peering into the unknown: p2p meets dtn

Jon Crowcroft ,Pan (Ben) Hui (*)
Augustin Chaintreau, Christophe Diot (!)
James Scott, Richard Gass (+)

(*) University of Cambridge
(+) Intel Research Cambridge
(!) Thomson Research, Paris

Infrastructreless Systems

- Three related infrastructureless systems
  - P2P
  - MANET
  - DTN
- Integrating P2P+MANET done (e.g. crosslayer)
- Integrating P2P+DTN new
- Similar *Architecture*
- Different *Performance* problems

Agenda

1. Background to DTN
2. Background to Haggle Architecture
3. Performance Questions and Answers

Part 1. DTN Origin/Confusion

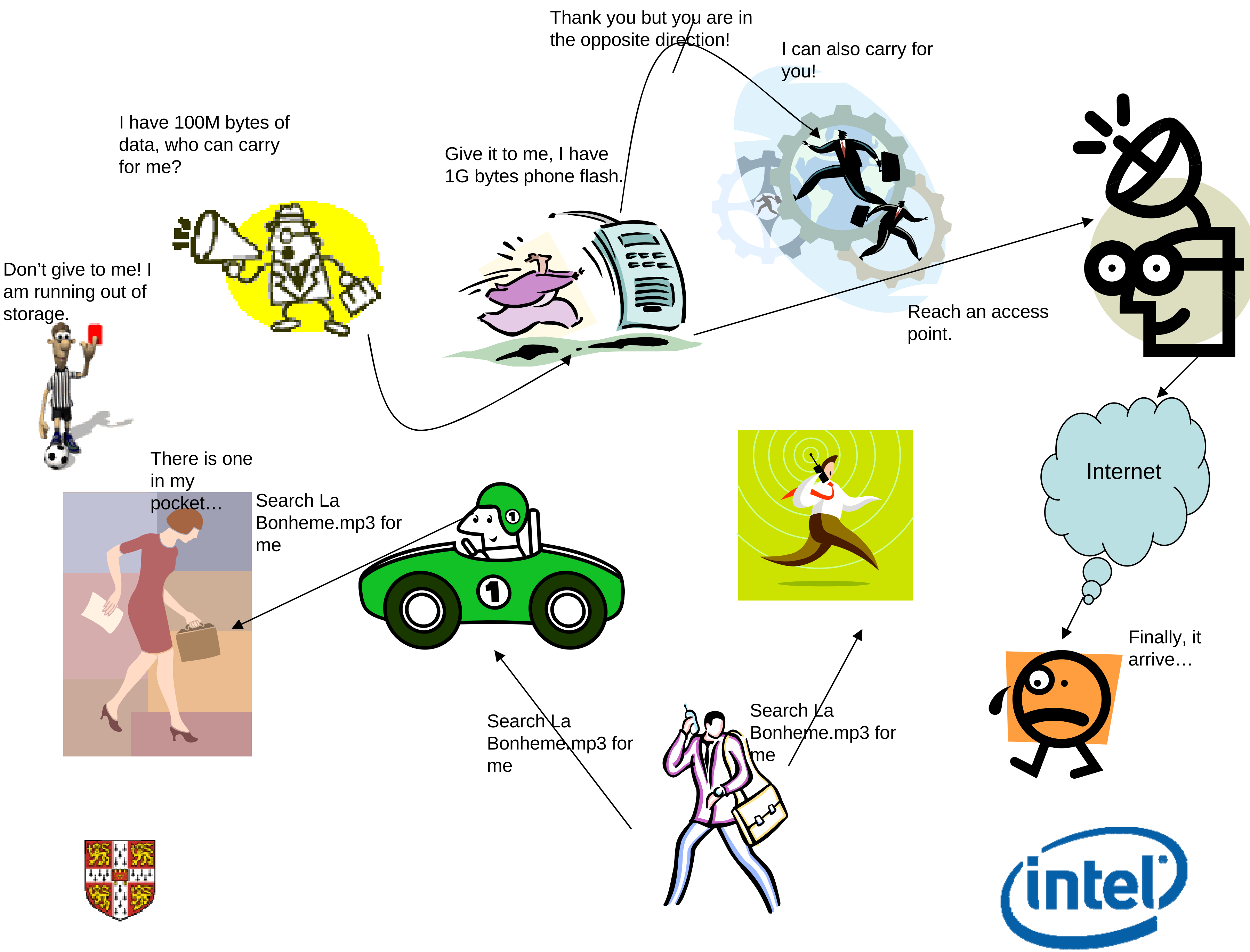
- DTN=Delay Tolerant Networks
  - E.g. Interplanetary Internet
  - D for Deep Space
- DTN = Disruption Tolerant Networks
  - E.g. Biosphere sensor nets
  - D for Deep Sea
- DTN = Disconnection Tolerant Networks
  - E.g. Tsunami, Terrorism, Katrina
  - D for Deep S**t ^H^H^H Poverty

My Kind of DTN: Haggle=Pocket Switched Networks

- Is there a packet in your pocket or are you pleased to see me?
- Crucially, concentrate on humans for now (or zebra, sheep, dolphins)

Pocket switched networks motivations

- Not always connected, “internet connectivity islands”
- Huge amount of untapped resources in portable devices
  - Local wireless bandwidth
  - Storages
  - CPUs
- A packet can reach destination using network connectivity or user mobility
- MANET ~= DTN [Fall]


Thank you but you are in the opposite direction!
I can also carry for you!
I have 100M bytes of data, who can carry for me?
Give it to me, I have 1G bytes phone flash.
Don't give to me! I am running out of storage.
Reach an access point.
Internet
There is one in my pocket…
Search La Bonheme.mp3 for me
Search La Bonheme.mp3 for me
Search La Bonheme.mp3 for me
Finally, it arrive…
intel

Applications

- Asynchronous, local messaging
- Automatic address book or calendar updates
- Ad-hoc Google
- File sharing, bulletin board
- Commercial transactions
- Alerting, tracking or finding people

- Inherently, P2P… … …

Part 2: Haggle Software Architecture

Contents

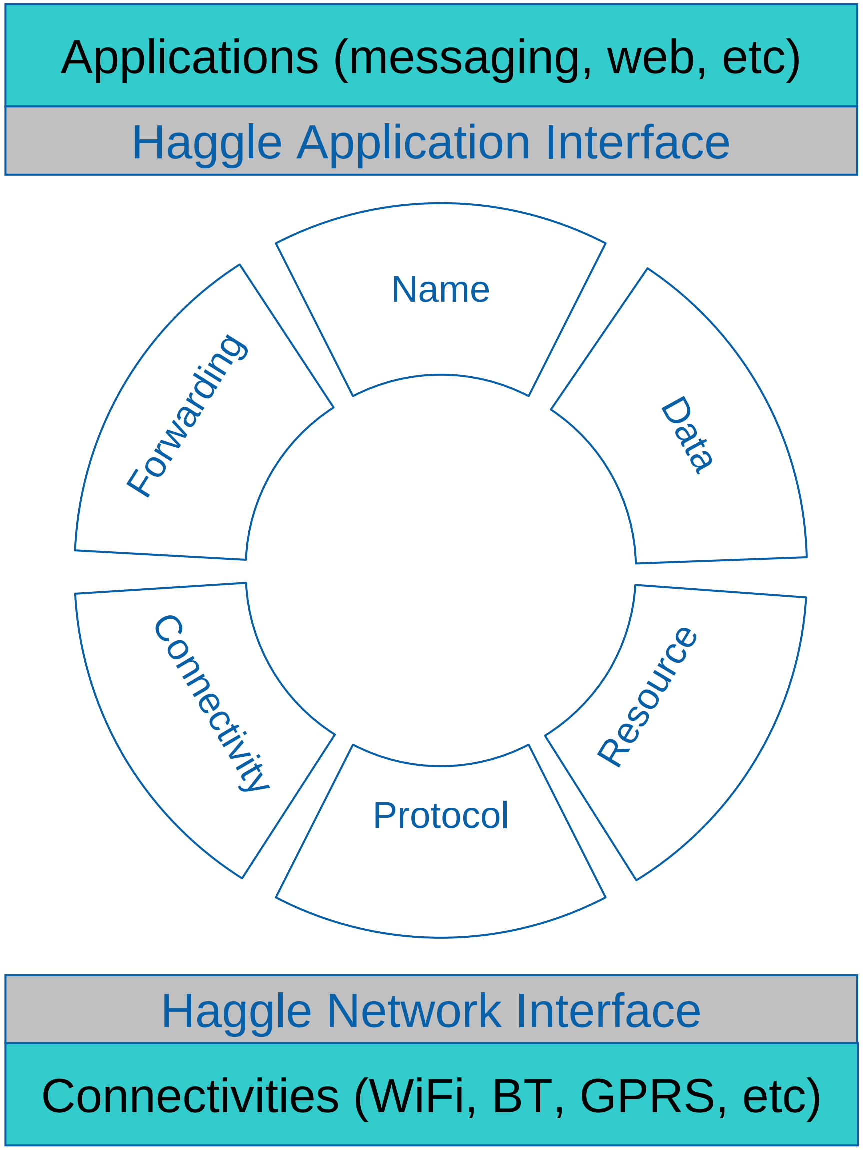
- Application interface changes
- Haggle managers
- Data representation
- Forwarding algorithms
- Resource management
- Further work/open questions

