Scalable IMAP Services

Theory, Practice, and Non-technical Issues

Brad Knowles

brad@stop.mail-abuse.org

with

Phil Pennock

phil.pennock@globnix.org

http://www.shub-internet.org/brad/papers/sistpni/

Overview

- Theory
  - IMAP Literature Survey
  - IMAP Server Review
    - Open Source & Commercial
- Practice
  - Scalable Architecture Review
  - Survey of Selected Installations
- Non-technical Issues
  - Access Model versus Protocol
  - Hidden Costs for online/IMAP service
  - AOL vs. GMail

Theory

- IMAP Literature Survey
  - Then
  - Now
- IMAP Server Review
  - Open Source
  - Commercial
  - Server Scalability Issues

IMAP Literature Survey: Then

- Grubb96
  - How to Get There From Here: Scaling the Enterprise-Wide Mail Infrastructure
- DeRoest96
  - University of Washington IMAP Cluster
- Klensin96
  - What a Public Operator May Need From Servers
- Stevens97
  - Serving Internet Email for 60,000
- Beattie99
  - Design and Implementation of a Linux Mail Cluster

IMAP Literature Survey: Now

- Books
  - Mullet2000
    - Managing IMAP, published by O’Reilly
- Dissertations
  - Siotos2004
    - Large Scale E-mail System
- Magazine Articles
  - Dribin2003
    - Large-scale mail with Postfix, OpenLDAP and courier
  - Bauer2003/2004
    - Paranoid penguin: secure mail with LDAP and IMAP, Part I & II
  - Marcotte2004
    - HEC Montréal: deployment of a large-scale mail installation

IMAP Literature Survey: Now

- Papers
  - Graham2000
    - 0 – IMAP in 90 Days or how to migrate 25,000 users to IMAP in three months
  - Knowles2000
    - Design and Implementation of Highly Scalable E-Mail Systems
  - Rodhetbhai2002
    - A High Performance System Prototype for Large-scale SMTP Services
  - Miles2002
    - A high-availability high-performance e-mail cluster

IMAP Literature Survey: Now

- Further Afield
  - Yasushi99
    - Manageability, availability and performance in Porcupine: a highly scalable, cluster-based mail service
  - von Behren2000
    - NinjaMail: The Design of a High-Performance Clustered, Distributed E-Mail System
  - Mislove2003
    - POST: A Secure, Resilient, Cooperative Messaging System
  - Jeun2003
    - A High Performance and Low Cost Cluster-based E-mail System
  - Risson2004
    - Email Storage: Towards a Robust Peer-to-Peer Design

Theory

- IMAP Server Review
  - What is Scalability?
  - Open Source
  - Commercial
  - Server Scalability Issues

What is Scalability?

- Horizontal scalability
  - No user data stored locally on a server
  - Adding a new server to the cluster entails
    - Installing and configuring OS
    - Installing and configuring Applications
    - Changing cluster and meta-data configuration to deliver load to new server
      - Should be do-able in a matter of minutes, with JumpStart-like services or disk cloning techniques
    - Equally easy to take old server out of production

What is Scalability?

- Vertical scalability
  - OS & applications have been optimized and configured so that each user places only small load on the server
    - You can get a lot more users per server
      - Managing a large number of servers becomes difficult and increases overall probability of significant failure in the system

IMAP Server Review: Open Source

- Washington University (WU)
  - Simple, many types of mailboxes, local & remote users, least scalable
- Courier-IMAP
  - More complex, Maildir only, local & remote users, horizontally scalable
- Cyrus
  - Most complex, Cyrus mailbox directory only, remote users only, vertically scalable

IMAP Server Review: Commercial

- Bynari Insight
  - Based on Cyrus
- Mirapoint Message Server Appliance
  - Based on Cyrus
- Samsung Contact Server
  - Previously HP OpenMail
- Sendmail Advanced Message Server
  - Based on Cyrus
- Stalker Communigate Pro
- Sun Java System Messaging Server
  - Based on Cyrus
- SuSE OpenExchange
  - Based on Cyrus

IMAP Server Scalability Issues

- WU
  - Supports many mailbox formats, employing more levels of abstractions, more complex internal architecture, and having a larger memory footprint
  - For the preferred mailbox format (.mbx)
    - Deleting a single message is expensive (the entire mailbox has to be re-written)
    - Entire mailbox has to be read in order to display a single message
      - Not a speed issue, but does impact memory utilization
    - Not NFS-friendly

