Web 2050: Our Decentralized Future

(Note: This is a slightly expanded version of the presentation "Web 2050", given on May 2, 2023 at The ACM Web Conference 2023, AT&T Hotel and Conference Center, Austin, TX)

Thank you so much for being here today, and thank you to Bebo White, Mark Schueler, Wendy Hall, and the ACM. ACM conferences have played a pivotal role in the history of the World Wide Web, and I'm honored to be here.

My name is Kevin Hughes, and I've been a Web developer for 30 years. You could say I'm not exactly old school, since - I helped build the school. But unfortunately I'm here to talk about the problems the Web faces today, and with the help of a little bit of history, how we might be able to solve them.
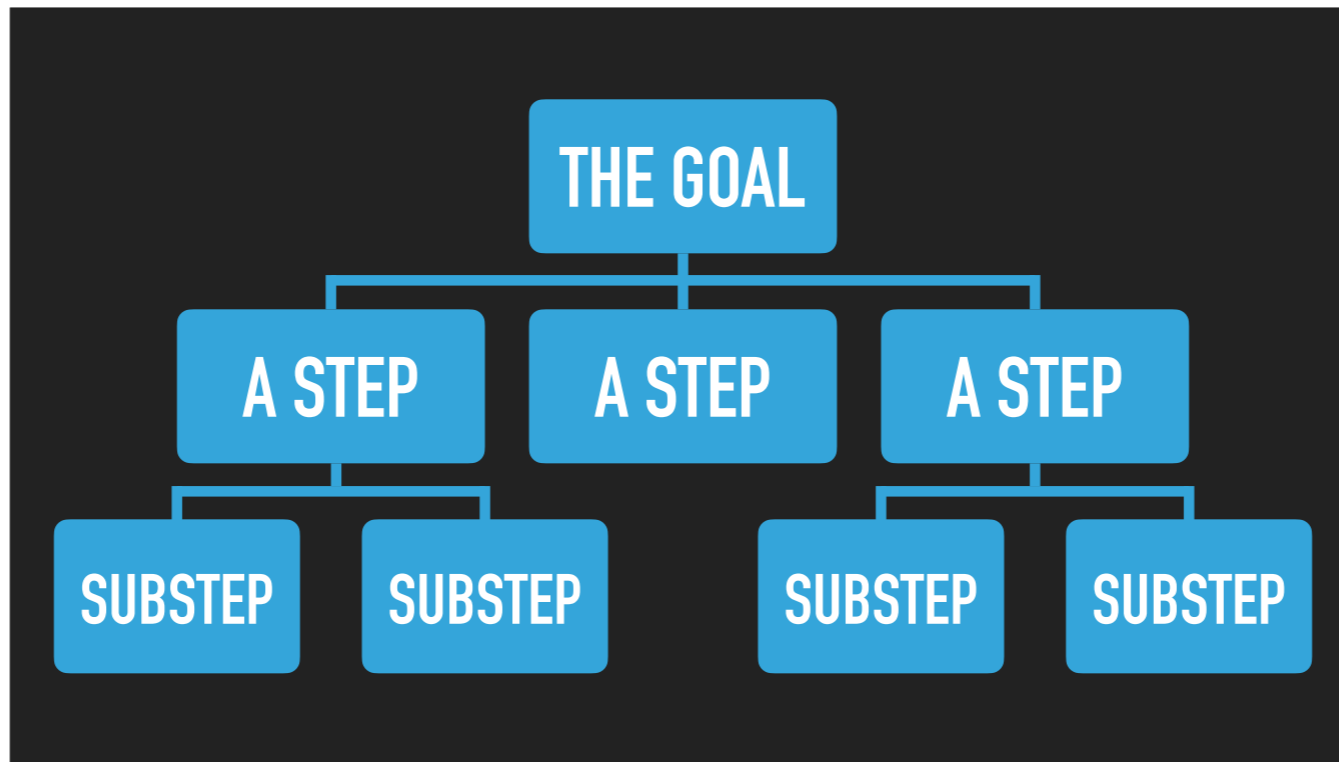
THE GOAL

When you first build a system you always start with some sort of initial requirements.
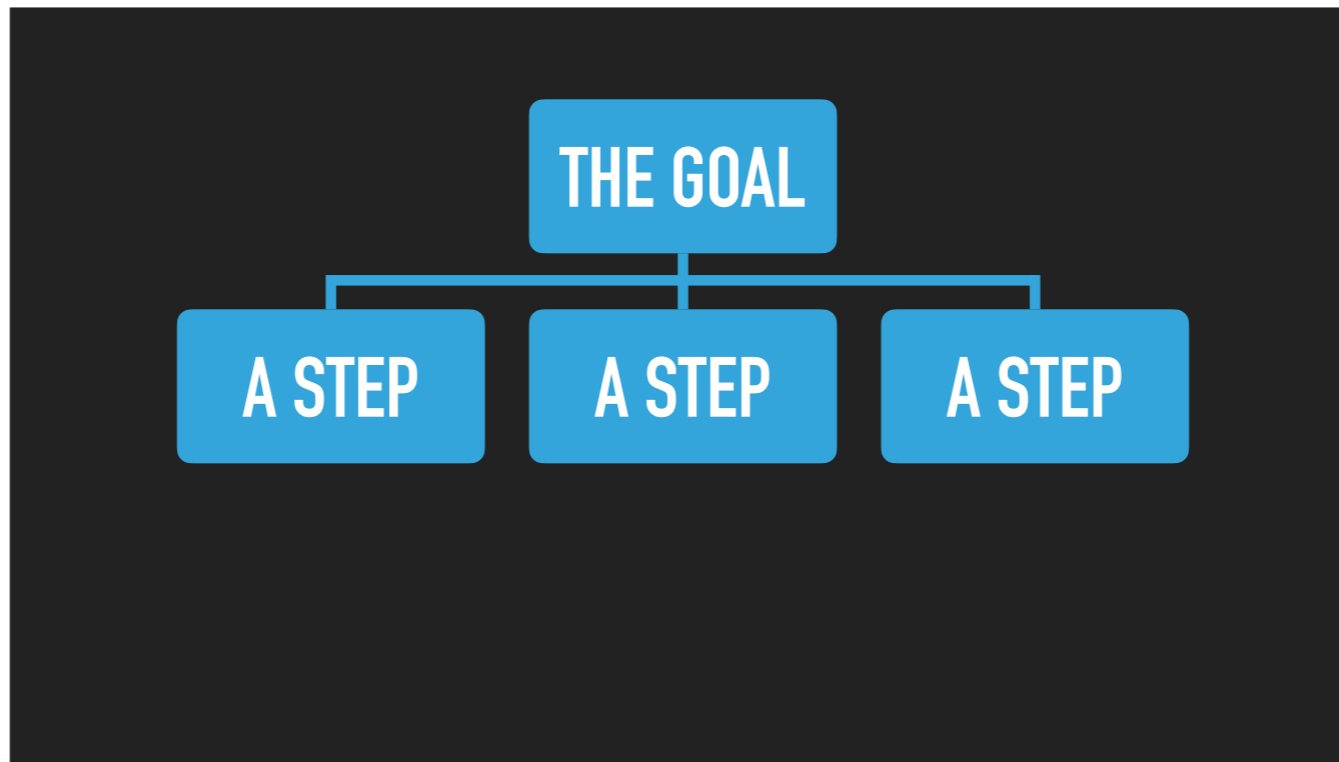
One often goes about this by breaking a problem down into individual steps.

In the software world this is achieved by creating an organization of smaller and smaller pieces of functionality that work together to achieve one's goals.

The problem is, is that over time, requirements change. Business needs change. Technology evolves, and society changes. Sometimes parts of the system need to be rewritten or removed.
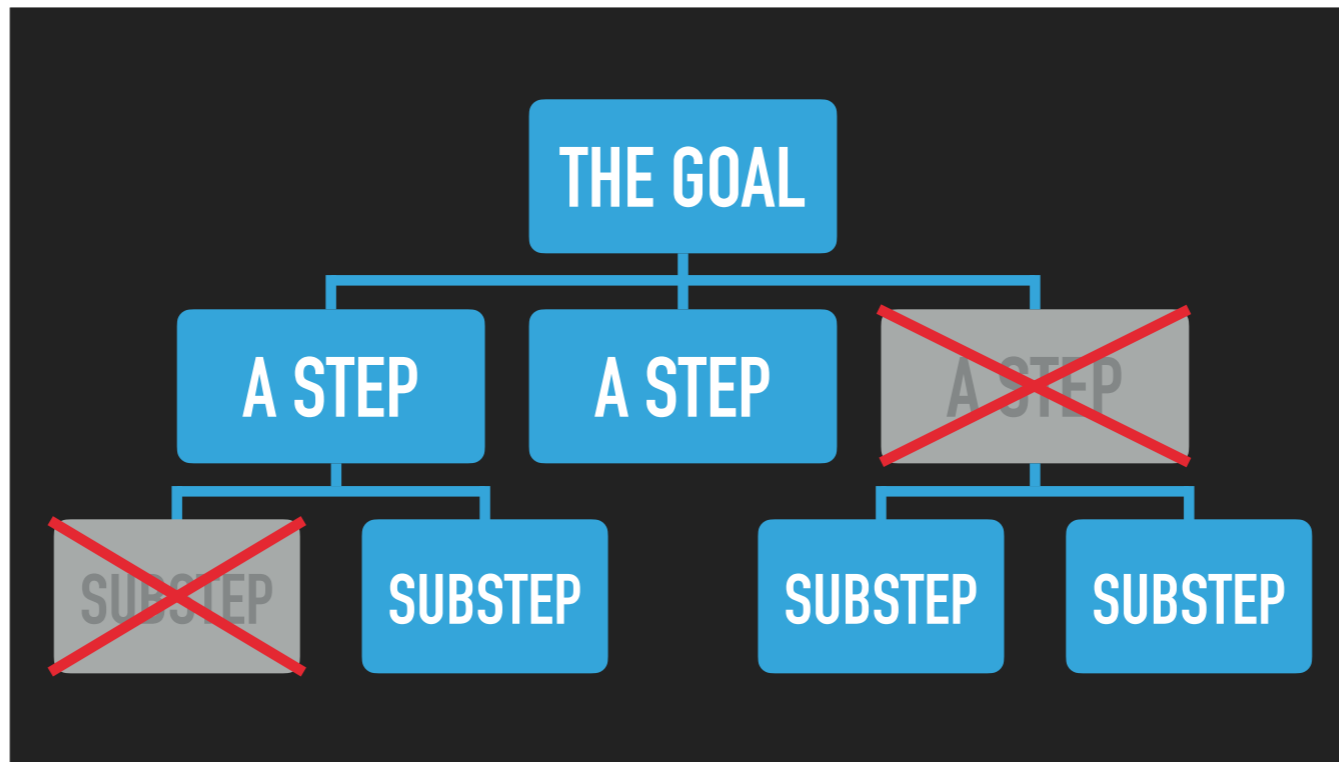
Sometimes bugs are found, requiring portions of the system to be fixed.
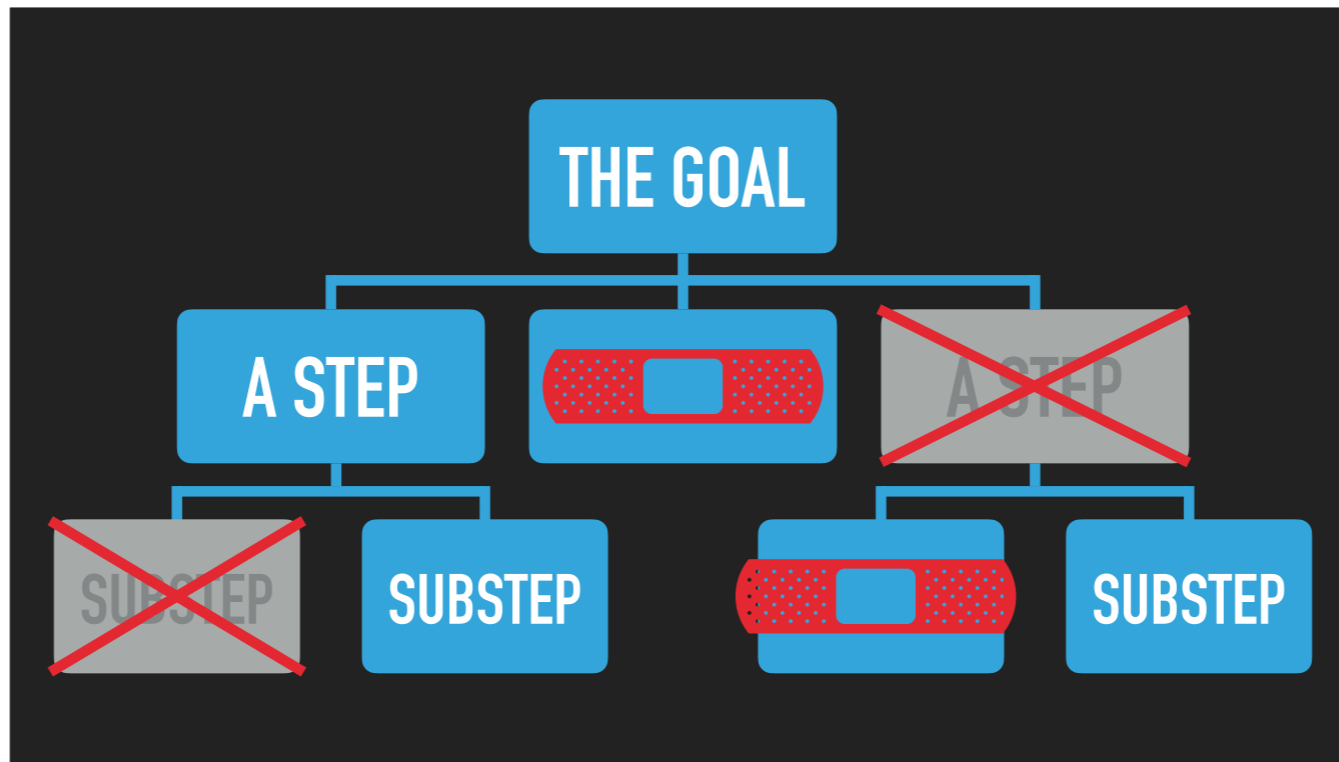
Sometimes in order to meet new requirements, parts of the system end up having to communicate with other parts in new and unforeseen ways.
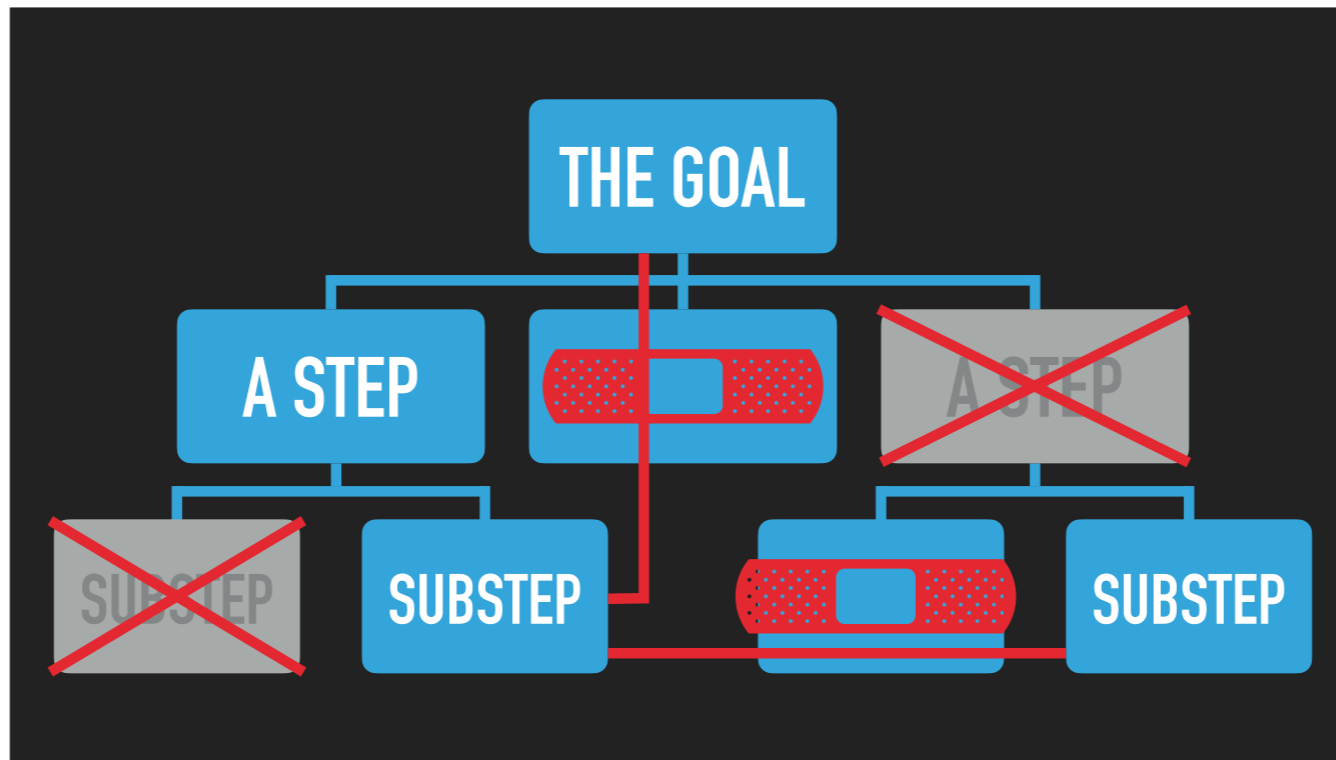
    Eventually the system often ends up looking very different from how it started - a bit like a Frankenstein monster - and as more and more changes get made to the system it becomes more complex, harder to maintain, and less able to evolve.

