

CS290i - Lecture 17 GoToMyPC Experience

Scalable Internet Services and Systems, Winter 2002

Thorsten von Eicken
Department of Computer Science
University of California at Santa Barbara

Talk today @3pm

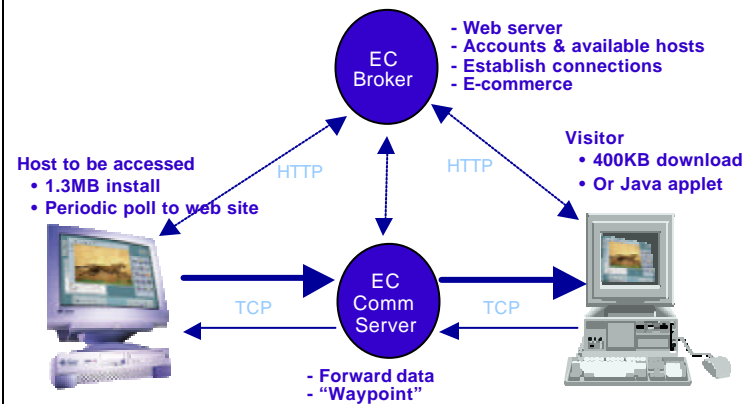
■ UCSB Department of Computer Science Colloquium

- Date: Wednesday, March 6, 2001
- Time: 3:00-4:00 PM
- Place: Engineering I, Room 2114
- Title: **An Architecture for Well-Conditioned, Scalable Internet Services**
- Matthew Welsh (Faculty Candidate)
- University of California, Berkeley
- <http://www.cs.berkeley.edu/~mdw/>

Expertcity Core Business

- **Web-based remote access solutions**
 - "Being able to remotely view and control any screen connected to the Internet..."
- **History**
 - Founded 1998, \$34M Sun/WitCap/BV
 - 100 employees
- **Target markets**
 - Technical support
 - Telework
- **Complete Web based ASP solution**
 - Download & go
 - Best screen sharing technology
 - Full communication solution
 - Cross-platform
 - Secure and Private

System Components



Research Issues

Real-time Data Compression

- Sophisticated image compression based on marks
 - 10 x better than GIF, 1000 x better than raw data
- Motion detection (can save another 100x)
- Scaling of images

Usability

- Fast – Easy – Secure
- Firewall friendly & automatic proxy detection
- Small code, portable, minimal dependencies
- Reliable & fault tolerant

Security

- End-to-end authentication
- End-to-end encryption
- Removing the servers as a point of vulnerability

Open Research Issues

Networking ("Overlay Networks")

- Choosing the best waypoint & connection
 - To get the best performance/price
 - To mask network faults
- Managing real-time traffic
 - When proxies are ubiquitous
- System Modeling
 - Understanding and simulating end-to-end system

Scaling

- Scaling across geographically diverse sites
- Without reducing reliability

Security

- End-to-end authentication & encryption

Multicast

- "Multicast" for presentation or Collaboration
- Collaborative applications

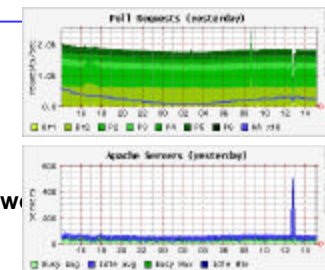
Reliability

Goal: site functional 99.9%

- E.g. <10 minutes down per week
- Incl. most sched. maintenance

Lesson: network, network, network

- Excellent connectivity required
 - the ISP's people&procedures matter more than the hardware
- Multi-homing, BGP
 - Fri 4:30pm: server crash (experiment)
 - Sat 8:40am: server crash (Solaris bug)
 - Sat 12:40pm: ISP router crash (Cisco...)
- Real solution: multiple sites & client-level fail-over



Primary-backup

- *Primary-backup fail-overs fail "monthly"*
- **List of failed fail-overs:**
 - Ethernet spanning tree
 - Cisco HSRP
 - ServerIron load balancer
 - Oracle/sun-cluster
 - T1 to VPN (OSPF)
 - DNS
- **Has not (yet) failed:**
 - Raid
- **Must exercise fail-over periodically**
 - causes it to fail during maintenance periods
- **Real solution: run multiple instances concurrently**
 - but, can't afford to test them unless there is no alternative