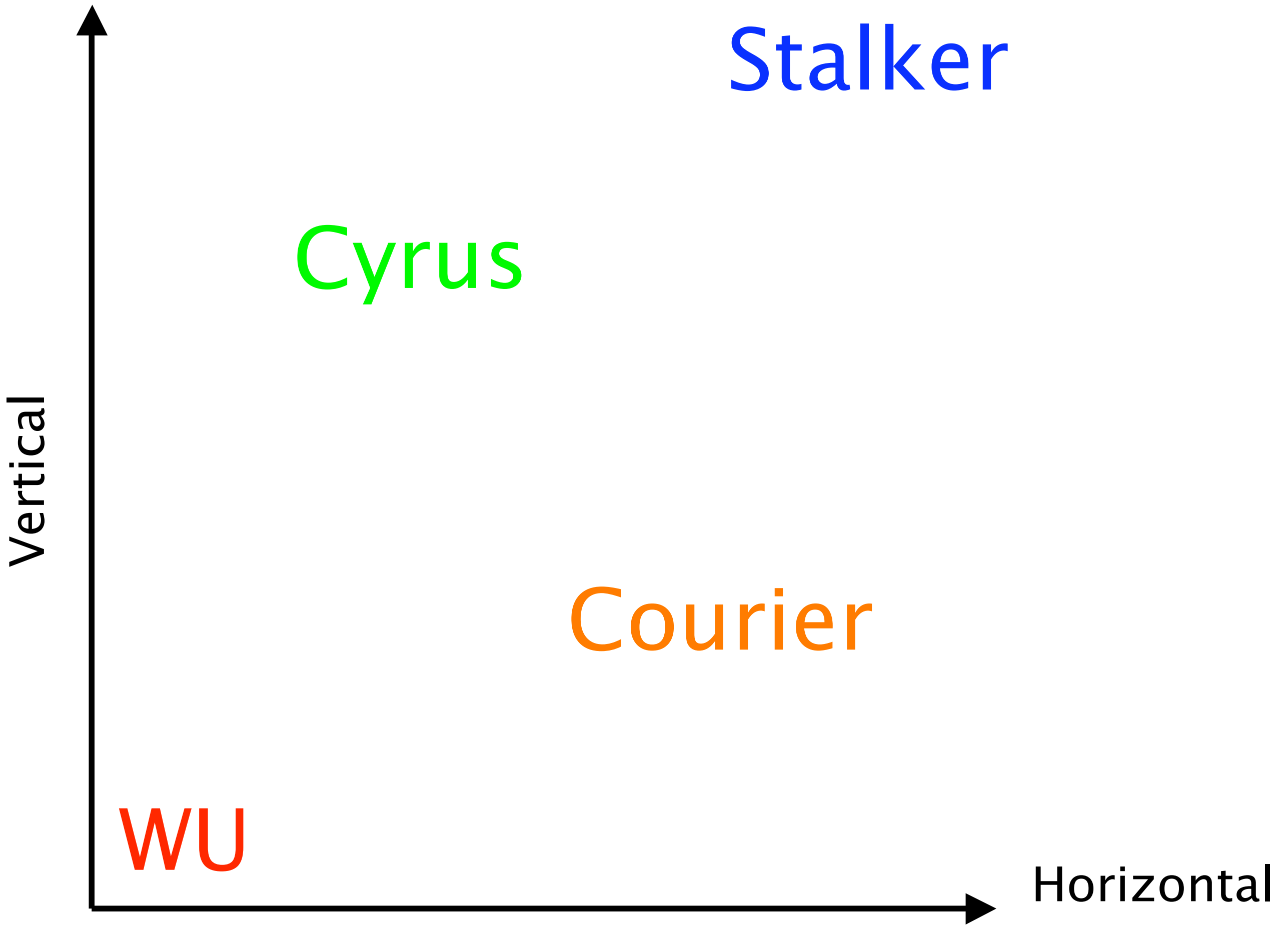
IMAP Server Scalability Issues

- Courier-IMAP (Maildir)
  - Must scan directory and `stat()` all files in order to get index
  - Must `open()` and `close()` each and every file in order to search mailbox
  - Files renamed to indicate status, which requires frequent directory re-scans
  - File names are very long, which causes iname caching structures to be invalidated
  - Mailbox directory structure is flat, which causes excessive delays when re-scanning or modifying mailbox with large numbers of messages
    - Also causes excessive synchronous meta-data update contention, exacerbated by excessive file renaming

IMAP Server Scalability Issues

- Cyrus
  - Depends on certain modern OS features (e.g., `mmap()`), so less portable
    - Also not compatible with NFS
  - Must `open()` and `close()` each and every file in order to do full-text search on mailbox
    - Only meta-data is in the index
    - However, this problem can be solved through the use of “squat” indexes for folders
  - Mailbox directory structure is flat
    - Causes excessive delays when modifying mailbox with large numbers of messages