    In the software industry, when this happens, we say that the system needs to be **refactored**.

    This means that the system needs to be reorganized to be more relevant, understandable, maintainable, and adaptable.

    I propose that the World Wide Web needs to be refactored.

THE WEB NEEDS TO BE
REFACTORED

To explain why, I'm going to take us on a few brief stops in the history of computers, the Internet, and the Web. This will help us understand how we got to where we are today. I'm then going to take us into the future, to 2050 - about 25 years from now - and provide a vision of how a refactored Web can help us in our daily lives.

So why do a "refactor"? What's wrong with the Web today?
I asked a machine learning algorithm that read the equivalent of 3 million books the other day, and this is what it told me:

WEB PROBLEMS

▸ Misinformation and disinformation

▸ Online privacy and security

▸ The digital divide

▸ Online harassment and bullying

▸ Monopolies and data control

Misinformation. Online privacy. The digital divide. Online harassment. Monopolies and data control.
Being a human, I'm going to break this list into two categories:

WEB PROBLEMS CAUSED BY
A LACK OF IDENTITY

…issues caused by, or related to, a lack of identity…

WEB PROBLEMS CAUSED BY
A LACK OF IDENTITY

WEB PROBLEMS CAUSED BY
A LACK OF EQUALITY

…and issues caused by, or related to, a lack of equality.

WEB PROBLEMS CAUSED BY
A LACK OF IDENTITY

WEB PROBLEMS CAUSED BY
A LACK OF EQUALITY

▸ Misinformation and disinformation

▸ Online privacy and security

▸ Online harassment and bullying

We lack an identity on the Web because of the organic way it evolved. When the Web began, everyone trusted each other, there was no reason not to. Only when email spam arose in the early 1990s did people realize that they needed to be able to restrict and filter incoming communications by identity.

And although misinformation may never go away, we can at least lessen it by providing a mechanism by which we can learn who to trust.

WEB PROBLEMS CAUSED BY
A LACK OF IDENTITY

▸ Misinformation and disinformation

▸ Online privacy and security

▸ Online harassment and bullying

WEB PROBLEMS CAUSED BY
A LACK OF EQUALITY

▸ The digital divide

▸ Monopolies and data control

We lack equality on the Web because of the way that computers and the Internet evolved: computing power and Internet access was once only available to organizations with a large amount of resources. Due to sheer economic and social inertia, this has changed very little, with the exceptions being the developments of personal and mobile computing.

So, lack of identity, and lack of equality.

WEB PROBLEMS CAUSED BY
A LACK OF EQUALITY

Let's get into how things got unequal first, and that takes us back to the Web's beginnings.

TED NELSON
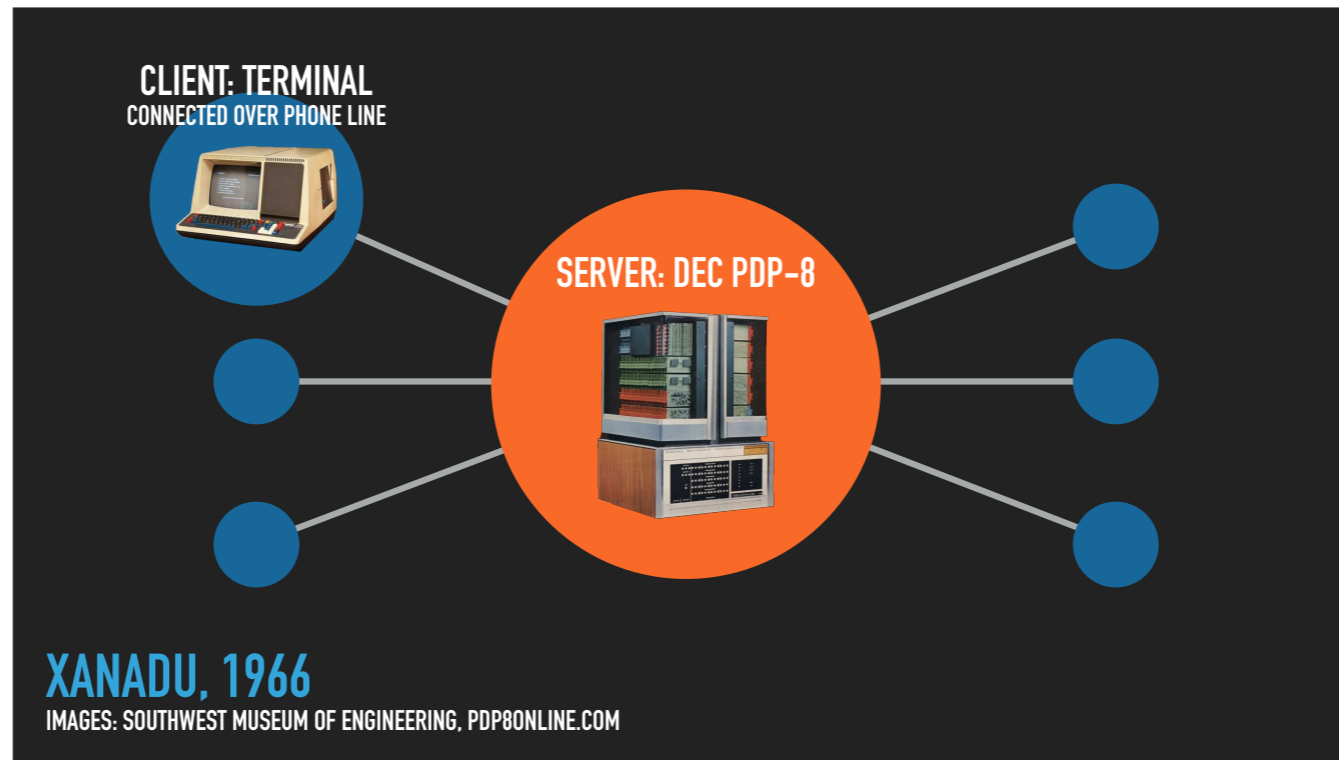IMAGE: STEPHEN CASS, <u>IEEE SPECTRUM</u>, 2018

We need to start with this guy, Ted Nelson, who invented the word "hypertext". Why him? Because he was one of the first inventors in the entire history of computing to have had no mathematic, scientific, or engineering background or training whatsoever. He was a **sociology student**, and thus treated computing not as a mathematic or scientific problem to solve, but as a way to benefit society in general.

In 1960 he began to ask how computers could be used to help people to be more creative, to help people learn and discover resources, and to help people make a living by selling their digital creations online.

By 1965 he began to develop a system to achieve these goals, which he called "Xanadu".

CLIENT: TERMINAL
CONNECTED OVER PHONE LINE

SERVER: DEC PDP-8

XANADU, 1966
IMAGES: SOUTHWEST MUSEUM OF ENGINEERING, PDP8ONLINE.COM

At the time, the idea was that people would log into a central computer - a **server** - via remote terminals, or **clients**. From there they could explore, annotate, edit, rearrange, create, publish, and sell hypertext documents.

His original ideas in 1960 were almost 20 years before mainstream personal computing and almost 10 years before the Internet. This vision, from 1966, reflected the available technology of the time. All of the computing power and data storage within the system was concentrated in a central minicomputer the size of a refrigerator.

Ted created many great quotes related to computer-mediated creativity, but to me his most prophetic and important quote is this one:

"IF THE BUTTON IS NOT SHAPED LIKE THE THOUGHT, THE THOUGHT WILL END UP SHAPED LIKE THE BUTTON."

TED NELSON, DREAM MACHINES, 1974

"If the button is not shaped like the thought, the thought will end up shaped like the button."

