# Personal Namespaces

Mark Allman
*International Computer Science Institute*

ACM SIGCOMM HotNets
November 2007

*"I want a thousand guitars*
*I want pounding drums*
*I want a million different voices*
*Speaking in tongues"*

# Naming Problems

- Naming network resources and services is a *big mess*

# The Solution

- Name everything “peachtree”

- QED

# Naming Problems

- Naming network resources and services is a *big mess*

# Naming Problems (cont.)

- Problem #1: names are obtuse
- Problem #2: names are hard to share

http://www.flickr.com/photo_zoom.gne?id=1131208946&size=o&context=photostream

# Naming Problems (cont.)

- Problem #3: names are gloablly unique, but ambiguous *to people*

- What is asu.edu?
  - Arkansas State University?
  - Alaska State University?
  - Arizona State University?
  - That fabulous little Division I-AA school in Boone, NC?

# Naming Problems (cont.)

- Problem #4: names are intolerant of location change

mallman@cs.ohiou.edu ma137591@ohiou.edu

mallman@lerc.nasa.gov mallman@bbn.com

mallman@grc.nasa.gov

mallman@icir.org

mark.allman@case.edu

# Naming Problems (cont.)

- Problem #5: naming is under nobody's control
  - *Service providers* play a part
    - E.g., "www.blogspot.com"
  - *Content providers* play a part
    - E.g., "MyGreatVacationPictures.html"
  - *Consumers* play a part
    - E.g., "Joe's Blog" in the bookmarks list

# Naming Problems (cont.)

- But, we cope .....
    - Address books
    - Bookmarks
    - “Mail this web page”
    - Clicking URLs
    - Google
    - Social networking sites
- Is there a better way?

# Related Work

- Much naming literature ....
  - How to name hosts
  - How to name services
  - How to name data
  - Personal or group naming realms

- All good stuff but issues remain

# A Naming Layer

- Perhaps what we need is a new over-arching namespace
  - Just an abstraction to existing namespaces

# A Naming Layer (cont.)

- Give users’ a way to name their own resources
  - Independent of resource/service location
  - With context sensitive names
  - Scoping defined by the user
    - Public vs. private

