Talk notes, to go with slides, version of 1/14/88

[Slide 1.] Owl, title, my name

background: I'm in my fourth year of working with Athena.
Previously did word processing, Multics, token rings, campus networks.

[Slide 1A] Topics

time plan: (Project, 5 min
System, 15 min
Lessons, 30 min
questions 10 min
60 min total)

[Slide 2.] Goal

apply engineering workstations to undergraduate (technical) education
NOT research in distributed systems (but a little does accidentally get done on the side.)

[Slide 3.] Athena Evolution

Method: IBM-MIT and MIT-DEC (two binary partnerships)
subsidy to allow living in the future of engineering w/s
[3 Overlays] Deploy while designing; three phases

Mention the NFS-like grant allocation procedure that gets funds to faculty who have teaching ideas. 125 such projects.

[Slide 1A--first repeat] (We move to topic 2)

[Slide 4.] Architecture: The Athena Workstation

PC vs Engineering Workstation choice
coherence with faculty/research computing base
graphics resolution, processor speed, memory size
OS capability
future convergence
size of present and future applications catalogs
price/function tradeoff
feature: high mips allows wasting on abstractions to gain coherence. (e.g., X)
lesson: the high performance of the high-end workstations produces unrealistic expectations for the low-end ones. (like a restaurant--you will usually be disappointed if you order the cheapest thing on the menu.)

[Slide 5.] Information Display

[Slide 6 with services overlay.] Architecture: The Major Services
Network/workstations/postoffice/NFS/RVD/Kerberos/ Hesiod/ServiceMgt/printers

[Each one is good for an hour talk in itself; keep this short; mention that there were papers at Winter 1988 USENIX conference!!]

[Slide 6 with storage services overlay.] Storage services

[Slides 7/8/9.] Architecture: The Storage Model

Reality takes into account the problems of scale
User view hides the reality
Shared storage required for system and class libraries, nice for personal files (debate on backup)

[SKIP! Slide 6 with Mail overlay.]

[SKIP! Slide 6 with Print overlay.]

[Slide 6 with Authentication overlay.]

[Slide 6 with Name overlay.]

[Slide 6 with Service Management overlay.]

[Slide 1A--2nd repeat] We move to topic 3

[Slide 10.] W/S UNIX; Coherence/Portability

What matters is the OS and display programming interfaces, not the instruction set.

[Slide 10 with EASY overlay]

[Slide 10 with HARD overlay]

[Slide 11] Hard

This is an example of a lesson learned, which actually takes us to the next section of this talk: problems encountered, lessons learned

[Slide 12.] The hard parts
scale is up two orders of magnitude, not one
the institution isn't uniformly prepared
the vendors don't know what this market is all about
the technology isn't ready
networking is much harder than people hope

[Again, each is worth an hour talk]
[Slide 13.] Scale (examples)
1 wizard/UNIX --> 0.01 wizards/UNIX
hand-tailoring (PC owner expects this, and UNIX makes it too easy)
versus central software update distribution. (The exceptions kill you)
problems:
keeping clocks coordinated
backup costs
synchronized network use
trouble propagates
electric power (3-phase neutral overload)
September registration spike
lesson: keep it simple
Example: software repair strategy--don't look at it to figure out what is wrong. Reload software; if it still doesn't work, call hardware repair
lesson: deployment in clusters much easier than at individual locations.
lesson: dedicated servers enhance availability, large number of small servers provides spare capacity

[Slide 14.] Unprepared institution (examples)
registrar: list of registered students
physical plant: machine room attention
physical plant: site preparation, ergonomics (lighting, table height)
administration: cost; not online themselves
administration: space for workstations
administration: pricing of services (network, storage, software, printing)
faculty: short on applications ideas
faculty: the faculty are, finally, mostly computer-literate but meanwhile the students became computer-fluent!
committee on discipline: hacking and copying

committee on privacy: clubroom atmosphere
libraries: network integration
legal office: site licensing
campus police: w/s theft
housing: space and policy
telecommunications: campus data net
graphic arts: laser printing facilities

********** related side topic
Controlling expectations
students: word processing; laser printers
faculty: 1. supercomputing, big memory, color, etc.
2. ability to run programs bought at a garage sale
Lerman: people will invest hours in a personal workstation but complain about lost minutes on a centrally provided facility.

University environment
Semester time gyroscope
Many subtle relationships in a University--who is in charge?

********** end of digression

[Slide 15.] Unprepared industry
site licensing; worry about unauthorized copies; need payment by use instead of by computer serial number
maintenance strategies (need blend of local expert plus help from co.)
network installation--you get to be the prime contractor
price/function tradeoff--need to hold function, drop price.
poorly understand market

[Slide 16.] Missing feature
need way to chain and lock the hardware
@P[Slide 16 with hole overlay]
@P[Slide 16 with Lock/Chain overlay]

@P[Return to slide 15]
removable balls in mice
Need laser printers built like an ATM
(locked panels, coin boxes, time-between-servicing)

********** another related side topic

System designers have very different requirements from students, so they don't work on the right things. We rarely ask students to work together, but education can benefit from 2-D and 3-D graphics. System designers usually work in teams, but hardly ever use graphics beyond text windows. So system designers spend all their time developing tools for working together and don't notice the unusability of the graphics packages.

********** end of digression

[Slide 17.] Unprepared Technology

general architecture: do it yourself, little guidance
network authentication
network service management
network naming
installation/configuration complexity (one wizard/machine)
remote file system is research topic, not
    fully-engineered, ready-to-use
terrorism (viruses, trojan horses) in shared information
user friendliness
(Three phase power problems)
ad hoc performance tools
too much interdependence of network services
information display technology is way ahead of programmability

[Slide 18.] Networking is hard

(slide includes obligatory incomprehensible network map)
physical media painful, expensive to install
too many standards, must bash heads technically
need "telephone company" mentality, not hacker/wizard approach
interesting problem, attracts wrong kind of people
not off-the-shelf at the system level (gateways, management, etc.)
Lerman: interfaces still expensive (especially for PC's)

[Slide 19.] Network lessons

[Firewalls Overlay] Don't forward trouble.
Firewalls contain trouble

[Broadcast Overlay] Floods of responses
sometimes appears to require forwarding or rerouting
damaged packet with maintenance/resend responses

[Quality Overlay]
lock up on physical disconnect or power glitch
wedge under massive collision
packets delayed in microcode buffers
intervendor transceiver inoperability
transceivers that draw more amps than specified

[Slide 20.] Unsolved problems
Others:
Building applications is much too hard.
Technology changes too fast for teachers to amortize development
Much iteration required for applications
Faculty incentives missing

[Slide 21.] Current Status
[Overlay highlights Athena's Technical Developments]
X window system--coherent interface to bit-mapped displays
Kerberos--network authentication ("who is the client?")
Hesiod--network naming ("where is my service?")
SMS--network service management for large scale
W/S UNIX--novice operator, minimal hand-tailoring, coherence