Application interface changes

- We take persistent data structures and app-level protocols away from the application and place this in Haggle
  - So Haggle can share appropriate data with other nodes
  - So apps are separated from connectivity-specific protocols
  - So interface with apps is asynchronous
- Interface is now oriented around placement of objects in tree structure inside Haggle, sending objects to destination “names” or soliciting objects

Haggle Managers

- Data: storing, linking together as trees, and searching persistent data objects
- Name: organising names of potential destinations for data
- Connectivity: proposing neighbour discovery tasks, estimation of costs of sending data, providing connections to neighbours on demand
- Protocol: using visible neighbours to mark names as “nearby”, sending and receiving data
- Forwarding: estimating benefits of sending data to nearby names
- Resource: for all tasks proposed, comparison of cost and benefits and decision of which tasks to perform now



Using Haggle for Ad Hoc Email

Intel Research Cambridge

Haggle Architecture

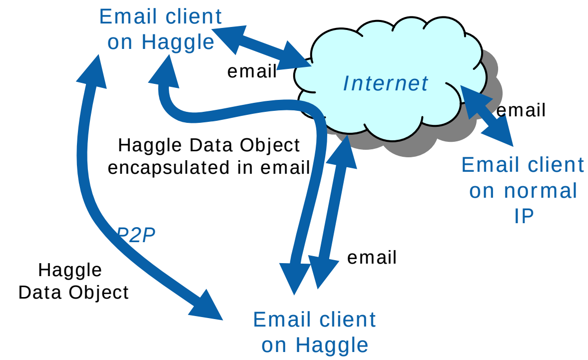
- A layerless networking architecture for mobile devices
- Separates out the “what” (e.g. a message), the “who” (James Scott; james.w.scott@intel.com) and the “how” (the end-to-end message path)
- Allows applications to become independent of infrastructure, and use ad-hoc or mobility-based connectivity as easily as Internet access
- Implemented in Java for portability; available as open source software

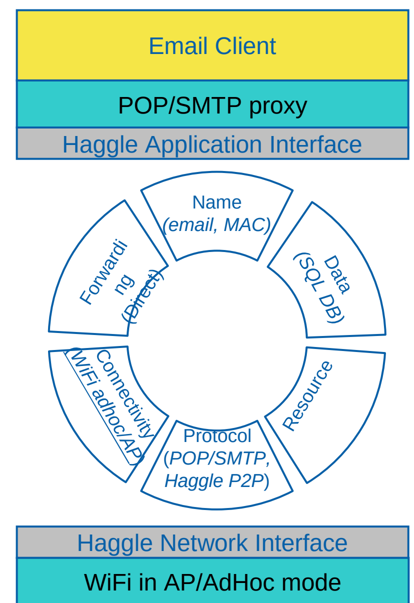
Application: Email

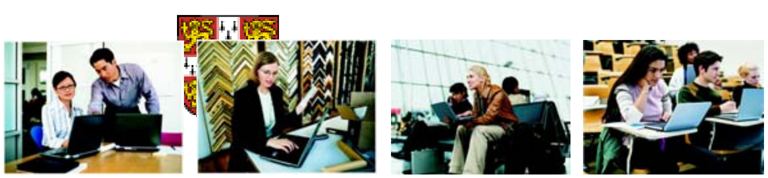
- Email uses a huge deployed infrastructure allowing asynchronous send and receive
- Haggle can make use of this infrastructure, as well as using local ad-hoc connectivity between Haggle-enabled nodes for when infrastructure is unavailable or broken
- Backwards compatible – can email people without Haggle as before, and use a POP/SMTP proxy on top of Haggle to support existing email apps

Implementation Using 802.11

- Data stored in SQL database
- Destination names map to both email and MAC addresses
- Forwarding directly to either name, or via store-and-forward
- Both POP/SMTP and a Haggle P2P protocols used
- 802.11 connectivity switched between ad-hoc and AP mode as forwarding load demands, with APs automatically assessed for Internet connectivity







Data Objects

Message

DO-Type	Data
Content-Type	Message/rfc822
From	James Scott
To	Richard Gass
Subject	Check this photo out
Body	[text]

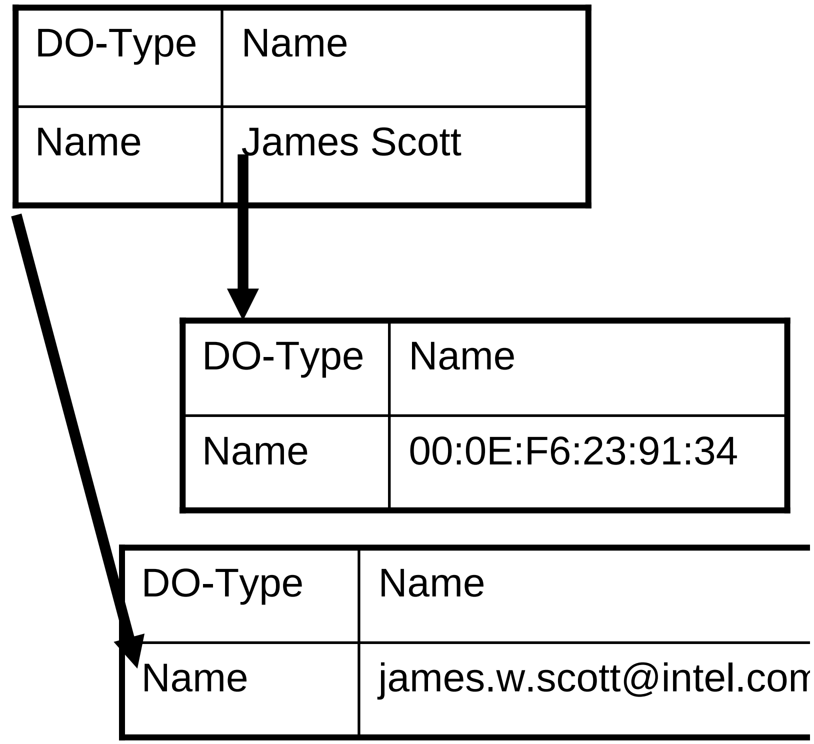
Attachment

DO-Type	Data
Content-Type	Image/jpeg
Keywords	Sunset, London
Creation time	05/06/06 2015 GMT
Data	[binary]