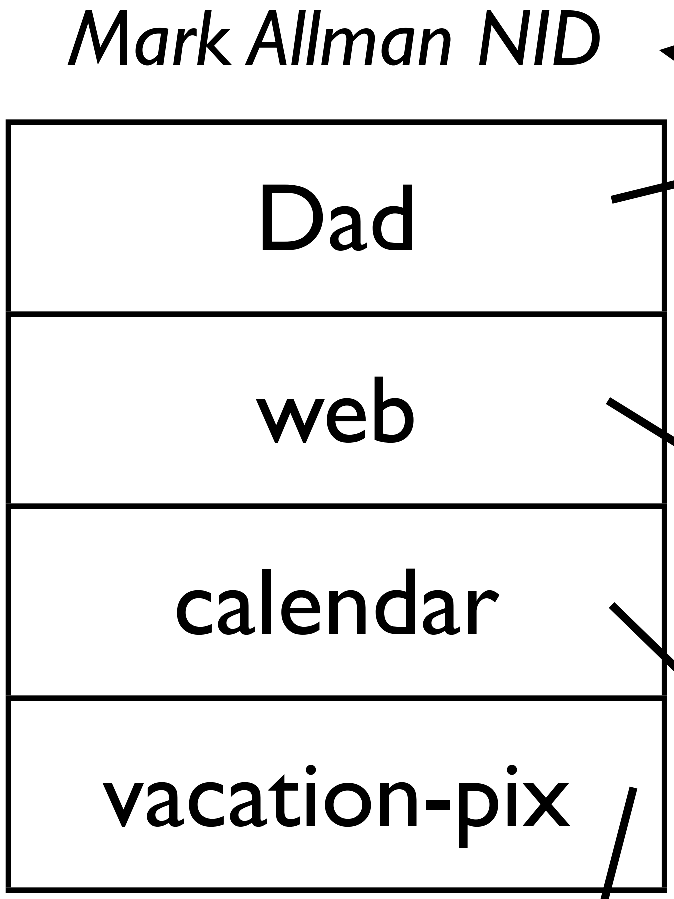
# Overview

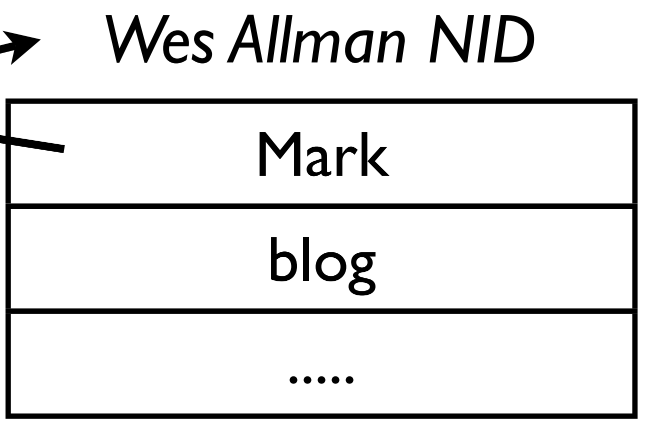
- Every *pnames* user gets a namespace
  - Identified by a *namespace ID* (NID)
    - NID is a hash of the public half of a locally generated keypair

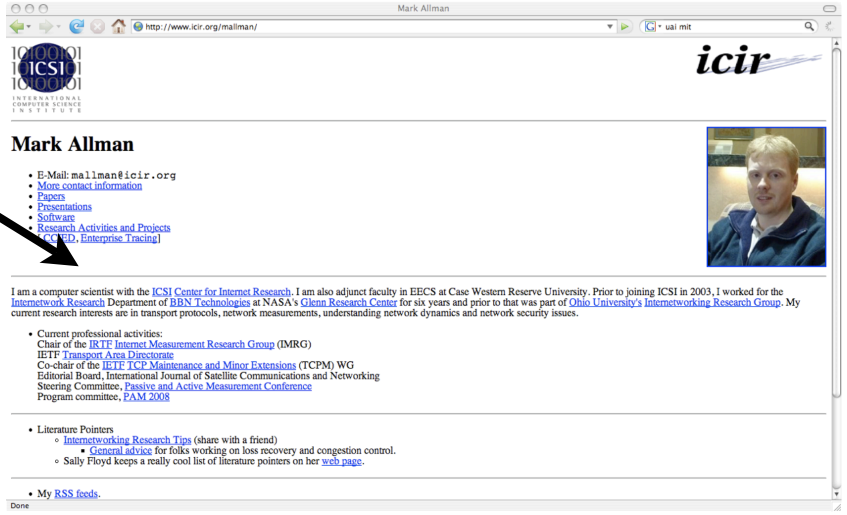
# Overview (cont.)

- Each namespace can contain:
  - Simple names
    - E.g., “calendar = webcal://cal.mallman....”
    - E.g., “email = mallman@icir.org”
    - E.g., “aim = myAIMhandle”
  - Pointers to other namespaces
    - E.g., “Joe = NID:7a6b623df1”

# Example







Mark Allman
http://www.icir.org/mallman/
uai mit
ICSI
INTERNATIONAL COMPUTER SCIENCE INSTITUTE
icir
Done

