

User-Relative Names for Globally Connected Personal Devices

Bryan Ford, Jacob Strauss, Chris Lesniewski-Laas,
Sean Rhea, Frans Kaashoek, Robert Morris

Massachusetts Institute of Technology

IPTPS

February 28, 2006

Local Device Connectivity: Easy

Want to trade files while waiting for a flight



Bob's laptop

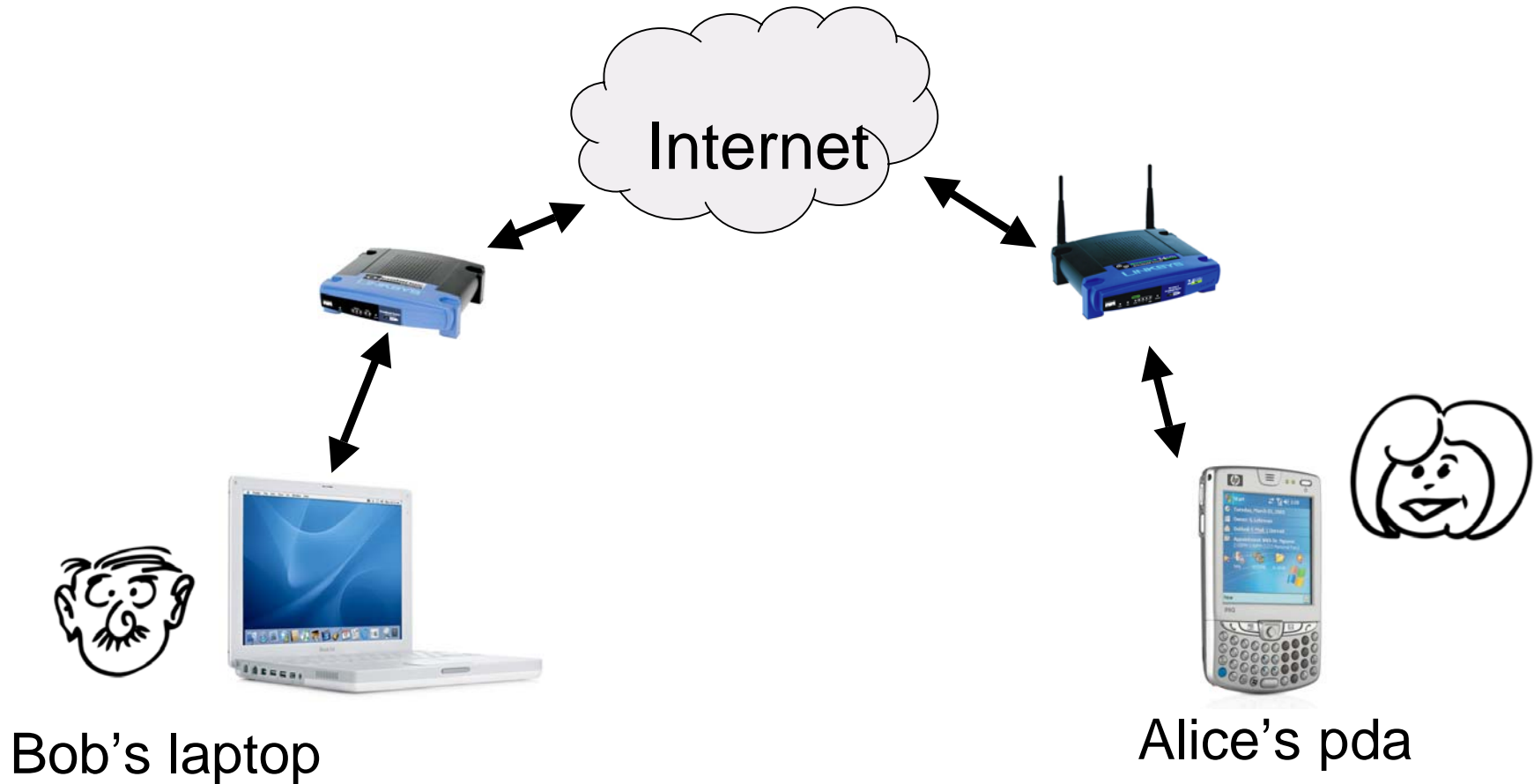
Simple Enough:
Bluetooth, wifi, usb, Bonjour