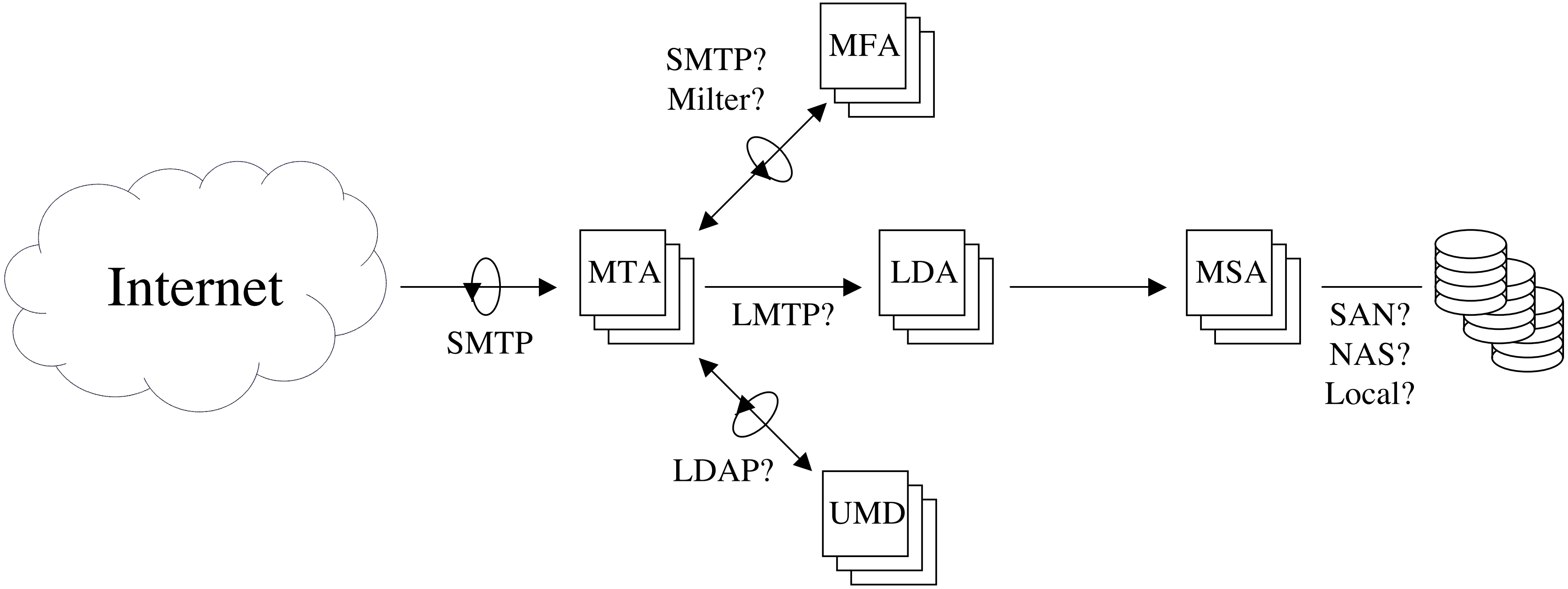
IMAP Server Scalability Chart



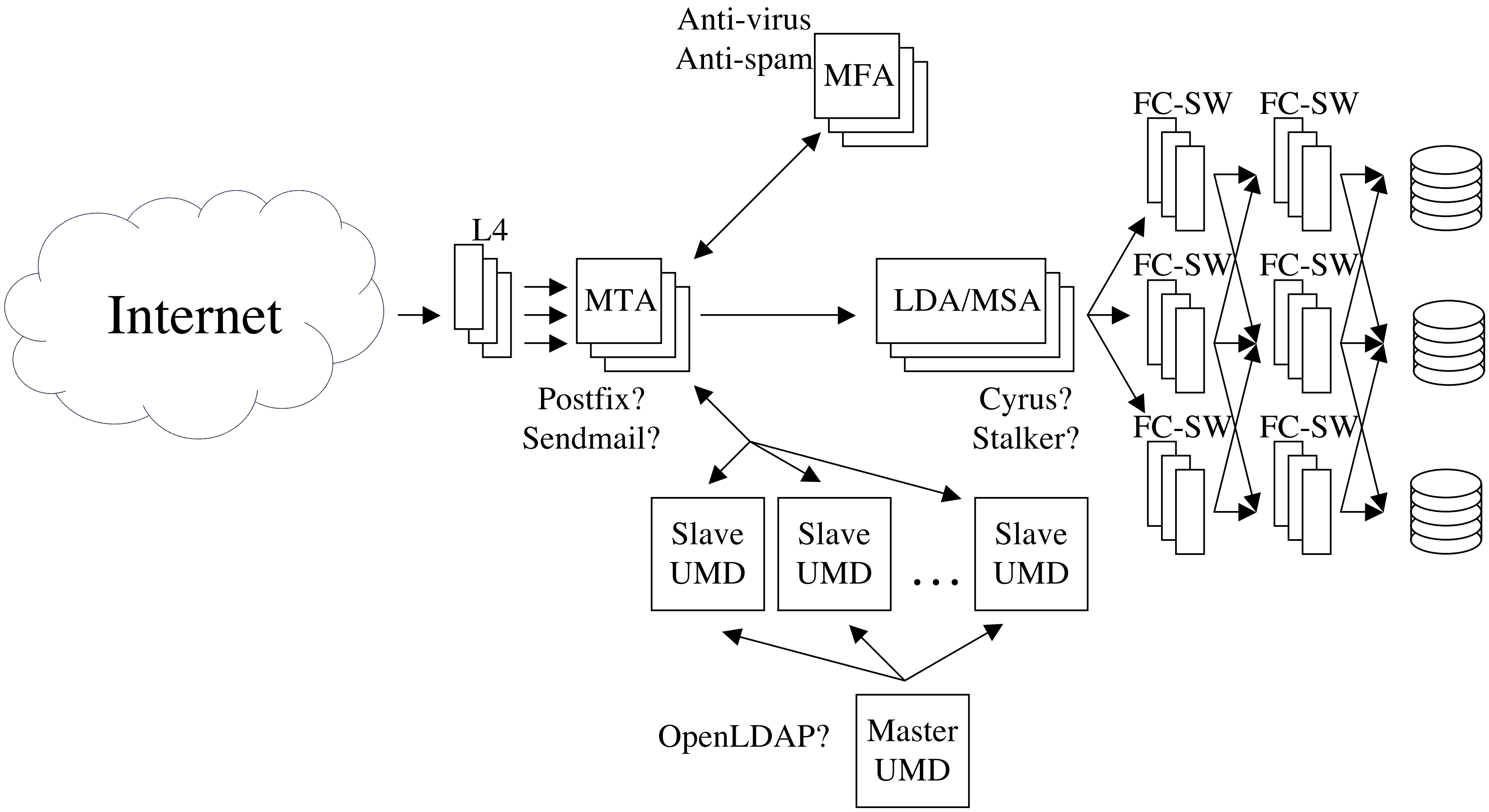
Practice

- Practice
  - Scalable Architecture Review
    - Storage and Retrieval
    - Functional and Detailed
  - Survey of Selected Installations
    - ISP/mail services provider in UK
    - Enterprise customer in Netherlands
    - University in Greece
    - University in Texas
    - Mail services provider in Australia

