Protium, an Infrastructure for Partitioned Applications

Cliff Young

Joint work with Lakshman Y.N., Tom Szymanski, John Reppy, David Presotto, Rob Pike, Girija Narlikar, Sape Mullender, and Eric Grosse

Bell Laboratories, Lucent Technologies

Copyright © 2001 Bell Laboratories

Protium

- Vision: Use any Internet-connected device in the world like one's office workstation.

  Is it easy to use your office phone?
  Is it easy to use your home phone?
  Is it easy to use your neighbor's phone?
  Is it easy to use a hotel phone?
  Is it easy to use a public phone?

  Now replace 'phone' with 'computer'.

- Services should reside in the network, not the computer.
- But networked applications are hard to write.

Assumptions

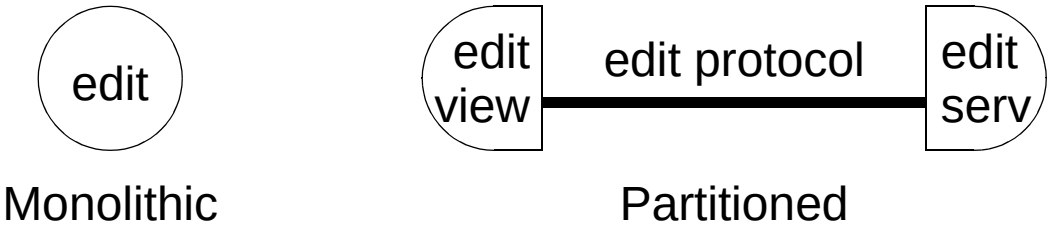
- Devices: Moore's law continues
  - Performance, power, cost, size improve exponentially
  - Number of devices per person will increase

- Communications:
  - Bandwidth gets better
  - Wireless coverage gets better
  - Latency does *not* get better

- Protium approach:
  - Partition applications into *service* and *viewer* pieces.
  - Build infrastructure that makes this easier.
  - Engineer for latency.

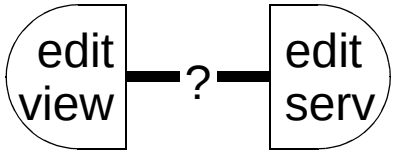
Partitioned Applications

- What happens when we split an application?



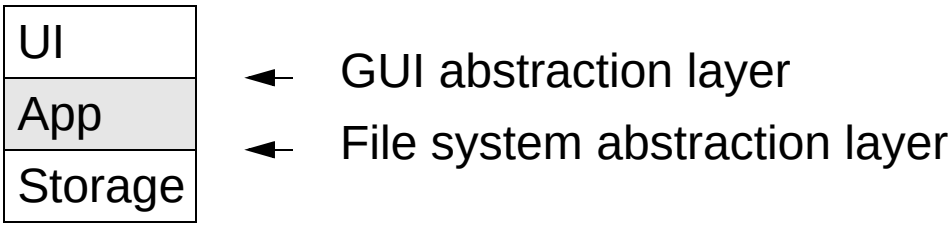
  - Viewers: soft state, disposable
  - Services: hard state, reliable, backed up, high availability

- The network split gives us a new place to put infrastructure:

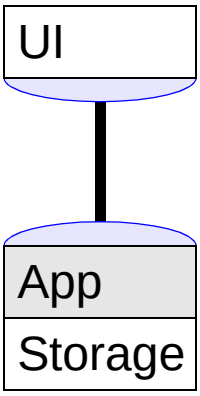


- Or a new abstraction/design layer

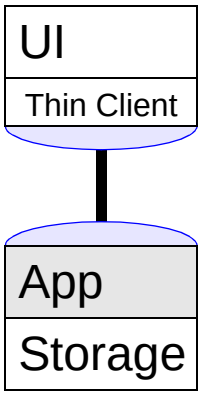
Partitioned Applications (2)



