NAF
NETWORK AUTOMATION FORUM

AC2 OPENING KEYNOTE:
Anna Claiborne

2003 - First Network Automation Journey

```
                @subtemp = split(" ", $temp[0]);
                $time = $subtemp[2];
                if(!defined($logStartDate)) { $logStartDate = $time }
                $agmIp = $subtemp[3];
                $zoneName = $subtemp[4];
                $ftype = substr($temp[2], 13, length($temp[2]));
                $temp[1] =~ m/(\w+\-Dynamic\-Filter)/i;
                $faction = $1;

                if($faction eq "Add-Dynamic-Filter") { $policyRule{ $temp[3] }++; }
                elsif($faction eq "Remove-Dynamic-Filter") { $issuer{ $temp[3] }++; }

                $srcIp = substr($temp[4], 15, length($temp[4]));
                $dstPort = $temp[8];
                $count = $temp[11];
                $attackGlobalId = ($temp[12], 17, length($temp[12]));

                $filterAction{ $faction }++;
                $filterType{ $ftype }++;
                $ip{ $srcIp }++;
                $attackId{ $attackGlobalId }++;
                $actionByTime{ "$time:$ftype" }++;
                $totalEntries++;

                print "ftype: $ftype $time\n";
        }
}
$logEndDate = $time;
close(FILE);

$totalSec = timeDeltaInSeconds($logStartDate, $logEndDate);

textResults(\$userSetType, \%filterType, \%ip, \%attackId, \%[illegible], \%[illegible]le, \%[illegible]\%filt[illegible] $totalSec);

sub textResults
{
        my ($userSetType, $filterType, $ip, $attackId, $actionByTime, $policyRule, $issuer, $filterAction, $totalSec) = @_;
        my ($key, $value);
        my $temp = 0;
        my $tempTotal = 0;

        while( ($key, $value) = each(%$filterType) ) { $tempTotal +[illegible]ue; }
        print "There were $tempTotal filters acted on.\n";
        while( ($key, $value) = each(%$filterType) )
        {
                $temp = ($value/$tempTotal)*100;
                print "$key = $value or ".substr($temp,0,4)."%[illegible] all acti[illegible] ta[illegible].\n";
        }

        print "\n\nFilter actions taken:\n";
        while( ($key, $value) = each(%$filterType) )
        {
                print "$value  $key\n";
        }

        print "\n\nTotals by Time\n";
        print "Time period: $totalSec\n";
        print "There were an average of ".substr($tempTotal/$totalSec,0,4)." filter rules changed a second.\n";
        print "There were an average of ".substr($tempTotal/($totalSec/60),0,4)." filter rules changed a minute.\n";
        print "There were an average of ".substr($tempTotal/($totalSec/120),0,4)." filter rules changed an hour.\n";

        print "\n\nTotals By Attack Id\n";
        print "There are ".keys(%$attackId)." unique attacks.\n";

        print "\n\nPolicies being acted on\n";
        while( ($key, $value) = each(%$policyRule) )
        {
                print "$key = $value\n";
        }

        print "\n\nThere are ".keys(%$ip)." unique IPs that have had filters acted on.\n";
        print "The top 10 IPs by activity:\n";
        $temp = 0;
        for my $value ( sort hashValueDesc (keys %$ip) )
        {
                print "$value: $ip{$value}\n";
                $temp++;

                if($temp > 10) { last; }
        }

        #print "<p><b>Lat Long Coordinates</b><br>";

        #while( ($key, $value) = each(%$latlongCoordinates) )
        #{
        #        print "$key = ($value)\n";
        #}
#}

sub timeDeltaInSeconds
{
        my ($logStartDate, $logEndDate) = @_;
        my @temp;
        my ($diffDay, $diffHour, $diffMin, $diffSec, $hour1, $min1, $sec1, $hour2, $min2, $sec2, $totalSec) = 0;
        my ($sec,$min,$hour,$mday,$month,$year,$wday,$yday,$isdst) = localtime(time);
        $year += 1900;
```

Why haven’t we seen full adoption of network automation, yet?

The 3 primary factors influencing network automation adoption:

History

Ecosystem

Perception

History

Who and what is the history of network automation?

The history of telecommunications is complex and contradictory:

Monopolies

Anti-competitive behavior

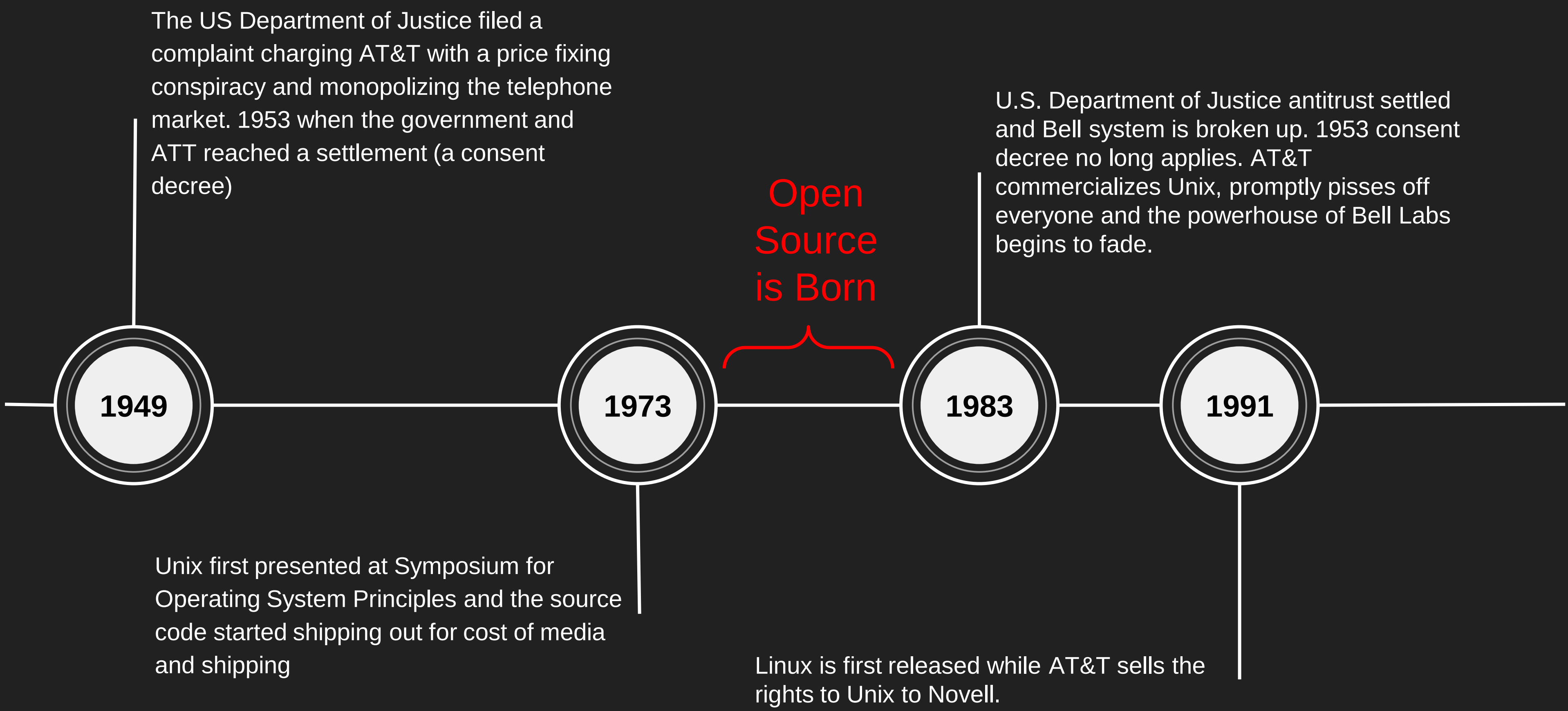
Origin of open source

Revolutionary innovation

The history of telecom is the history of the Bell System because it was a monopoly.

1953 consent decree: the origin of open source software and community

How government regulation and AT&T lawyers inadvertently started one of the most important movements in tech.



1949

The US Department of Justice filed a complaint charging AT&T with a price fixing conspiracy and monopolizing the telephone market. 1953 when the government and ATT reached a settlement (a consent decree)

1973

Unix first presented at Symposium for Operating System Principles and the source code started shipping out for cost of media and shipping

Open Source is Born

1983

U.S. Department of Justice antitrust settled and Bell system is broken up. 1953 consent decree no long applies. AT&T commercializes Unix, promptly pisses off everyone and the powerhouse of Bell Labs begins to fade.

1991

Linux is first released while AT&T sells the rights to Unix to Novell.

Bell Labs Innovation

1949 - Transistors: The foundation of solid state electronics and all technology we enjoy today.

1947 - Cellular concept: The first cellular network was installed in Chicago in the 1970s.

1954 - Solar cell: The first conversion of light to electrical energy.

1958 - Laser: Which as first called an optical maser…

1962 - Telstar 1 satellite: The first communications satellite in orbit enabling calls world wide.

Programming languages: B, C, C++, S, SNOBOL, and AWK

Radio astronomy: Developed at Bell Labs to find static in overseas radio communications and now powers the search to extra-terrestrial life.

Bell Labs has also produced scientist who earned 10 Nobel Prizes and 5 Turing Awards.

“The most impactful product Cisco ever released was certifications. They convinced an entire generation that the right way to do things was to type commands into a network device and feel special because of it.” - *Former Google Employee Who Was There*

Result: the perfect storm to accelerate the compute automation ecosystem and delay network automation ecosystem

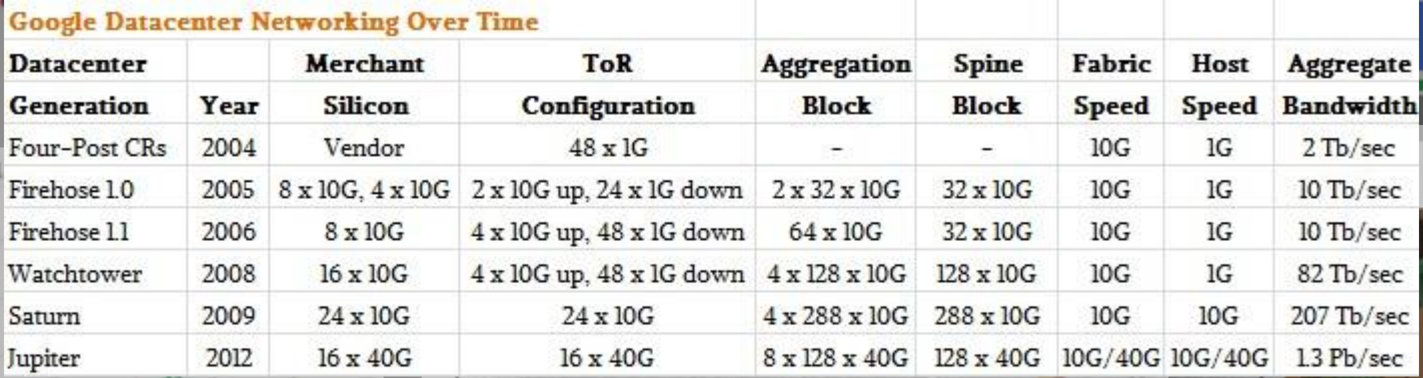
Compute

- Open Source/Free OS independent of hardware platform
- Open source software ecosystem
- Community support