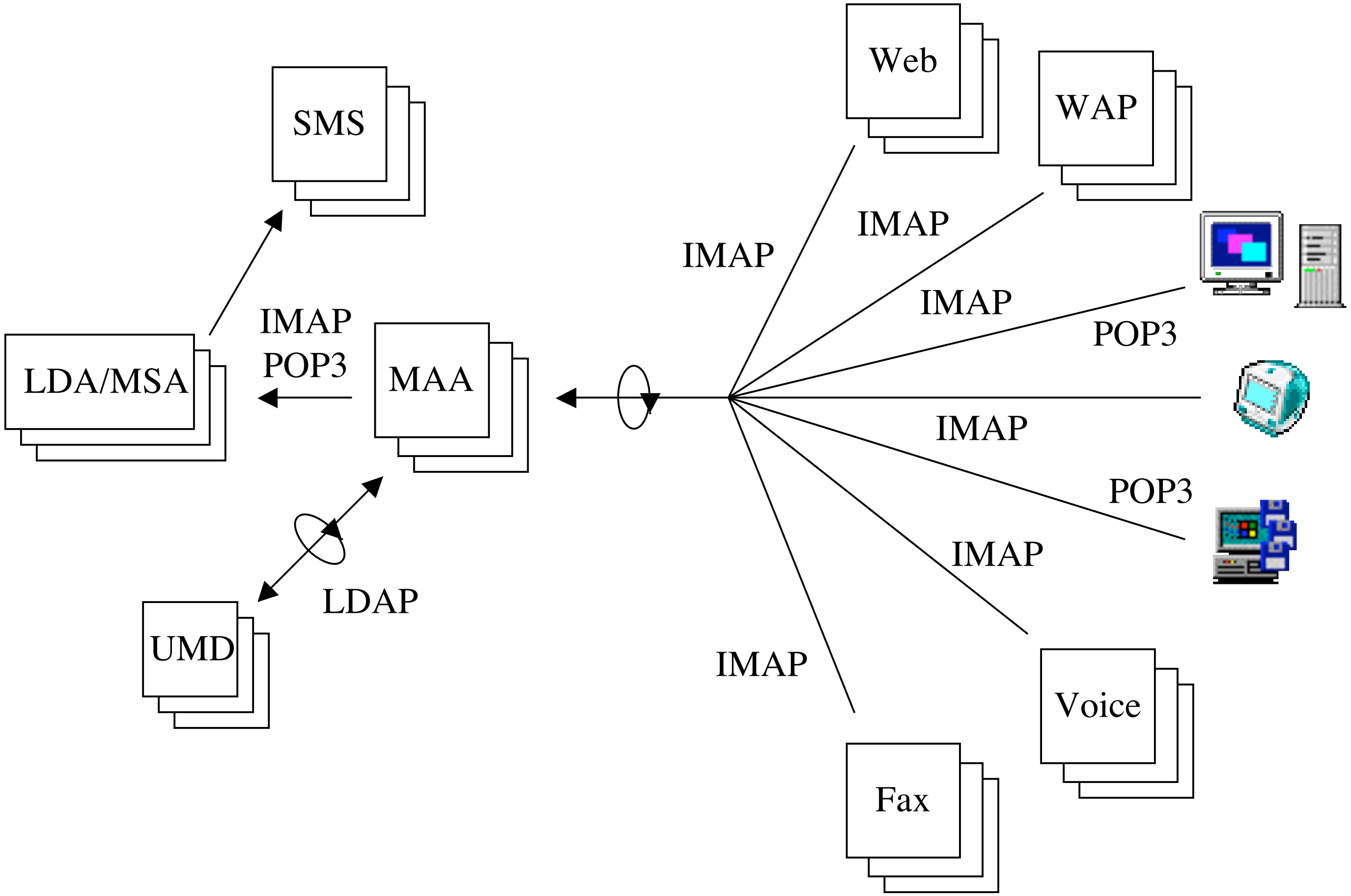
Functional Architecture: Storage



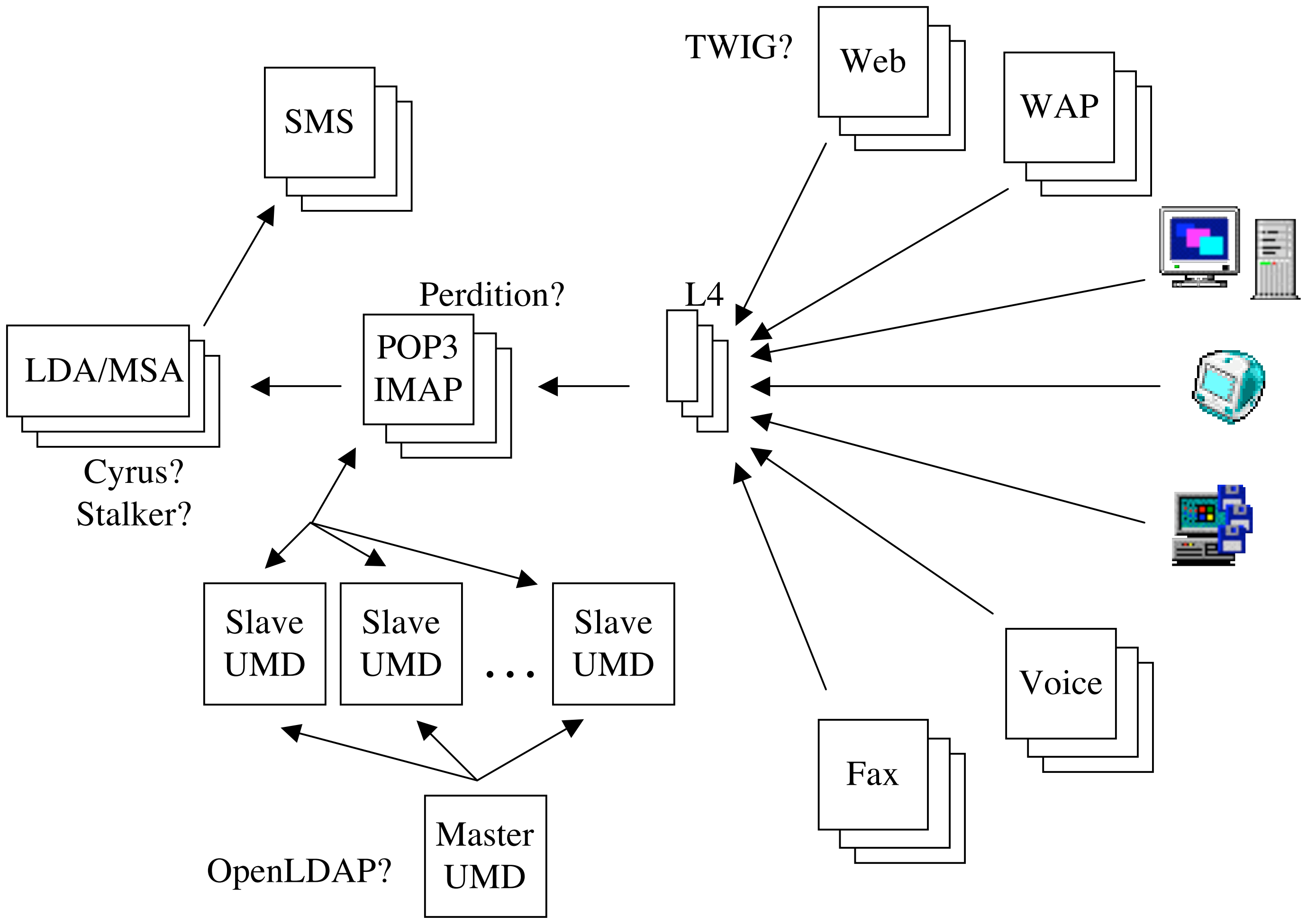
Detailed Architecture: Storage



Functional Architecture: Retrieval



Detailed Architecture: Retrieval



TWIG?

Web

WAP

SMS

Perdition?

L4

LDA/MSA

POP3
IMAP

Cyrus?
Stalker?

Slave
UMD

Slave
UMD

. . .

Slave
UMD

Voice

Fax

OpenLDAP?

Master
UMD

Scalable Architecture Summary

- Single Points of Failure (SPOFs) are our worst enemy, so we identify and eliminate all possible SPOFs
  - All components are at least duplicated, replicated, clustered, and operated in active/active high-availability/load-balancing mode
    - May alternatively be N+1 or N+M redundance, if duplication is not feasible
  - Failure of any one component can be routed around by other components in the system

Scalable Architecture Summary

- All devices have at least one cluster-mate
  - Primary function is to monitor mate(s) and take over all functions in case of failure

    -or-
  - Primary function is to monitor systems to which load is being distributed, and redistribute if failure is detected
  - Secondary function is active/active load-balancing with cluster-mate(s)

Scalable Architecture Summary

- All devices have
  - Redundant power supplies
    - Connected to separate redundant UPSes
    - On different circuits
      - Watch your phase variance!
- All devices on network
  - Support multiple IP addresses per NIC
  - Have at least two NICs per network
- All storage network devices
  - Use FC-SW to prevent cascade failure

Scalable Architecture Summary

- All Layer-4 Load-Balancing Switches
  - Distribute incoming load to Front-end Processors/Proxies
    - Inbound mail handlers
    - IMAP/POP3 proxies
    - Webmail servers
    - Etc…
  - Monitor cluster-mate(s) for failure and take over all functions if necessary
  - Detect failure in FEPs and redistribute

Scalable Architecture Summary

- All Front-end Processors
  - Short-circuit and offload all possible work from back-end message storage/access servers
    - E.g., anti-virus and anti-spam scanning, etc…
  - Connect to User Meta-Data servers to find out where to route remaining traffic
  - Distribute remaining traffic to appropriate back-end MSS
  - Detect failure in connected systems and re-route as appropriate

Scalable Architecture Summary

- All Message Store/Access Servers
  - Clustered with Veritas Cluster software
  - Use Veritas Volume Manager (VxVM) to manage all storage devices for user data
  - Use Veritas Filesystem (VxFS) for all user data storage
  - Are at least dual-connected to all storage networks
  - Connected to all message store contents
  - Technically capable of serving all user mailboxes
    - Mailbox/server affinity maintained in UMD servers, which are also used to redirect traffic to alternate servers if primary mailbox server is unavailable or overloaded

Scalable Architecture Summary

- All data storage devices use
  - RAID-1 where maximum reliability is needed
  - RAID-1+0 where performance is needed
  - RAID-5 where disk storage capacity is needed
    - Or where tests prove that there is little or no penalty for using RAID-5 instead of RAID-1
  - Multiple pre-defined hot-spare devices per cabinet
  - Disk devices which can be hot-plugged and reconfigured on-the-fly
  - Battery-backed non-volatile write-back storage cache
    - Must be mirrored internally
    - Should be able to be partitioned and statically allocated per storage volume to be exported

Practice

- Survey of Selected Installations
  - ISP/mail services provider in UK
  - Enterprise customer in Netherlands
  - University in Greece
  - University in Texas
  - Mail services provider in Australia

Selected Installations

- ISP/mail services provider in UK
  - Almost 200k user registrations in first year (2000)
    - Later sold retail ISP, and ADSL reseller/LLU telco businesses
  - Now has over 200k web services business customers
  - Original architecture straight out of DIHSES
    - Load with initial set of customers was not measurable
  - Unfortunately, mail services outsourcing didn’t work out in a suitable timeframe
    - Dot-bomb crash
    - Customers did not see value of managed services when compared to free services