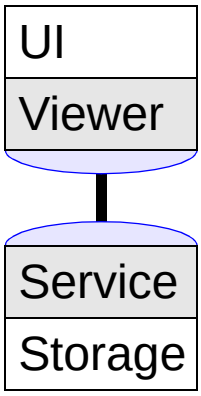
Local case



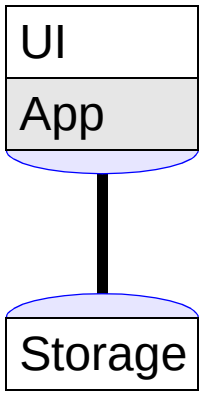
Remote FB



Thin Client



Protium

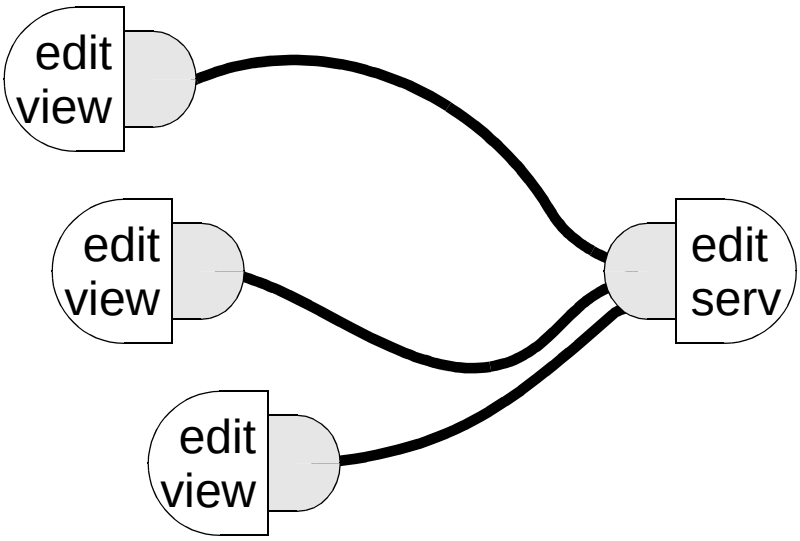


Distributed FS

- Previous: Network at existing abstraction boundary
- Protium: Network inside the application

Glens and Glendas

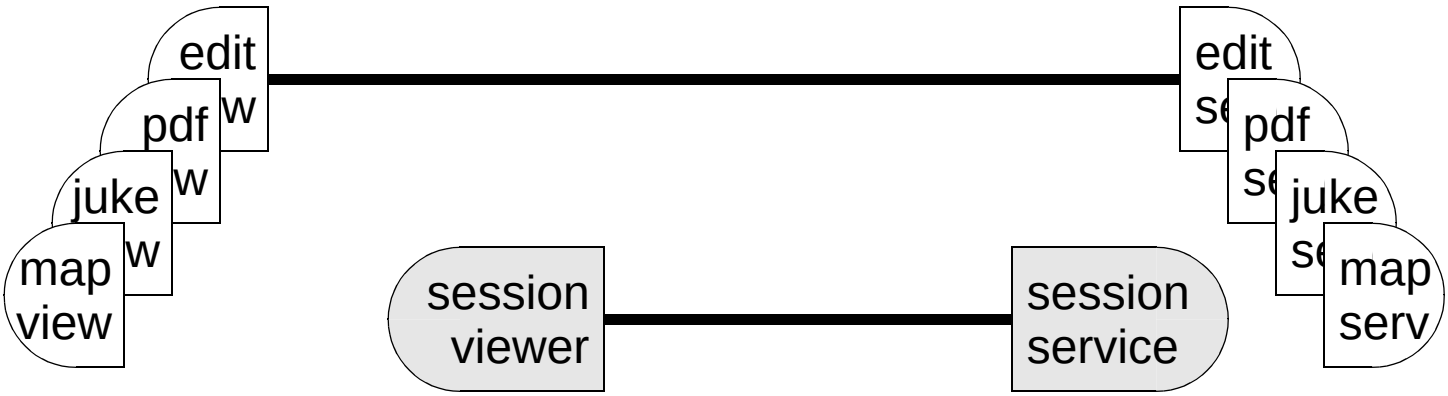
- Multiplex multiple viewers using a single service



- Use a *dynamic primary* consistency strategy for responsiveness
  - 3 message types: reads, lightweight writes, heavyweight writes
  - Infrastructure snoops application messages
- Simplifies service construction