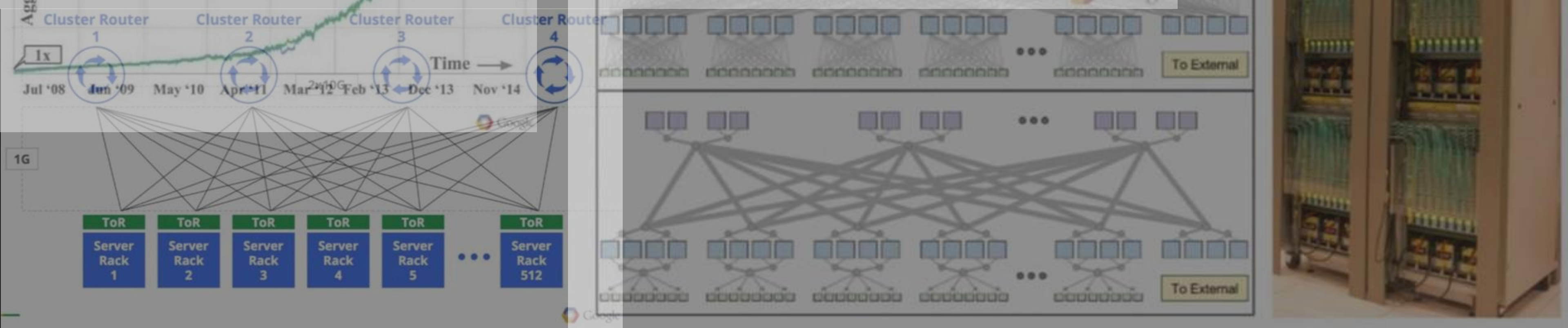
Network

- Closed Source/Paid OS bound to hardware platform
- Limited vendor software ecosystem
- Vendor support

Google Escapes CLI Tyranny

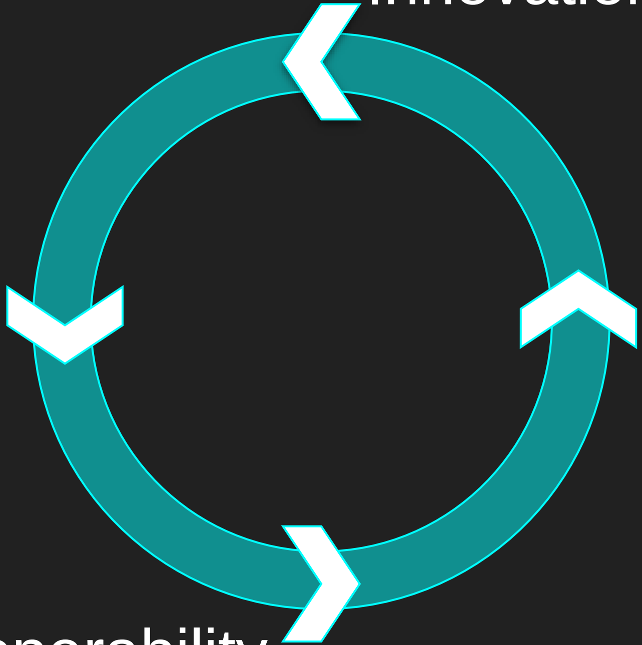


Google Datacenter Networking Over Time								
Datacenter		Merchant	ToR	Aggregation	Spine	Fabric	Host	Aggregate
Generation	Year	Silicon	Configuration	Block	Block	Speed	Speed	Bandwidth
Four-Post CRs	2004	Vendor	48 x 1G	-	-	10G	1G	2 Tb/sec
Firehose 1.0	2005	8 x 10G, 4 x 10G	2 x 10G up, 24 x 1G down	2 x 32 x 10G	32 x 10G	10G	1G	10 Tb/sec
Firehose 1.1	2006	8 x 10G	4 x 10G up, 48 x 1G down	64 x 10G	32 x 10G	10G	1G	10 Tb/sec
Watchtower	2008	16 x 10G	4 x 10G up, 48 x 1G down	4 x 128 x 10G	128 x 10G	10G	1G	82 Tb/sec
Saturn	2009	24 x 10G	24 x 10G	4 x 288 x 10G	288 x 10G	10G	10G	207 Tb/sec
Jupiter	2012	16 x 40G	16 x 40G	8 x 128 x 40G	128 x 40G	10G/40G	10G/40G	1.3 Pb/sec



Ecosystem

The ecosystem flywheel effect



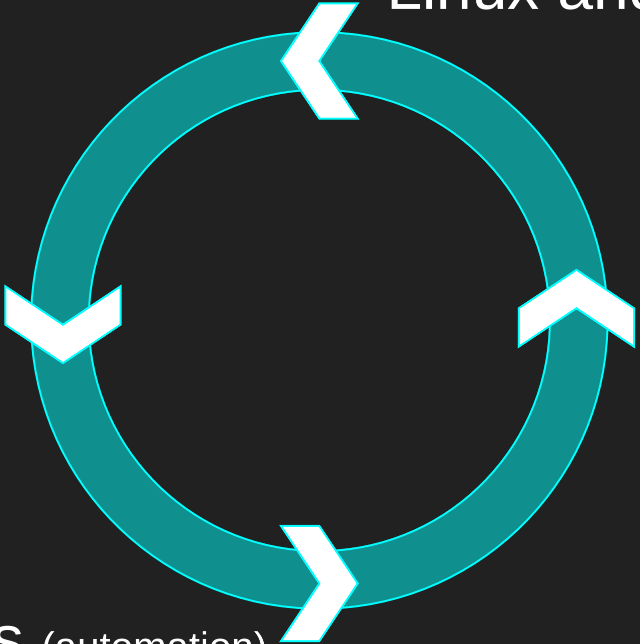
Innovation

Investment

Lower Costs

Interoperability

The ecosystem flywheel effect applied to compute and software



Linux and open source

Cheaper hardware

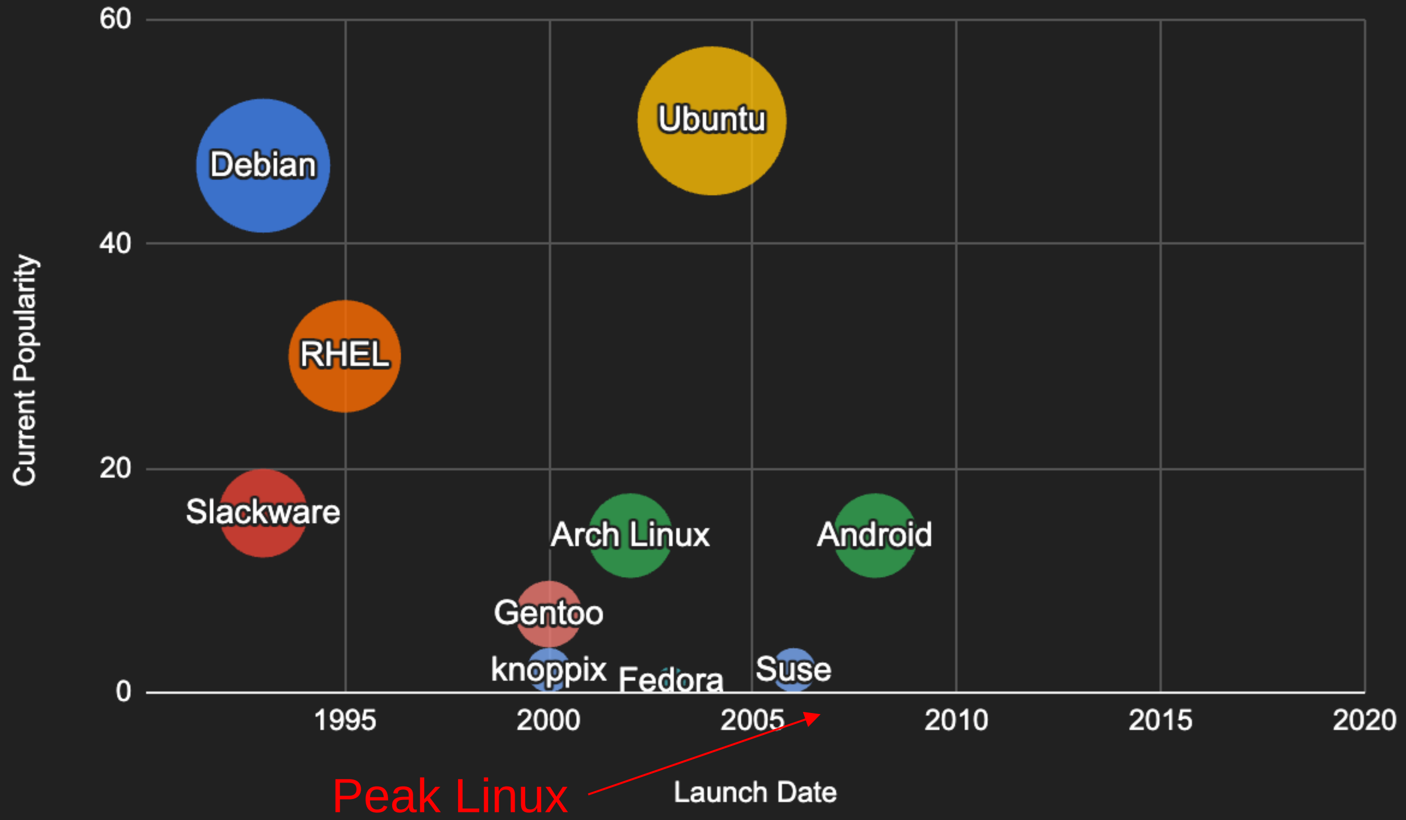
APIs (automation)

Venture capital

Linux is historically is a huge innovation and driver of innovation, automation and lowering costs in the compute space because:

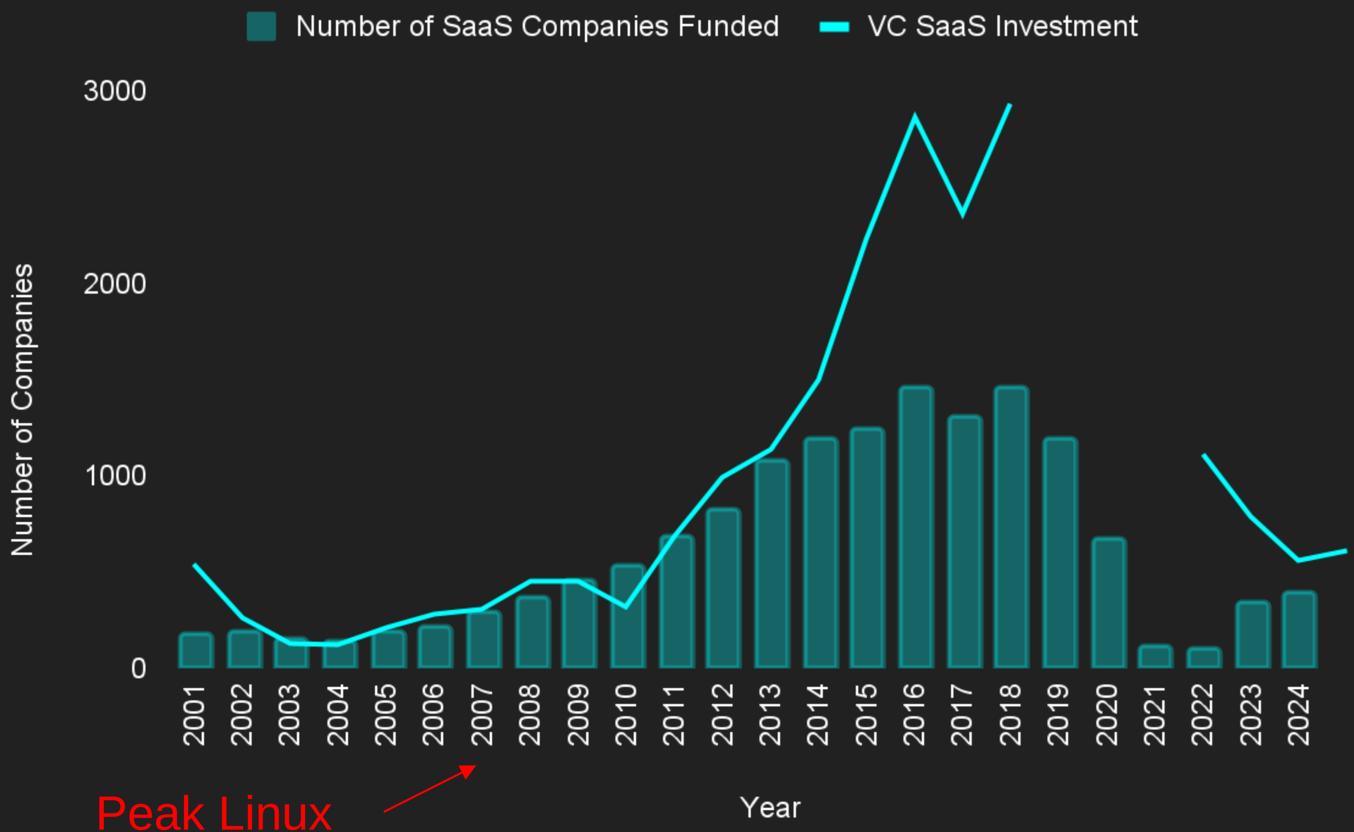
1. It’s open source.
2. It’s free.
3. It’s supported by the user community.
4. It runs on everything (present and future).