Apps can insert DOs, “claim” DOs, link DO to DO, etc

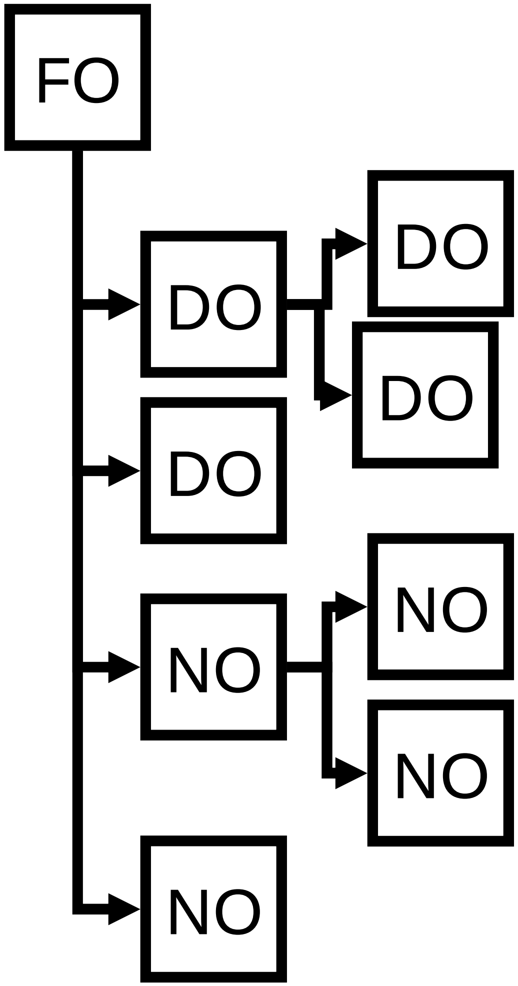
Name Objects

- Special class of DO used for naming destinations for data
- Names and links between names obtained from
  - Applications
  - Network interfaces
  - Neighbours
  - Data passing through
- Used as destinations for sent data

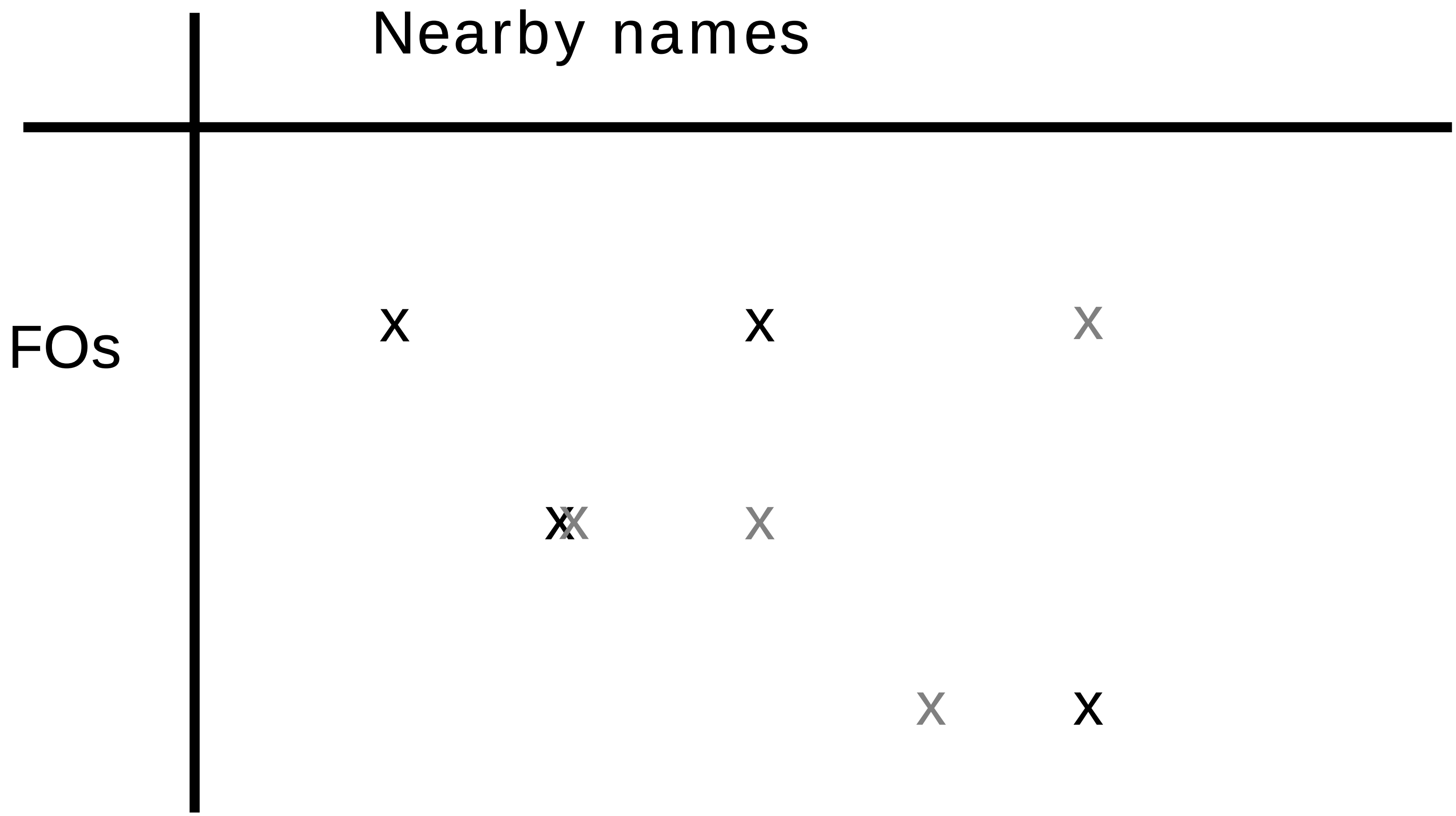


Forwarding Objects

- Special class of DO used for storing metadata about forwarding
  - TTL,expiry
  - Security
  - Destination NOs
  - Data being carried
  - Forwarding algorithm hints (e.g. who it's already been delivered to and when, suggested route, etc)



Forwarding Algorithms - Benefit



Resource Management – Cost/Benefit

- Task(int getBenefit(), Cost getCost())
- Cost = {Energy, Time on network X, Money}
- Benefit determined by forwarding algorithm
- Cost determined by Protocol and Connectivity

- Resource manager does cost/benefit comparison using some utility function
  – Should take into account FO priority (= app, user priority), as well as scarcity of resources

Part 3: Performance: Measuring Human Mobility

Mobility is a double-edged sword, it potentially increases the bandwidth, but also provides challenges for communication.