Selected Installations

- Enterprise customer in Netherlands
  - Around 3000 “local” customers, ~7000 world-wide
  - Original architecture based on departmental all-in-one servers
    - E.g., Sun E4500, E6000, E10k, etc…
    - Running Solaris 2.4, 2.5, 2.5.1, 2.6, and 7
      - Starting to think about how to roll out Solaris 8 at the same time Sun started shipping Solaris 9

Selected Installations

- Enterprise customer in NL, page 2
  - Not vertically scalable
    - Too many functions overloaded on one system
      - E.g., shell access, home directory service, development, e-mail, NFS, Oracle, etc…
    - If a department grew or shrank, old hardware was not able to scale up or down with them
      - Large departments became small but still had big machines
      - Small departments grew big but still had to try to cram everything onto small servers

Selected Installations

- Enterprise customer in NL, page 3
  - Not horizontally scalable
    - User account data stored in NIS
      - NIS not scalable in and of itself
        - » Especially on the WAN
      - Could not be replaced by NIS+
        - » Due to use of old machines/OS versions and requirement to continue to support old machines/OS versions currently in the field
    - Actual user files stored locally
      - If a user moved from one group to another, files had to be copied, mail messages could be lost during transition, etc…
      - If user required extra storage but it was not available, it had to be provided via NFS mounts from other servers
        - » All servers ended up cross-mounting all other servers

Selected Installations

- Enterprise customer in NL, page 4
  - Expensive to support
    - Lots of old hardware required expensive support contracts
      - Sun E10k alone was over 1m Euro per year
    - Lots of expensive software contracts required to continue operations on old hardware
      - Oracle licenses even more expensive
    - Lots of administrator overhead required to keep old machines running
      - No time to install and configure modern network monitoring/administration toolkits
      - No time to do anything pro-active

Selected Installations

- Enterprise customer in NL, page 5
  - Consolidation desperately needed
  - Long-term solution for e-mail
    - Management decreed long-term move to Microsoft Exchange
      - Microsoft Exchange already in use for senior management and marketing
      - Initial entry cost was low
      - No consideration given to TCO if deployed company-wide

Selected Installations

- Enterprise customer in NL, page 6
  - However, Exchange was not feasible in short-to-medium-term
    - Technical staff proposed Unix-based mail cluster using
      - Inexpensive front-end hardware
      - Same back-end storage hardware as already decided (and paid for) by other projects
        - » I was already on-staff doing unrelated work, so my time was “free”

Selected Installations

- Enterprise customer in NL, page 7
  - Short/medium-term solution
    - Working with R&D, initial proposal was pretty much straight out of DIHSES
      - However, we discovered that Network Appliance NFS servers had already been procured for message store
        - iSCSI and DAFS were still in development, and not planned for support on the hardware we had
        - Cyrus-based products do not work on NFS
      - Budget was later determined to literally be zero
        - No new hardware could be bought
        - All software had to be freely available, or available through existing contracts

Selected Installations

- Enterprise customer in NL, page 8
  - Second proposal substituted Courier-IMAP for Cyrus-based commercial product
    - User meta-data directory server was OpenLDAP (testing)
      - » Company already had NIS -> LDAP migration planned and underway
    - MTA was sendmail
      - » Planning for future anti-virus/anti-spam processing where it should be more scalable than postfix
    - Front-end proxy was Perdition
    - Hardware was ten Sun Ultra 10 servers
      - » Found in a closet, hidden and unused for years
      - » Half the machines stripped to make five better equipped servers
      - » Two FEPs, three MSSes

Selected Installations

- Enterprise customer in NL, page 9
  - Annual Enterprise-wide TCOs
    - Open-source

– Software License	None
– OS License	Already paid
– Hardware	
» Five Sun Ultra 10	Already paid
– Personnel	Known
– Total	Very little

Selected Installations

- Enterprise customer in NL, page 10
  - Annual Enterprise-wide TCOs
    - Oracle database-oriented mail system
      - Software License high
        - » Believed to be > 1 million Euro/year
      - OS License known
      - Hardware
        - » Two full Sun V880 back-end servers med-high
        - » Two full Sun V480 front-end servers medium
      - Personnel known
      - Total less than Exchange
        - » Had to be less
        - » Otherwise Oracle would never have pitched it

Selected Installations

- Enterprise customer in NL, page 11
  - Annual Enterprise-wide TCOs
    - Microsoft Exchange

– Software License	
» Initial pitch	35 Euro/user/month
» Adjusted w/ real data	75 E/u/m
– OS License	?
– Hardware	
» Dozens of servers (~10x)	?
– Personnel	
» Lots of additional staff	?
– Total	> 3m Euro/year
» Adjusted w/ real data	> 8m Euro/year

Selected Installations

- Enterprise customer in NL, page 12
  - Status as of the time I left
    - Management in shell-shock over Exchange cost
      - They thought it might be expensive, but that much?!?
    - Management didn’t believe open source TCO
      - Nothing could possibly be that cheap and still work, right?!?
  - Meanwhile, open source implementation benchmarked
    - Strong evidence to indicate that it would be able to easily handle ~3000 LAN users
    - Architecture demonstrated to easily extend to multiple LAN clusters, ~7000 world-wide WAN users
      - All the real magic is in the LDAP database

Selected Installations

- University in Greece
  - University of Athens <http://email.uoa.gr/overview/>
  - Project started in 1999
    - Initial target user base of several thousand students & faculty
    - Projected growth to ultimately include over one million secondary education students

Selected Installations

- University of Athens, page 2
  - Architecturally similar to DIHSES
    - Sendmail (MTA), Cyrus (MSS), OpenLDAP (UMD), Perdition (POP/IMAP proxy), SquirrelMail (webmail), mailbox storage on SAN (EMC)
    - Custom development
      - » Integration of Cyrus and OpenLDAP
      - » Cyrusmaster administration tool
      - » All code available as open source

Selected Installations

- University in Texas
  - Project started in 1997
    - Started with ~9k students
  - Current back-end hardware in use since 1999
    - ~15k students plus all faculty and some staff
    - Sr. Administration and most staff on Exchange
  - Will start migrating to new hardware in 2005

Selected Installations

- University in Texas, page 2
  - Architecture is very similar to DIHSES
    - postfix, Cyrus, LMTP, Veritas VxFS, Veritas VxVM, separate inbound and outbound mail relay server clusters
    - SpamAssassin, postgrey, ClamAV, McAfee uvscan
    - LDAP used on front-end mail routers to determine final back-end destination
      - Student/faculty Cyrus-based system
      - Sr. Administration/staff Exchange server
    - No proxy

Selected Installations

- University in Texas, page 3
  - Current primary mail/message-store server
    - Sun Enterprise 250
      - Six internal SCSI hard drives used for OS and temporary storage
        - » Three volumes mirrored with Veritas VxVM
        - » UFS used for root volume
        - » UFS+Logging used for other volumes
      - External Sun StorEdge 3500 storage array for mailbox storage
        - » RAID-5+0 (RAID-5 in hardware + RAID-0 in software using VxVM)
        - » Veritas VxFS used for message store filesystem

Selected Installations

- University in Texas, page 3
  - Auxilliary servers
    - Outbound mail relay is Sun V120
    - Inbound mail router is Sun V120
    - Anti-spam/anti-virus processing on Sun V210
      - In combination with a Tipping Point appliance at the DMZ
    - Post-queue processing on Sun V120
      - Because they defer on Cyrus users over-quota instead of bouncing