Linux reaching innovation critical mass



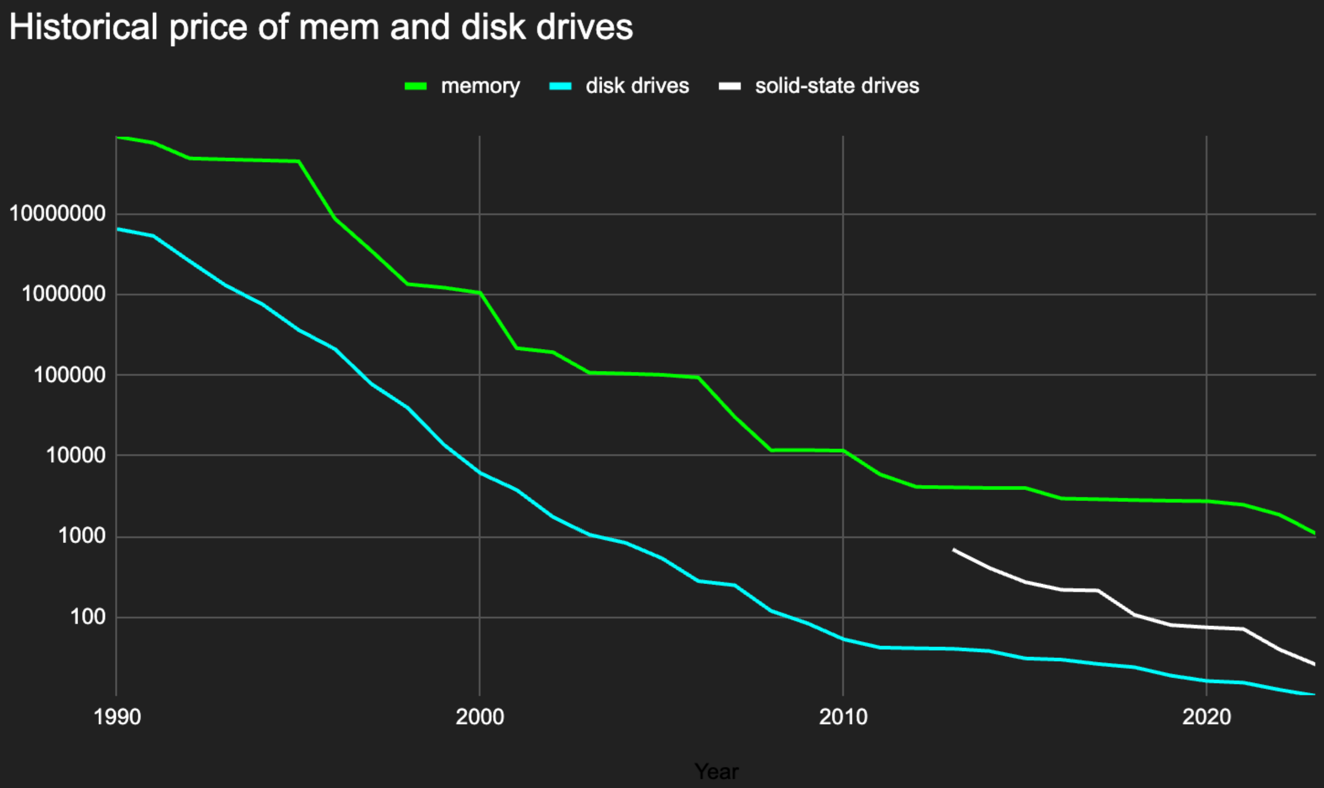
Peak Linux

Investment following innovation

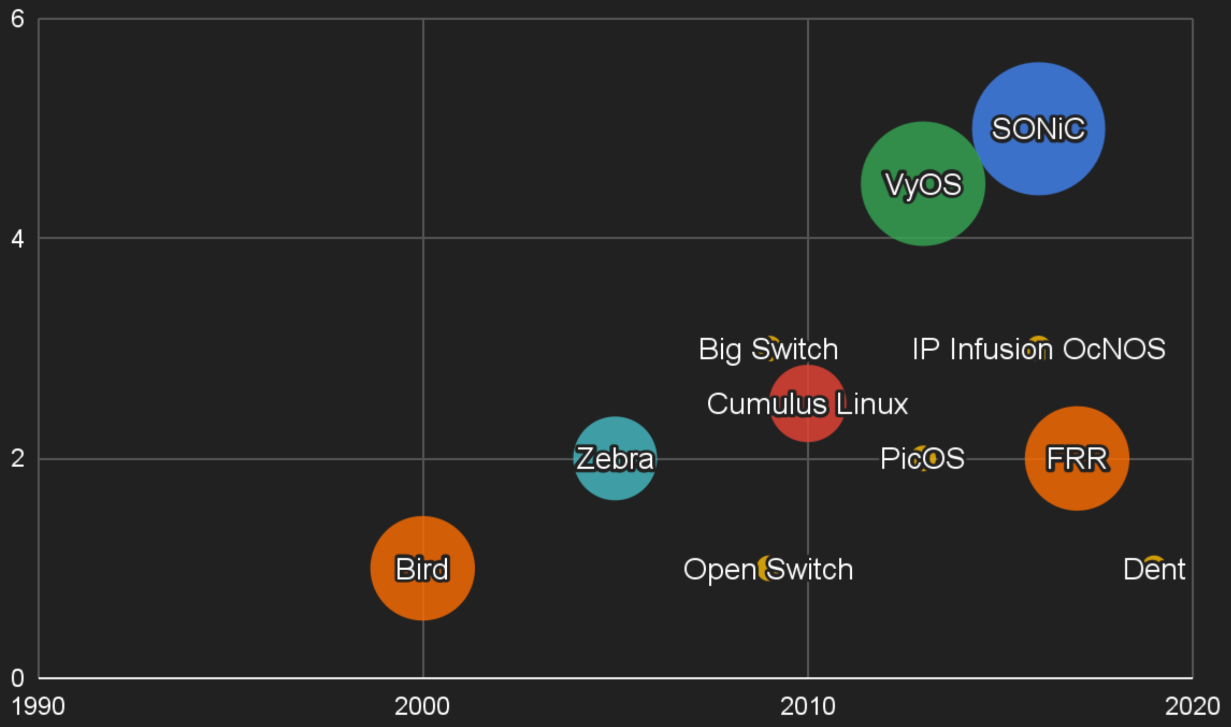


Peak Linux

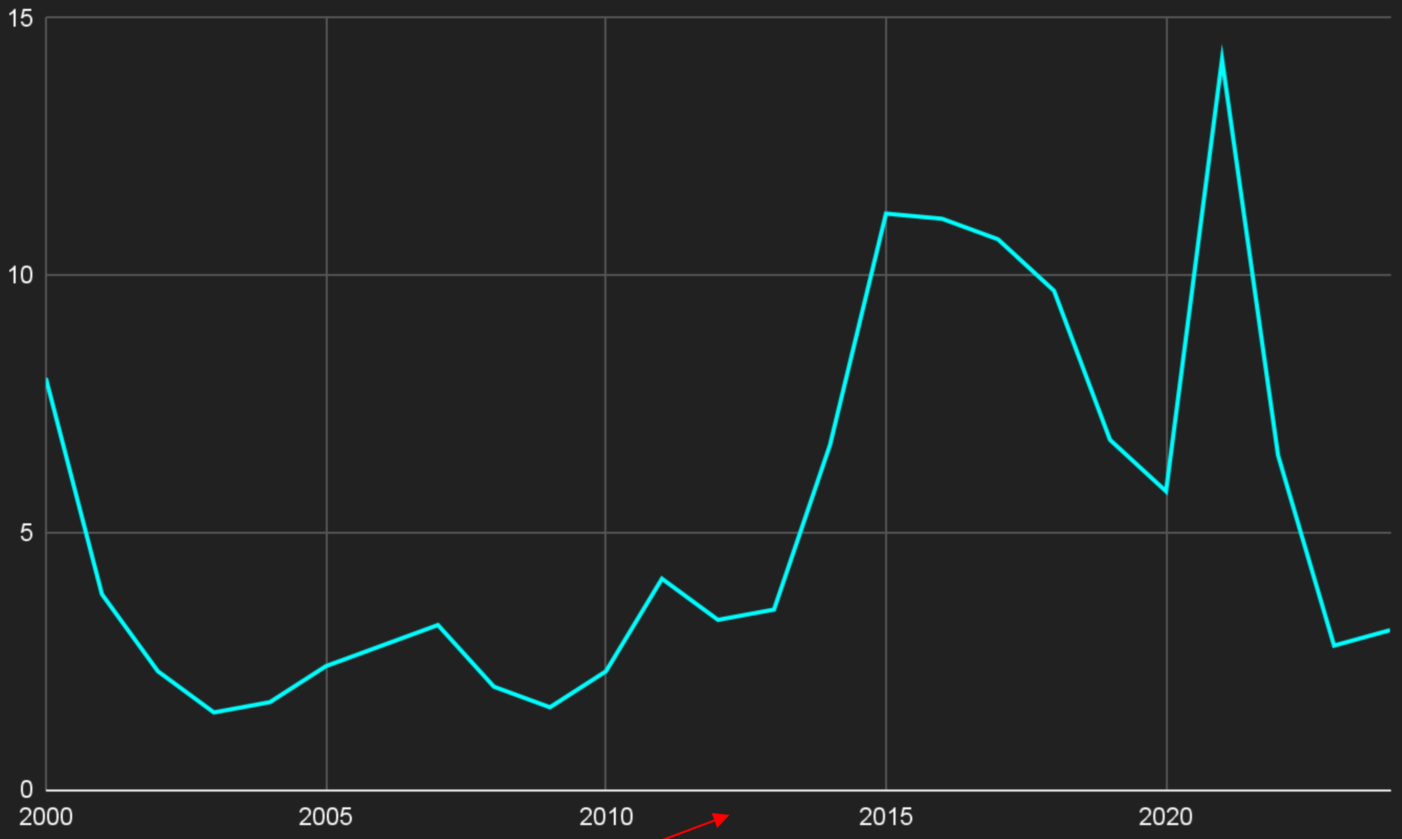
Ever declining costs of compute



Network Operatings Systems reaching innovation critical mass?