This does not apply to just buttons or user interfaces. It applies to programming languages, information design, computer architectures, physical controls, all aspects of technology.
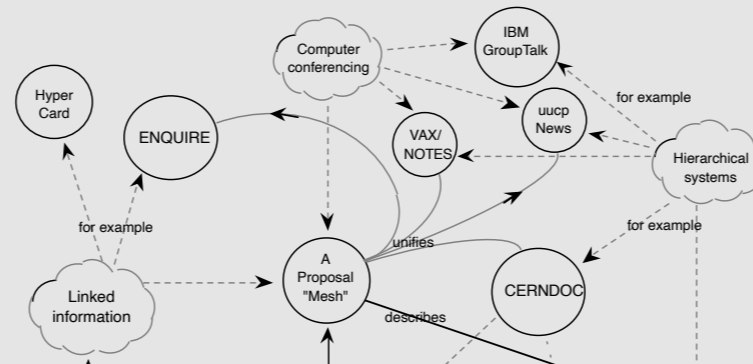
If we create technology that is not complementary to how we think, we will end up creating technology that will mold us in its image. If you create any usable thing, I ask that you repeat this quote like a mantra.
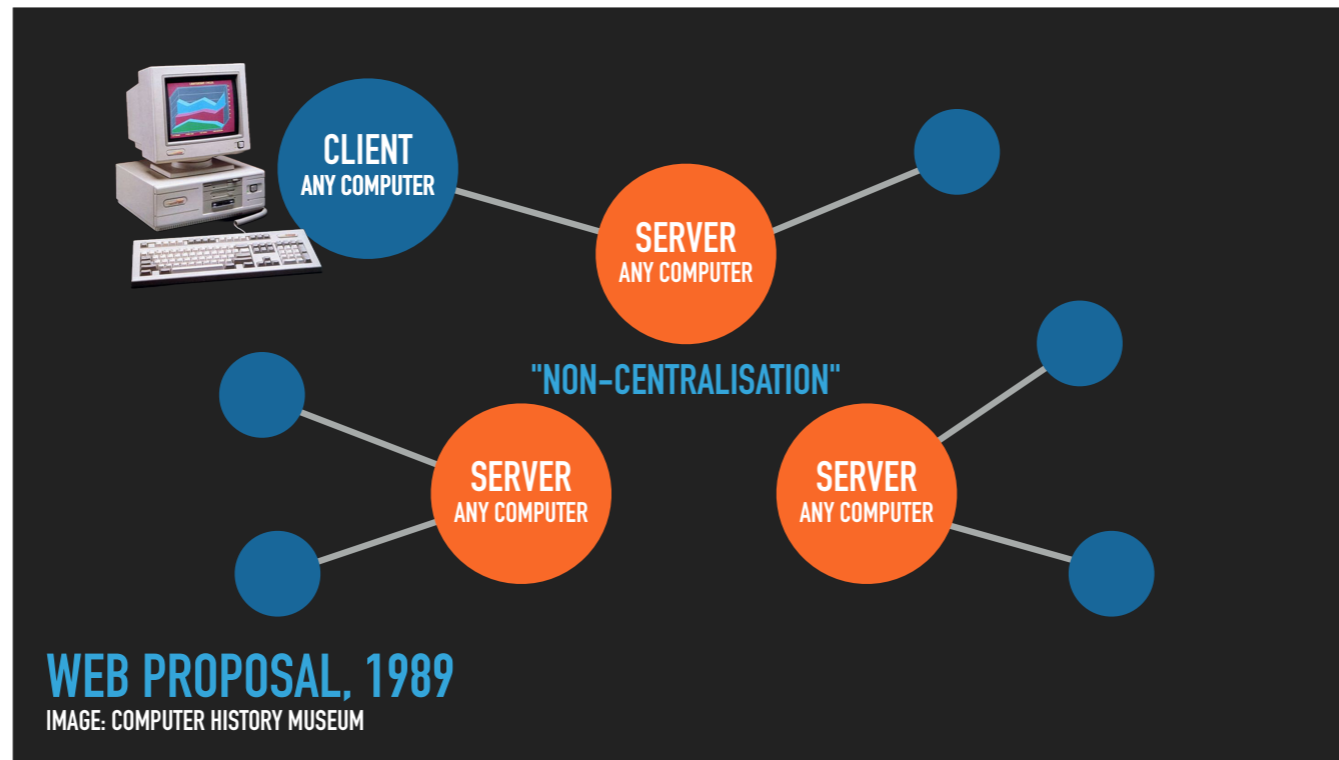
Let's move on to 1989, to Tim Berners-Lee's seminal proposal document at CERN that marked the official birth of the World Wide Web. About 30 years after Ted Nelson's initial thoughts on Xanadu, the idea of a networked hypertext-based system looked like this.

WEB PROPOSAL, 1989
IMAGE: COMPUTER HISTORY MUSEUM

The main difference between this diagram and the Xanadu diagram is that instead of remote terminals that had no computing power, users now had access to personal computers running software that would retrieve documents wholesale over a network from one or more servers.

This meant that one could access the system from many different types of computers. It also meant that one could serve documents from many different types of servers. There was no one central system. And not only was this kind of "non-centralisation" (as Tim called it) a requirement, it also implied that the network would need to support the transfer of much more data, while requiring more computing power on both the client and server sides.

While this was a great step forward for personal computing, it was designed to work for a group of organizations which had access to relatively uncommon amounts of computing power and network bandwidth. Although "non-centralised", the system was developed for a high-performance computing environment, and therein lies the beginning of our problems.
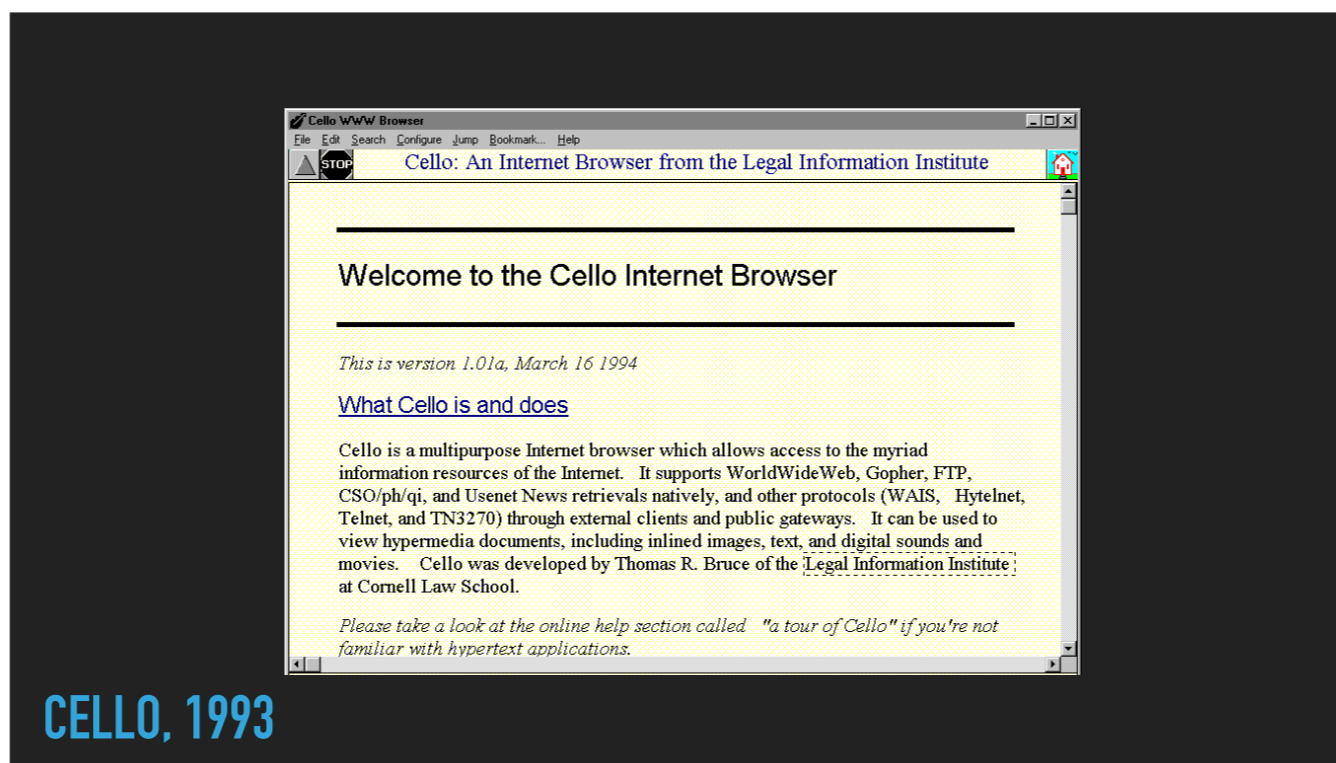
CELLO, 1993

Released four years later in 1993, the Cello Web browser was not only the first Web browser to run on Windows, it was the first Web browser to run on computers that were already in people's houses. In 1993 over 25 million people ran Windows, which had a near total marketshare at the time. And most home PC users had a modem that would allow them to connect online via their phone line.

So thirty years ago the conditions that allowed a person at home to access the Web were ideal, even if it was realistically from only one hardware and software platform.

Now that millions of people could read and interact with Web content, what about the server side? In 1993, where was the equivalent solution to allow everyday people to create and serve content on the Web? The short answer is: it didn't exist, and because of this, I will make the case that it still doesn't exist today, at least not in the way that it should.

1993 WEB SERVER REQUIREMENTS

In 1993, what did it take to run a Web server?

1993 WEB SERVER REQUIREMENTS

▸ A domain name (usually)

First, it required a domain name to make the site's address usable by humans. At the time there was no easy method for individuals to acquire domain names, which was something only companies and organizations did at the time.

1993 WEB SERVER REQUIREMENTS

▸ A domain name (usually)

▸ Responsive, fast, 24/7 Internet access

Second, it required responsive, fast, 24/7 Internet access, which was also beyond the capabilities of typical home users.

1993 WEB SERVER REQUIREMENTS

▸ A domain name (usually)

▸ Responsive, fast, 24/7 Internet access

▸ 24/7 power

Third, running a server required more energy needs than the average household might be willing to pay, at a time when the only always-on technology in one's home was the refrigerator. So not only was it costly, it did not meet the social expectations of the time.

1993 WEB SERVER REQUIREMENTS

▸ A domain name (usually)

▸ Responsive, fast, 24/7 Internet access

▸ 24/7 power

▸ Costly computing power

Fourth, to be fast and responsive, servers at the time ran on larger industrial-class computers which typically cost a minimum of dozens of thousands of dollars.

All of this is why in the early 1990s all Web developers were either in research organizations and computer companies, or were university students. They were the only people in the world that had access to this unique combination of technologies. And even then, at most universities these resources were only available to declared computer science students.

1993 WEB SERVER REQUIREMENTS

▸ A domain name (usually)

▸ Responsive, fast, 24/7 Internet access

▸ 24/7 power

▸ Costly computing power

▸ **Friendly environment**

Finally, what has been lost to history is that in those days running an unsanctioned Internet service in a business or university as a non-computer science student could be considered a subversive, almost punk act - you could be fired or expelled. So the conditions for running a Web server were not only rare, they were hostile. Yet even in this environment we envisioned a day when every person could run their own Web server. Sadly, in repressive regimes the environment for running a Web server remains hostile to this day.

So, besides research organizations and universities, who else ran Web servers at this time? Why, startups did, of course!

1996 STARTUP
SERVER ROOM
IMAGES: KEVIN HUGHES

Most startups in the 1990s built what were called machine rooms or server rooms, which ran shelves of computers configured to be Web servers that sometimes provided other Internet services like email as well. This haphazard state of affairs was typically the norm in the early half of the decade. But obviously after a few years of this it became an inadequate solution, and as we move into the early 2000s we can see how explosive Web growth fueled inequality on both the client and server sides.

THE 2000s: THE RISE OF CLIENT-SIDE INEQUALITY

▸ Cable: a consumer-oriented business model

By the mid-1990s companies were scrambling to find ways to bring Web access into people's houses. The method that won out was via the cable TV connection that most folks already had. But the cable TV companies treated Internet access like television - a one-way communications medium parceled out to an audience of passive consumers.

THE 2000s: THE RISE OF CLIENT-SIDE INEQUALITY

‣ Cable: a consumer-oriented business model

‣ No business incentive to increase upload speeds

It fit perfectly with their existing 50-year-old business model and operations, so while download speeds were enough for people to view images and text on Web pages, they were given only just enough upload speed to submit the contents of an online shopping cart.