Alice's pda



Remote Connectivity: Hard



Remote Connectivity: Hard

Today Alice could:

1. Register for a globally unique DNS name
alice576-pda.homeip.net
2. Set up dynamic DNS daemon (or MobileIP)
3. Tell Bob about Alice's pda's DNS name
4. Configure NATs to allow incoming connections
5. Repeat each step for each device
6. Ignore security concerns
7. <http://alice576-pda.homeip.net/presentation/>

Bob

Too many steps

Other approaches

- Local discovery - Bonjour
- Bluetooth
- IPv6
 - Large addresses
- Mobile IP
 - Dynamically change IP address, home agent
- HIP
 - Public keys as addresses
- DNSSEC with updates
 - Tell DNS to update IP address
- SDSI
 - User defined names and delegation

Partial solutions requiring more configuration!

Remote Connectivity: Hard

Goal:

1. <http://pda.alice/presentation>



Bob's laptop



Alice's pda

Solution: User Information Architecture (UIA)

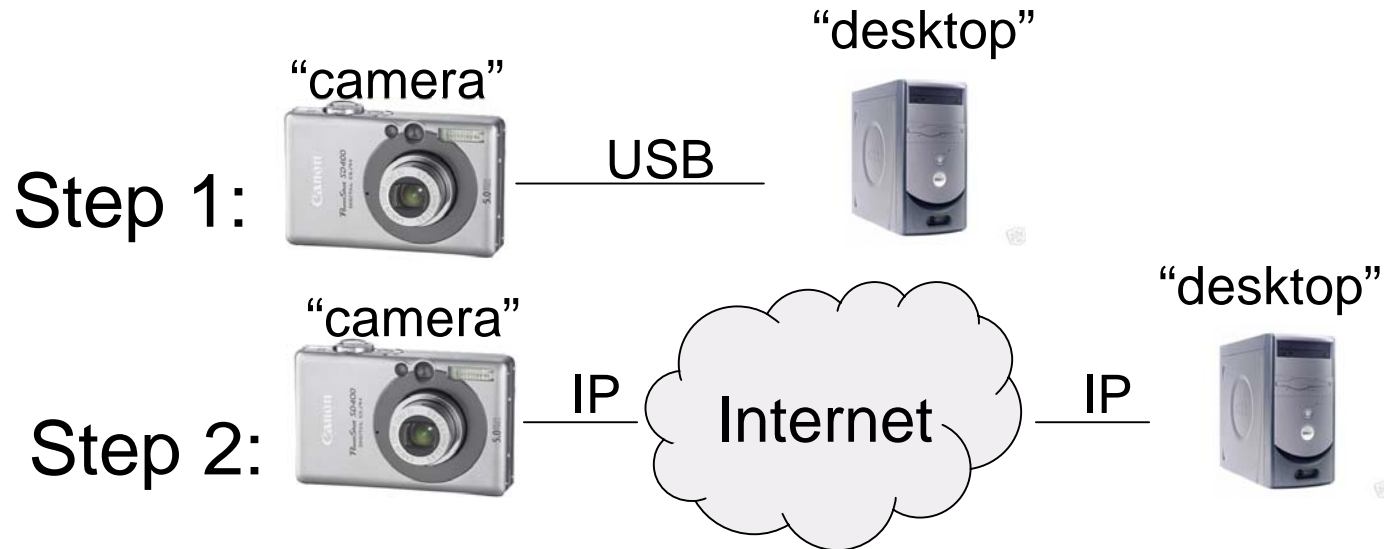
Talk Outline

- Key Concepts of UIA
- Naming System Design
- Prototype
- Conclusions & Future Work

