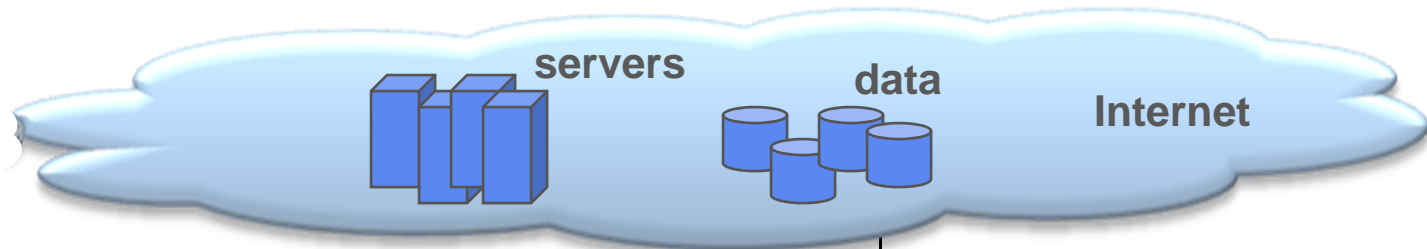# PRPL: A Virtual Data System

Monica Lam

Stanford University

# Vision



servers

data

Internet

PC,TV
at home, on the road,
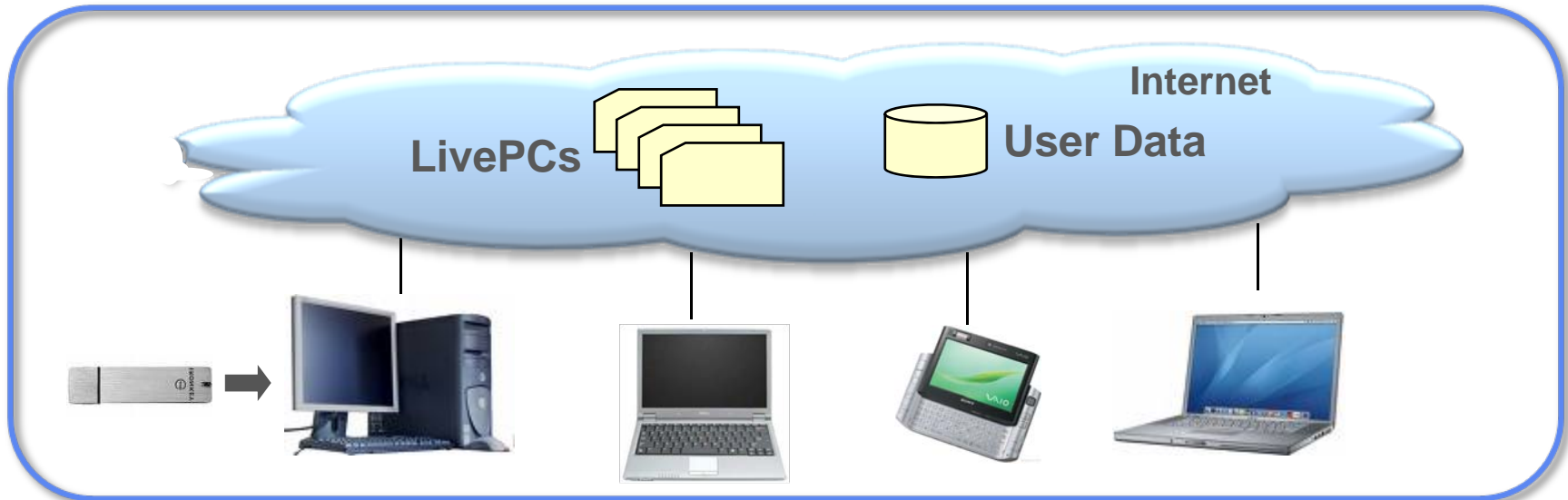in hotels, on the plane