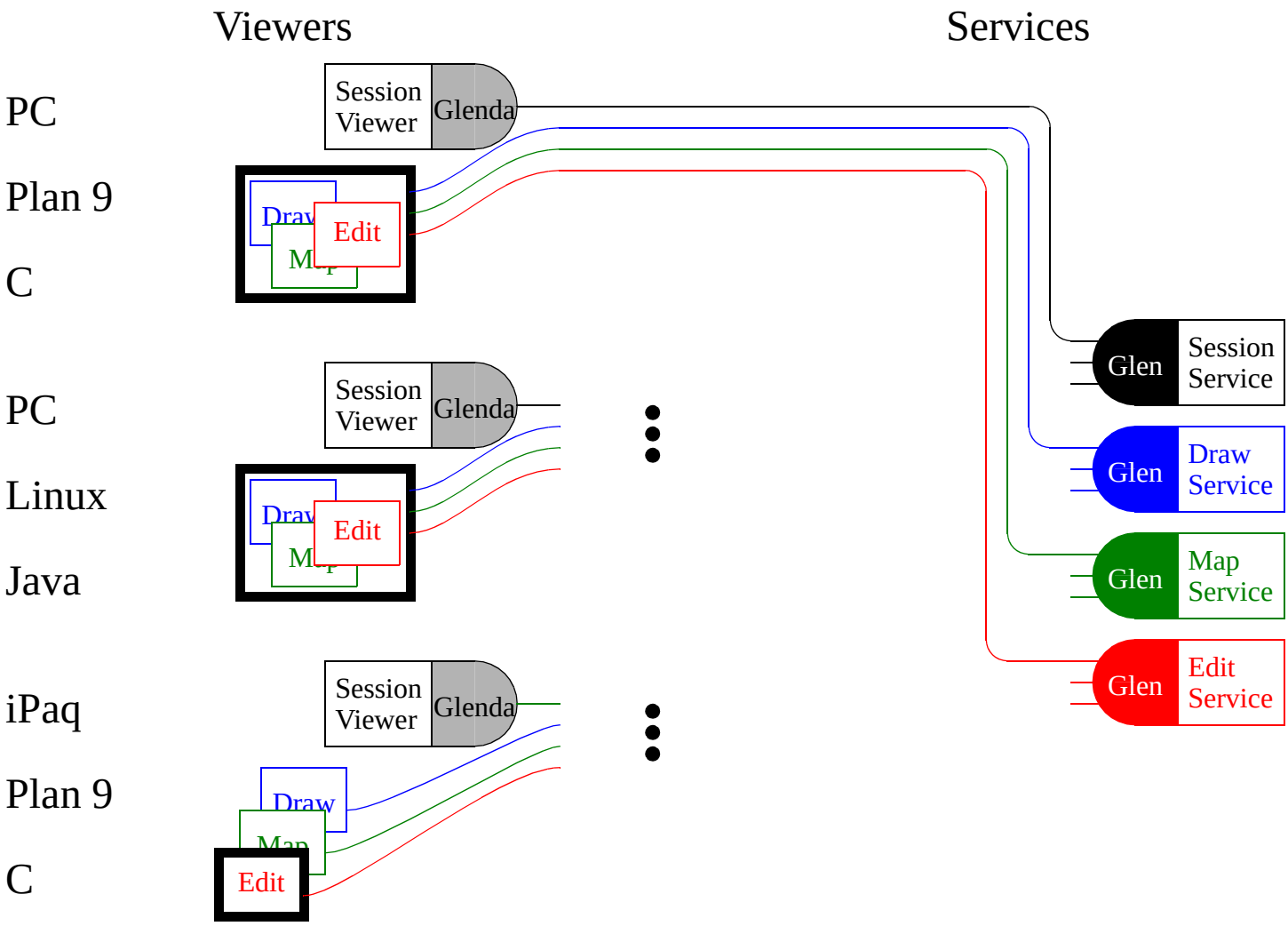
Session Management

- Roughly, a “session” is a desktop.
- What keeps track of my desktop?