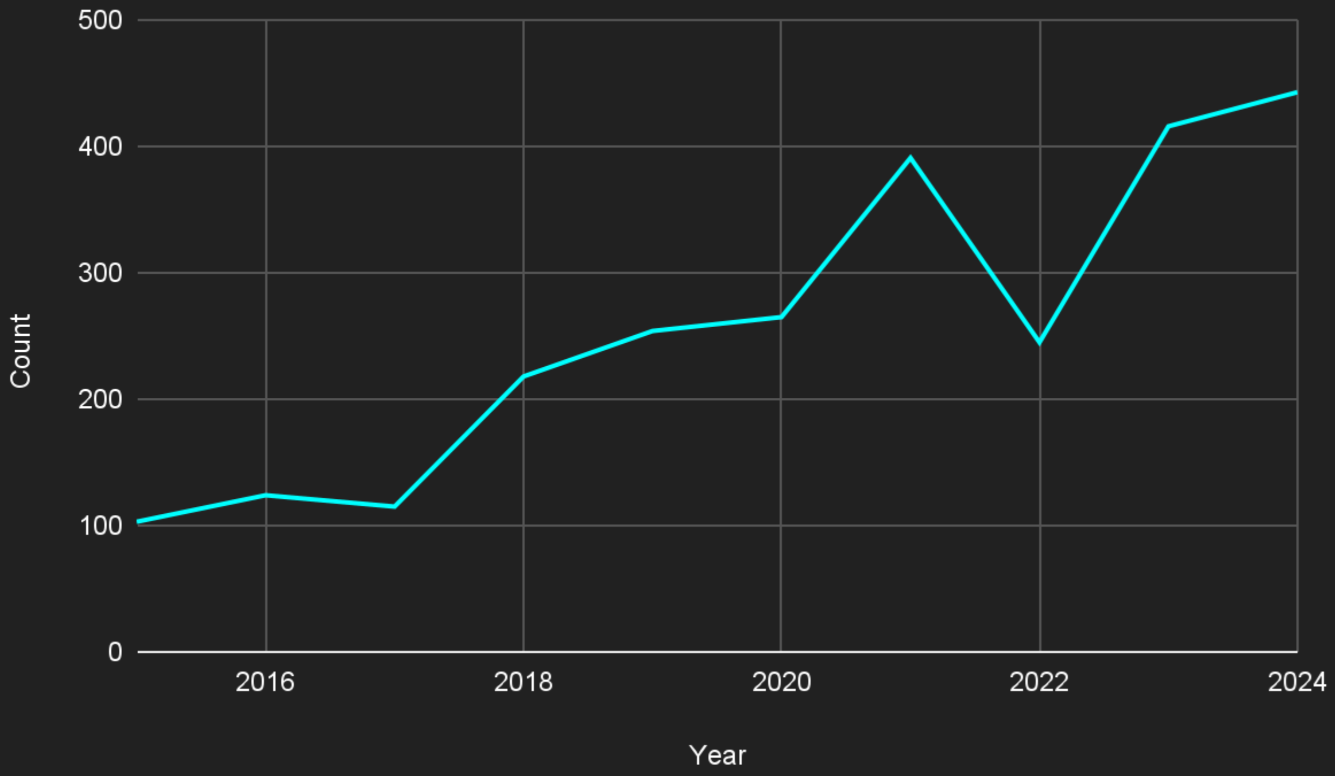


VC Investment in Telecommunications



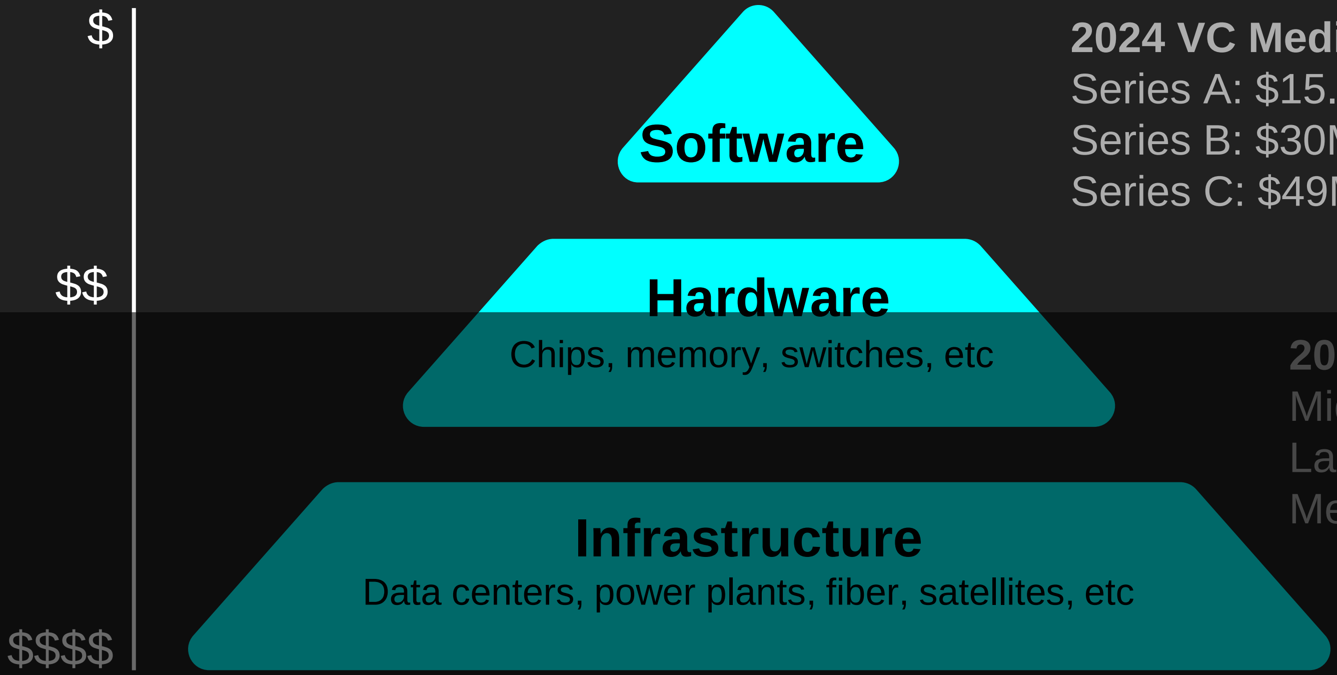
Peak NOS?

Network Automation Repos on Github



7355 automated food delivery companies and we still don’t have network automation….

Technology Investment Hierarchy



Software

Hardware

Chips, memory, switches, etc

Infrastructure

Data centers, power plants, fiber, satellites, etc

2024 VC Median Investments
Series A: $15.1M
Series B: $30M
Series C: $49M

2024 Private Equity Buckets
Middle Market: $50M-$500M
Large: $500M-$5B
Mega: >$5B

Different investor profiles yield different results

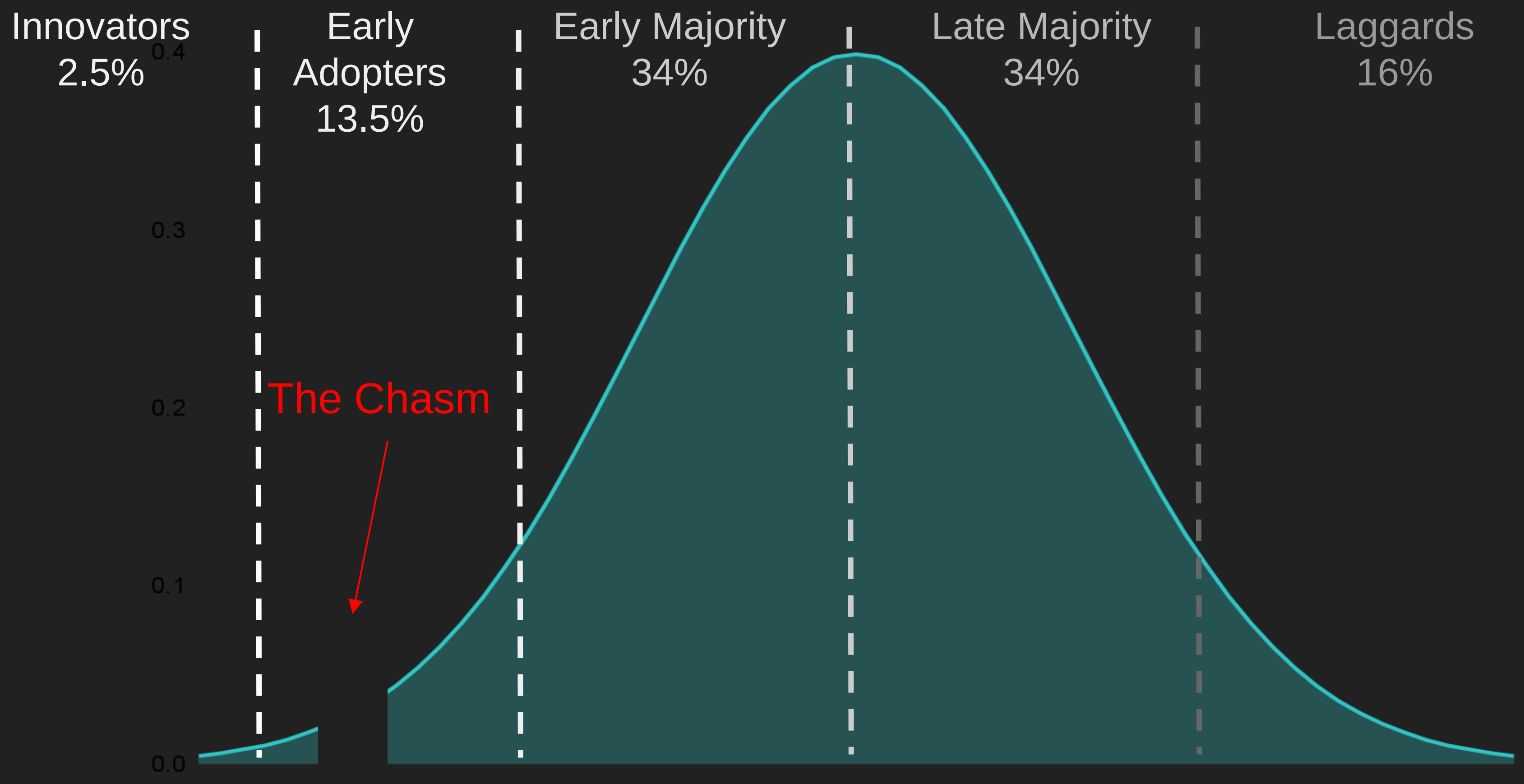
	Venture Capital	Private Equity
Target investments	New tech and growth industries	Anything
Percentage acquired	Minority stake	>51% for controlling interest
Investment Dollar Amounts	Smaller	Larger
Deal structures	Equity	Equity and debt
Company stage	Immature	Mature
Risk Tolerance	High - many small bets and expect few to yield big returns	Low - few very large bets and expect a return on nearly all
Operational Involvement	Low	High
Value Creation	Growth	M&A, Financial Engineering, Growth

Perception

“Just 5 more minutes…”

Every new “revolution” takes much longer than it seems, and it also impossible tell where you are in much of that journey.