Selected Installations

- University in Texas, page 5
  - New primary mail/message-store server
    - Sun V440
      - Clustered (with SunCluster) with second V440 for fail-over
        - » Other V440 will normally be used for unrelated NFS services
      - Internal hardware RAID controllers used for OS + temporary storage
        - » Filesystem as yet unconfirmed
      - External Sun StorEdge 6920 storage array for mailbox storage
        - » RAID configuration as yet untested
        - » Veritas VxFS still probably used for message store filesystem

Selected Installations

- Mail services provider in Australia
  - Fastmail.fm <http://www.fastmail.fm/pages/fastmail/docs/about.html>
    - Provides variety of account types
      - Free, $14.95 one-time fee, $19.95/yr, and $39.95/yr
      - Up to 2GB mailbox storage, 250MB file storage, domain hosting, IMAP & POP access, webmail, multiple aliases, outbound mail server, etc…
    - Largest known Cyrus installation in the world
      - Currently about half a million customers
      - Annual growth rate of ~200%
        - » I.e., they roughly triple in size every year
    - Strong supporters of open source/free software community

Selected Installations

- Fastmail.fm, page 2
  - Hardware
    - Mail storage
      - IBM xSeries x235, dual Intel Xeon processors, 6GB of RAM, ServerRAID 5i controller, UMEM non-volatile RAM drive for ReiserFS journals, RAID-5 SCSI drive arrays
    - Web/SMTP servers
      - White box, various configurations
    - All moving components redundant and hot-swappable
      - Fans, HDDs, PSUs, etc…

Selected Installations

- Fastmail.fm, page 3
  - Software
    - OS is Linux 2.6 (RedHat?) with custom kernels
    - Filesystem is ReiserFS
    - Postfix, Cyrus, Apache, Perdition, SpamAssassin, ClamAV, plus custom code
      - Most custom code written in Perl
      - Some custom code written in C for speed
      - Much custom code contributed back to the community
    - MySQL with InnoDB back-end for user meta-data

Selected Installations

- Fastmail.fm, page 4
  - Operations
    - Hardware
      - Most machines located in New York Internet Datacentre
        - » Four primary back-end mail servers
        - » One beta back-end mail server
        - » Two front-end web/proxy/encryption servers
      - One backup server in Texas
      - One emergency backup server in Europe?
    - Software
      - Checks entire system every two minutes for failures (including sending itself e-mail and confirming delivery within 30 seconds)

Selected Installations

- Fastmail.fm, page 5
  - Personnel
    - Two founders
      - Jeremy Howard (AU)
        - » Part-time, Manager for Messagingengine back-end
      - Rob Mueller (AU)
        - » Full-time, Manager for Fastmail front-end service
    - One support person (India)
      - Full-time
    - Three programmers (two full-time in AU, one part-time in US)
    - Various volunteer contributors to community (e.g., wiki, blog, etc…)

Non-technical Issues

- Access Model versus Protocol
  - Online vs. Offline
  - IMAP vs. POP3
- Hidden Costs
  - Requirements for long-term storage
  - Law enforcement access/abuse
  - Innocent third parties endangered
- AOL vs. GMail

Axiom

- E-mail is the ***ONLY*** universal mission-critical application
  - Each person/group will have various mission-critical applications
  - Lower-level services mission-critical, because mission-critical applications depend on them
    - E.g., network, power, etc…
  - But the only application that ***everyone*** depends on universally is e-mail

Access Model

- Offline
  - Message flow
    - Mail delivered to user mailbox
    - User logs on to download mail
    - User deletes mail from server
    - User logs off
    - User reads mail locally
      - May file to subfolder, may choose to delete
      - May log back on to send responses
      - May choose to send responses next time mail is checked

Access Model

- Online
  - Message flow
    - User logs on first thing in the morning
    - Mail delivered to user mailbox
    - User reads mail
      - May file to subfolder
      - Very unlikely to delete mail
    - User sends responses
    - User checks mail again
    - User may log off when they leave to go home

Access Model

- Observations
  - Offline
    - All permanent storage occurs on user's local computer
      - User responsible for all backups
    - User not typically logged on for long periods of time
    - User usually only logged on once at a time
    - If service crashes
      - User has only lost access to mail that has not yet been downloaded and maybe ability to send new mail

Access Model

- Observations
  - Online
    - All permanent storage occurs on server
      - Copies of messages may be cached locally
      - Service responsible for all backups
    - User typically logged on all day
    - User likely to have multiple simultaneous sessions logged on
      - Some protocols or clients depend on this
    - If service crashes
      - User has lost ***all*** access to ***all*** mail

Access Model

- Implications
  - Offline service provision requires relatively little resources per customer
    - Users not logged on for long periods of time
    - Most storage is transient and requires less reliability to provide adequate service
    - Example
      - You’re a cable company
      - If you’re broken, users can go watch TV somewhere else
        - » No one is going to die if you wait until it is convenient to fix whatever the problem is

Access Model

- Implications
  - Online service provision requires much more resources per customer
    - Users usually logged on all day
    - Very little storage is transient and much greater reliability is required
    - Example
      - You’re the power/telephone company
      - If you’re broken, users probably cannot get power/telephone somewhere else
        - » Someone may very well die if you delay fixing the problem

Protocol

- POP3
  - Typically used as an offline protocol
  - Doesn’t support multiple simultaneous logins
  - Many POP servers do not handle large mailboxes well
  - Most POP providers do draconian things
    - Disable “leave on server”
    - Prevent excessively frequent logons
    - Purge mailboxes of old mail
    - Provide only small mailboxes
    - Allow only small messages to be sent/received

Protocol

- IMAP
  - Typically used as online protocol
  - Multiple simultaneous logons implied
    - May be required by some IMAP clients
  - Using reasonable mailbox format, handles large mailboxes fine
  - Most IMAP providers are limited in the resource restrictions they can place on customers
    - All mail is almost always left on server
      - Unless user chooses otherwise
    - Users frequently logged in all day, if not permanently logged in

Access Model vs. Protocol

- Offline/POP
  - Old model, old technology
  - Well understood
  - 99% or even 95% availability may be perfectly suitable
- Online/IMAP
  - Not as old, not as well understood (wrt Internet)
  - Storage requirements 10x to 100x or more for same number of customers
  - Typically requires 10x or even 100x other resources to provide same level of SLA
  - Requires much higher SLA to be adequate
    - 99.99% or even 99.999% may be necessary
      - Each additional 9 costs another 10x to 100x to provide

Service Model

- You’re in Florida
  - Hurricane Nellie is bearing down on you
    - This is the fifth category four hurricane of the year
  - Who do you want providing your mission-critical service?
    - Power/telephone company?
    - Cable company?
- E-mail is mission-critical
  - Who do you want providing your service?