THE 2000s: THE RISE OF CLIENT-SIDE INEQUALITY

▸ Cable: a consumer-oriented business model

▸ No business incentive to increase upload speeds

▸ Asymmetrical Internet access becomes the norm

The first cable modem to be based on industry standards was introduced in 1999, and it provided an asymmetric Internet connection. **Asymmetric** means that one's download speeds are different from one's upload speeds, while having a **symmetric** connection means that one's download speeds are equal to one's upload speeds.
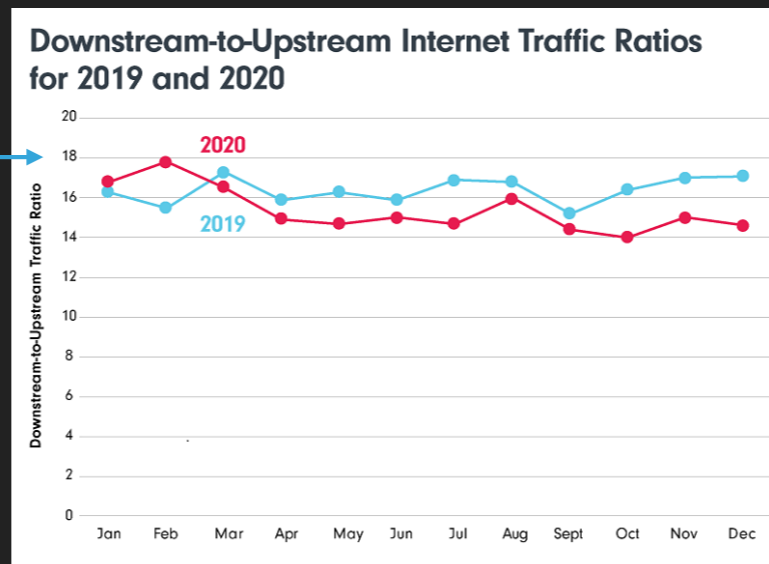
In 1999, the average person was given a download speed of 56 Mbps and an upload speed of 3 Mbps. That's a download to upload ratio of 18, meaning that upload speeds were just **6%** of download speeds. For most people globally, this download to upload ratio has not meaningfully changed in 25 years.
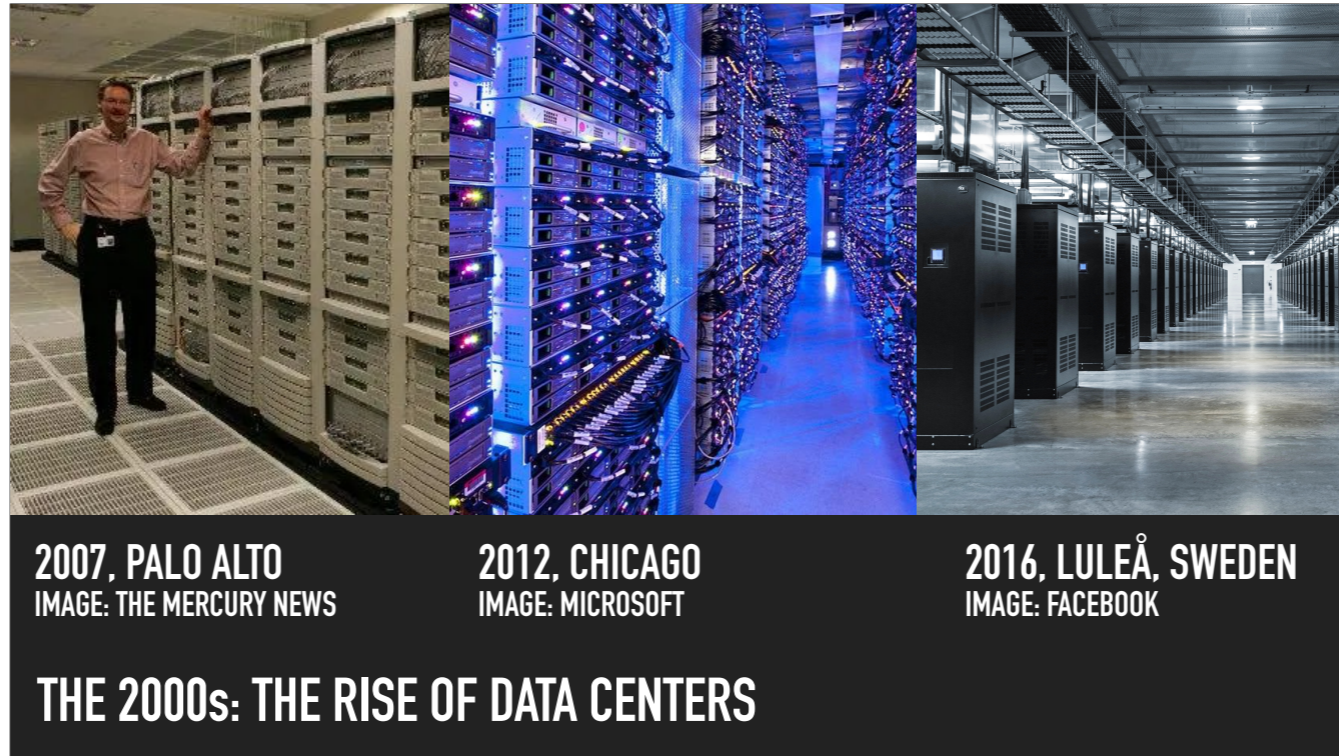
2007, PALO ALTO
IMAGE: THE MERCURY NEWS

2012, CHICAGO
IMAGE: MICROSOFT

2016, LULEÅ, SWEDEN
IMAGE: FACEBOOK

THE 2000s: THE RISE OF DATA CENTERS

On the server side, as the Web grew, the need for larger and more powerful servers arose, such that companies began to put racks of Web servers into large industrial locations called **data centers**. At first, existing warehouses and lofts were converted into data centers, and as the 2000s progressed, completely new types of buildings were designed and built to handle huge amounts of bandwidth, storage, computing power, and noise and heat generated by thousands of servers.

THE CONSUMER WEB, TODAY
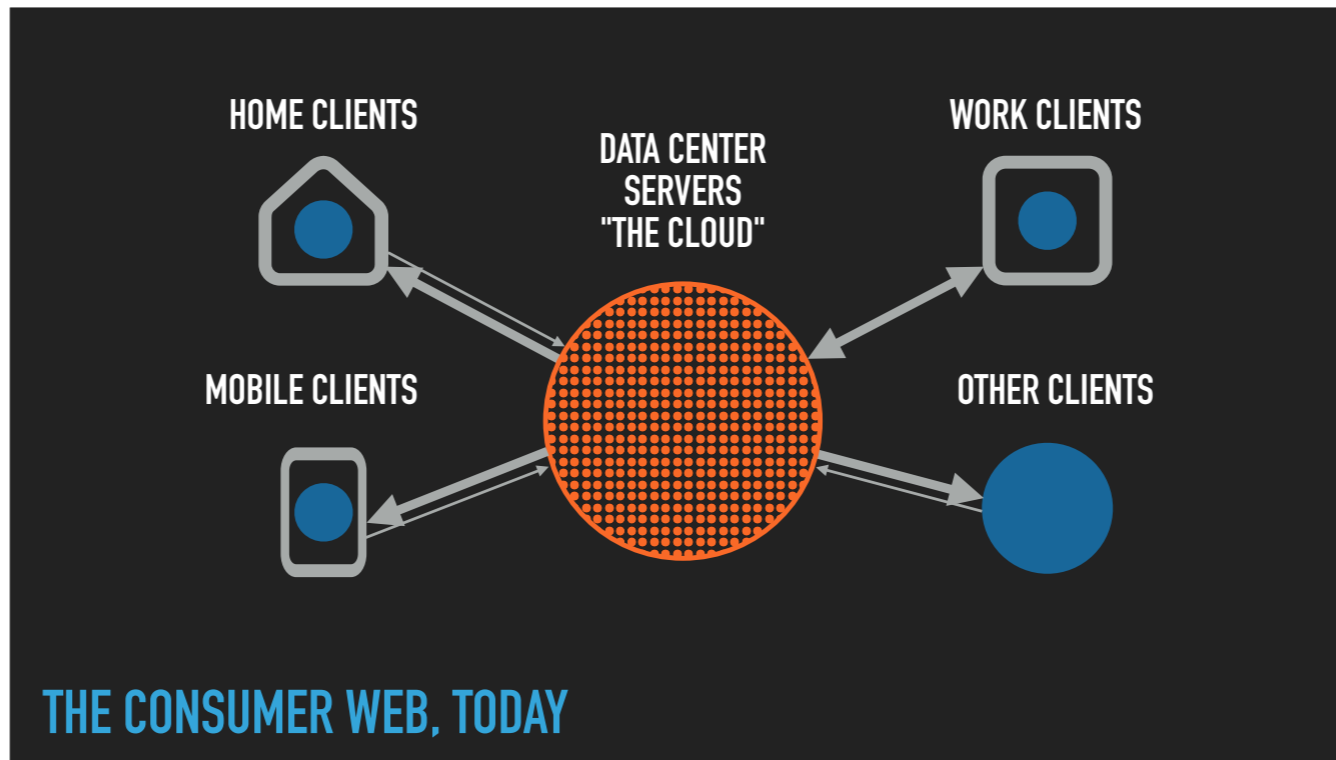
Here's a client and server diagram of how the Web exists today for most people in the world. Today home and mobile Web clients are used from mobile phones and computers using asymmetric connections, and they are used to access very dense clusters of Web servers concentrated almost solely in data centers. At work your organization may be able to afford a symmetric connection to run its own servers, but even internal corporate networks are run today in data centers, or what is often referred to as "the cloud".
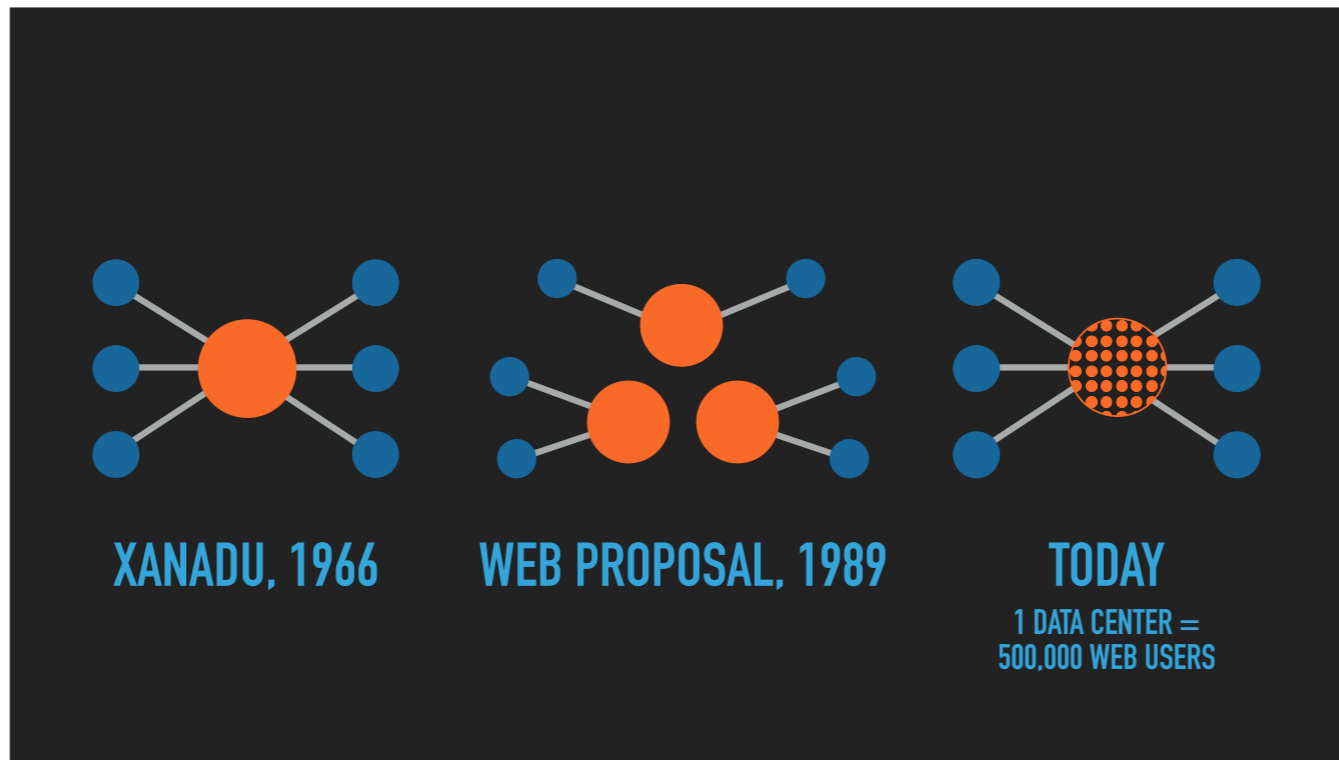
The average data center today contains **100,000** servers.

XANADU, 1966
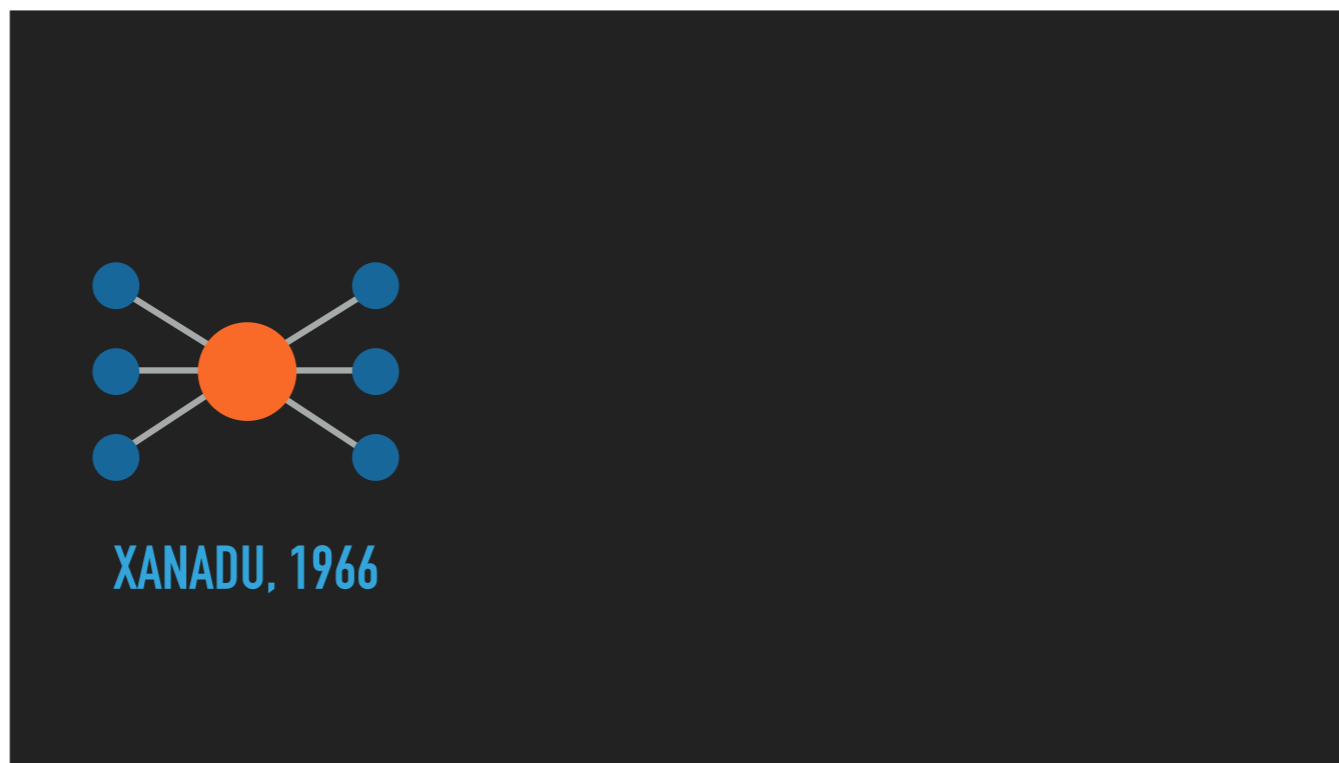
To recap: here's the Xanadu client-server diagram from 1966.