Personalize the generic PC,
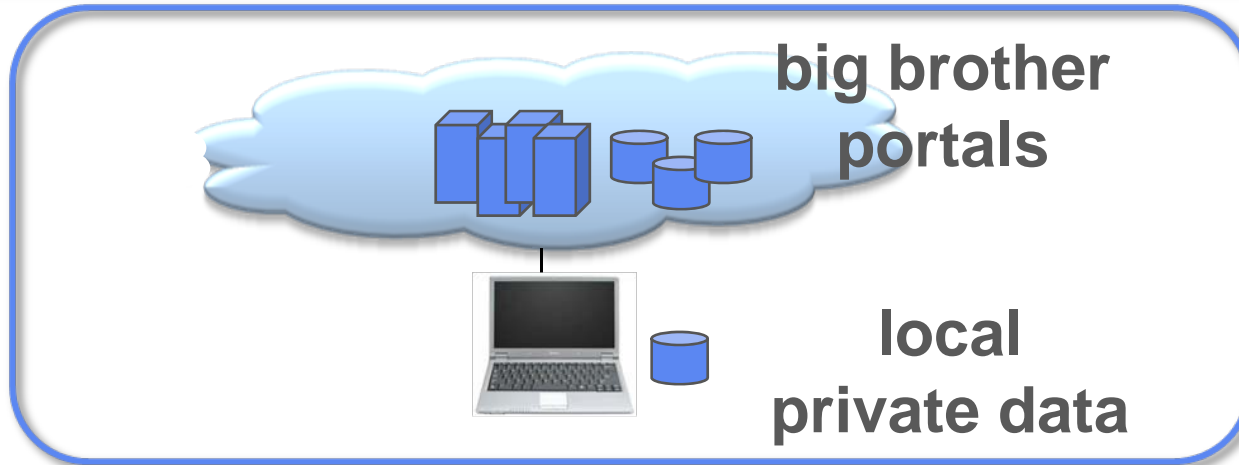borrow the power, display, keyboard, memory, etc

My key, cache, window into my
digital ID, digital personality, digital assets,
and the internet

# Background: First Approximation



- Collective project, Moka5
- LivePCs: managed virtual machines in the cloud

# Cloud Computing Today



**big brother portals**

**local private data**

- Big-brother portals: gmail, yahoo mail, flickr
  - Easy to use, free, quick software updates
  - They own our data – monetize via advertisement
  - Centralized facility to service many small groups
    - Requires a lot of resources to scale the services
    - Low resolution photos
- But hardware is cheap!