Hidden Costs

For Online/IMAP Service

- Requirements for long-term storage
- Law Enforcement
  - Access
  - Abuse
  - Other issues
- Provider abuse
- Innocent third-parties endangered

Hidden Costs

- Requirements for long-term storage
  - System requirements
    - Need to be able to recover from operator/admin error
  - User requirements
    - This is probably the sole repository of all e-mail
      - Must be able to recover from user error
  - Also Sarbanes-Oxley and other legal requirements
    - May be required to store all e-mail for seven years (or more)

Hidden Costs

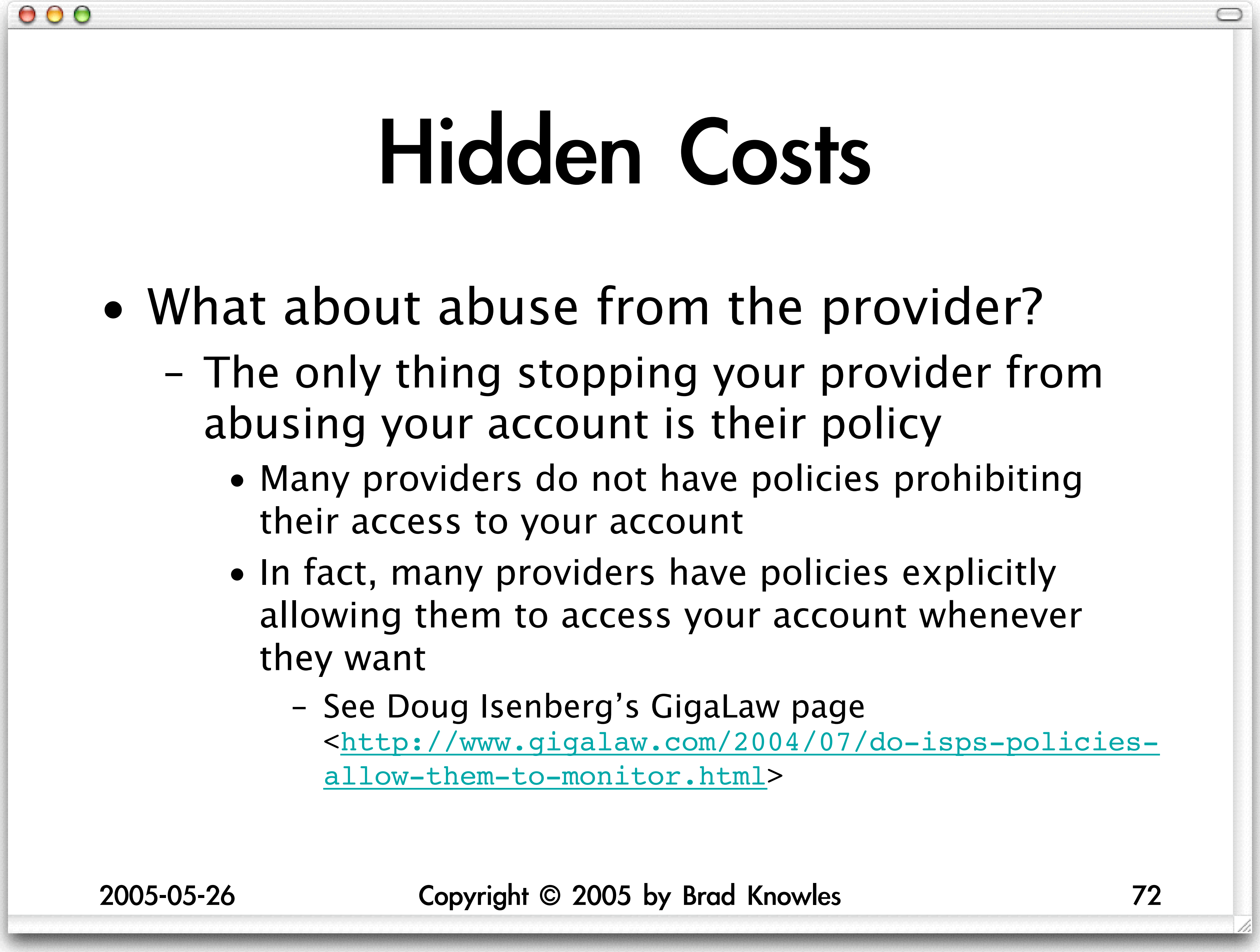
- Law enforcement access
  - Very high standard of proof required before law enforcement can legally enter your home and gather evidence against you
  - Much lower standard of proof required to obtain evidence from facilities outside your home
    - In many cases, all they have to do is ask
      - Your provider may hand over all your stored e-mail
        - » May also set up processes to capture all incoming/outgoing e-mail in real-time
      - Your provider may well hand over your hardware
        - » As happened recently to an Italian activist Group
      - Provider prohibited from saying anything to you, even if they opposed the action with all legal measures

Hidden Costs

- Law enforcement abuse
  - Official “fishing expeditions”?
    - Some official doesn’t like your organization
      - Such as the Dutch “What the Hack” group?
    - The government itself hates you?
      - Maybe you’re on a McCarthy-ist “Red List”?
    - History of paying commercial providers for information they could not legally gather themselves
  - Personal abuse of law enforcement power for financial reward?
    - Some cops are also crooks
      - Sell your personal information to private investigators
      - Sell your personal information to identity thieves

Hidden Costs

- Law enforcement issues
  - What about EU privacy guidelines?
    - What happens when a US law enforcement agency acts against a service provider in the US against an EU citizen?
    - What happens when a US law enforcement agency acts against an EU service provider against a US citizen?
    - What happens when an EU law enforcement agency acts against an EU service provider against a US citizen?
    - What happens when an EU law enforcement agency acts against a US service provider against an EU citizen?
  - What happens when EU law conflicts with US law?
    - Whose laws do you want to break?
    - Do you want to be caught in the middle?



Hidden Costs

- What about abuse from the provider?
  - The only thing stopping your provider from abusing your account is their policy
    - Many providers do not have policies prohibiting their access to your account
    - In fact, many providers have policies explicitly allowing them to access your account whenever they want
      - See Doug Isenberg’s GigaLaw page <http://www.gigalaw.com/2004/07/do-isps-policies-allow-them-to-monitor.html>

Hidden Costs

- Innocent third parties endangered
  - Third parties may well send you information that is sensitive
    - If that information had been stored on your private machine in your own home, it may have been difficult or impossible for law enforcement to “go fishing”
    - If that information is stored in your mailbox at your service provider, that may be fair game
  - You not only risk all your own private personal information that is stored centrally, you also risk potential private information from any third party who may send you mail

Hidden Costs

- Innocent third parties, page 2
  - You might think to use encryption to protect any potential third parties
  - However, the mere presence of encryption or encryption software may be taken to be an admission of guilt
    - C|Net article by Declan McCullagh “Minnesota court takes dim view of encryption” <http://news.com.com/2100-1030_3-5718978.html>

AOL vs. Google

- AOL
  - Architecture & Premise
  - Privacy
  - AOL Mail
  - What AOL Gets Wrong
- Google
  - Architecture
  - Premise
  - Privacy Issues
  - Gmail
  - Corporate Motto “Don’t Be Evil”