Mark Allman

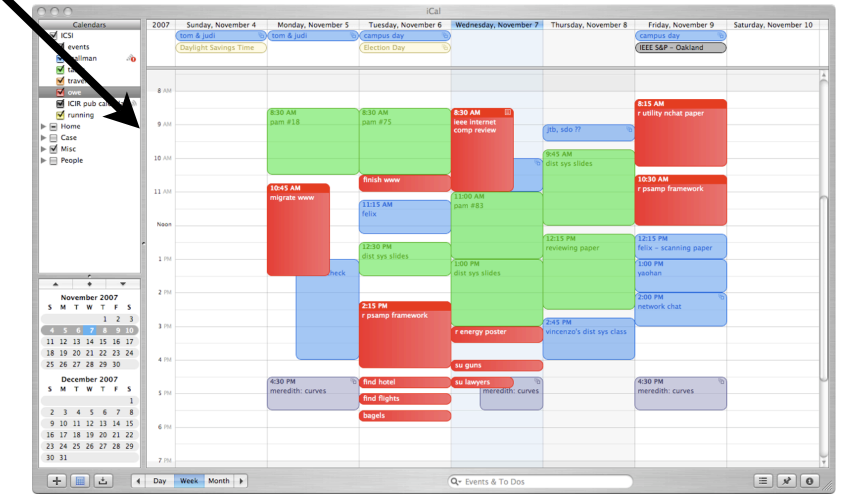
- E-Mail: mallman@icir.org
- More contact information
- Papers
- Presentations
- Software
- Research Activities and Projects
- [ CCIED, Enterprise Tracing]

I am a computer scientist with the ICSI Center for Internet Research. I am also adjunct faculty in EECS at Case Western Reserve University. Prior to joining ICSI in 2003, I worked for the Internetwork Research Department of BBN Technologies at NASA's Glenn Research Center for six years and prior to that was part of Ohio University's Internetworking Research Group. My current research interests are in transport protocols, network measurements, understanding network dynamics and network security issues.

- Current professional activities:
  Chair of the IRTF Internet Measurement Research Group (IMRG)
  IETF Transport Area Directorate
  Co-chair of the IETF TCP Maintenance and Minor Extensions (TCPM) WG
  Editorial Board, International Journal of Satellite Communications and Networking
  Steering Committee, Passive and Active Measurement Conference
  Program committee, PAM 2008

- Literature Pointers
  - Internetworking Research Tips (share with a friend)
    - General advice for folks working on loss recovery and congestion control.
  - Sally Floyd keeps a really cool list of literature pointers on her web page.

- My RSS feeds.



Allman

# Example (cont.)

- Mark can use:
  - Dad:blog
- Wes can use:
  - Mark:vacation-pix
  - Mark:web

# Sharing Names

- Scheme #1: sharing through a DHT
  - Hash key composed of NID, name and type
  - Value is the actual resource name

```
key = MA_NID + ”blog” + ”rss”
put (key,“http://foo.blogspot.com/feed.xml”)
blog_url = get (key)
```

# Sharing Names (cont.)

- This scheme works on a *per-name* basis, not a per-namespace basis
  - I.e., lookups do not retrieve all the names someone makes available

# Sharing Names (cont.)

- Scheme #2: names could be swapped without using a DHT
    - However users now informally share data

# Bootstrapping

- Key problem: NIDs are *more obscure* than any other sort of name we already use
  - Makes them *harder* to share
  - Makes pointer records crucial to the system

# Bootstrapping (cont.)

- Swap NIDs any way information is shared now
  - Email signatures, web pages, vCards, etc.
- Could also setup *pnames* registeries where users could add pointers via some web page
  - E.g., "WellKnownRegistry:MarkAllman"

# Reliability

- Adding a new naming layer adds a new point of failure

- We combat this in two ways:
  - DHTs are robust
  - Heavily use caching and pre-fetching

# Security

- Names can be validated as belonging to the given namespace by checking signatures
- Given crypto usage is required we can encrypt records we want to share only privately
- DHT is robust to some kinds of DoS attacks since it is a *distributed* data structure
    - Of course, different attacks may leverage homogeneous software in a tightly coupled system

# Final Thoughts

- Pnames is a simple new abstraction
  - Do we need it?
  - Is it solving a problem we have?
  - Are our current coping mechansisms Good Enough?

“Is there anybody alive out there?”

