Research, Startups, and Risk

Brad Smith
Overview

• Why startups matter in UC.
• Cenus… a startup howto (or hownot:)
  – Technology
  – Product development
  – Funding
• Managing risk.
What is Purpose of UC?

- Research
- Teaching
- Public Service.
- Creation
- Dissemination
- Application…
- …of new knowledge.

Important goal of applying new knowledge is...

Technology transfer is important function of UC.

Startups are an important vehicle for tech transfer.

Opinion: students are the best agents of tech transfer!
UC in World of Higher Ed

• Carnegie Foundation (www.carnegiefoundation.org) classifies degree-granting institutions in the U.S.
  – Tribal
  – Special Focus
  – Associate
  – Baccalaureate
  – Master’s
  – Doctorate
    • Doctoral/Research
    • Research University (high research activity)
    • Research University (very high research activity) - “R1”

• In general, UCs are “R1s”.
• Opinion: R1’s are knowledge engines for industry and society.
Importance of UC as R1?

- There are 4,391 degree-granting colleges or universities in the U.S.
- How many are R1s?
  - 96 (2.2%) Research Universities (very high research activity).
  - 63 (1.4%) of these are public.
- UCSC’s role
  - 1 of < 2.2% dedicated to expanding human knowledge.
  - 1 of < 1.4% who serve this role solely for the public good.
  - 8 of which are UC’s that are near/at the top of this list.

- **Tech transfer is core to UC, and UC is one of few doing this**
- **Opinion: UC’s tech transfer role is incredibly important!**
Cenus
Summary

• Startup called Cenus Technologies
• Content routing
• Chief Architect from 2000-2002
• Based on research with Prof. JJ Garcia-Luna
• Have not been involved since ~2003
• Cenus is still in business…
Cenus
The Technology
The Internet

• What is an invariant of the Internet?
• It grows.
Client-Server

• Clients communicate directly with server
• All requests traverse the Internet
• Average latency is half the Internet’s diameter
• As the Internet grows this becomes painful
  – High latency → loose customers
  – High jitter → poor streaming performance
  – Single-point of overload and failure
Client Server in the Internet

• What’s the problem?
• Performance always gets worse.
Problem with Client-Server

• Clients communicate directly with server
• All requests traverse the Internet
• Average latency is half the Internet’s diameter
• As the Internet grows this becomes painful
  – High latency $\rightarrow$ loose customers
  – High jitter $\rightarrow$ poor streaming performance
  – Single-point of overload and failure
Solution - Caching

• Store copy of fetched data “locally”
• Satisfy future requests from cache
• Previous uses: memory, disk
• Effectiveness depends on
  – Locality of reference (requested object requested recently?)
  – Differential in access times (how far to the server?)
  – Request rate (how many clients does cache see requests for?)
  – …only control last one
Jim Gray - Storage Latency: How Far Away is the Data?

Andromeda

Tape /Optical Robot

10^9

Disk

10^6

Memory

100

On Board Cache

2

On Chip Cache

1

Registers

Springfield

This Campus

This Room

My Head

Pluto

2,000 Years

1.5 hr

10 min

1 min

October 14, 2010

TIM 101
Web Caching

- Clients talk to cache
- Cache fetches content for client
- Subsequent requests satisfied by cached content
- Weak cache coherency (TTL)
- Goals – improve performance, save external bandwidth

**Effectiveness of cache measured by hit rate**
- *is a function of request rate…*
- … increase chance that content requested recently enough.
Standalone Caches not Effective

- Need very high request rate for effective caching
- Place caches high up network hierarchy
- Expensive
  - Big cache machine
  - High bandwidth cache connectivity
- Same problems - latency, jitter, reliability
- From many loaded web servers…
- …to one loaded cache server
Hierarchical Caching

• Attempts to resolve hit rate/performance trade-off
• Organize caches in hierarchy
• Only root cache fetches content from server
• In effect, aggregates request streams over set of caches
Results

• Performance is worse(!)

• Multiple caching hops exacerbates…
  – Latency
  – Jitter
  – Reliability
Our Original Insight

• Limitations of client-server model
  – On-demand caching attempts to address

• Simple caching depends on very high locality of reference
  – Web access does not have adequate locality of reference

• Hierarchical caching involves too many hops
  – Attempts to address this have not been effective

• Object routing
Internet Routing

• Telephone routing
  – Phone number specifies route
  – Statically configured route to each telephone

• Internet routing
  – IP address says nothing about location
  – On-going computation of route to each host
  – Have developed very efficient solutions
Object Routing

• View object (web page) as destination

• Compute route to each object on on-going basis
  – Closest instance of that object ("anycasting")

• Organize caches in overlay network
  – Can’t require object routing in all routers ("boil the ocean")

• Object router
  – Place next to each cache
  – Configured as parent of its cache
Object Routing

- In effect, a hierarchy…
  - With at most 2 hops
  - Where 2\textsuperscript{nd} hop always “has content”…
  - …and incurs least possible latency in the 2\textsuperscript{nd} hop
- I.e. hierarchy with bounded overhead(!)
- Request streams aggregated over all cache sites…
- …high “cache cloud” hit rate(!!).
- Effectiveness depends on performance differential…
Object Routing Protocol

• Very efficient
  – Unpopular objects, low update rates, wide distribution
  – Popular objects, high update rates, limited distribution
  – Supports request rate to pull all Internet content into caching system!