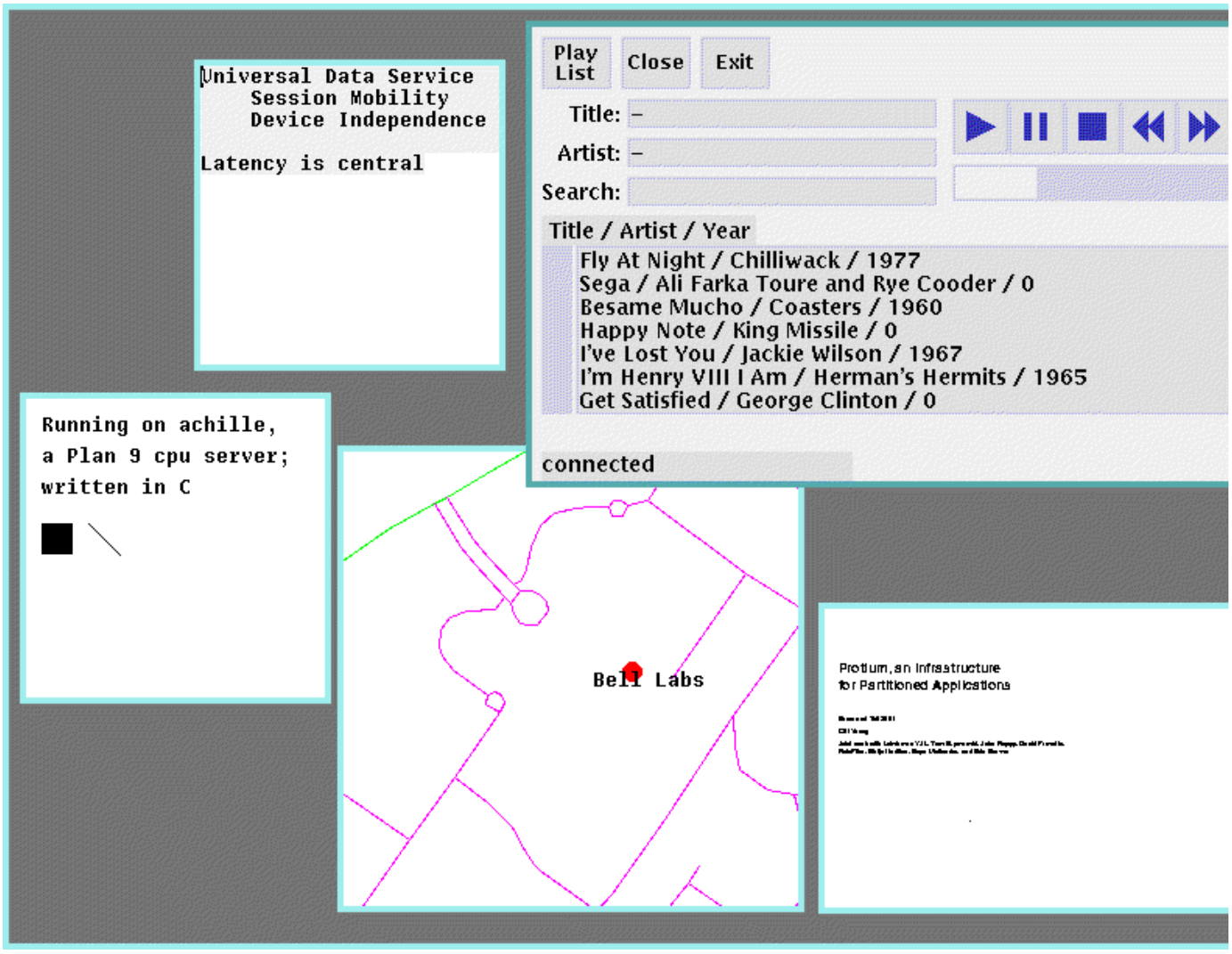
- Also tracks window operations, launches new apps
- Adaptation to different devices
- Window manager looks up, initializes the other viewers