# Two Tiers → Three Tiers
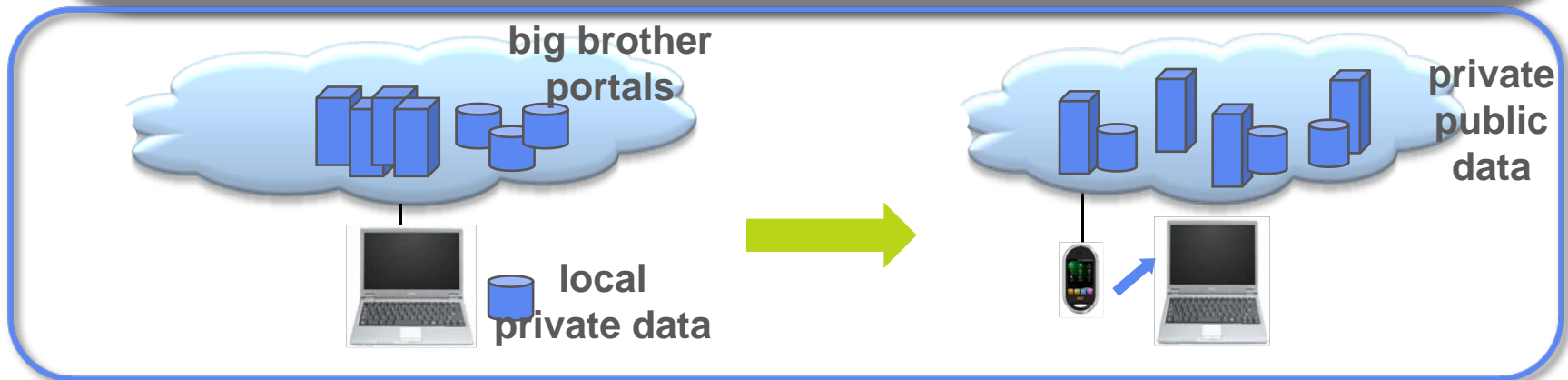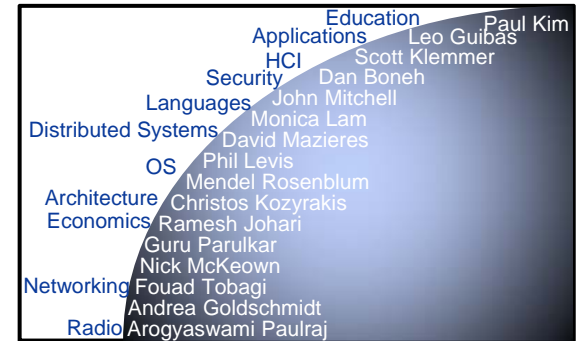


- Data infrastructure
  - Separate ownership from storage vendors & application providers
  - (Caching) on home servers, with data backup provided by the ISPs

- Computation infrastructure
  - Provide "backup" computation for cell phones at a nearby server
  - Service the long-tail of distributed applications

- Fluidity across devices
  - Data and computation caching in the hierarchy

## Infrastructure for Three-Tiered Applications

- Openness → protection against mobile code
- Protection against design / coding errors
  - Data leakage, fault containment
- Good user experience on a handheld
- Energy efficiency
- Distributed, location-aware applications

# Research Strategy

- Collaboration:
  - Whole system design
    - Computation server cache in network
  - Unifying principles cross domains
    - Information tracking
  - Inspirations across domains
    - New data system → new user interfaces

- Combines clean-slate and pragmatic thinking
  - Inter-operability with existing infrastructure

- Hard-core science + usability
  - Distributed file systems (read-only sharing)
  - Encryption that supports sharing

- End-to-end development from application to networking



Education
Paul Kim
Applications
Leo Guibas
HCI
Scott Klemmer
Security
Dan Boneh
Languages
John Mitchell
Distributed Systems
Monica Lam
David Mazieres
OS
Phil Levis
Mendel Rosenblum
Architecture
Christos Kozyrakis
Economics
Ramesh Johari
Guru Parulkar
Nick McKeown
Networking
Fouad Tobagi
Andrea Goldschmidt
Radio
Arogyaswami Paulraj

# Potential Contributions

- Disruptive system architecture concepts
  - Healthy competition with long-tail services
  - Fundamental computer science technology for the masses
- Standards: web browsers, Android II.
- Tools and frameworks for existing systems
- Cross-domain innovations
  - human resources and courses
- Innovative applications

# Examples of Success Scenarios

- Secure communication with banks and health-care providers
- Personal virtual living rooms a la Second Life for families/groups
  - Create a "social network" of their choice
  - Share large amounts of data and code safely.
- Perfect recall from our phone
  - "All our experiences can be saved", Reddy 1997
  - Locate a photo from 15 years ago

# PRPL: Virtual Data System

# Do you know where your data are?

NFS, AFS, coda
CVS, SVN

Samba



Email
- Cached on different devices
- Online and offline access
- Fine-grain sharing
- Information hidden in attachments
- Why application-specific synchronization?