• Very robust
  – Benefits depend on local cache.

• Requires huge (billions) routing tables
  – Map object to destination
  – Developed proprietary solution
  – ~10-20bytes per entry
  – Constant access time (10’s of instructions)
Technology Summary

• Hugely scalable, weak coherency, on-demand caching

• Applications
  – Peer-to-peer
  – Web caching
  – Domain Name Service (DNS)

• Futures
  – Strong cache coherency (e.g. distributed filesystems)
What were the Risks?

• How will object routing perform?
  – Analysis
  – Simulation
  – Prototype

• How handle huge routing tables?
Cenus
Product Development
Cenus History

• Late 1999 - Developed object routing ideas

• 1/2000 - Cenus Technology calls
  – Existing company from Florida in on-line video distribution
  – Existing investors all private, “angel” investors
  – Looking for technology

• 2000 - Worked with Cenus to get further investment
  – No luck with VCs
  – Expanded group of angels
Cenus History (cont.)

- 2001 - Development in Utah
- 2002 - Development in Scotts Valley
- 2003 on
  - Basic technology developed
  - Attempt to find market
Cenus Product Development

• During .com boom needed very little… today need prototype

• Concept
  – “Mock-up”
  – Illustrate concept with scripted demo

• Prototype
  – Open source tools and FreeBSD
  – Ex-Novell engineers
  – Primitive build technology
  – Performance problems
Cenus Product Development

• Production
  – Ported to Solaris
  – Replaced Novell engineers with Silicon Valley engineers
  – Resolved performance problems

• Challenges
  – Routing algorithms are very subtle
  – Routing is tough to debug
  – Routing is not sexy to demo
  – Object routing table design is very abstract…
What were the Risks?

• 2000 was boom time in Silicon Valley
  – Engineers were expensive.
  – Space was expensive.
  – \textit{Located in Scotts Valley and used Novell engineers}

• Novell engineers were not Valley engineers
  – DOS/Windows perspective
  – Primitive software engineering technologies
  – Robust product development models
  – \textit{Didn’t know how to work on Valley time.}

• Marketing...
  – Spent our time looking for investors
  – No time identifying customers
Sources of Funding

- Venture Capitalists
- Angel Investors
- Grant funded research
Venture Capitalists

• Professional investors
• Invest other people’s money
• Provide
  – Connections (talent, partners, services)
  – Discipline
• Require
  – A (large) share of the company
  – Control of the company
How VCs Work

• They aren’t technical
• Herd mentality
• Funding process
  – Rounds
  – Lead and secondary investors
• They prefer large investments
• They don’t say “no”
• They don’t sign NDAs
• They will use what you teach them
What Gets VC Attention

• An unfair advantage
  – A good idea
  – A good team
    • Reputation
    • Skills

• “Buzz”
  – Something that gets investors attention
  – Well known team members
  – In a currently hot field
How to Work With VCs

• You need an introduction
• You need something they can’t buy elsewhere cheaper
  – Management team
  – Critical technical talent
• In the end you’ll be depending on their integrity!
  – Reputation
  – Gut sense of trust
• Remember - they’re accountable to their investors
Angel Funding

• Wealthy individuals
• Not professional investors
  – Less involvement in business
  – Less help with connections
  – Don’t do discipline
• Don’t require as large a share of company
• Don’t get as involved in company
• Can be impediment to VCs
Academic Research

• Grant funded research

• Technology ownership
  – Funding agency may claim some ownership
  – University claims ownership
  – Typically inventor is given share (UC > 30%)
  – In the end, not a big issue
    • First to market is the big issue

• Great for developing a prototype
What are the Risks?

• VCs
  – Will give too little…
  – …and take too much

• Angels
  – Will scare away VCs

• Academic research
  – Getting funding for the research you want…
Cenus
Lessons
Risk at Cenus

• What were the risks and how did we deal with them?
  – Technology… did a reasonably good job.
  – Marketing… barely thought about it.
  – Development costs… attempts to mitigate were not effective.
  – Environment… .com crash.
  – Funding… we depended on Angels (exacerbated last three risks)

• We gambled...

• *You don’t succeed in risky environments by gambling!*
Risk in General

• To succeed in a world of risk…
  …you don’t live with it,
  …you eliminate it.
Cisco Network Management and Operations Lab
Thank you!

brad@soe.ucsc.edu