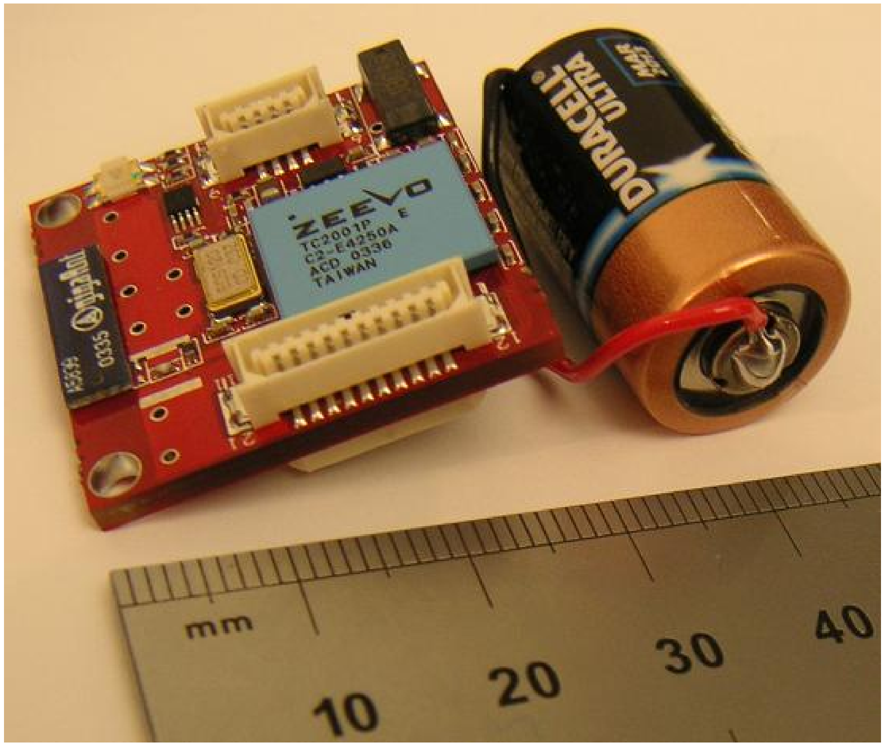
Why measure human mobility?

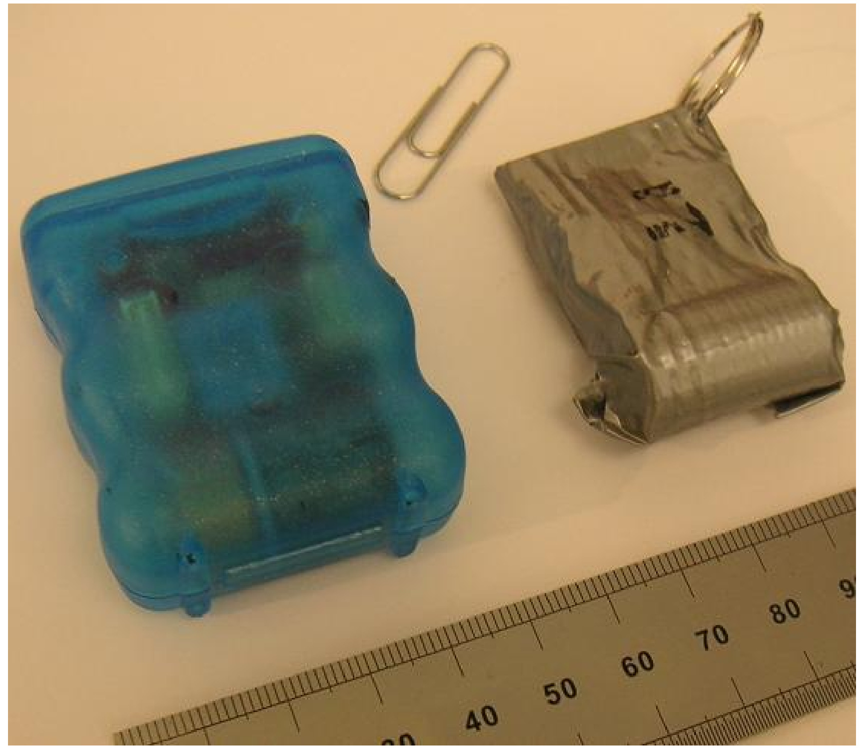
- Mobility increases capacity of dense mobile network [tse/grossglauser]
- Also create dis-connectivities
- Human mobility patterns determine communication opportunities
- And discover social groupings - see later for resource allocation (e.g. spectrum)

Experimental setup

- iMotes
  - ARM processor
  - Bluetooth radio
  - 64k flash memory
- Bluetooth Inquiries
  - 5 seconds every 2 minutes
  - Log {MAC address, start time, end time} tuple of each contact

Experimental devices





Infocom 2005 experiment

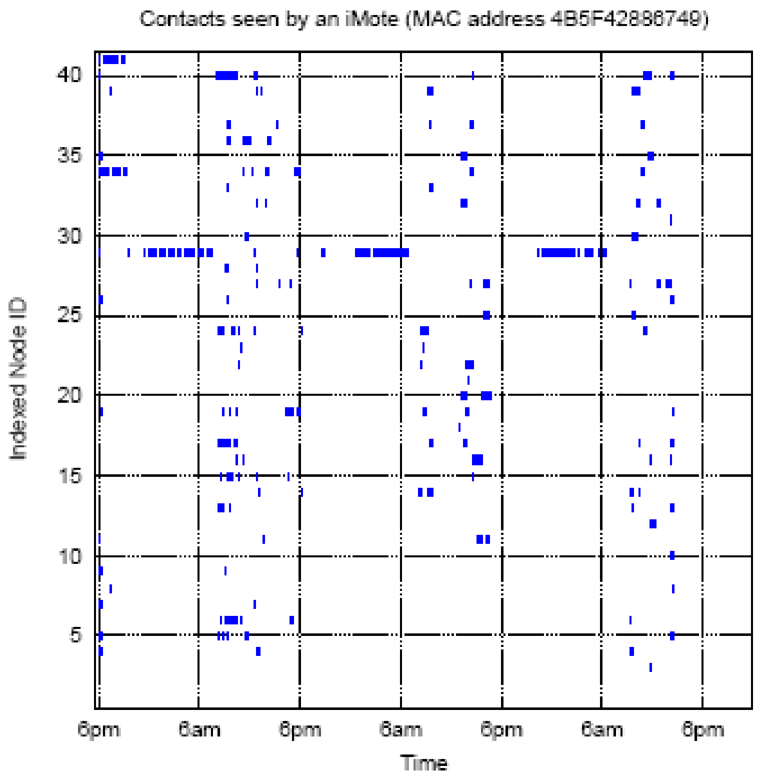
- 54 iMotes distributed
- Experiment duration: 3 days
- 41 yielded useful data
- 11 with battery or packaging problem
- 2 not returned

Brief summary of data

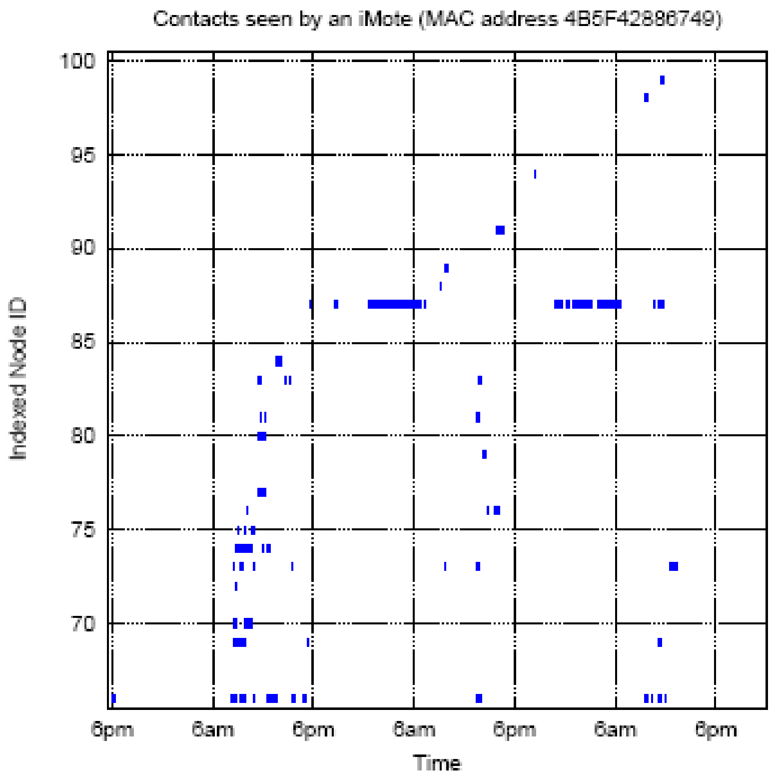
- 41 iMotes
- 182 external devices
- 22459 contacts between iMotes
- 5791 contacts between iMote/external device
- External devices are non-iMote devices in the environment, e.g. BT mobile phone, Laptop.