XANADU, 1966   WEB PROPOSAL, 1989
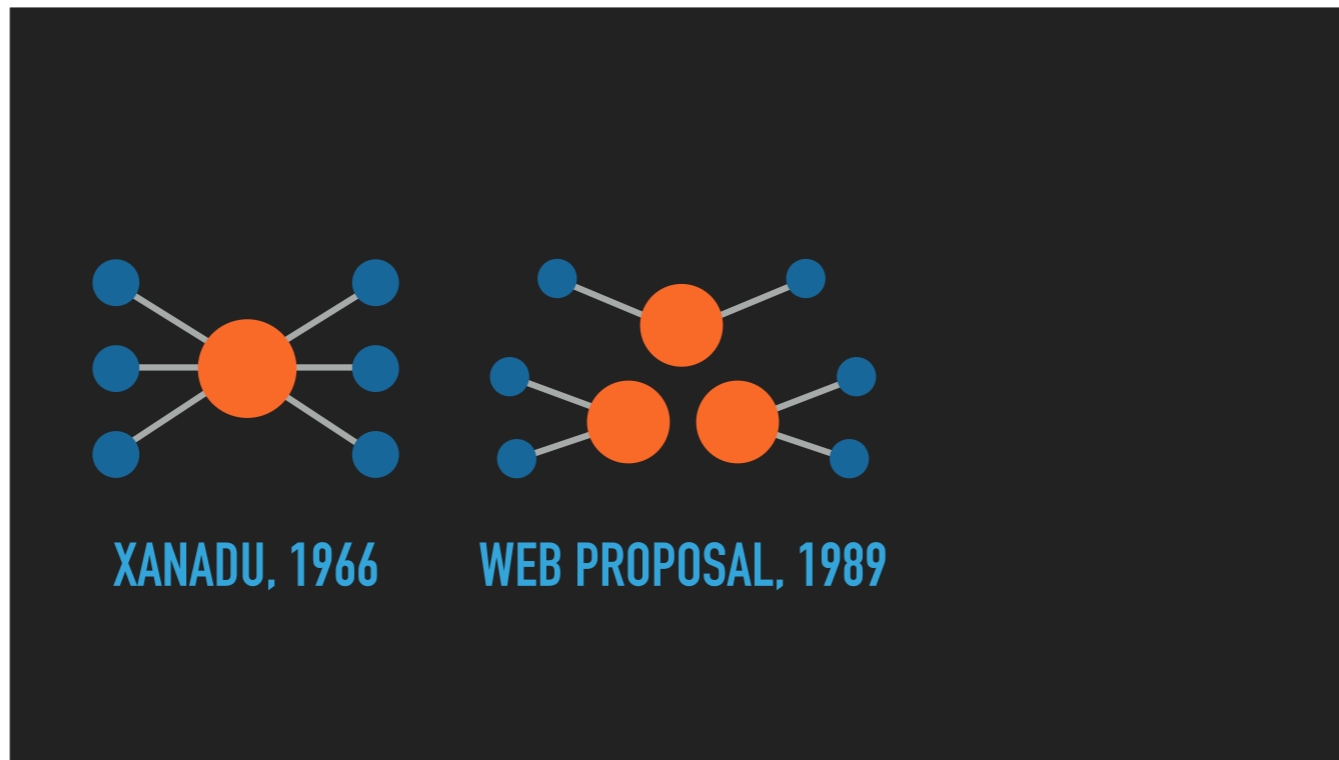
And here's Tim's original Web diagram from 1989.

And here's the Web as it is used today. After almost 60 years, have we made progress?

Today there are 8,000 data centers in the world, or roughly 1 data center for every 500,000 Web users globally. Combined, these centers use over 2% of the world's energy. This is what today's centralized, consumer-oriented Web looks like.

In summary, networking infrastructure as it exists today is **inequitable**. It is realistically one way and consumer-oriented. True symmetrical, fast, and reliable networking exists only for those who can pay for it and we take that for granted.

IN SUMMARY

‣ Networking infrastructure as it exists today is inequitable

  ‣ It is one way and consumer-oriented

‣ It is not treated as a necessary universal service

The Internet is not treated as a utility, as a necessary universal service.
There are no modern standard defined levels of Internet access, and companies are unwilling to provide the data needed to improve it at a global scale.

IN SUMMARY

▸ Networking infrastructure as it exists today is inequitable

  ▸ It is one way and consumer-oriented

  ▸ It is not treated as a necessary universal service

  ▸ For 25 years, one lane goes 6% the speed of the other

It's like we've built a global highway system in which one lane goes 6% of the speed of the other. **You cannot build critical infrastructure this way.**

IN SUMMARY

▸ Networking infrastructure as it exists today is inequitable

 ▸ It is one way and consumer-oriented

 ▸ It is not treated as a necessary universal service

 ▸ For 25 years, one lane goes 6% the speed of the other

▸ Web infrastructure as it exists today is inequitable

 ▸ You must pay to run a Web server

Web infrastructure as it exists today is inequitable. You must pay to run a Web server, because residential networking is inadequate.

IN SUMMARY

▸ Networking infrastructure as it exists today is inequitable

  ▸ It is one way and consumer-oriented

  ▸ It is not treated as a necessary universal service

  ▸ For 25 years, one lane goes 6% the speed of the other

▸ Web infrastructure as it exists today is inequitable

  ▸ You must pay to run a Web server

  ▸ Issues of monopolies and data control

And if you depend on a Web server run by a company, that brings about issues of monopolies and data control. The content you create, upload, and sell is always handled by a third party, and is always subject to their specific requirements and policies. In most cases, when you upload something to a Web site, they legally reserve the right to do anything they want with your content, and it is often difficult, if not impossible, to permanently remove something you've uploaded to a third party Web site.

WEB PROBLEMS CAUSED BY
A LACK OF IDENTITY

I've talked about how Web infrastructure has become unequal; now I'm going to talk about how the Web has suffered from a lack of individual identity.

"On the Internet, nobody knows you're a dog."

In mid-1993 *The New Yorker* published this cartoon by Peter Steiner: "On the Internet, nobody knows you're a dog." Not only was it one of the first indications that the Internet had reached the levels of popular media, it placed the idea in people's minds that the Internet was a haven of anonymity. After 20 years, it remained the magazine's most reprinted cartoon.

Thirty years later, while there are many Web sites that provide anonymity, there are now also many that require verification of one's identity.
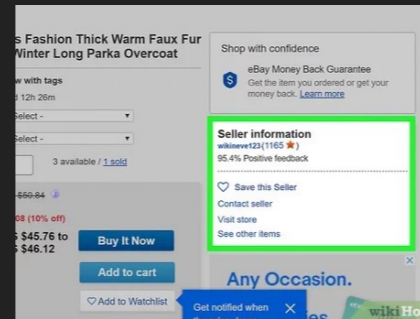
MODERN IDENTITY VERIFICATION

ON TWITTER

Elon's blue check disaster is getting worse
Celebrities are refusing to pay, and the trolls keep trolling.

Elon Musk
@elonmusk · Follow

Replying to @StephenKing
We need to pay the bills somehow! Twitter cannot rely entirely on advertisers. How about $8?

7:16 PM · Oct 31, 2022

212K   Reply   Copy link

Read 30.2K replies

IMAGE: VOX.COM

ON EBAY

IMAGE: WIKIHOW.COM

ON FACEBOOK

Verify Your Accounts on Facebook and Instagram
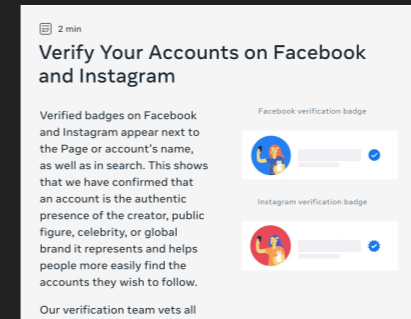
IMAGE: FACEBOOK.COM

. . .BUT NOT FOR COMMON HUMAN TASKS

And as you can see, this verification is often up to the whims of the company doing it, and sometimes you have to pay for it.
And of the sites that do offer identity verification, most are related to ecommerce or social media, not to common human tasks.
What do I mean by "common human tasks"?

WHAT DO HUMANS DO?

In modern societies, as a human being, as you live your life, you will likely do most of these things:

WHAT DO HUMANS DO?

▸ You are born

▸ You go to school

▸ You vote

▸ You go to college

▸ You graduate

▸ You get a job

▸ You drive a car

▸ You pay taxes

▸ You pay bills

▸ You see a doctor

▸ You travel overseas

▸ You get married

▸ You buy or rent a house

▸ You sell a house

▸ You parents pass away

▸ You retire

▸ You make a will

▸ You die

You are born, you go to school. You vote, you go to college. You graduate, you get a job. You drive a car and pay taxes and bills. You see a doctor and travel overseas. You get married, buy or rent a house, maybe sell a house.

Your parents pass away. You retire, you make a will, and you die.

All of these things require verification of your identity, and in many countries, such verification is *legally required*.

If we wish to create humane technology, it should at least help us do these things. Yet even today doing them with the assistance of online services can still be difficult, frustrating, expensive, and sometimes impossible.

Compounding the issue are technologies (such as machine learning) that are becoming exponentially more effective in fooling (sometimes by mimicking) people to reveal or gain access to private credentials.

IDENTITY IS A SLIDING SCALE

FULLY ANONYMOUS
- Zero verification
- Might not be human

HUMAN ANONYMITY
- Proven human
- Support groups
- Victim protection
- Repressed groups

FULLY VERIFIABLE
- Unique human
- Notarization
- Government services
- Legal, ecommerce

We need a system that recognizes that identity exists on a sliding scale, from being fully anonymous to being fully verifiable, because this is how the world works. It's about being clear about identity, whether it's verifiable or not. This builds trust and strengthens online communities. The ability to build a safe anonymous online community is just as important as the ability to build a safe verifiable online community. If we built a Web that was identity-aware in this way, we could then build systems on top of it to bring identity awareness to all Internet services.

As the great American abolitionist and orator Frederick Douglass said over 150 years ago, "Where there is no truth, there can be no trust, and where there is no trust there can be no society."

Here are a few scenarios that an identity-aware Web could allow:

IDENTITY-AWARE SCENARIOS

▸ Private, verified, and safe communications groups between relatives, colleagues, friends and friends of friends

Private, verified, and safe communications groups between relatives, colleagues, friends and friends of friends.

IDENTITY-AWARE SCENARIOS

▸ Private, verified, and safe communications groups between relatives, colleagues, friends and friends of friends

▸ Online legal transactions between multiple parties on your behalf

Online legal transactions between multiple parties on your behalf.

IDENTITY-AWARE SCENARIOS

▸ Private, verified, and safe communications groups between relatives, colleagues, friends and friends of friends

▸ Online legal transactions between multiple parties on your behalf

▸ Access to medical records across multiple healthcare organizations

Access to medical records across multiple healthcare organizations.

IDENTITY-AWARE SCENARIOS

▸ Private, verified, and safe communications groups between relatives, colleagues, friends and friends of friends

▸ Online legal transactions between multiple parties on your behalf

▸ Access to medical records across multiple healthcare organizations

▸ The legal transfer of digital assets and credentials upon death or other criteria

The legal transfer of digital assets and credentials upon death or other criteria.

IDENTITY–AWARE SCENARIOS

▸ Private, verified, and safe communications groups between relatives, colleagues, friends and friends of friends

▸ Online legal transactions between multiple parties on your behalf

▸ Access to medical records across multiple healthcare organizations

▸ The legal transfer of digital assets and credentials upon death or other criteria

▸ **Verifiable chains of ownership of digital assets**

Verifiable chains of ownership of digital assets.

IDENTITY-AWARE SCENARIOS

- Private, verified, and safe communications groups between relatives, colleagues, friends and friends of friends
- Online legal transactions between multiple parties on your behalf
- Access to medical records across multiple healthcare organizations
- The legal transfer of digital assets and credentials upon death or other criteria
- Verifiable chains of ownership of digital assets
- Online voting and use of government services

Online voting and use of government services.

In addition to all of this, an identity-aware Web could allow people to permanently and verifiably remove any content they have created at or uploaded to any Web site.

So we've talked about how the Internet - and the Web - could be more equitable, and more identity-aware. Where do we begin to fix these things?

NEXT STEPS

A good approach when refactoring a system is to look at the original requirements and then move forward from first principals, keeping what has worked to date and discarding what no longer works for us.

Throughout history the main goal of Web-like systems can be summed up in two words: **resource sharing**.

NEW GOALS FOR A RESOURCE-SHARING SYSTEM

The term "resource sharing" does not imply a non-commercial system. It refers to a system designed to allow people to distribute, navigate, and discover information.