Expertcity Outages

<ul style="list-style-type: none"> ■ 7/26/00: 5 minutes <ul style="list-style-type: none"> ● Routing problems ■ 8/8/00: 5 minutes <ul style="list-style-type: none"> ● Hosting facility router down ■ 10/17/00: 1 minute <ul style="list-style-type: none"> ● T3 looped ■ 11/9/00: 1 hour <ul style="list-style-type: none"> ● Qwest announces route ■ 2/6/01: 10 minutes <ul style="list-style-type: none"> ● GoTopower cord pulled ■ 2/6/01: 1.5 hr maintenance <ul style="list-style-type: none"> ● Eth span tree/uplink fast ■ 2/26/01: 2 minutes <ul style="list-style-type: none"> ● GoTobeta JVM restart ■ 3/5/01: 2 minutes <ul style="list-style-type: none"> ● GoTobeta JVM restart 	<ul style="list-style-type: none"> ■ 3/20/01: 20 minutes <ul style="list-style-type: none"> ● DB fail-over failed ■ 3/21/01: 7/15 minutes <ul style="list-style-type: none"> ● DB E-cache failure ■ 3/23/01: 1.3 hours <ul style="list-style-type: none"> ● Router packet multiply ■ 4/3/01: 2 minutes <ul style="list-style-type: none"> ● Failed serverIron fail-over ■ 4/3/01: 5 minutes <ul style="list-style-type: none"> ● GoTo Solaris crash ■ 5/8/01: 2 minutes <ul style="list-style-type: none"> ● DTS JVM restart ■ 6/6/01: 2 minutes <ul style="list-style-type: none"> ● DTS JVM restart ■ ... ■ 1/14/02: 6 hrs site unstable <ul style="list-style-type: none"> ● DB overload due to marketing
---	---

Comm Server / Waypoints

- **Function**
 - Real-time streams between users A and B are routed via intermediate application-specific router
- **Benefits**
 - Required in the presence of firewalls/proxies
 - ◆ HTTP/TCP is the internet foundation, not IP
 - Often better when connecting "foreign" hosts
 - Excellent for debugging
 - Can record data for auditing
- **Disadvantages**
 - Worse when connecting "close-by" hosts
 - Additional point of failure
 - Bandwidth costs

Comm Server performance requirements

- **Latency?**
 - Added to each "packet"
- **Number of connections?**
 - < 4KB/s each

Comm Server Design

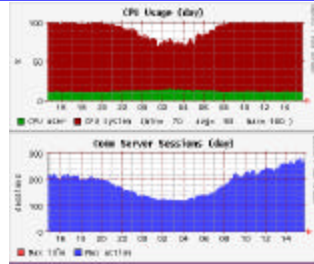
■ 1U rack mount PC

- 933Mhz P-III, 512MB, 60GB
- FreeBSD 4.x

■ Single-threaded C++ program

- Non-blocking I/O
- While (1)
 - ◆ Poll(readable sockets)
 - ◆ Accept()
 - ◆ Read()
 - ◆ Move data around
 - ◆ Poll(writeable sockets)
 - ◆ Write()
 - ◆ Once every 30 seconds:
 - ! check for timeouts
 - ! dump state to log file

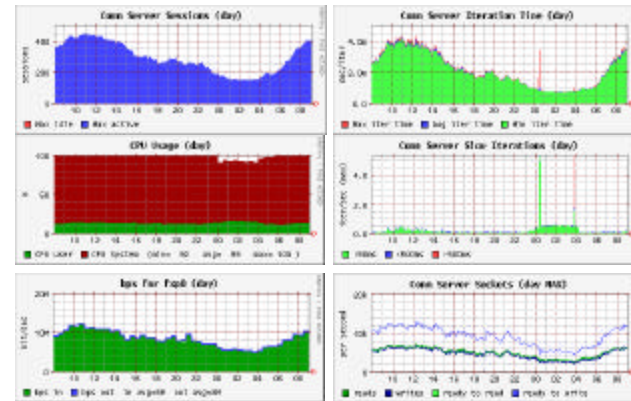
■ Implemented by Albert Alexandrov



- Poll(list of all sockets)
 - slows down linearly with number of sockets
- Workarounds:
 - Solaris: /dev/poll
 - FreeBSD: k-queues

13

Performance Behavior



14

Performance History

■ Comm Server evolution

Version	OS	Mhz	Max Sessions	Iteration	100% CPU	Limiting Factor
Perl	Solaris	450	30	50.0 ms	15 sess	user-level CPU usage
C++	Solaris	450	120	10.0 ms	70 sess	system CPU usage
C++	FreeBSD	933	200	0.8 ms	170 sess	accept()
C++ opt	FreeBSD	933	?	1.8 ms	190 sess	Marketing

■ Some unexpected problems:

- Too few accept() per iteration
- Nightly backups / tripwire scans
- Admin communication to web server
- Local file system flush
- -> All are outside the "critical path"

15