Distributed file systems
- Single-image view
- Too hard to deploy & use

# Do you know where your data are?

NFS, AFS, coda
CVS, SVN

Samba



**Social networks:**
- Large-scale sharing
- Data are distributed by apps
- Control over our data
- Control over our social relations
- No offline access
- The rich gets richer scheme

**Browser & mail clients**
- Mega web-access programs
- Must secure against data leakage

**Common problem**
- Data navigation on the cell phone

# PRPL: PRivate-PubLic data index



NFS, AFS,
CVS, SVN

Samba

PRPL

A unified view of data
Separate data ownership, storage, applications
Secure, fine-grain sharing
Device-independence: caching
Interactive data navigation with semantic-web queries

# Key Technologies

- PRPL data appliance
  - Individual mini file server, http server, imap server, web crawler, semantic-web indexer that keeps all shared data current
  - Encryption-based sharing
  - Target of attacks: prevention of coding errors that leak data
  - Dynamically generated user interfaces
    - Based on semantic information
    - Tailored to individual users and display technology
- APIs to support inter-operability and extensibility
- Identity management
- Distributed software management
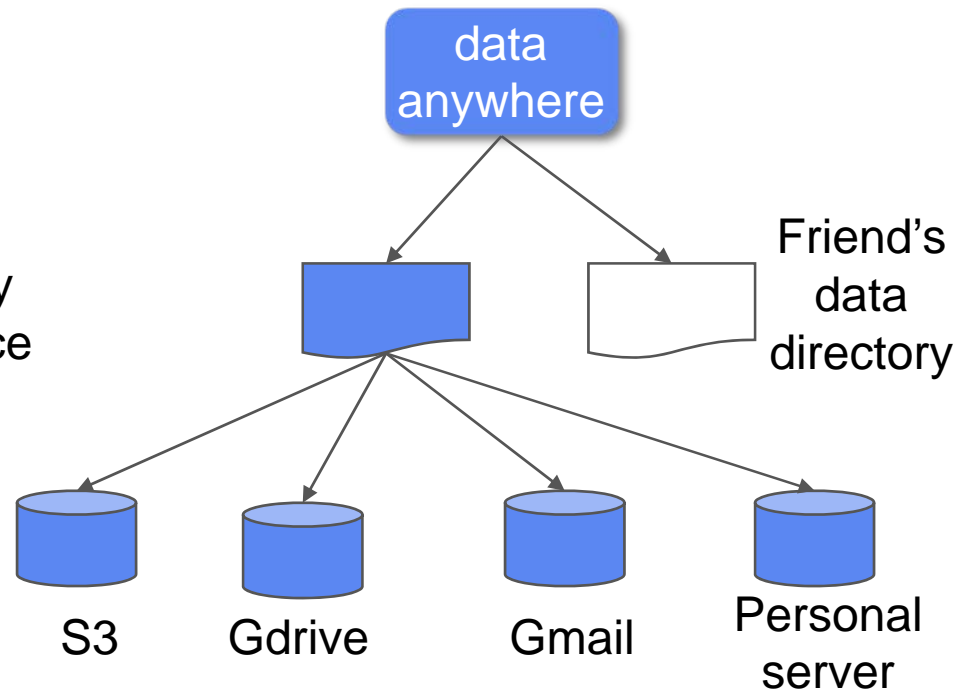- Computation resource management in the cloud