Demo Structure

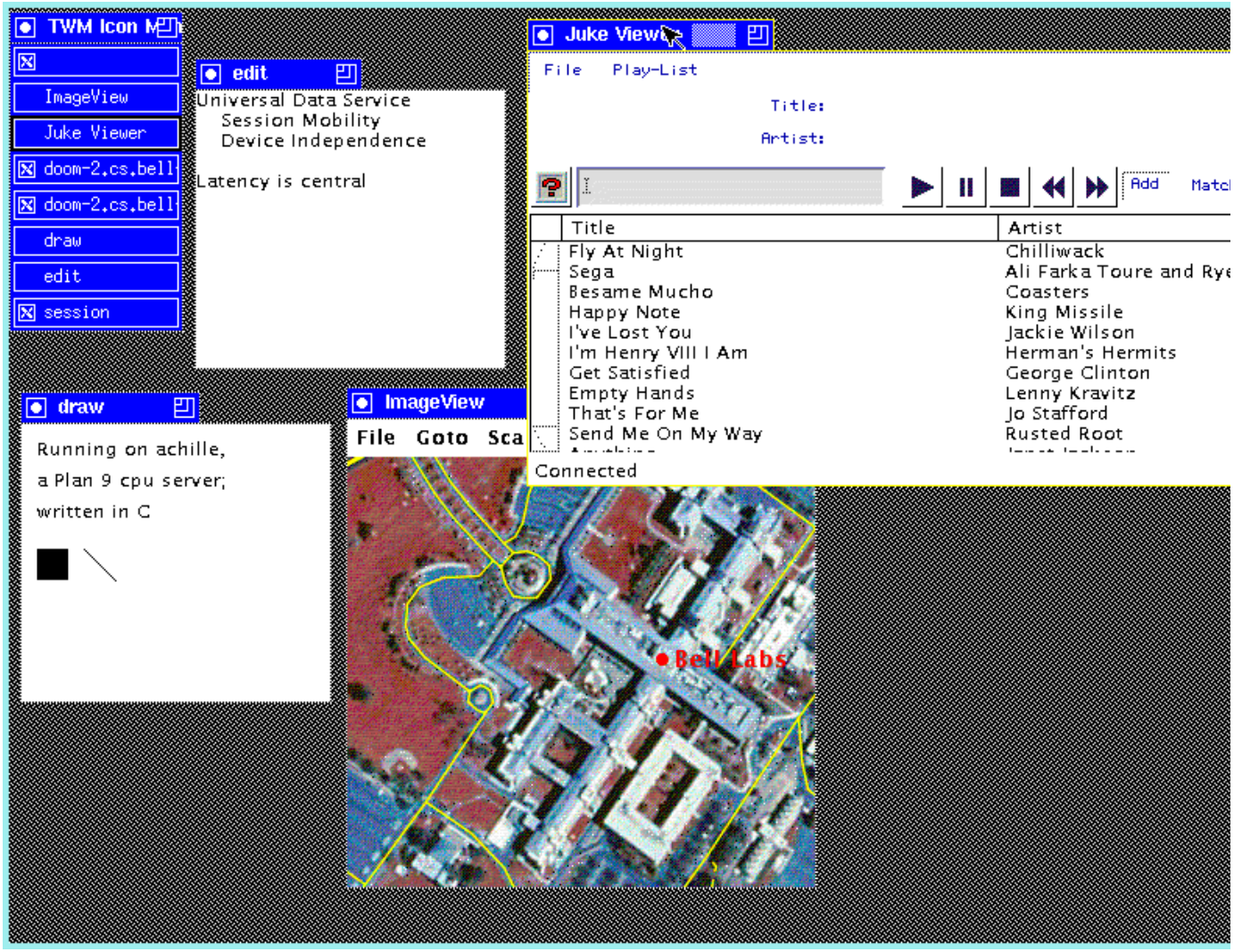


(all connections are TCP/IP)