Contacts seen by an iMote

iMoites

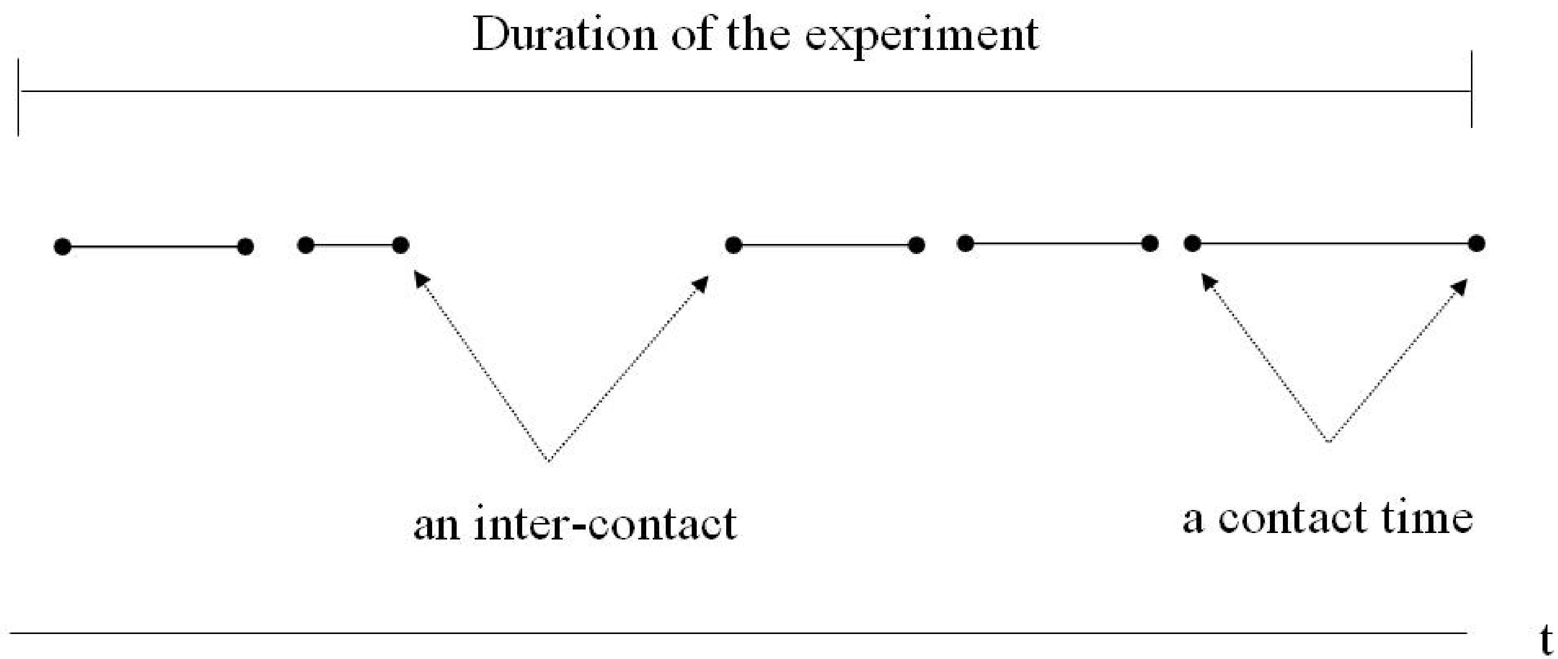


External Devices



Analysis of Conference Mobility Patterns

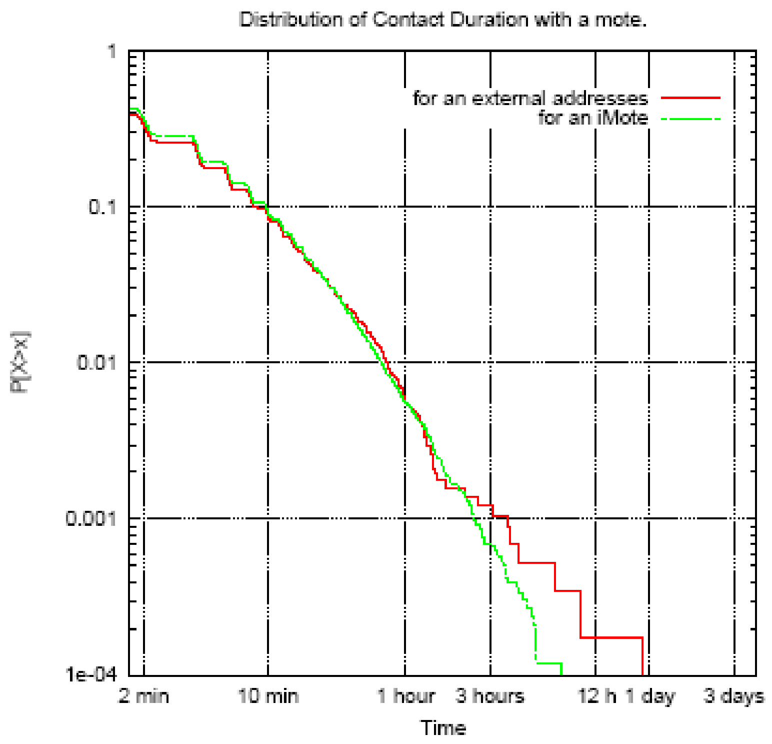
Contact and Inter-contact time



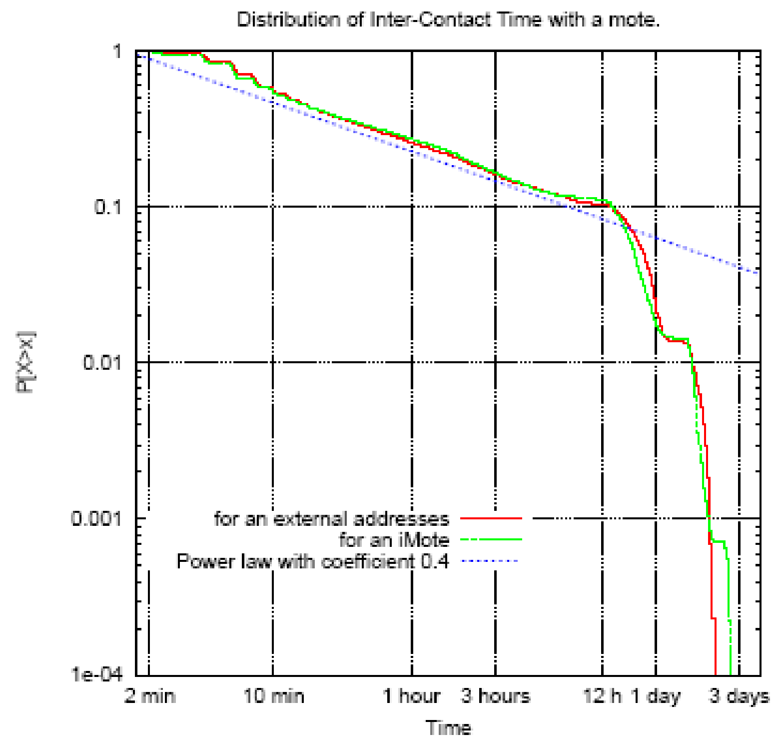
- Inter-contact is important
  - Affect the feasibility of opportunistic network
  - Nature of distribution affects choice of forwarding algorithm
  - Rarely studied

Contact and Inter-contact Distribution

Contacts



Inter-contacts



What do we see?