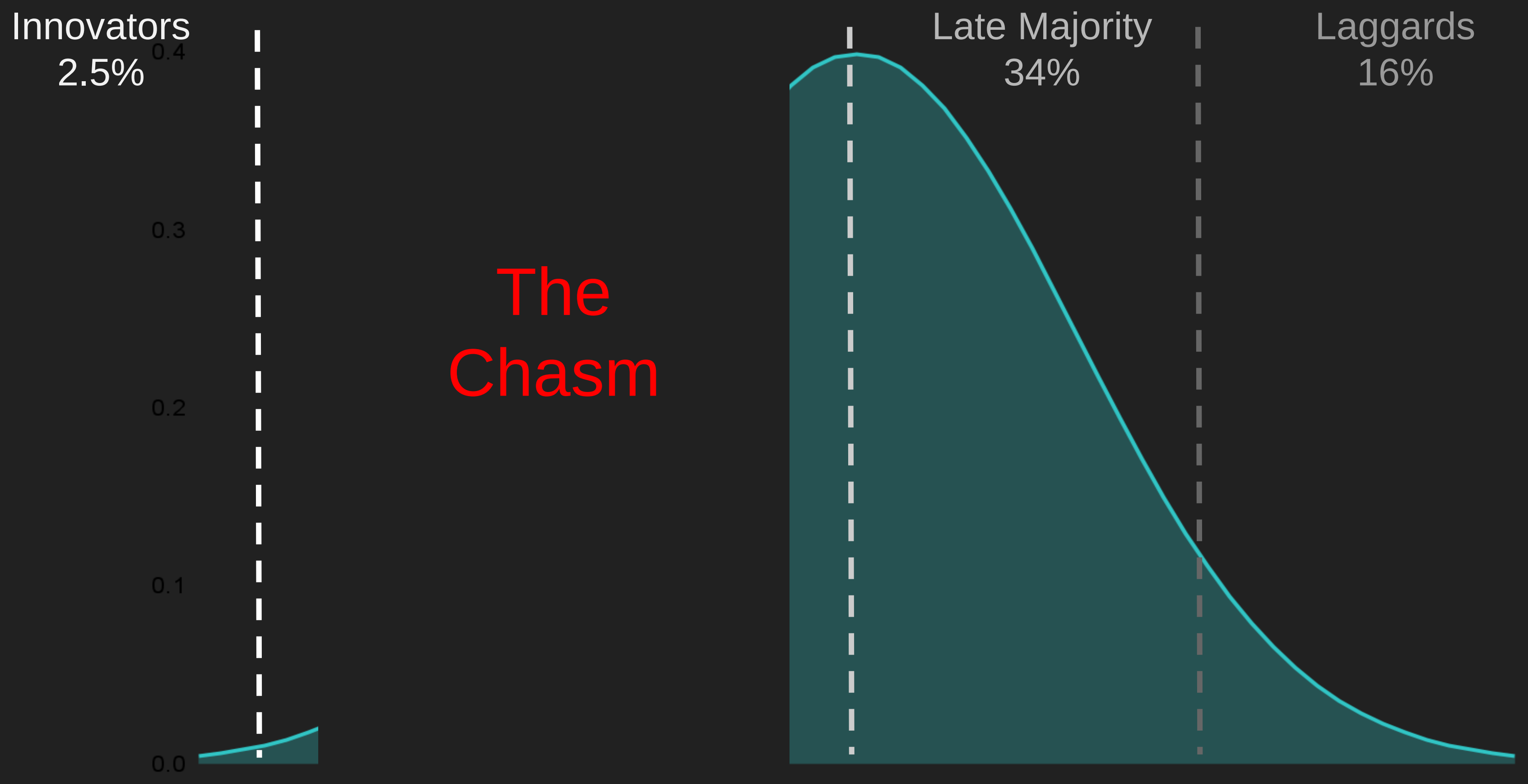
Innovation Adoption Curve



"There's just not that many videos I want to watch."

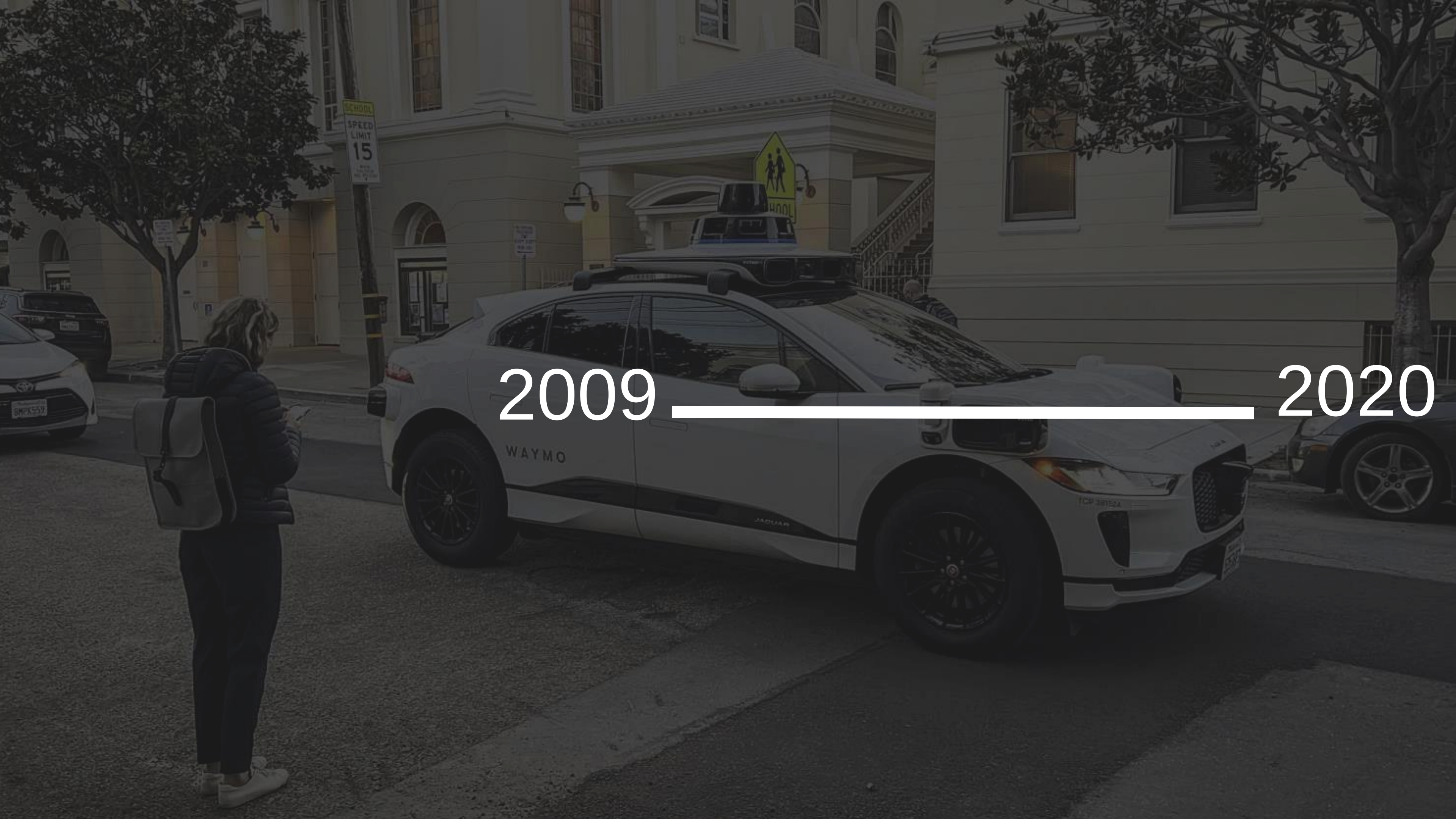
— CTO and co-founder expressing concerns about his company’s long term viability

Realistic Innovation Adoption Curve

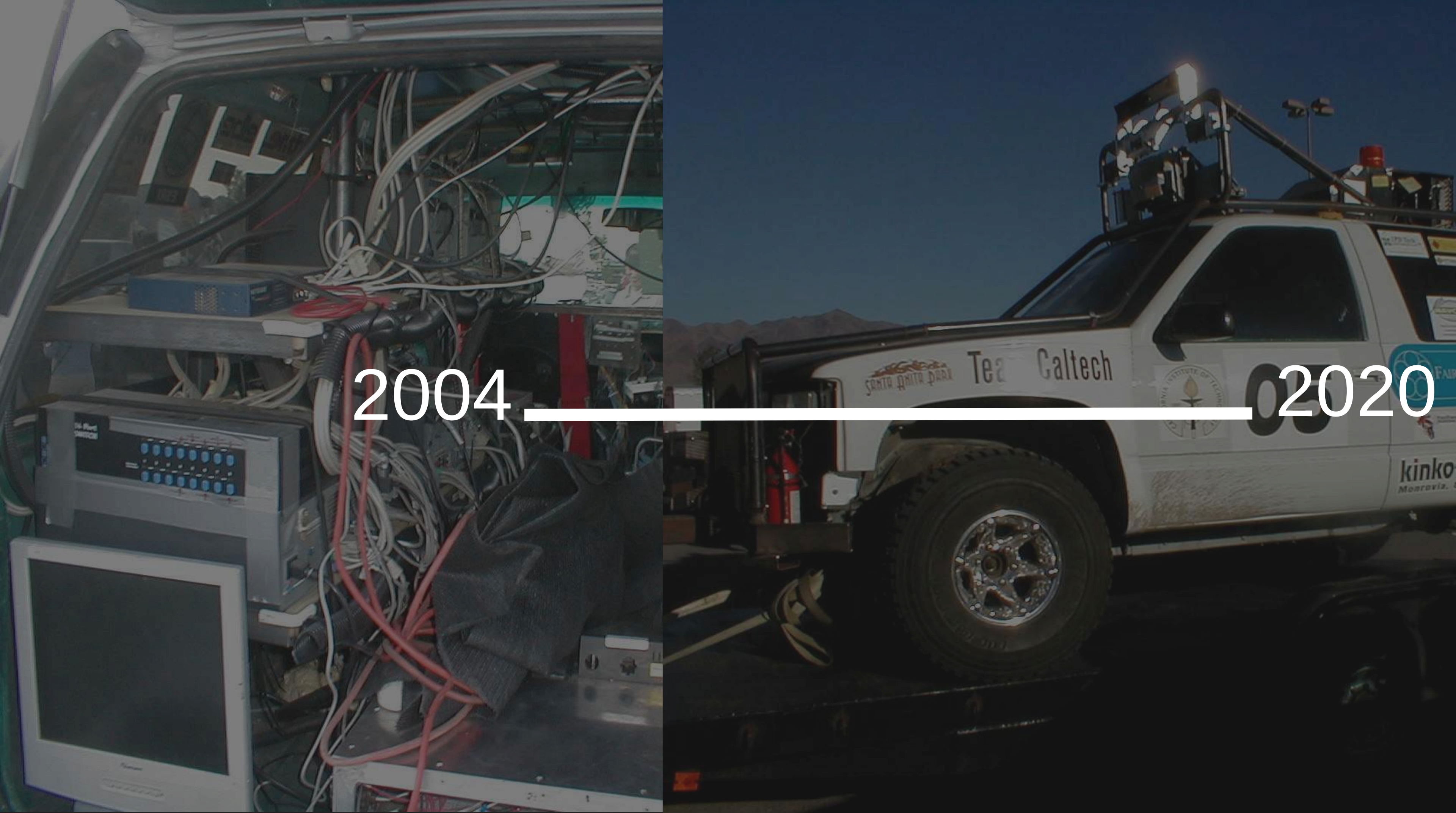


Technology evolves slower than its perceived pace.