Four Key Concepts in UIA

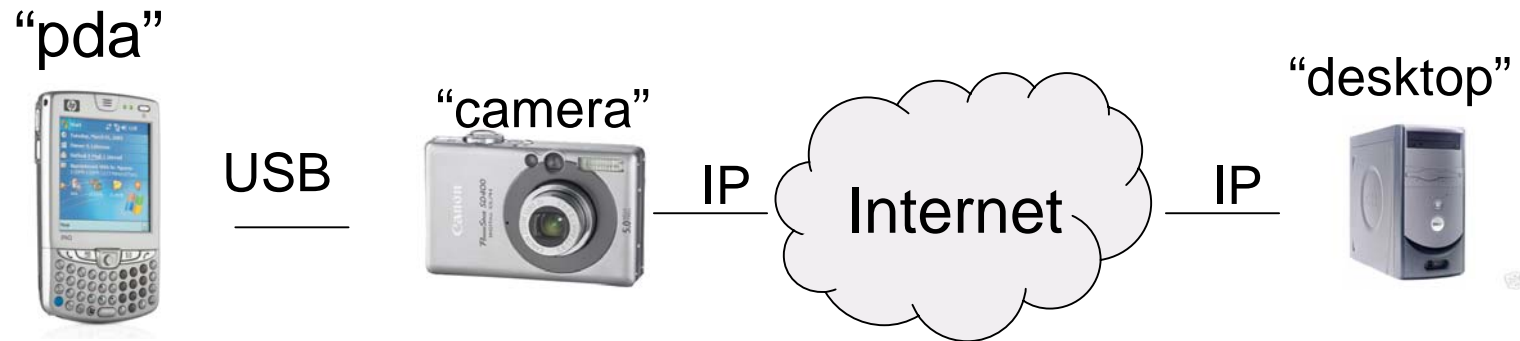
- Local Introduction—Remote Access
- Personal Device Clusters
- Ad-Hoc Naming—Social Networks
- Routing via Opportunistic Rendezvous

Local Introduction—Remote Access



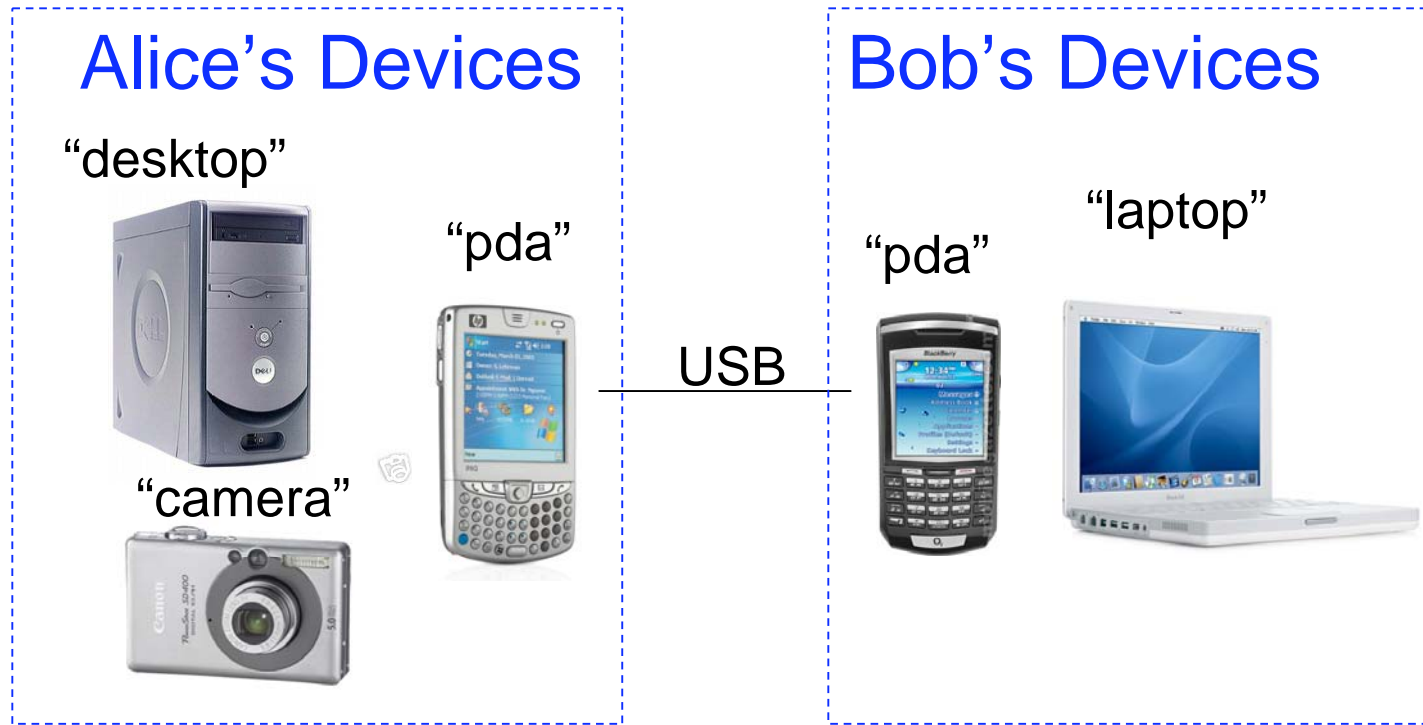
1. Alice introduces devices at home
 - Each device's key is its EID [HIP, SFS]
 - Exchange keys, assign names
2. Alice takes camera on trip
 - Uploads images to desktop at home
 - Have name for peer, can authenticate it