- Power law distribution for contact and Inter-contact time
- Both iMotes and external nodes
- Does not agree with currently used mobility model, e.g. random way point
- Power law coefficient < 1

Implication of Power Law Coefficient

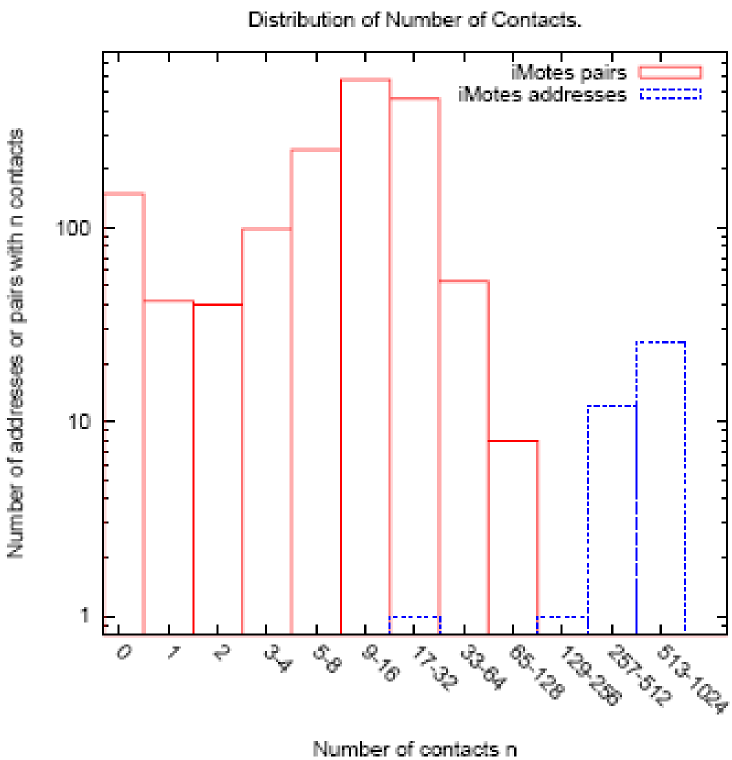
- Large coefficient => Smaller delay
- Consider 2-hops relaying [tse/grossglauser] analysis [TechRep]
- Denote power law coefficient as α
- For $\alpha > 2$

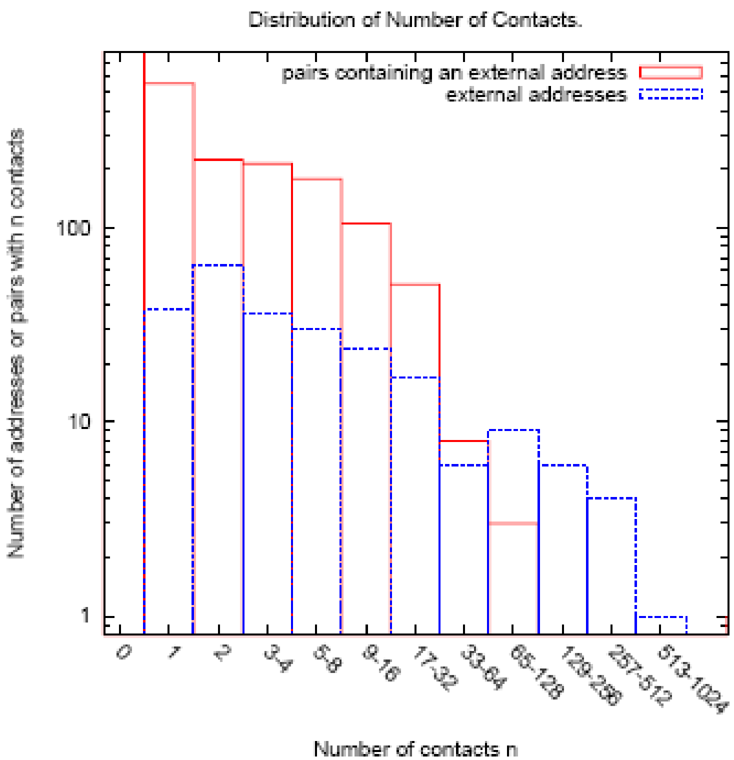
  Any stateless algorithm achieves a finite expected delay.
- For $\alpha > (m+1)/m$ and #{nodes} $\geq$2m :

  There exist a forwarding algorithm with m copies and a finite expected delay.
- For $\alpha < 1$

  No stateless algorithm (even flooding) achieve a bounded delay (Orey's theorem).

Frequency of sightings and pairwise contact

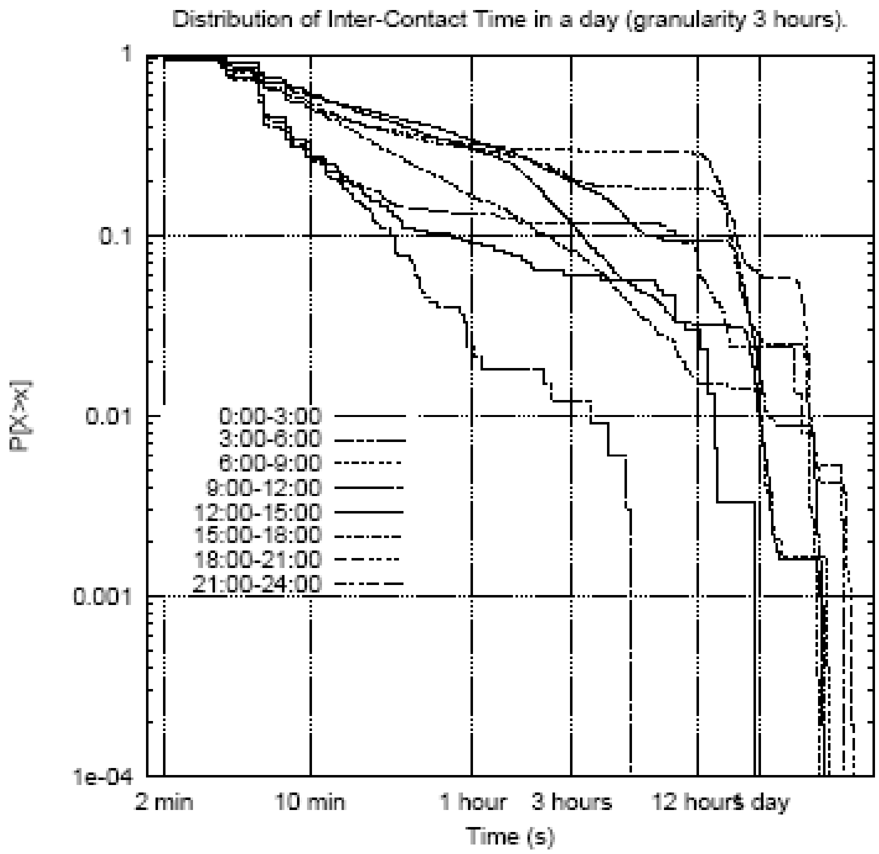




What do we see?

- Nodes are not equal, some active and some not
  – Does not agree with current mobility model, equally distributed.
- iMotes are seen more often than external address
- More iMotes pair contact
  – Identify Sharing Communities to improve forwarding algorithm

Influence of time of day



What do we see?

- Still a power law distribution for any three-hour period of the day
- Different power law coefficient for different time
  - Maybe different forwarding algorithm for different time of the day

Consequences for mobile networking

- Mobility models needs to be redesigned
  - Exponential decay of inter contact is wrong
  - Mechanisms tested with that model need to be analyzed with new mobility assumptions
- Stateless forwarding does not work
  - Can we benefit from heterogeneity to forward by communities ?
  - Should we consider different algorithm for different time of the day?