And everything we have today is built on the backs of giants.

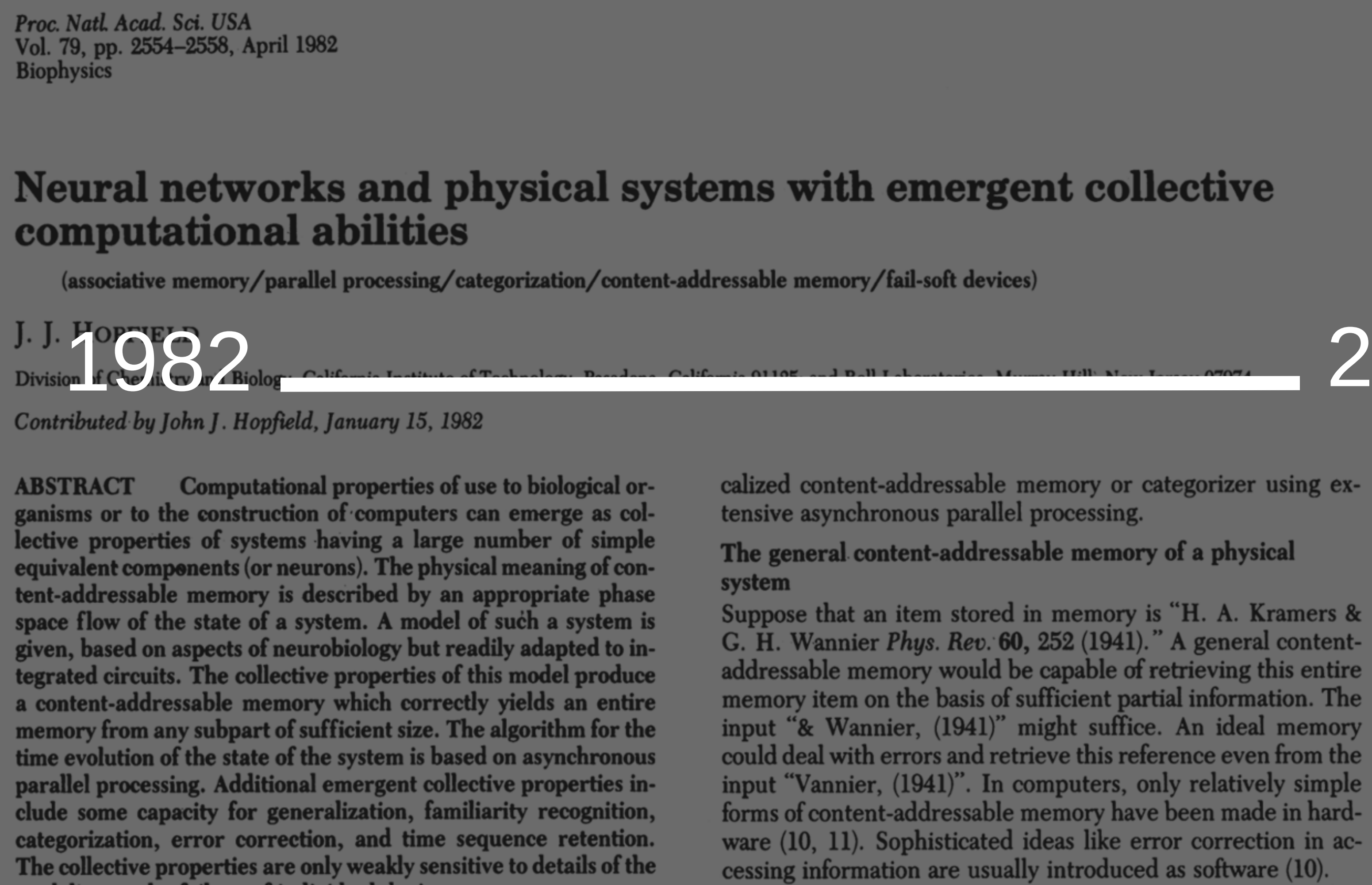


2009 — 2020



2004

2020



1982

2020

Proc. Natl. Acad. Sci. USA
Vol. 79, pp. 2554–2558, April 1982
Biophysics

Neural networks and physical systems with emergent collective computational abilities

(associative memory/parallel processing/categorization/content-addressable memory/fail-soft devices)

J. J. Hopfield

Division of Chemistry and Biology, California Institute of Technology, Pasadena, California 91125; and Bell Laboratories, Murray Hill, New Jersey 07974

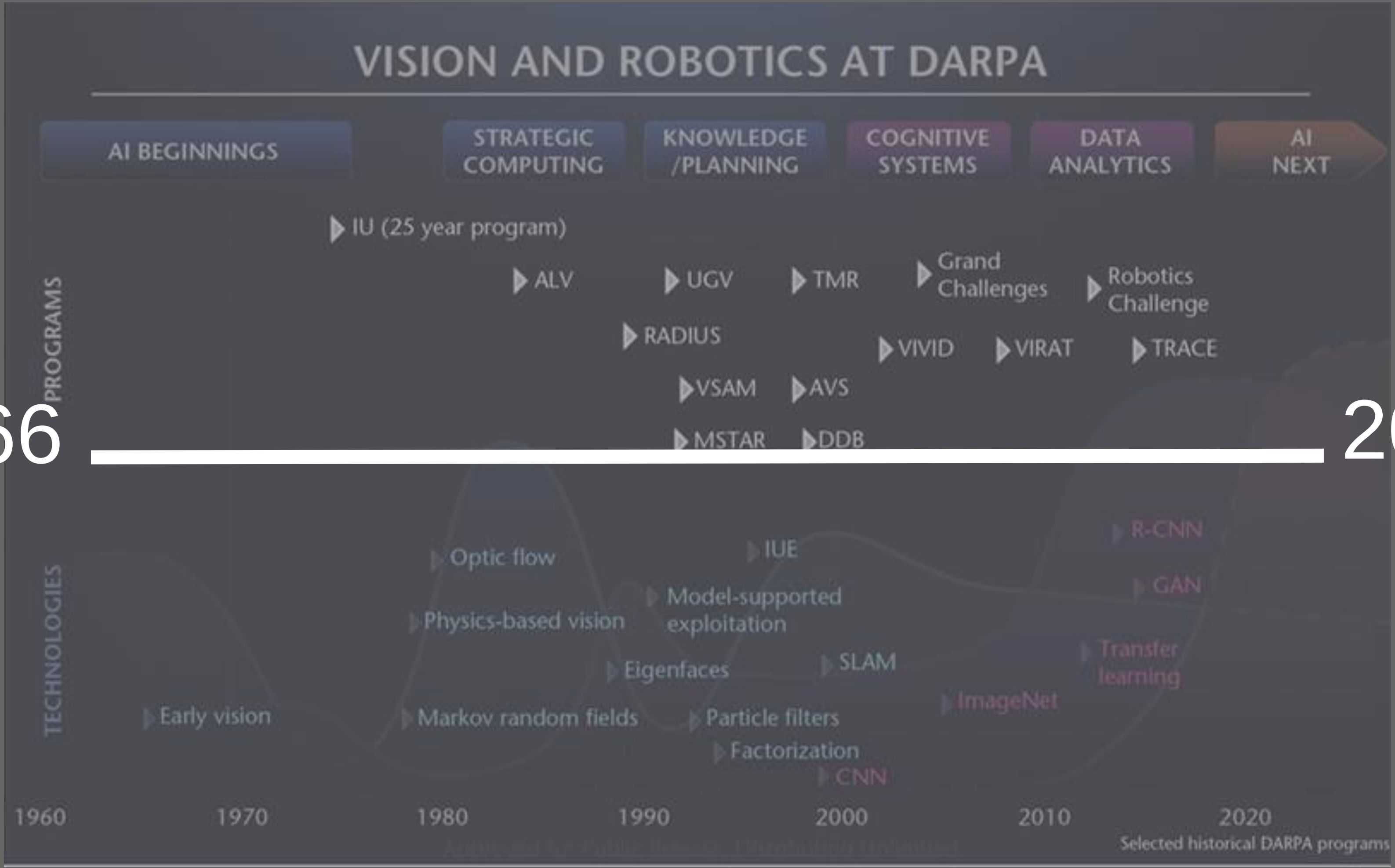
Contributed by John J. Hopfield, January 15, 1982

ABSTRACT Computational properties of use to biological organisms or to the construction of computers can emerge as collective properties of systems having a large number of simple equivalent components (or neurons). The physical meaning of content-addressable memory is described by an appropriate phase space flow of the state of a system. A model of such a system is given, based on aspects of neurobiology but readily adapted to integrated circuits. The collective properties of this model produce a content-addressable memory which correctly yields an entire memory from any subpart of sufficient size. The algorithm for the time evolution of the state of the system is based on asynchronous parallel processing. Additional emergent collective properties include some capacity for generalization, familiarity recognition, categorization, error correction, and time sequence retention. The collective properties are only weakly sensitive to details of the

calized content-addressable memory or categorizer using extensive asynchronous parallel processing.

The general content-addressable memory of a physical system

Suppose that an item stored in memory is "H. A. Kramers & G. H. Wannier *Phys. Rev.* **60**, 252 (1941)." A general content-addressable memory would be capable of retrieving this entire memory item on the basis of sufficient partial information. The input "& Wannier, (1941)" might suffice. An ideal memory could deal with errors and retrieve this reference even from the input "Vannier, (1941)". In computers, only relatively simple forms of content-addressable memory have been made in hardware (10, 11). Sophisticated ideas like error correction in accessing information are usually introduced as software (10).