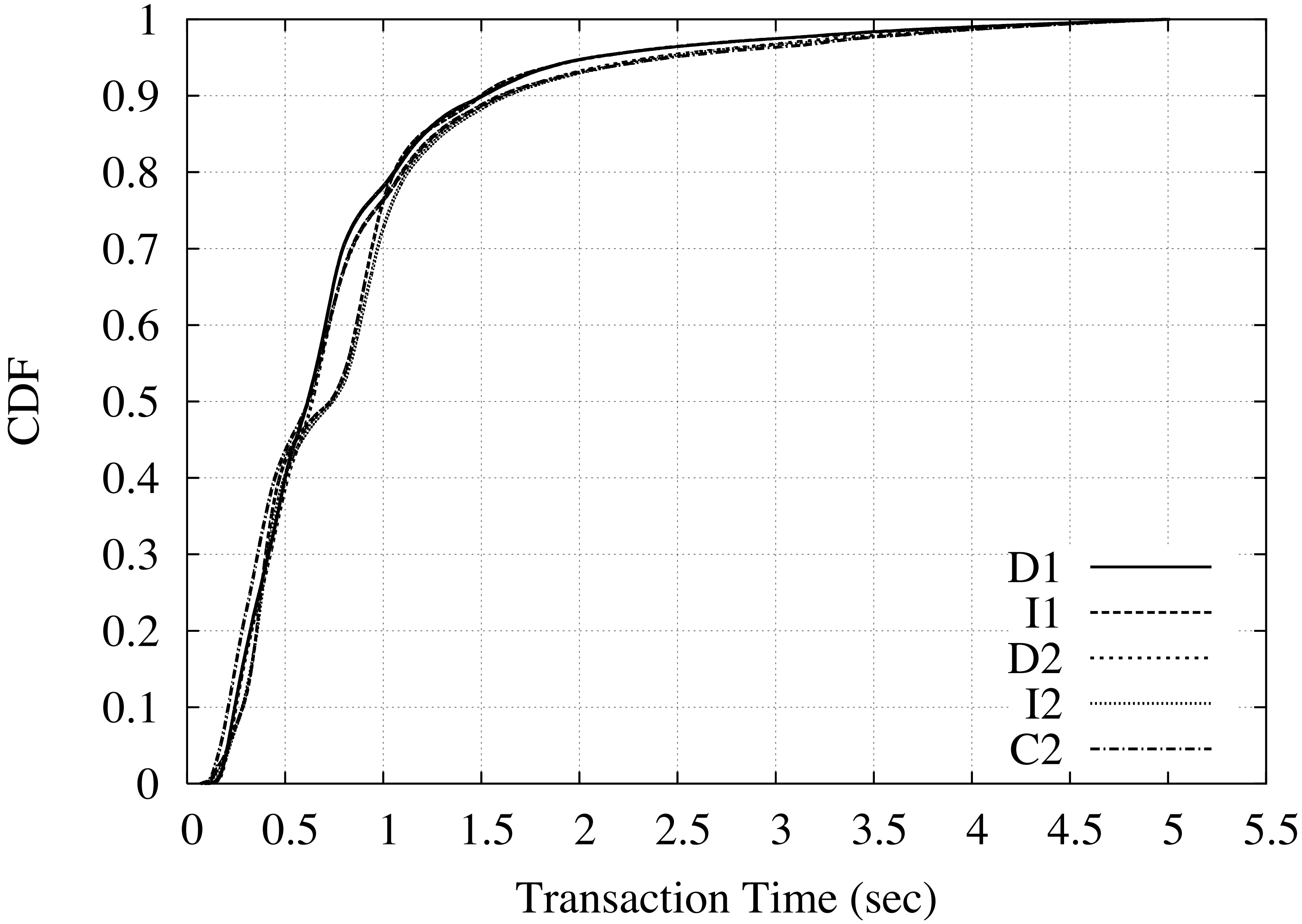
# Extra Slides

# A Different Namespace

- Pnames is not strictly scoped to *people*
  - E.g., organizations could build a namespace

- What are the implications?
- Who is authoritative if “google.com” resolves differently in different environments?

# OpenDHT Performance



Brian Beck, Benjamin Golub, Scott Reid, Mark Allman. *On the Performance of OpenDHT*. October 2007. Under submission.