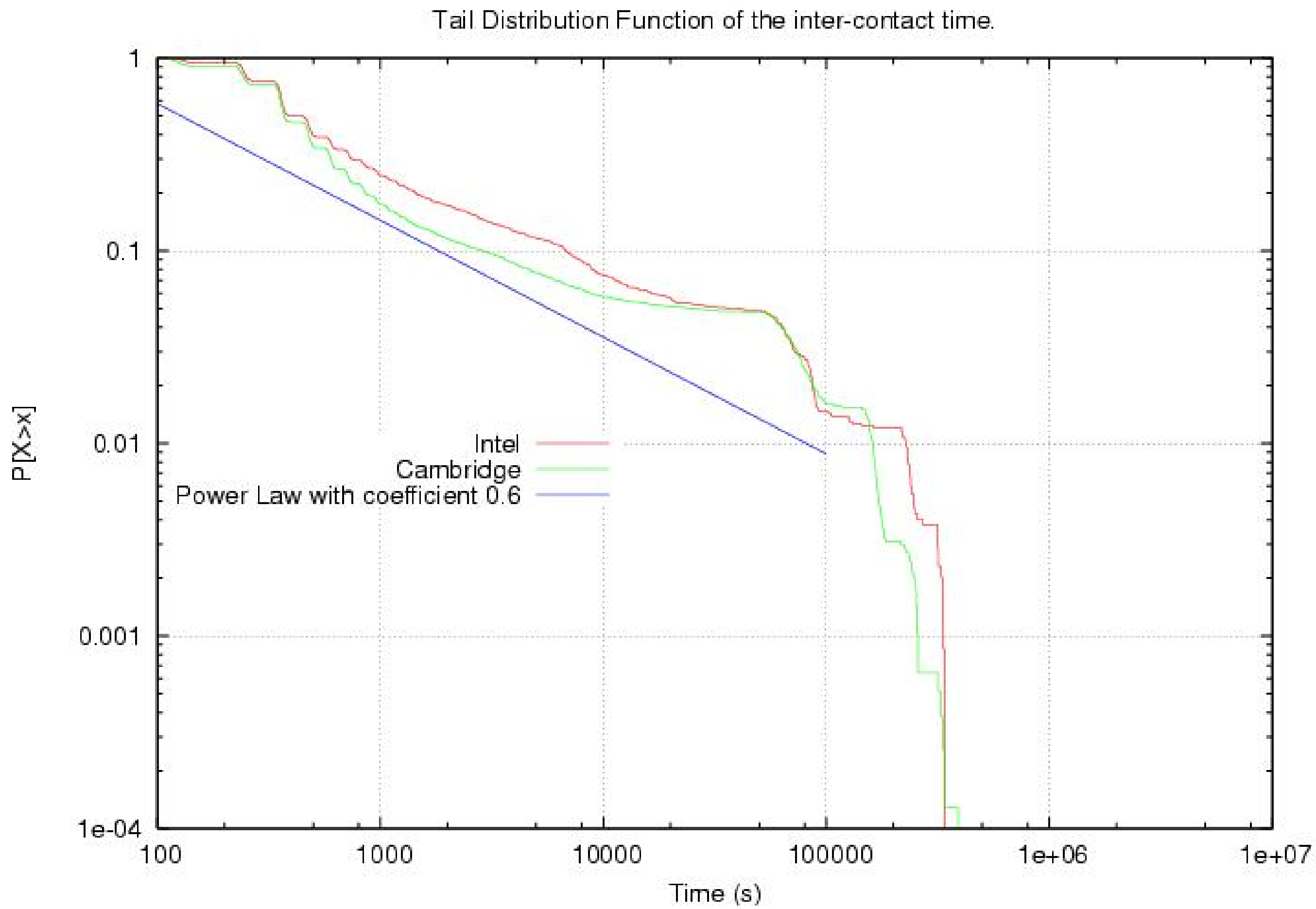
Future Work

- Continue mobility measurement in different network environments+mathematical analysis
- Uncovering realistic mobility/social net models
- Design and evaluate forwarding algorithms:
- Social Group/Network may serve:
- Forwarding, Content and Trust
- Prototyping PSN applications, e.g. distributed file sharing (index+search – hence “haggle” = ad hoc google☺)
- Usability (e.g. feedback/visualisation)