AOL

- Architecture & Premise
  - AOL is the only Online Service Provider left
    - CompuServe, Prodigy, GEnie, etc… all folded or got bought
  - Started out on Stratus mainframes as the only fault-tolerant hardware that really worked at the time
    - Had previous experience with Tandem, but despite claims, didn't provide real fault-tolerance at the time AOL was making their choice
  - Maintained mainframe/fault-tolerant methodology

AOL

- Privacy
  - AOL takes privacy seriously
    - One of the strongest privacy policies in the business
      - People get fired for first-time violations of user privacy
  - AOL doesn't really do their own search
    - They outsource that to other firms
  - AOL does do extensive data mining regarding usage patterns
    - Tracks every click, every mouse movement, every character typed, for ~25% of all customers
      - Information is anonymized
      - Looking for data indicating that common operations are too hard, require too many clicks
    - AOL does also tie private user information to advertising
      - All work done in-house, never sold or exposed to advertisers

AOL

- AOL Mail
  - Never tries to correlate private information in mail folders with personal consumer information
  - Does delete messages
    - Unread messages are deleted after 30 days
    - Messages that are read deleted after one day
    - Messages that are read and marked “keep as new” are deleted after seven days
    - Messages deleted by the user are immediately removed
    - Of course, these defaults can be changed, within limits
  - Provides AOL client, webmail, POP, and (now) IMAP access
    - Online access model
      - AOL is an Online service provider, has the correct mindset

AOL

- AOL Mail
  - Keeps only one backup
    - Database-structure mail system
      - Alternates between two sets of database servers
    - Reclaims free space every night
    - System backup only, not accessible to users
  - Long-term storage & backups is up to the customer
    - Use AOL Filing Cabinet
- Retention & backup policy explicitly chosen to avoid entanglements with law enforcement
  - If law enforcement presents legally binding request to obtain all mail for a user, AOL can only provide what is currently visible in the user mailbox, plus what may not have been reclaimed from the heap since the previous night

AOL

- What AOL Gets Wrong
  - Anti-spam system is too complex
    - Too easy for a user to accidentally report legitimate mail as spam
    - Burden of proof is on the operator of the sending system
      - Can have catastrophic results on entire ISPs and businesses, due to stupid acts on the part of a few AOL customers
    - Silently throws away any e-mail that has even the slightest hint that it might potentially be spam, without recourse from the user
      - If that was a legitimate business offer, your company may go bankrupt because you didn’t see it
  - Should not be deleting any user e-mail unless explicitly directed to do so
    - Give the user a mailbox quota and let them deal with overflowing mailboxes

AOL

- What AOL Gets Wrong
  - Support system is too rigid and complex
    - If you call for help, you might as well be talking to a robot
  - Still try desperately to keep everyone in the “Walled Garden”
    - Try too hard to stick to 100% proprietary interfaces and actively prevent interoperability with anyone else
  - But they desperately want to bring in new Internet customers
    - To replace all the dial-up users that are converting to broadband and switching providers
    - New customers are going to want interoperability
      - Want to use one chat or e-mail client that works everywhere
    - Result is a service suffering from multiple personality syndrome
      - GNN.com would have been a good, but got canned years ago
      - AIM.com perhaps a better fix?

Google

- Google architecture is based on clustering, replication, distribution, and load-balancing
  - If a layer four switch goes down, that’s okay because they’re always configured in pairs and the second one will take over
  - If a given front-end web server goes down, that’s okay because the layer four load-balancing switches will direct the traffic elsewhere
  - If a given back-end database server goes down, that’s okay because the front-end web servers will direct their traffic elsewhere
  - If a given cluster goes down, that’s okay because the geographic load balancing system will direct the traffic elsewhere
  - If all of Google is down, that’s okay because there are plenty of other web search engines

Google

- Google’s premise — it’s “just” search
  - If two users do the exact same query at the same time and get two different answers, that’s okay
  - If the same user does the same query twice in a row and gets two different answers, that’s okay
  - After all, it’s “just” search
    - As far as the users are concerned, there’s nothing mission-critical here

- Google is “just another” web search/services company

Google

- Privacy issues
  - Google remembers every search you’ve ever done
  - Google Toolbar tracks every URL you visit
    - If any are not indexed by Google, it adds them to the list
    - If you go to a private web page that is password protected, the contents will now be indexed by Google
  - Google proxy
    - Compresses results for increased speed
    - Connected to other Google proxy services around the world
      - Bi-directionally? Upstream proxy caches your cookie?
        - » Private information for some users has been exposed to others, because the proxy still appeared to be logged in as the other user
    - Pre-fetches many URLs for every web page you visit
      - If one of those URLs was for a “delete” button on a webmail provider somewhere else, all your mail may be gone as soon as you view the index page

Google

- Privacy issues
  - Google recently bought Axciom
    - A data mining company
    - Embroiled in numerous privacy scandals
    - Sells information to law enforcement and commercial customers
      - InfoBase, the largest collection of customer behaviour
      - Personicx, tracks specific consumer behaviour of almost every household
        - » Including income, shopping, and bank balance information
      - Provided personal information to Transportation Security Administration CAPPS-II project
      - Would have been primary source of information to Total Information Awareness project

Google

- Gmail indexes all private content
  - Combines that with information about the consumer to tailor advertisements
  - Gmail never deletes anything, even if you ask it to
    - It just hides it from you so that you don’t see it anymore
  - Gmail only provides web and POP access
    - However, they have an online access model
      - Requires online-style operations mindset
    - But Google is “just another” web search/services company
      - Corporate mindset is more like cable than power/telephone company

Google

- “Google expects itself to be in the enviable, and profitable, position of being the largest personal information repository on the planet” <http://www.politechbot.com/pipermail/politech/2004-April/000574.html>
- Google is the commercial equivalent of the FBI Carnivore program or the NSA Echelon system
  - Not only do they have all known public information about you
  - They also have all your private information that has ever passed through your mailbox
  - And they have much less legal restrictions on what they do with that information <http://www.epic.org/privacy/gmail/foirequest.html>

Google

- The only protection you have is their supposed corporate motto “Don’t Be Evil”
  - But corporate mottos have changed in the past
  - Corporations with a good history of privacy protection have been sold in the past to those that are bad
    - Once your privacy has been violated, it can never be recovered

- Just because you can do something, does not necessarily mean you should

Acknowledgements

- Phil Pennock
  - If he hadn’t written his white paper, this talk would never have happened <http://www.sage.org/whitepapers/pennock.mm>
- Amos Gouaux
  - Provided much encouragement, pointers, information, and links to other sites
- Apostolos Siotos
  - Lots of good questions, provided useful information about University of Athens
- Jeremy Howard
  - For extensive details of the operations of fastmail.fm

Acknowledgements

- Bram Moolenaar
  - For poking and prodding me along into agreeing to do this talk
- Mark Crispin
  - For his discussion of mailbox formats at <http://www.washington.edu/imap/documentation/formats.txt.html>
- Sam Varshavchik
  - For his discussion of Courier-IMAP at <http://www.courier-mta.org/mbox-vs-maildir/>
- Many fellow SAGE members posting to the sage-members mailing list