Personal Device Clusters



- Alice gets a new PDA, introduces to camera
 - Exchange keys, assign names
- Devices gossip known names, keys, etc.
 - Desktop learns about PDA from camera
 - And vice-versa
 - Future changes on one device propagate to others
- $O(n)$ not $O(n^2)$ introductions

Ad-Hoc Naming—Social Networks



- Users name their friends' clusters
 - Bob names Alice's devices via suffix ".alice"
 - E.g. "camera.alice", "pda.alice", "desktop.alice"
 - And vice versa: "laptop.bob", "pda.bob"
- Mechanism is still physical introduction

Routing via Opportunistic Rendezvous



- PDAs have moved, both now behind different NATs
 - Each reconnect to previous neighbors (if available)
- Use scoped flooding to find each other
 - Leverages stable hosts as rendezvous points (desktop)
 - Use desktop to forward packets or punch holes in NATs
 - Unlike Mobile IP, no need to configure a home agent

Talk Outline

- Key Concepts of UIA
- Naming System Design
- Prototype
- Conclusions & Future Work

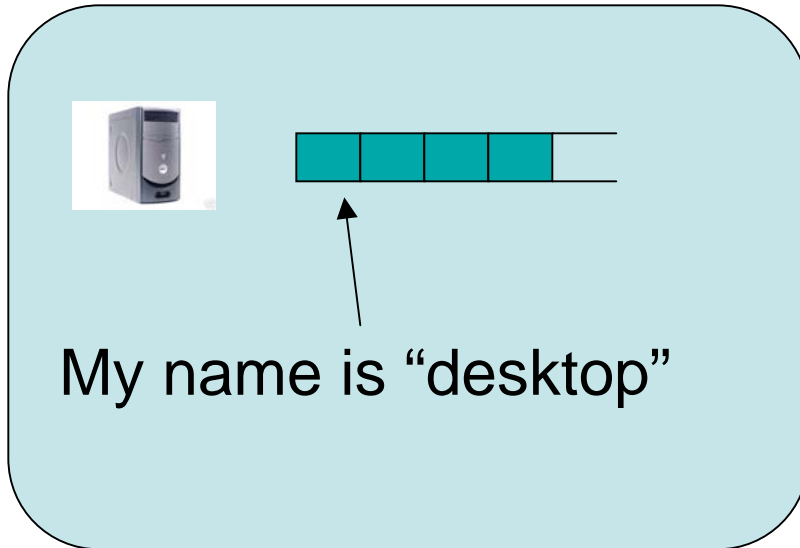
Naming Challenges

- Strong Bindings, User Relative
 - Analogy: Cell phone address book
- Make each change on only one device
 - System automatically propagates to others
- Support Partitioned Operation
 - Resolution doesn't depend on full connectivity