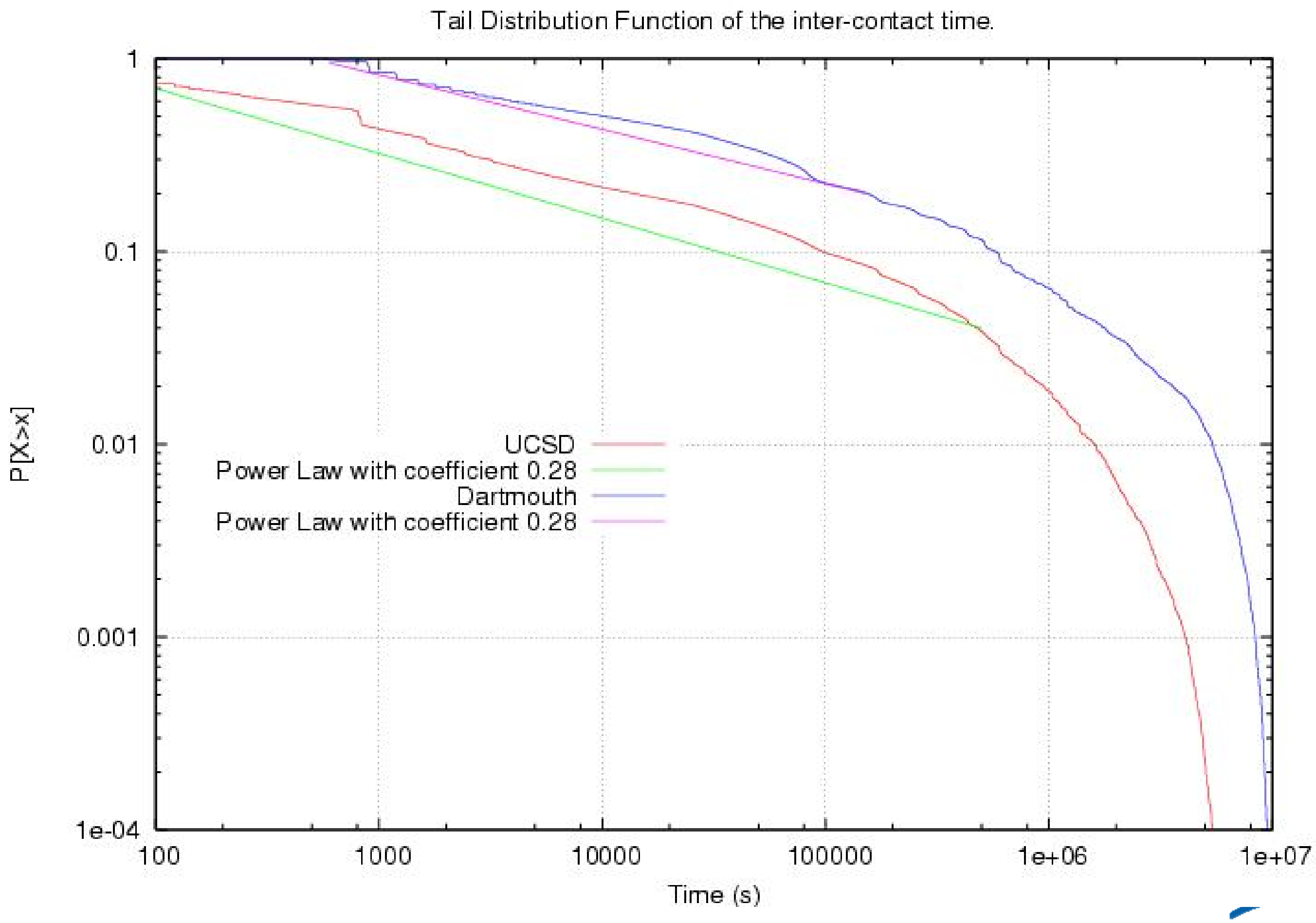
Thank You

Jon.Crowcroft@cl.acm.ac.uk

Inter-contact for Workplace and University Environment

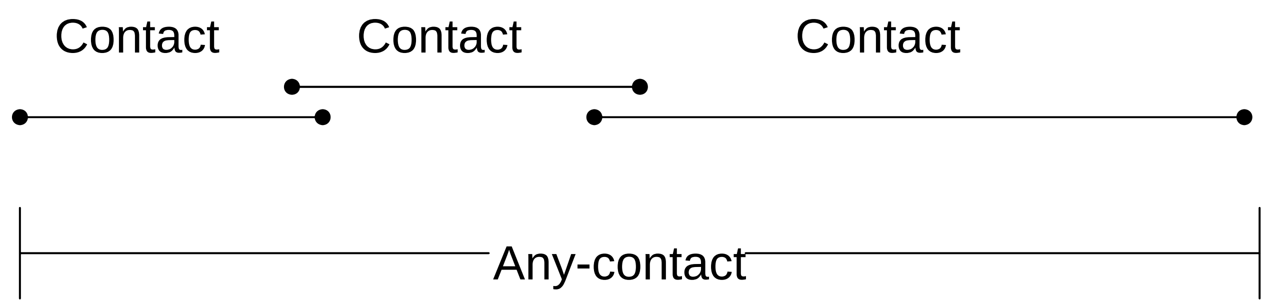


Inter-contact time for WiFi traces

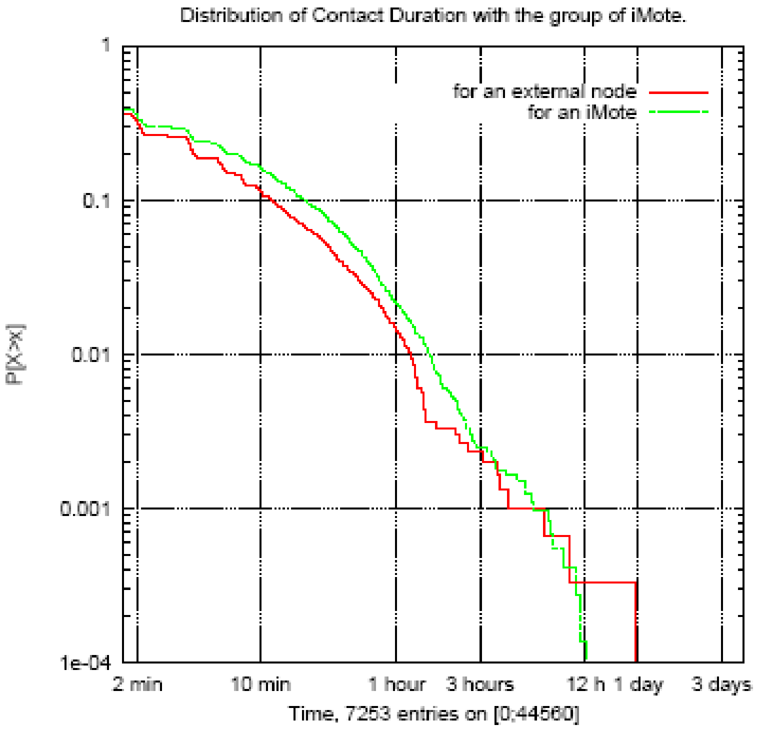


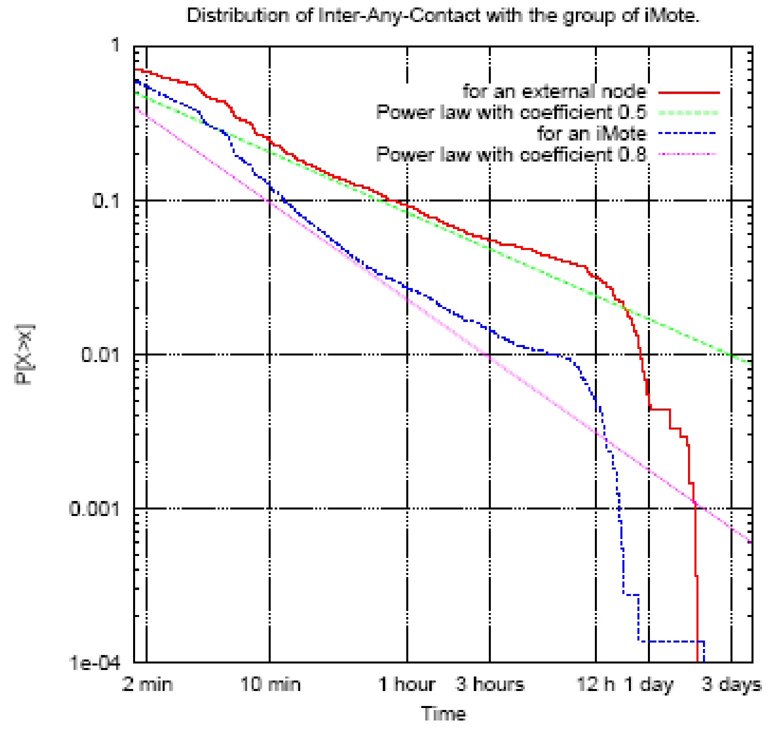
Any-contact and inter-any-contact

- Any-contact : the duration of staying with at least one node
- Inter-any-contact : the duration between two any-contact

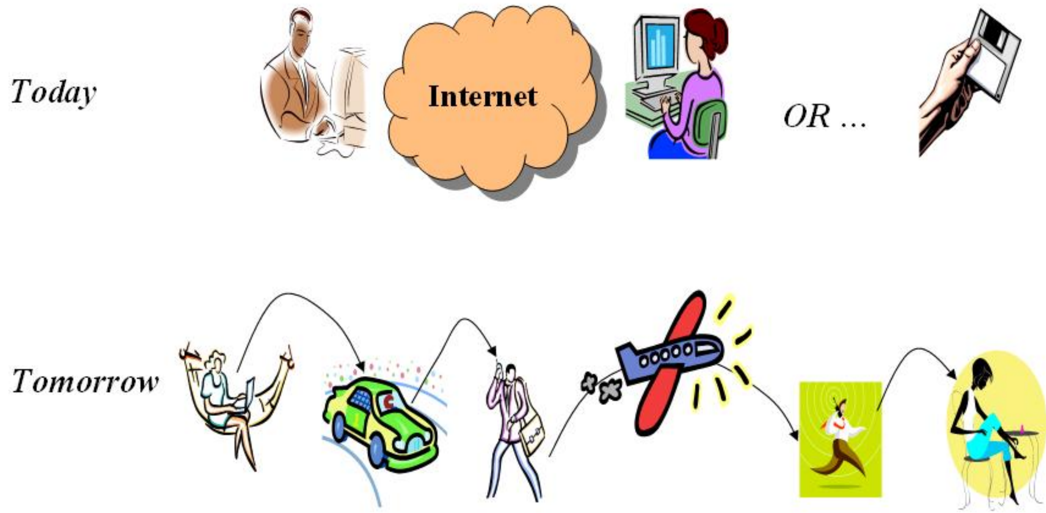


Any-contact and Inter-any-contact distribution





Pocket switched networks



- Make use of global, local network connectivity and user mobility
- Under more general
  - MANET
  - DTN [Fall]