So let's rework the Web's original requirements given today's technological advancements and social expectations.

NEW GOALS FOR A RESOURCE-SHARING SYSTEM

▸ Public and private world wide resource sharing for everybody

Public and private world wide resource sharing for everybody. Originally the Web was made to work within and for organizations with specialized resources. This must be explicitly extended to all individuals, especially those who have no such resources.

NEW GOALS FOR A RESOURCE-SHARING SYSTEM

▸ Public and private world wide resource sharing for everybody

▸ Resource sharing within and among individuals, groups, organizations

Resource sharing within and among organizations, individuals, and groups. This incorporates the notion of group and organization-based resource sharing, such as group communications and business-to-business transactions.

NEW GOALS FOR A RESOURCE-SHARING SYSTEM

▸ Public and private world wide resource sharing for everybody

▸ Resource sharing within and among individuals, groups, organizations

▸ Everyone may have a verifiable identity or be anonymous

This is a new goal. **The identity of Web users may be verifiable as well as anonymous.** The system must be identity-aware on multiple levels.

NEW GOALS FOR A RESOURCE-SHARING SYSTEM

▸ Public and private world wide resource sharing for everybody

▸ Resource sharing within and among individuals, groups, organizations

▸ Everyone may have a verifiable identity or be anonymous

▸ All resources may be associated with one or more identities

This is also a new goal. **All resources may be associated with one or more identities.** The system must be able to verify ownership, or the lack of it, at all levels.

NEW GOALS FOR A RESOURCE–SHARING SYSTEM

▸ Public and private world wide resource sharing for everybody

▸ Resource sharing within and among individuals, groups, organizations

▸ Everyone may have a verifiable identity or be anonymous

▸ All resources may be associated with one or more identities

▸ Everybody should have access to and run a Web browser

Everybody should have access to and run a Web browser. For almost 5 billion people - over half the world's population - this is true.

NEW GOALS FOR A RESOURCE-SHARING SYSTEM

▸ Public and private world wide resource sharing for everybody

▸ Resource sharing within and among individuals, groups, organizations

▸ Everyone may have a verifiable identity or be anonymous

▸ All resources may be associated with one or more identities

▸ Everybody should have access to and run a Web browser

▸ Everybody should have access to and run a Web server

Finally, surprisingly a new goal for a resource-sharing system, one that has never really been made explicit in the history of computing: **Everybody should have access to and run a Web server.** Let me expand on this for a moment.

EVERYONE SHOULD HAVE ACCESS TO AND RUN A

WEB SERVER

OR, "DO YOU RENT YOUR COMPUTER?"

This can no longer be a controversial requirement - people **want** to put content on the Web, but they pay other companies to do it for them: 40% of Web sites run on WordPress, which takes care of many of the technical and infrastructure needs required to run a Web server.

But in the same way that you fully own your **personal computer** and can do whatever you want with it, you should fully own your **personal server** and do whatever you want with that.

Today people can pay for their phones and laptops in installments if they choose to. But what if the only way to have your own computer was on a subscription basis, and if you failed to pay, it would be taken away from you with all the data on it deleted? Maybe you pay for your phone this way, but do you rent your laptop in this way? Think about how silly that sounds.

For the average global citizen that has a Web site, this is the only choice that they have.

You may say, "but there are plenty of ways to make a Web site for free online". Yes, to a degree, as long as the company that allows you to stays in business and as long as you follow their policies.

But as we expand the notion of the Web server to become a personal computing device designed to handle all aspects of resource sharing in your life, you will begin to see how limiting the current state of things actually is.

And thus begins our process of refactoring the Web.

THE CONSUMER WEB, TODAY
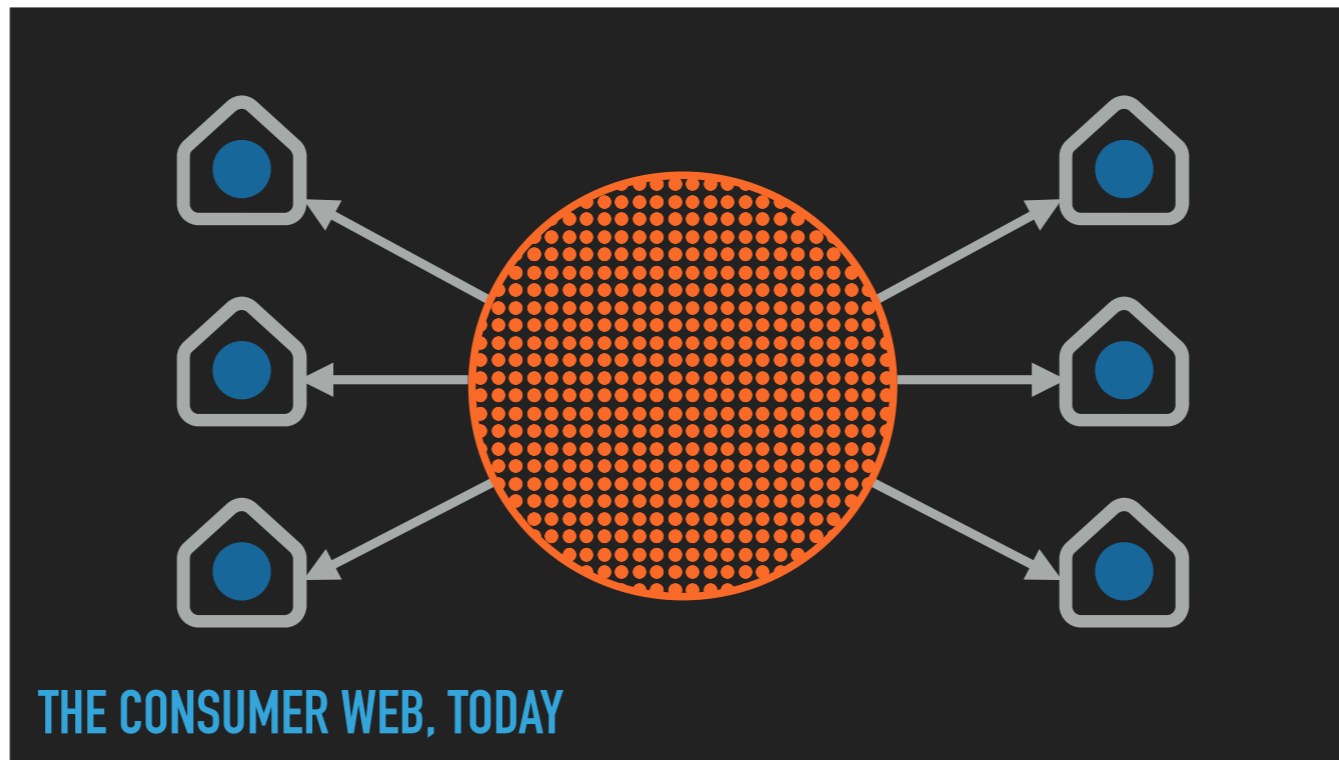
You've seen this before, how the Web exists today for the vast majority of people around the world. It looks unequal, doesn't it? Little bits of personal computers running Web clients, and massive giant clusters of corporate-owned Web servers.

If the infrastructure is unequal, we must make it equal. We can do this by examining everything that a data center does today, and then putting that functionality into the hands of everybody. This is the refactor: **we blow up the data center.** Here's what that looks like.

THE REAL WORLD WIDE WEB
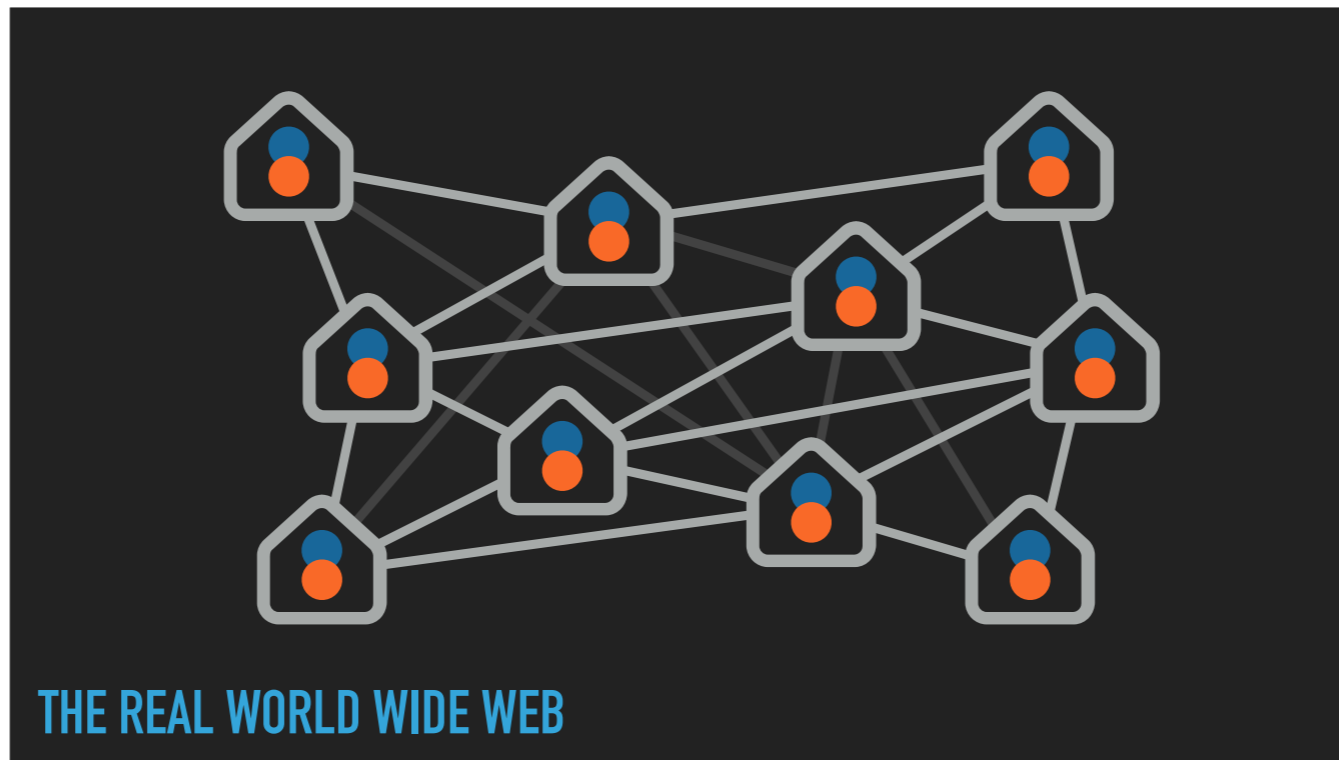
      In this vision of a future Web, everyone has a Web server. But it's not just that, far from that. In the same way that every person connected to the Internet already has a router, modem or dish that manages their physical network connection all day every day, they also have a personal server that manages all of their resource sharing needs. Just as your digital life incorporates a **personal computer** today, it will incorporate a **personal server** tomorrow.

      This is akin to giving everyone in the world the modern equivalent of their own printing press, with all of the human potential that the development of the original printing press unleashed hundreds of years ago.

      And as you'll see, it is this idea that can help solve the problems of inequality and identity that plague us today.

YOUR SERVERS TODAY

The social expectation for running a personal server already exists today. In fact, chances are, you already run a number of servers in your household - in your wi-fi router, streaming box, game console, and smart TV. But all of these servers were designed for the **consumption** of content.

None of the servers currently running in your house were made for you to design, control, edit, authenticate, publish, sell or lease content and services that you create and provide.

What we call a Web server now - some combination of hardware and software, regardless of form or location - must drastically evolve to fit modern needs, to the same degree that the "dumb phone" evolved into the "smart phone" of today.

Residential technology is the technology of society. It creates assumptions in ourselves about everyday living. For over 100 years since the radio, people have assumed that they are nothing more than an audience in their own homes. But what if we lived in a society that assumed first that we had imaginations, that we were creators, inventors, writers, artists, entrepreneurs, or actually social people instead?

THE PERSONAL SERVER

▸ Energy efficient, inexpensive

▸ Send/distribute/sell content

Imagine an energy-efficient, inexpensive device - in your house, apartment, or RV that serves as the main storage and control point for all digital media and services that you wish to sell or publish, including Web sites, images, music, videos, ebooks, software, services, livestreams, anything, on your own terms.

THE PERSONAL SERVER

▶ Energy efficient, inexpensive

▶ Send/distribute/sell content

▶ Identity verification

It serves as the primary authentication point that verifies whether or not a digital creation - or any piece of information on the Internet, such as a post, comment, or reaction - originated with you, and controls how accessible it should be to the public.

THE PERSONAL SERVER

- ▶ Energy efficient, inexpensive
- ▶ Send/distribute/sell content
- ▶ Identity verification
- ▶ **Secure data archive**

It serves as the verifiable, secure archive for all of your critical data, such as medical, legal, and financial information, which you can share immediately as needed with other verified and trusted organizations, on your own terms. It helps you notarize information, whether it's a work of art, a legal document, or a vote.

THE PERSONAL SERVER

▶ Energy efficient, inexpensive

▶ Send/distribute/sell content

▶ Identity verification

▶ Secure data archive

▶ For all Internet services

Imagine being able to combine the power of multiple devices of those you trust - family and friends - to share real-time news and media directly with each other, to trade and sell resources within your group, or to form cooperatives to buy and sell goods and services collectively.