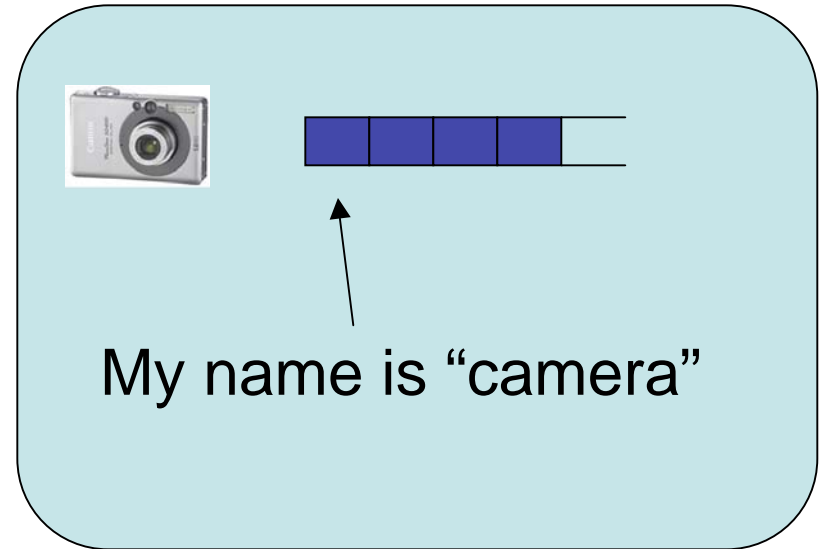
Naming System Outline

- Each device has a public/private key
 - Goal is to map human-readable name to key
- Devices store namespace changes in append-only local log
 - Ex: Alice's PDA has key ID:1234...
 - Ex: merge local namespace with key ID:4321...
- Locally Replicate other devices' logs
 - Resolve Names by consulting own log and local replicas

Example: Alice Introduces Camera and PC



Alice's desktop



Alice's camera

Example: Alice Introduces Camera and PC

This Device: "desktop" ID:4321..
Other Device: "camera" ID:1234...

Merge devices?

yes



Alice's desktop

This Device: "camera" ID:1234...
Other Device: "desktop" ID:4321...

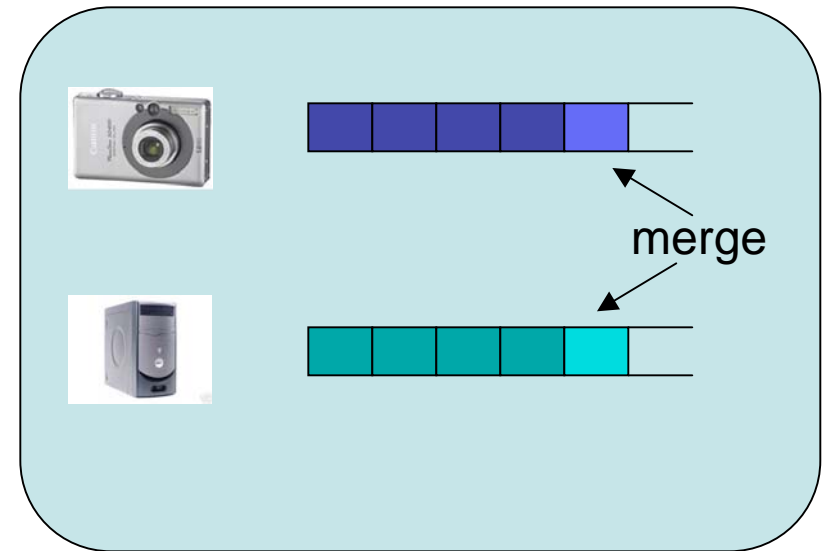
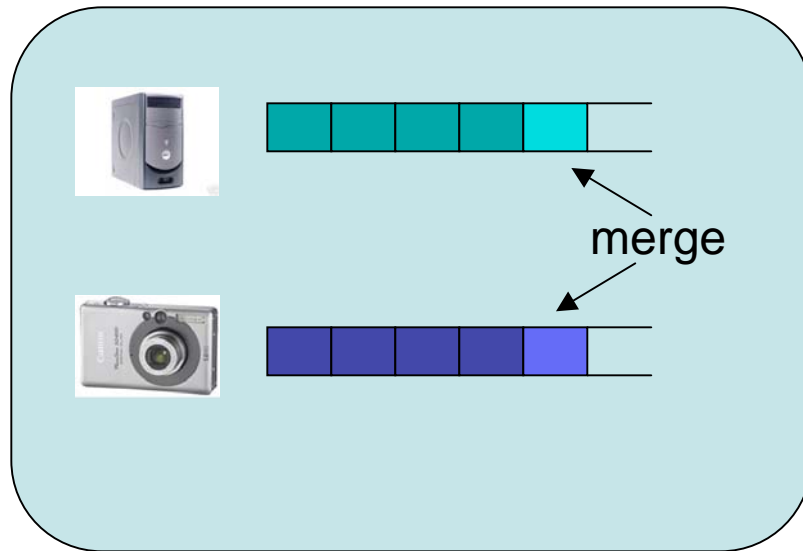
Merge devices?