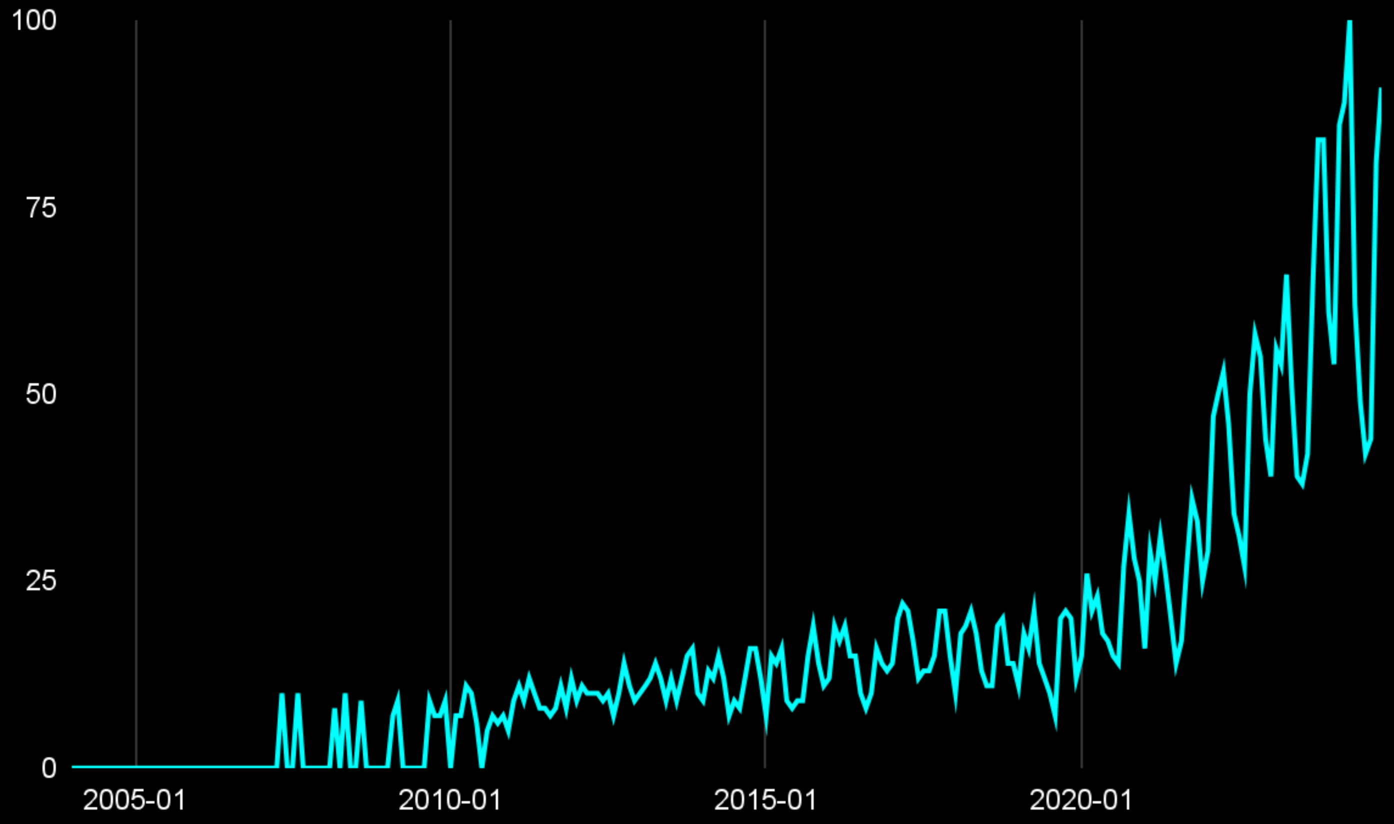

VISION AND ROBOTICS AT DARPA
AI BEGINNINGS
STRATEGIC COMPUTING
KNOWLEDGE /PLANNING
COGNITIVE SYSTEMS
DATA ANALYTICS
AI NEXT
PROGRAMS
IU (25 year program)
ALV
UGV
TMR
Grand Challenges
Robotics Challenge
RADIUS
VIVID
VIRAT
TRACE
VSAM
AVS
MSTAR
DDB
TECHNOLOGIES
R-CNN
Optic flow
IUE
GAN
Model-supported exploitation
Physics-based vision
Transfer learning
Eigenfaces
SLAM
ImageNet
Early vision
Markov random fields
Particle filters
Factorization
CNN
1960
1970
1980
1990
2000
2010
2020
Selected historical DARPA programs
Approved for Public Release, Distribution Unlimited

1966 2020

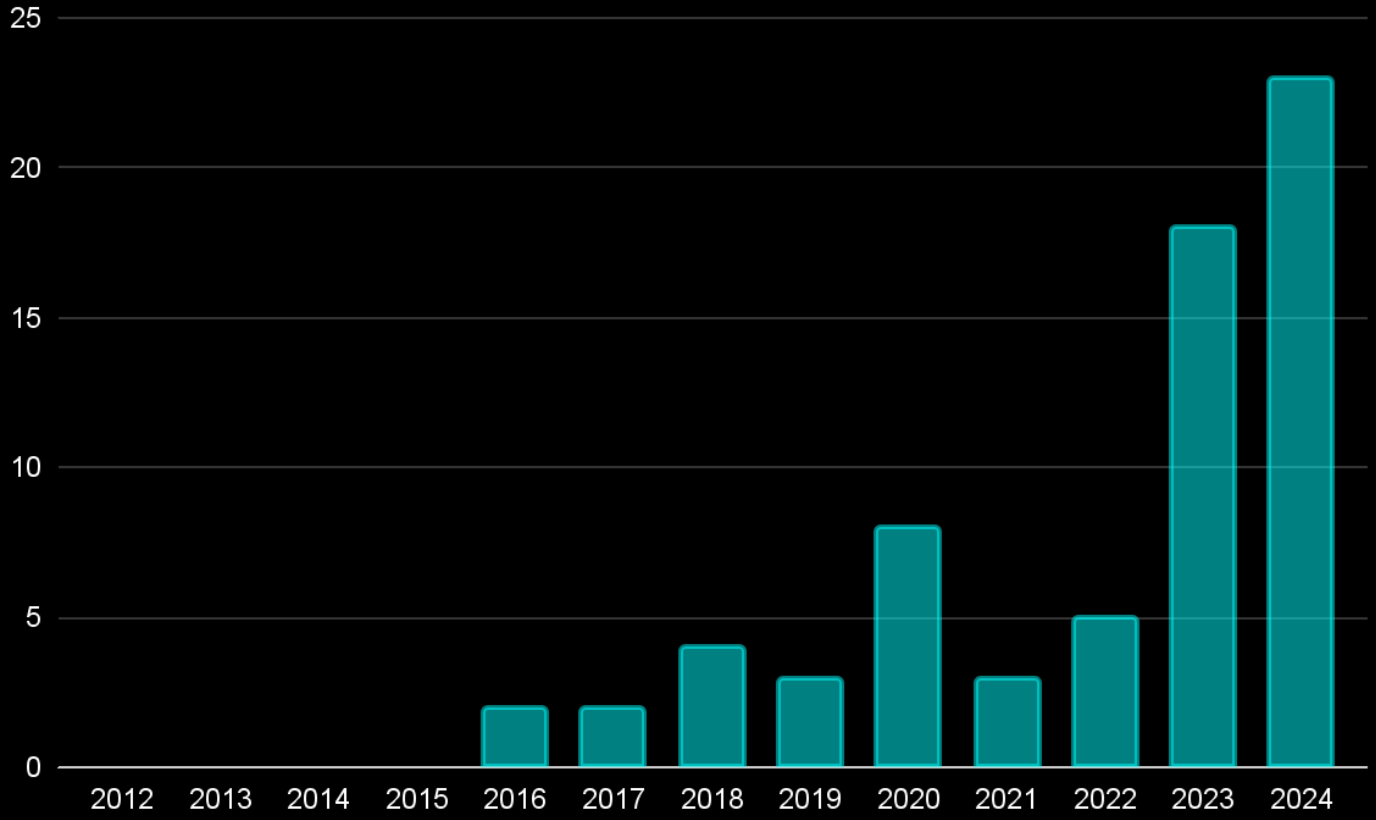
Additional Signals

If it’s difficult to see objectively where we are on the adoption curve, what other signals can we look at to determine adoption and ecosystem growth?

Google Trends - Network as a Service



Books Published on Network Automation



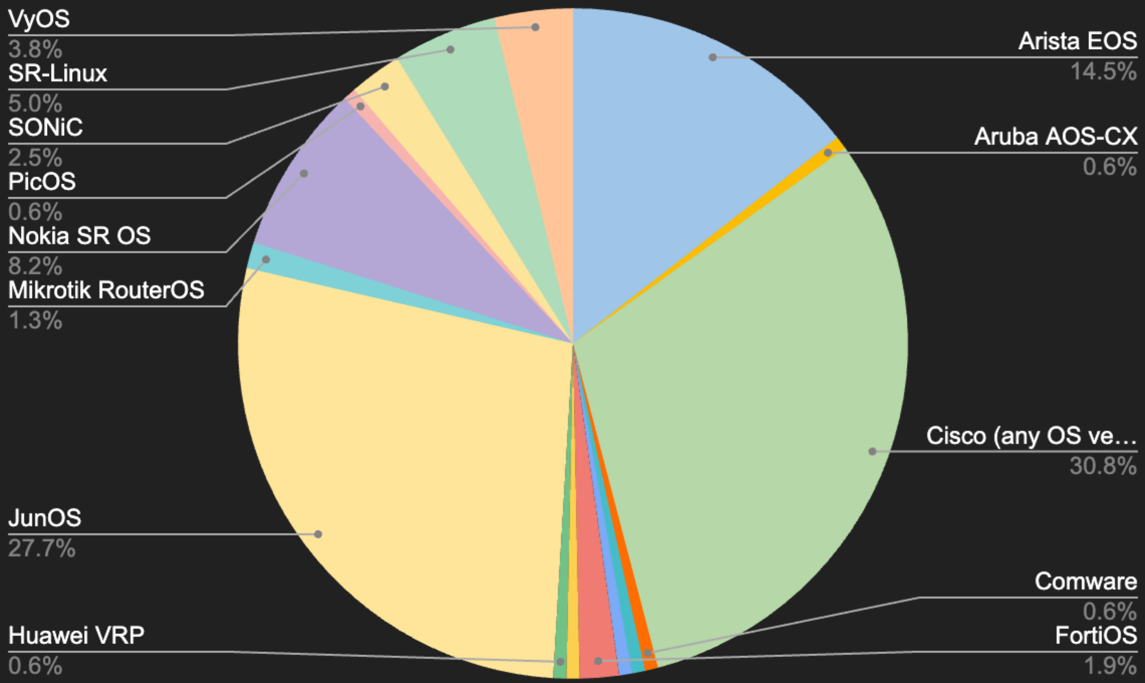
Hold onto your butts.

Operating Systems in use

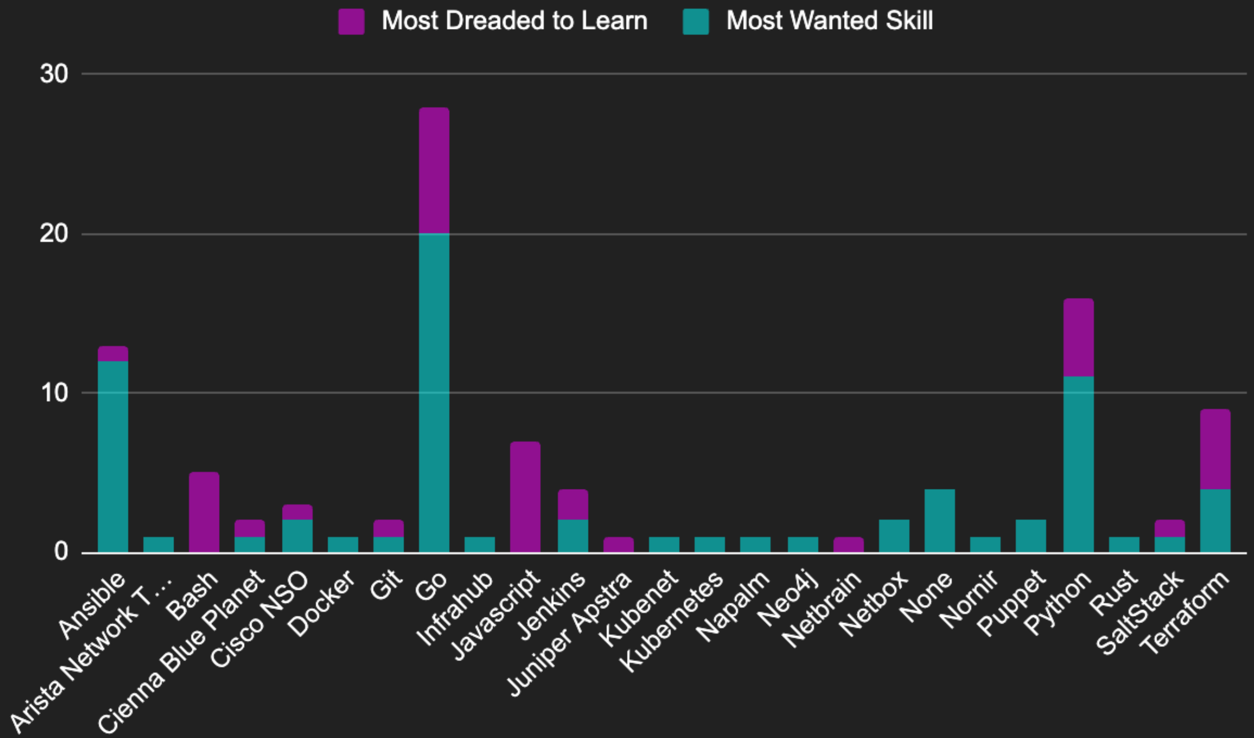
Key Findings:

~13% of operating systems used everyday for networking activities are open source

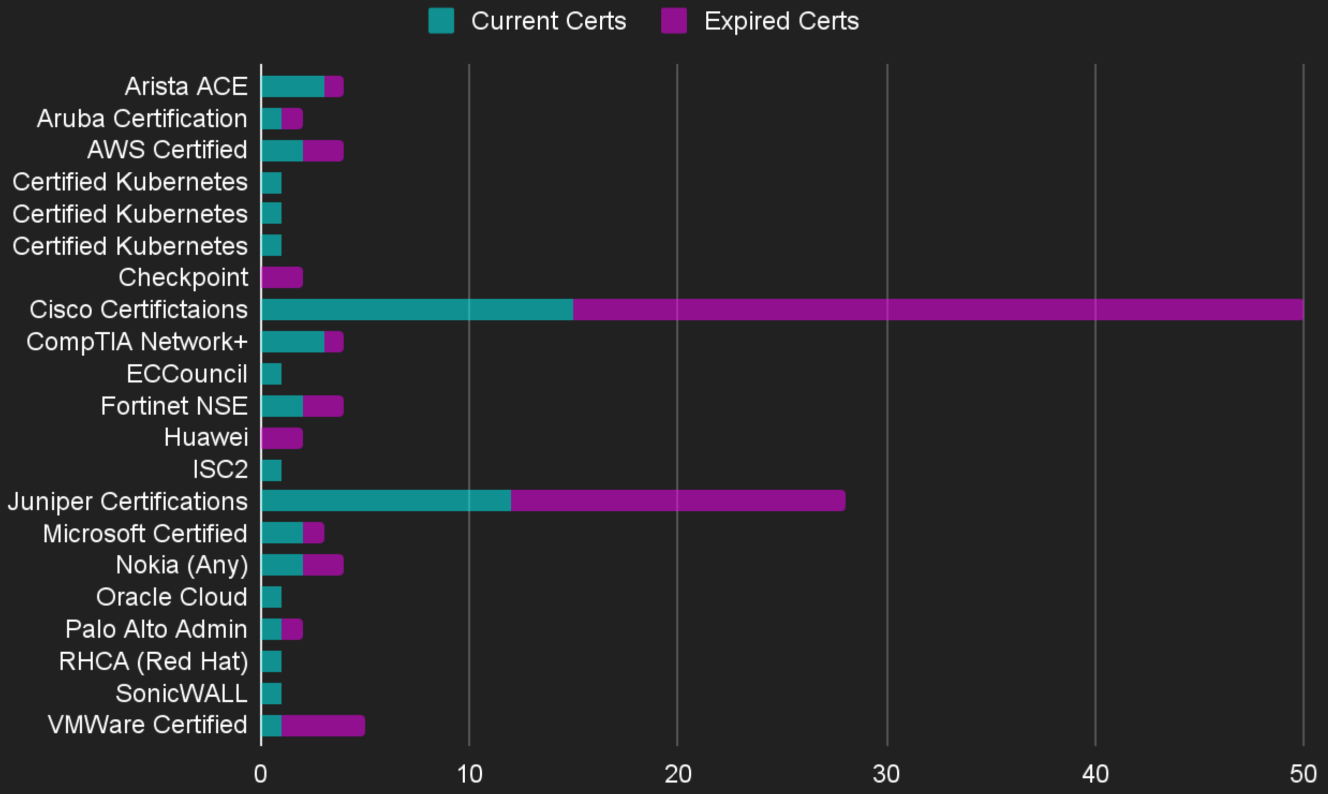
10 years ago this would have been considerably less



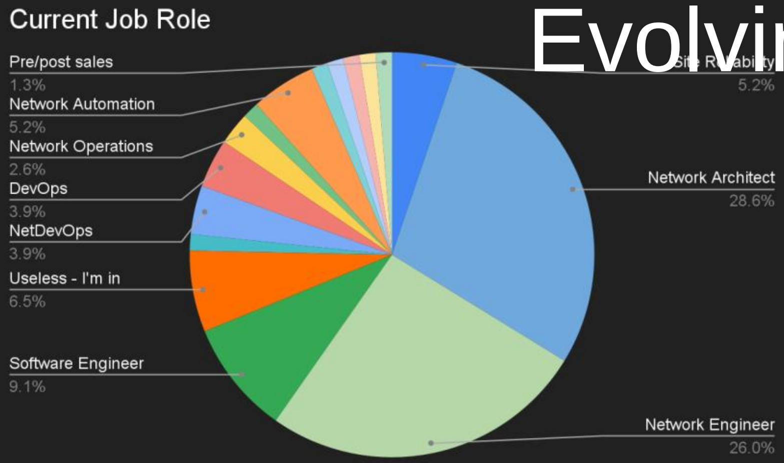
Which skills are hot, and which are not

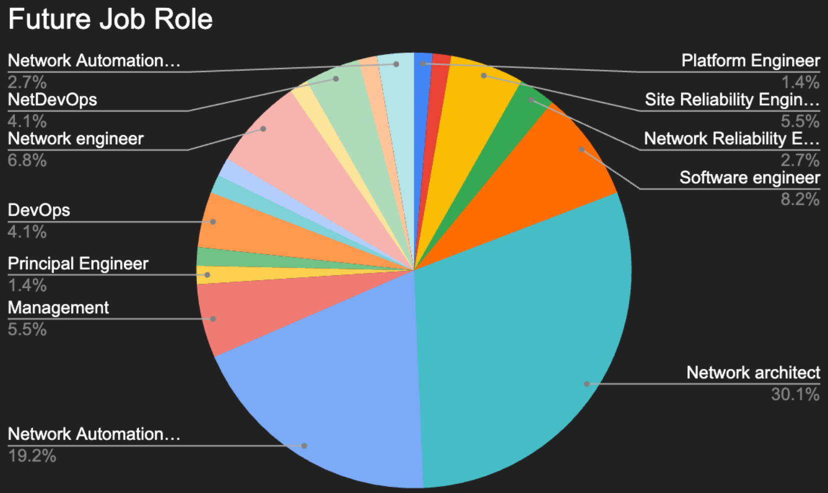


Certifications



Evolving Job Roles





Key Findings:

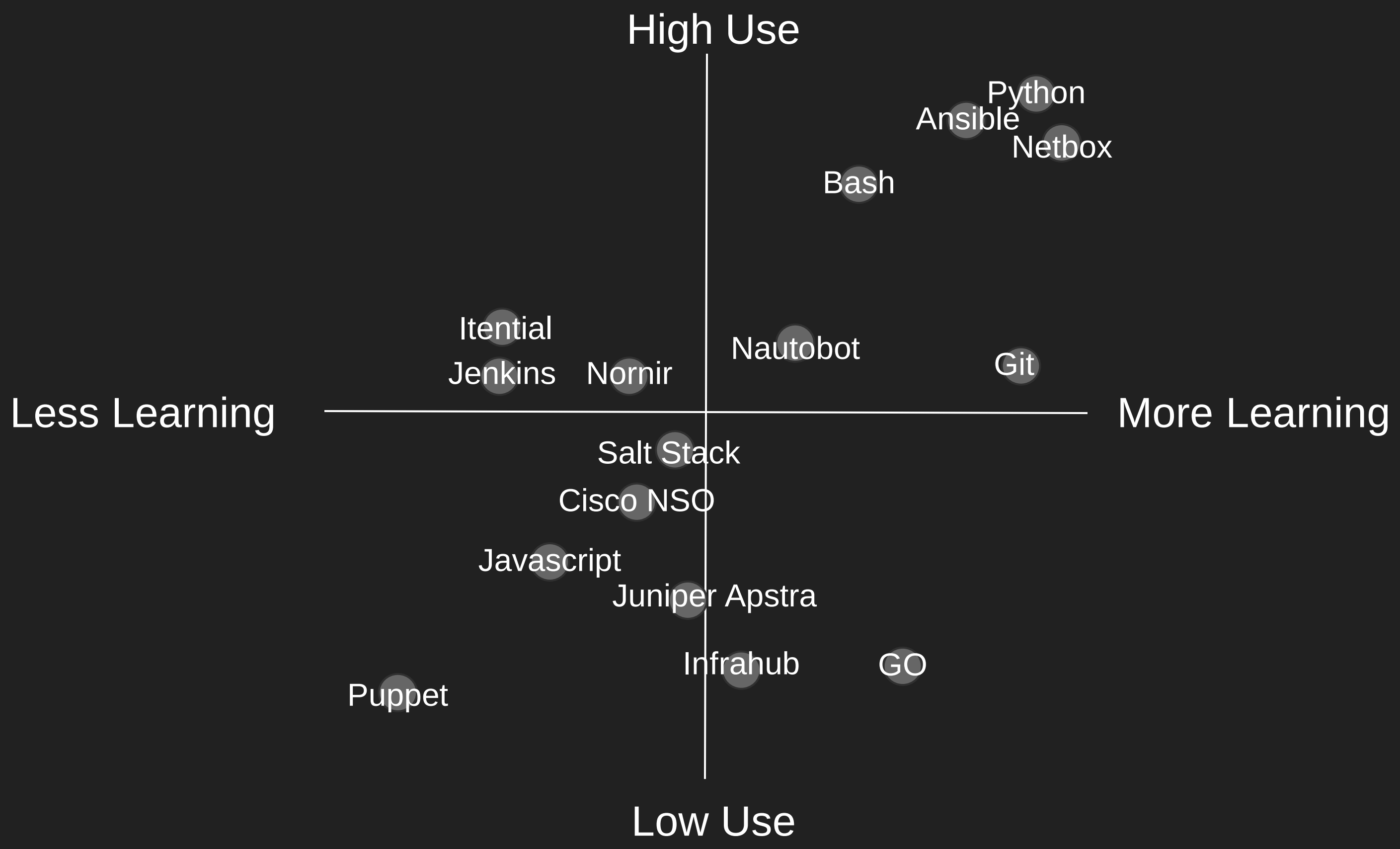
20% shift away from traditional job titles towards hybrid job titles

14% increase looking to move to Network Automation Engineer

2.8% added the same unlisted role: Network Automation Architect

0% selected Network Reliability Engineering for either current, or desired role

Tooling Trends



Barriers

- Networking software began without programmatic interfaces in mind, making automation difficult and delaying uptake
- Investment in innovation around network infrastructure and some hardware is large and requires a risk averse class of investors
- Network automation is in the middle of adoption curve, so it feels farther away than it is

Accelerators

- Increasing usage of open source operating systems for network devices and open source software being created in the network ecosystem
- People closest to the problem are developing crossover skill sets in software and network that will help further increase system interoperability and automation
- Network automation is in the middle of adoption curve, so it feels farther away than it is

How can you accelerate network automation up the adoption curve?

✓ Start an open source project or contribute to one
✓ Start a software company
✓ Learn crossover skill sets
✓ Ensure everything you create is programmatically accessible and demand it from your vendors

Viva la network automation revolution!

We must keep pushing the ecosystem flywheel with maximum force to reach full adoption of network automation.