Imagine not having to pay a hosting firm to run your Web site and receive your email, or a cloud service to share, store, secure and authenticate your content and personal information, or a data center to run your business. Imagine not having to use an endless parade of companies as middlemen in order to distribute, publish, sell, and verify images, videos, writings, software, services, and goods that you create.

You could do all these things yourself using one easy-to-use device that you have complete control over and can secure and authenticate. One day, this device could very well be your wi-fi router, HomePod, or an open-source relative of them.

This is the promise of a true decentralized Web - one that is designed to empower individuals, not monopolists, venture capitalists, financial institutions, or tech bros.

THE PERSONAL SERVER: 2050 SCENARIO

Let's zoom ahead 25 years and imagine the Web of 2050, made up of a global network of personal servers.

It's May 2, 2050. You just moved into a new place that already had fiber installed. Your Internet Service Provider gave you a personal server device, which was included free with the service and is yours to keep.

You open the box. The device itself is small, about the size of a large grapefruit. As it never draws more than 50 watts of energy (and uses less than 10 watts when idle), you hook it up to your solar window array, plug it into the Ethernet jack in the wall, and turn it on.

THE PERSONAL SERVER: 2050 SCENARIO

▸ Configuration

You connect to the server's private wireless network to begin the setup process using a Web browser or your voice. The device checks for and installs software updates, and then asks you to enter or say your name as the primary identity associated with the device as well as your preferred language.

THE PERSONAL SERVER: 2050 SCENARIO

▸ Configuration
▸ Identity verification/installation

Now you're asked to authenticate yourself. You look down at the top of the device and show both hands to it. The device uses some combination of your features, such as your fingerprints, iris patterns, and facial structure to generate a globally unique and verifiable identity which will be used by default in all services that run on the device. You may then specify a number of backup authentication methods, such as other identities that can provide authentication for you in case of emergency.

You then tell the server to install this identity on other devices on your private network, such as your phone or laptop. Now these devices can use this identity when using services like email or the Web.

THE PERSONAL SERVER: 2050 SCENARIO

▶ Configuration

▶ Identity verification/installation

▶ Domain name setup

Immediately your server obtains its own unique randomly generated domain name, but the setup process allows you to buy new domain names to associate with the device. The domain names you set up are automatically associated with your identity.

THE PERSONAL SERVER: 2050 SCENARIO

‣ Configuration

‣ Identity verification/installation

‣ Domain name setup

‣ **Site design**

As you are selling things on a Web site, you set up services using your custom domain name. Using a Web browser optionally assisted by your voice, you create an email account for your domain and design your Web site. In minutes you've got an email address and a working retail Web site. It can make you money, but it cost you nothing, and had zero energy impact on the environment.

Now let's take a look at some of the Internet services your personal server provides. All services are Web applications that are compiled into WebAssembly, which is a W3C standard format for programs that can be written in any programming language and run on any type of hardware.

These services include:
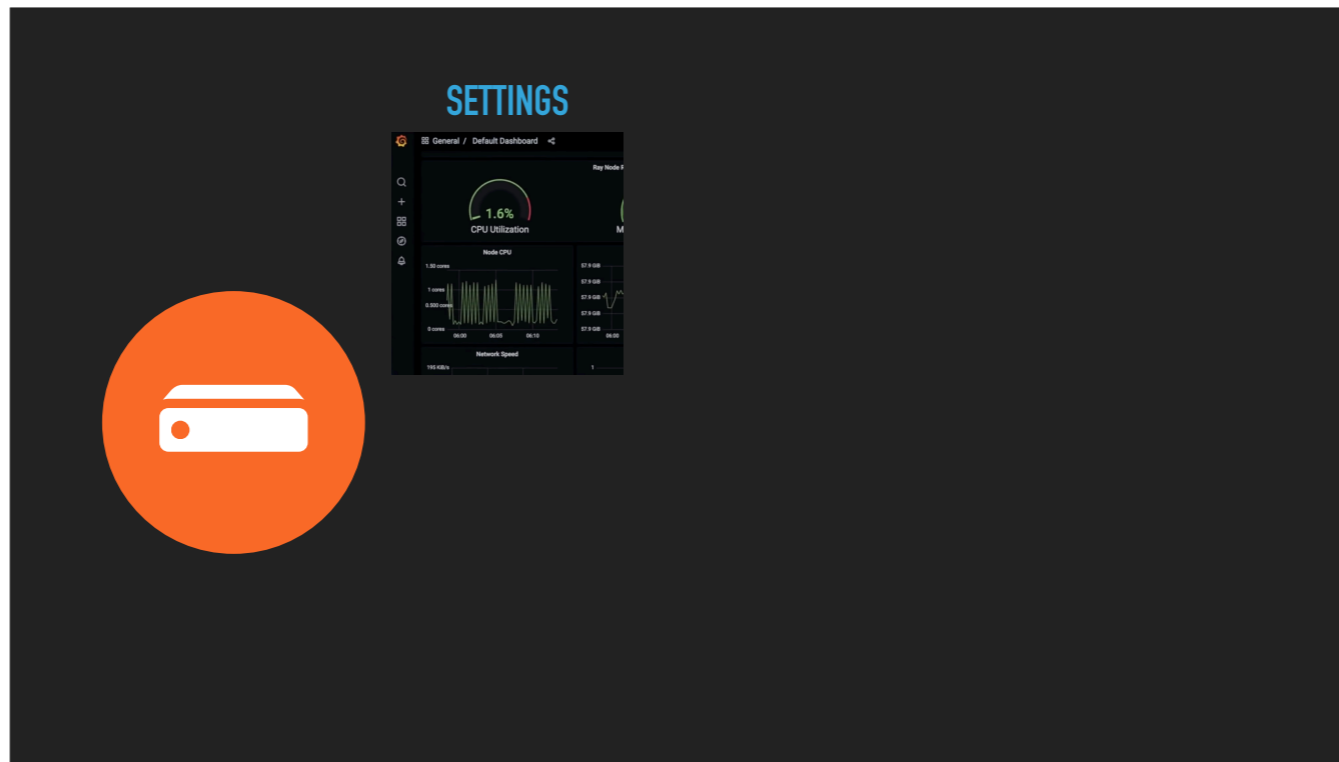
An overall configuration and management interface, perhaps similar to the Settings app on iPhones or System Settings on macOS.
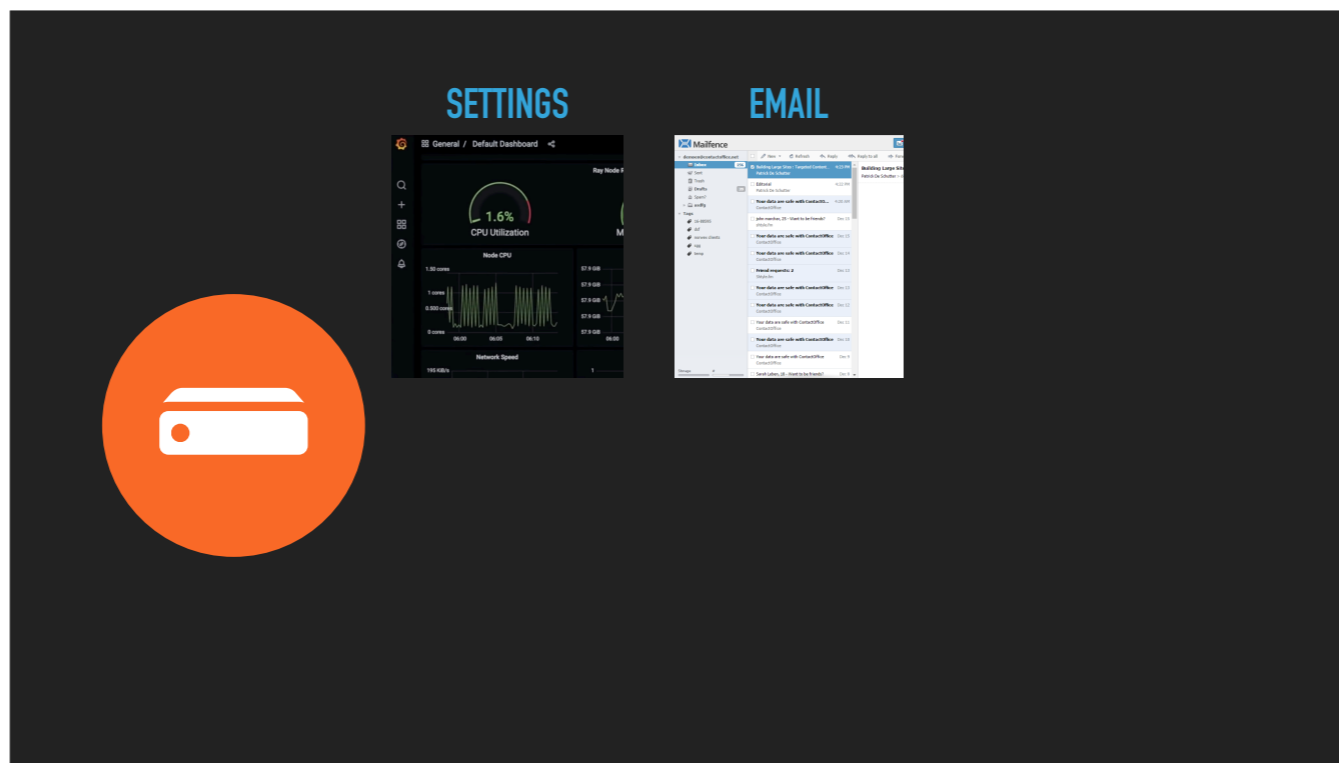
An email server and Web client.
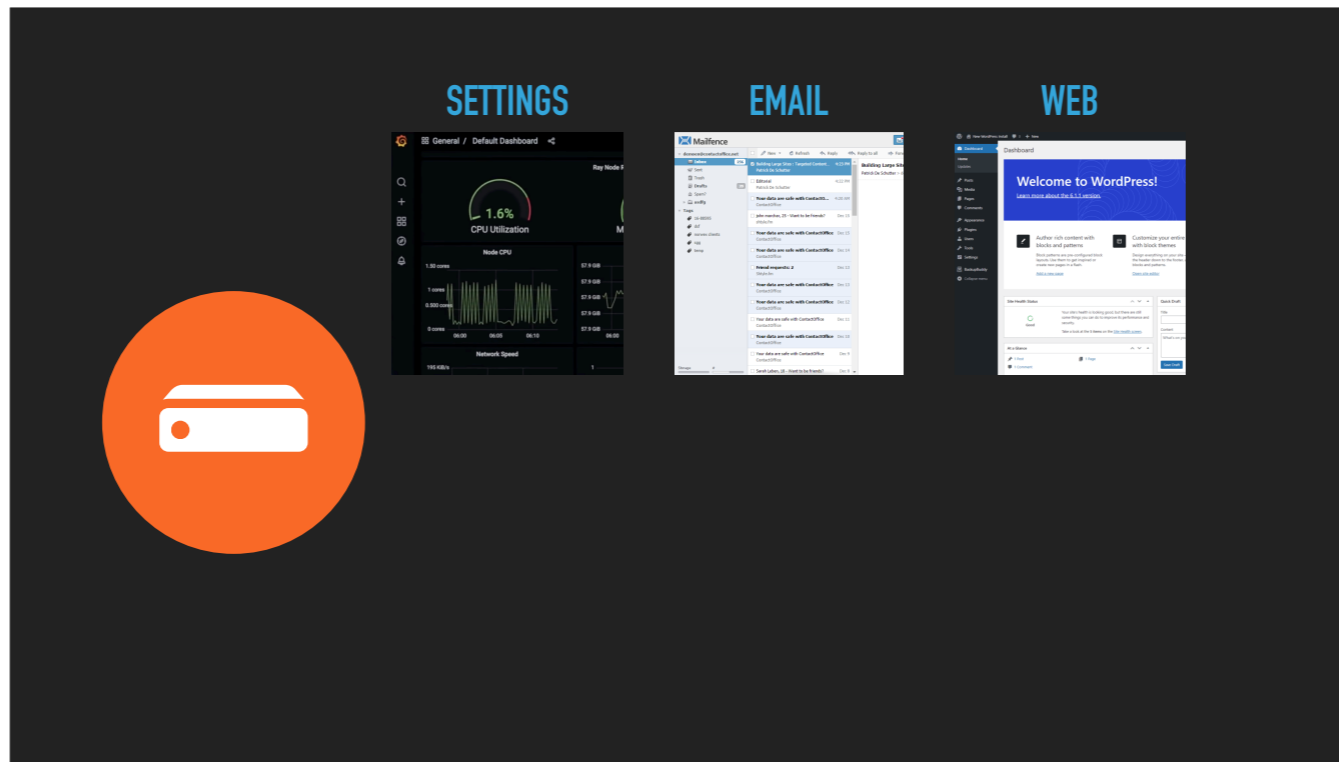
No-code tools for Web site and application creation, similar to WordPress and Squarespace.
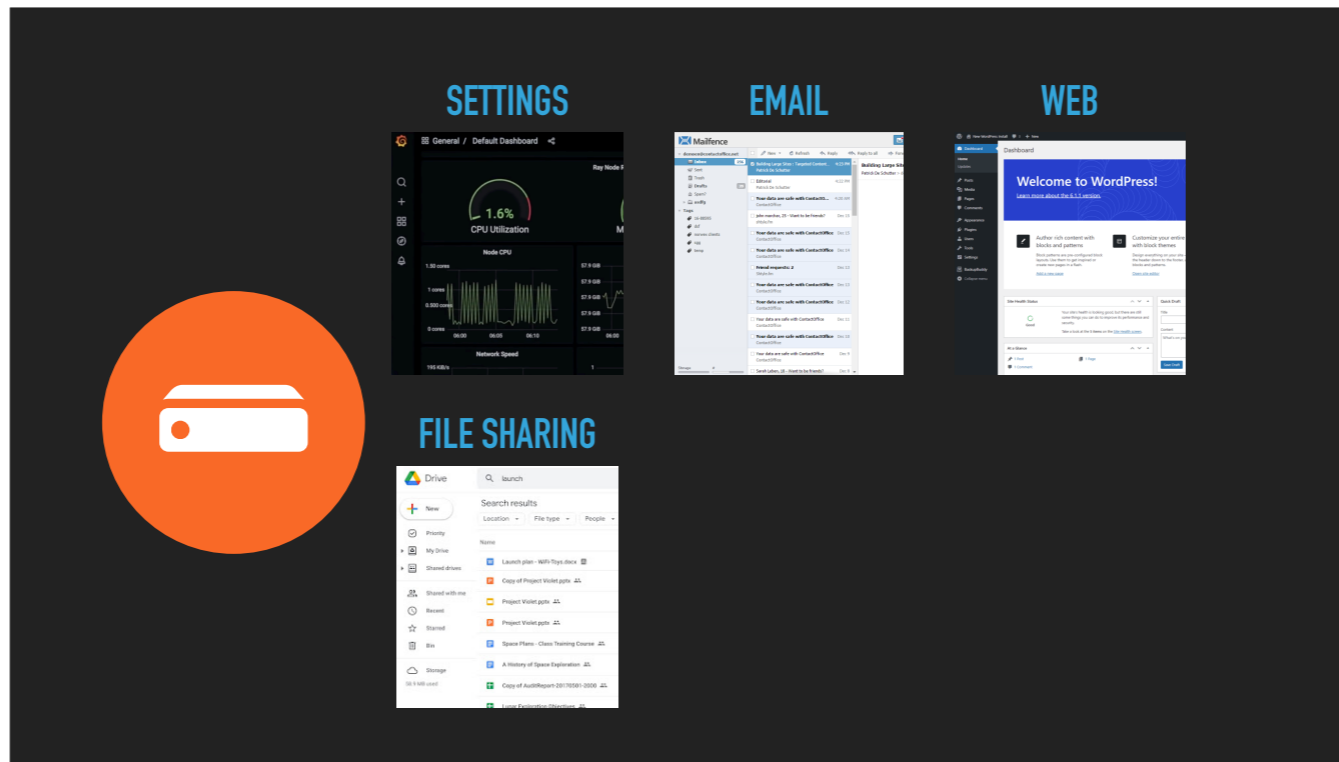
A filesystem-like document storage and sharing service, like Google Drive.