# Early Prototype Design

Personal data view
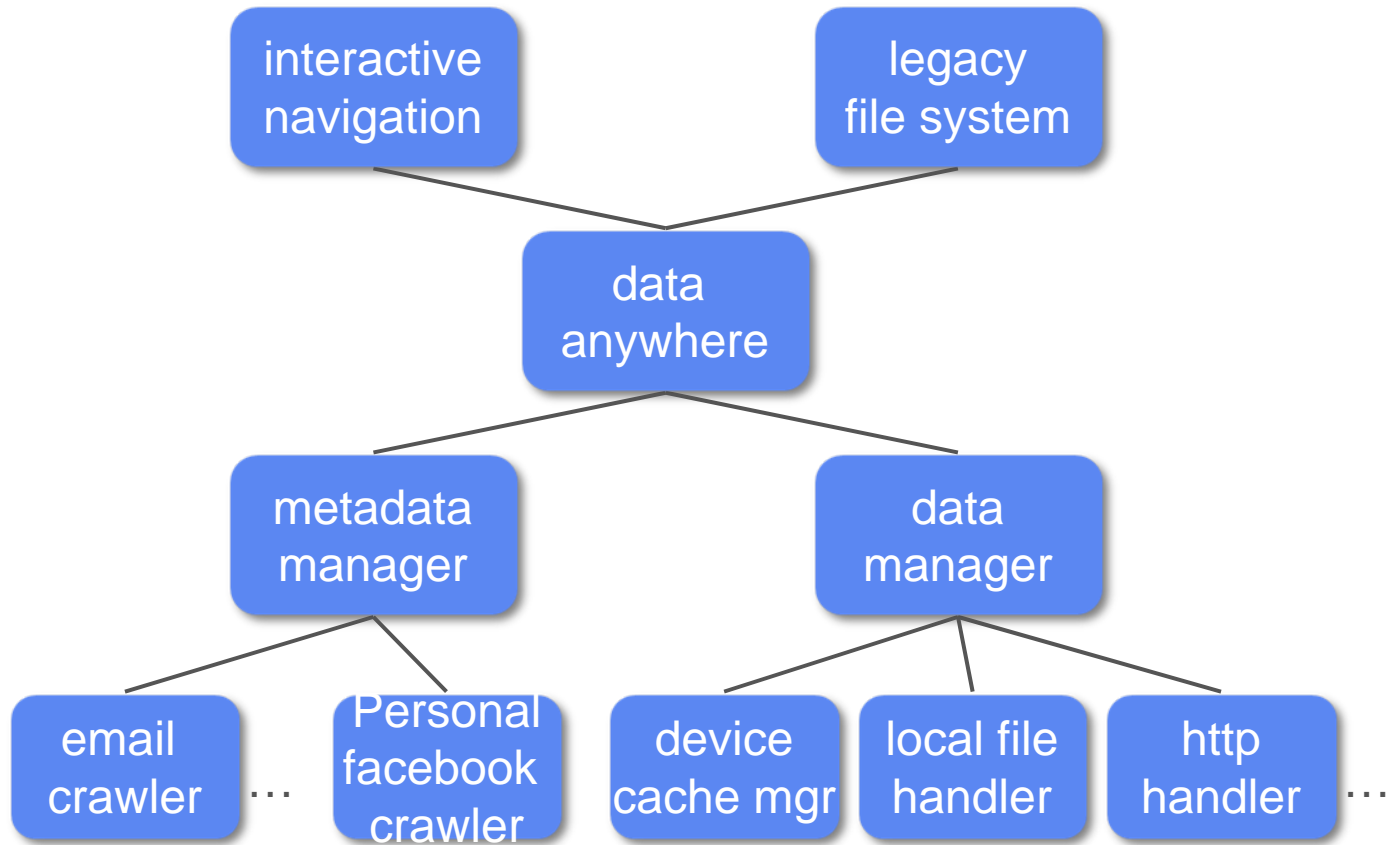-Cached
-Source independence
-Semantically indexed

Personal data directory
- Storage independence

Physical storage

data anywhere

Friend's data directory

S3  Gdrive  Gmail  Personal server

# System Components

# Information Tracking

- Goal: Data from one principal not leaked to other principals
  - Email: Prevent mail to be forwarded to the wrong person
  - Browser: Information from a bank on a browser cannot be stolen
  - PRPL: financial, health info shared only with trusted principals
- Technique
  - Reduce the trusted base to
    - operations on sensitive resources & privilege-escalation routines
- Combining approaches at different levels
  - Language provides guarantees inside processes (Lam)
  - OS handles persistent data & sandboxes unknown processes (Mazieres)
  - Network provides trusted messages (Mazieres)
  - Hardware tracks low-level data efficiently (Kozyrakis)

# User Interface

- Semantic-web based navigation
- Automatic generation of UI based on meta information
- Interactive generation of display
    - based on the display size, individual capabilities and preferences
- Multi-modal inputs
    - Speech is more competitive on cell phones.

# Conclusions

- Clean-slate research opportunity

- Three-tiered architecture
  - Interesting, challenging research topics
  - Innovations through collaboration