yes



Alice's camera

After Introduction and Gossip



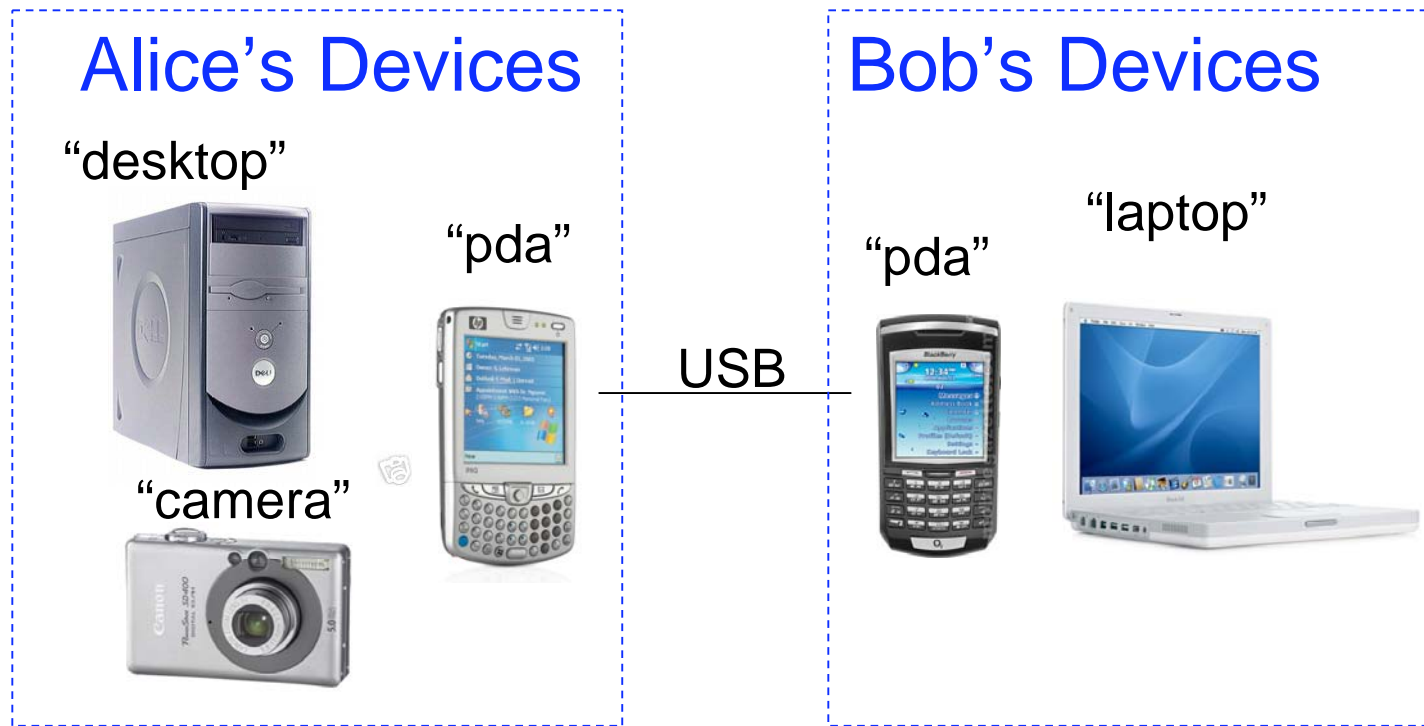
Alice's desktop



Alice's camera

Result: treat each others namespace as equal

Example 2: Social Network Links



Example 2: Social Network Links

Neighbors:
"bob" "pda" ID:5678

Create link to "bob"?

yes



Alice's pda

Local Neighbors:
"alice" "pda" ID:4321

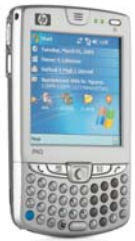
Create link to "alice"?