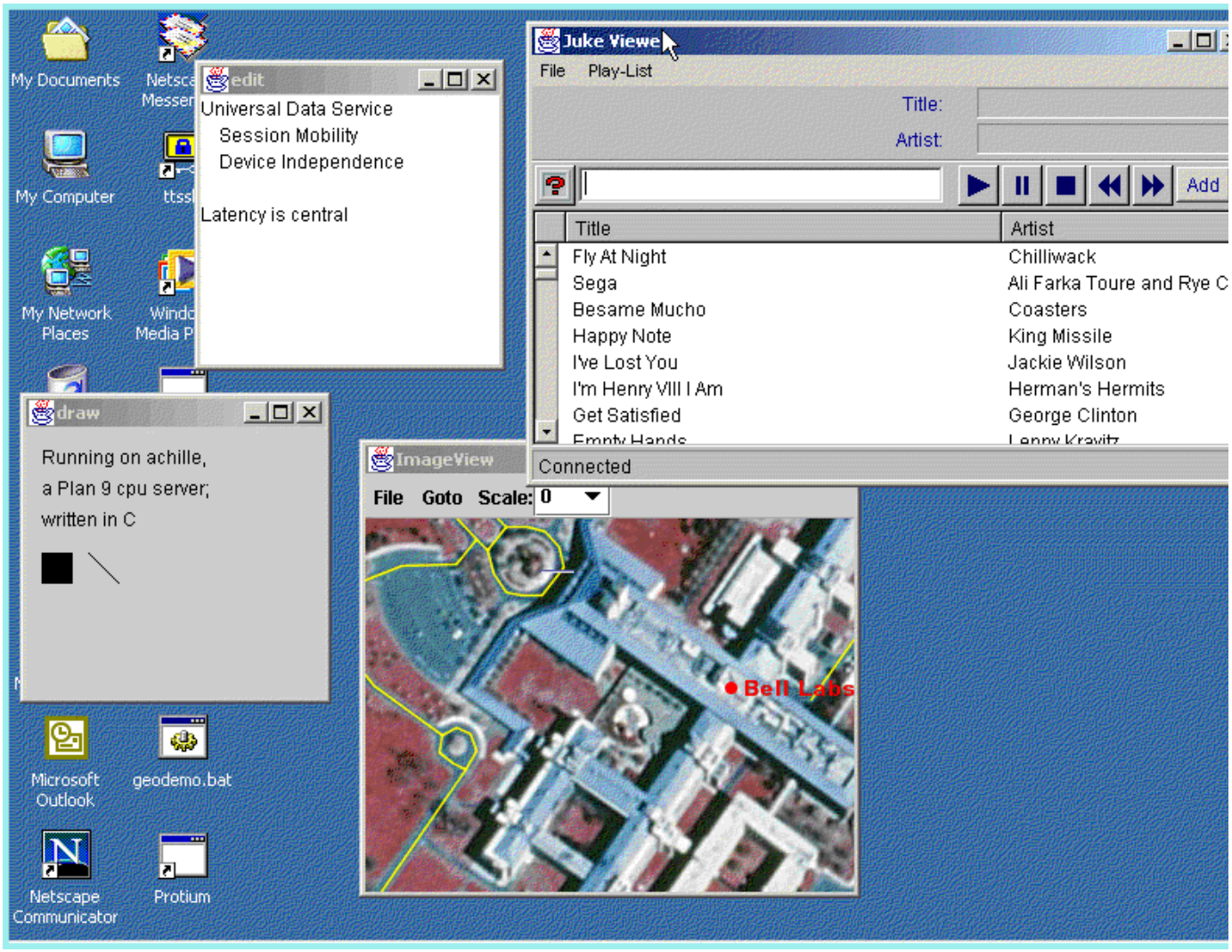
Demo (Plan 9 Screen)



Demo2 (Linux Screen)



Demo3 (Windows Screen)



What Next?

- More and better infrastructure
  - Security: authenticate and encrypt
  - Logging, undo/redo, transactions
  - Caching, prefetching, disconnection(?)
  - Naming
  - Managing heterogeneous server machines
  - Realtime
- Other viewer platforms
  - Text: fallback; allows scripting
  - Voice? HTML?
- More applications: PDA, conferencing, games

Pieces and People

Infrastructure	**Cliff Young**
Jukebox	**Girija Narlikar, Rob Pike**
PDF	**John Reppy**
Map	**Tom Szymanski, Lakshman Y.N.**
Edit and Draw	**Cliff Young**
iPaq Port	**Sape Mullender, David Presotto**

Anticipated Questions

- Please compare to
  - VNC, Citrix ICA, Sun Sunrays, Microsoft NetMeeting
  - The X Window System, Sun NeWS
  - NFS, AFS, Coda, Locus/Ficus
  - Groove Networks, bmypc, seven
  - Berkeley's Ninja project
  - .NET, Java
- Do I have to rewrite every application?
- Writing viewers seems hard.
- Why bother? The network is getting better.