - Eric Brewer, Mike Demmer, Rabin Patra (UCB)
  - Bob Durst, Keith Scott (MITRE)
  - Kevin Fall, Melissa Ho (Intel Research Berkeley)
  - Sushant Jain (UW Intern)
  - S. Keshav (U Waterloo)
  - Ting Liu (Princeton)
- Other DTN (IPN) People:
  - Vint Cerf (MCI)
  - Scott Burleigh, Adrian Hooke (NASA/JPL)
  - Forrest Warthman (Warththman)
  - Stephen Farrell (Trinity College, Ireland)
  - The *dtn-interest* mailing list

Intel Research Berkeley

http://www.dtnrg.org

Thank You!

http://tier.cs.berkeley.edu

Intel Research Berkeley