yes

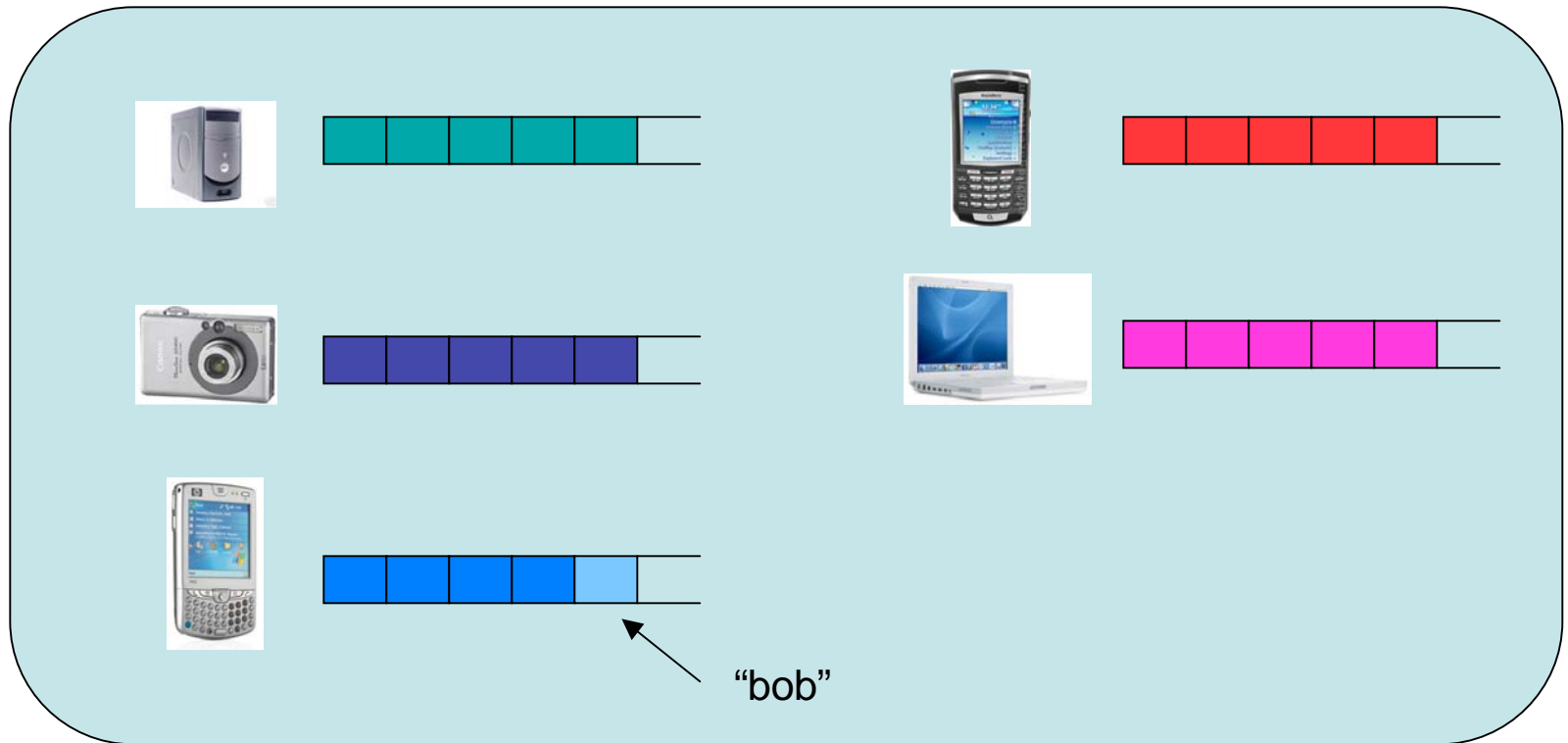


Bob's pda

Replicate Neighbor Records



Alice's
pda



Result: use names with suffix ".bob"

Talk Outline

- Key Concepts of UIA
- Naming System Design
- **Prototype**
- **Conclusions & Future Work**

UIA prototype

- Runs on Linux and OS X
- Routing
 - EID is hash of public key
 - Send(message, EID)
 - Message = Recv()
- Naming
 - Merge(other device)
 - Link(friend's device, my name for friend)
 - EID = resolve("name")

UIA prototype

- Supports legacy interfaces
 - Embed EIDs into IPv6 addresses
 - Intercept packets via tun device
 - UIA nicknames as DNS entries
 - Intercept and resolve via local proxy
 - Works with unmodified apache, ssh, firefox

Future Directions

- Scalability vs Robustness tradeoffs
 - How far to gossip?
 - How many devices to find a stable rendezvous?
- Authorization and Access Control
 - Can *.Bob use my printer?
 - Deny packets from devices I cannot name?
- Privacy
 - Separate family names from work names

Summary

- UIA simplifies personal device connectivity
 - Local introduction is root of all trust
 - Names are user-relative
 - Updates entered anywhere -- no master device
 - No NAT wrangling
 - No explicit key management