SETTINGS

EMAIL

WEB

FILE SHARING
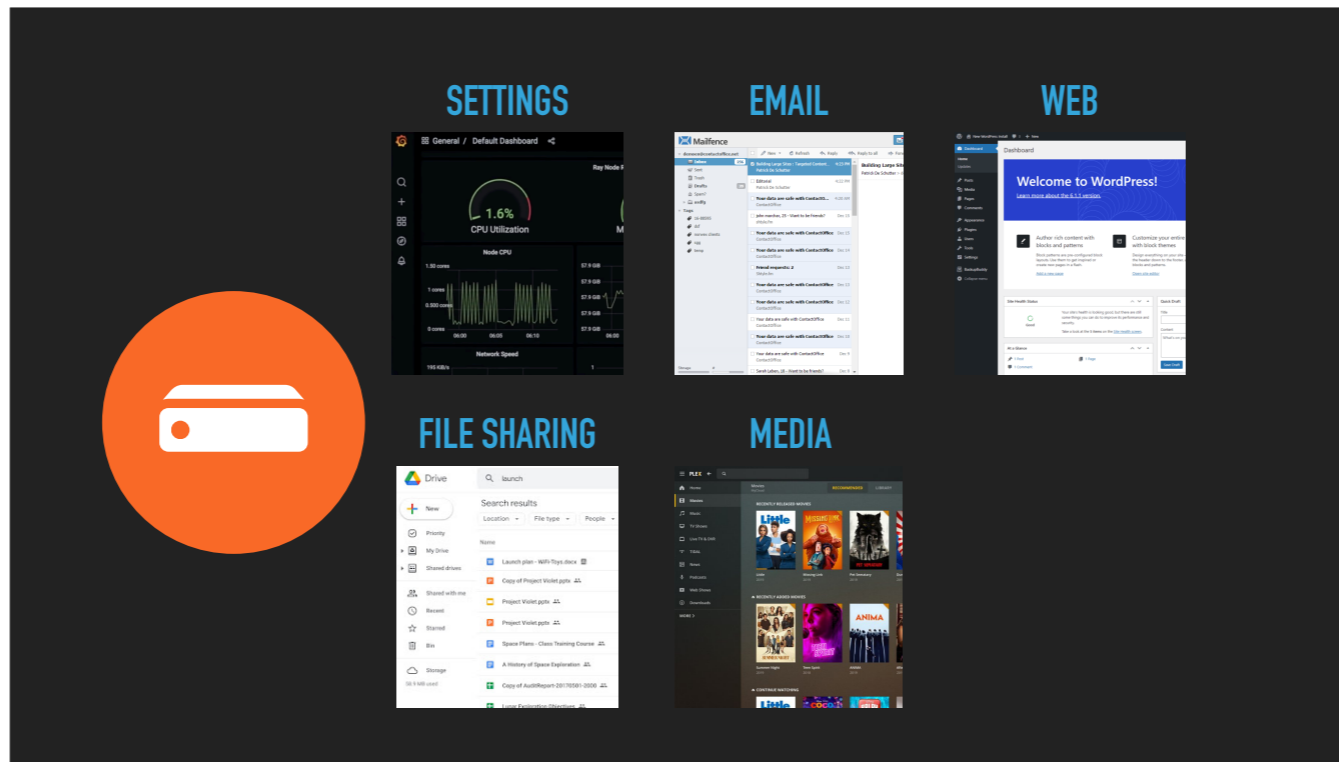
MEDIA

A media streaming, storage, and sharing service, similar to Google TV or Plex.
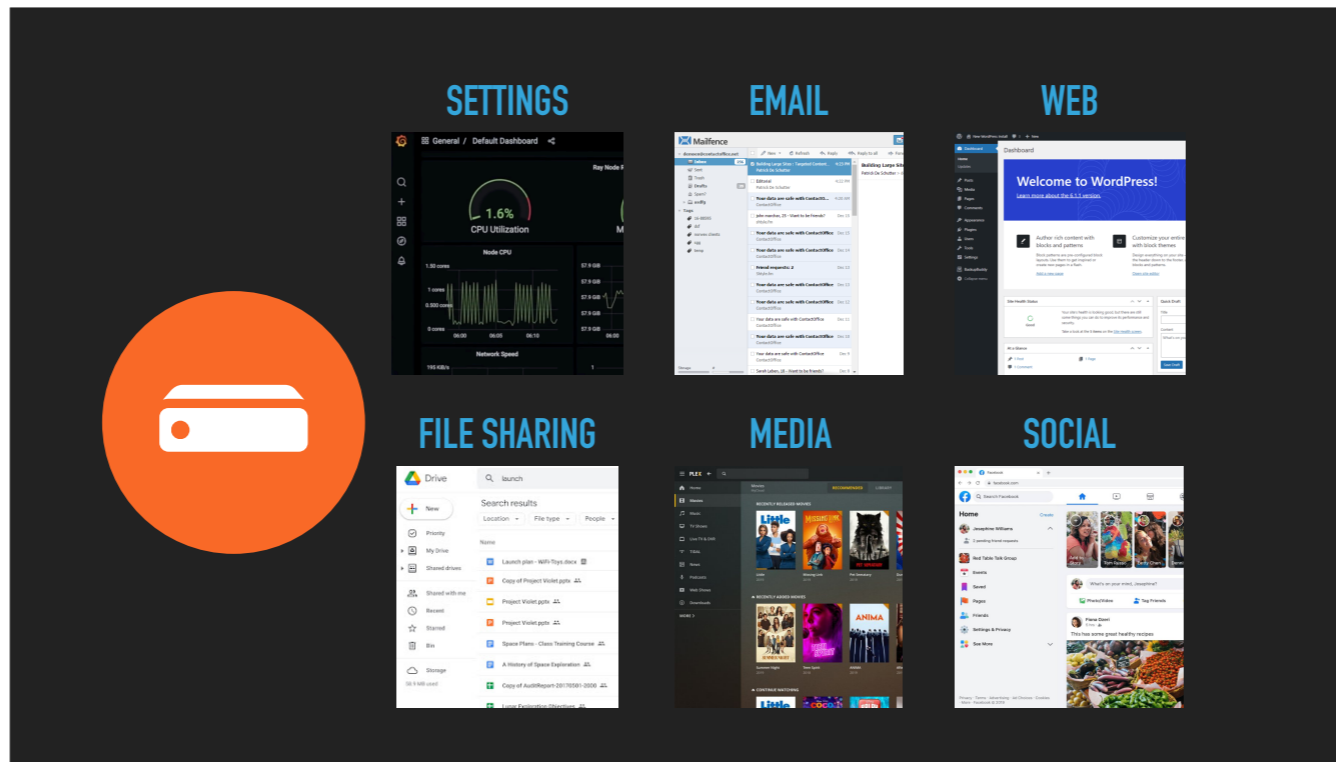
A rich media communications service, like Facebook, Messenger, Twitter, and Instagram. This service would allow you to create public and private social news feeds and rich media messaging for distribution to verified as well as anonymous individuals worldwide. These feeds or chats could run on multiple personal servers at once in a synchronized manner among its participants in a process called **federation**.
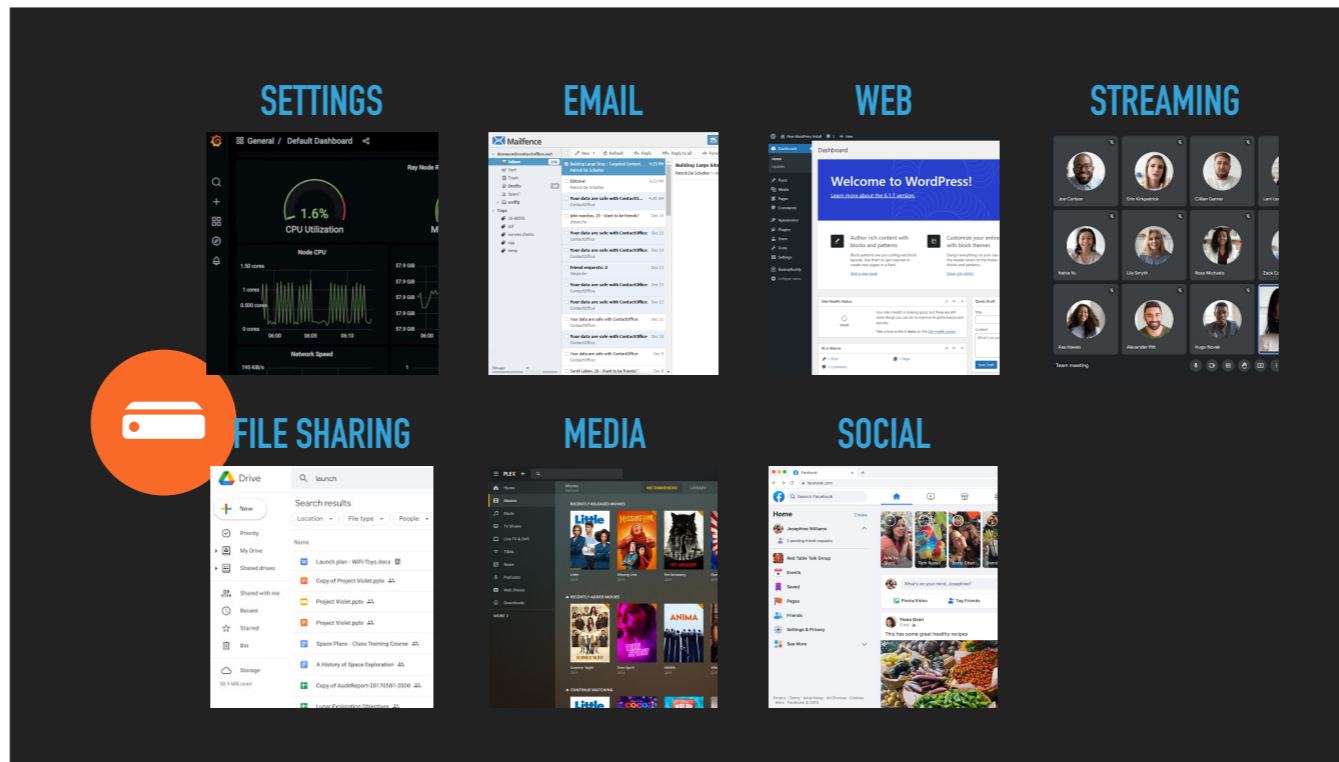
Also included would be streaming services for broadcasting and receiving live audio and video streams. This would be used to allow voice calls, videoconferencing, livestreaming, collaborative performances, and jam sessions.
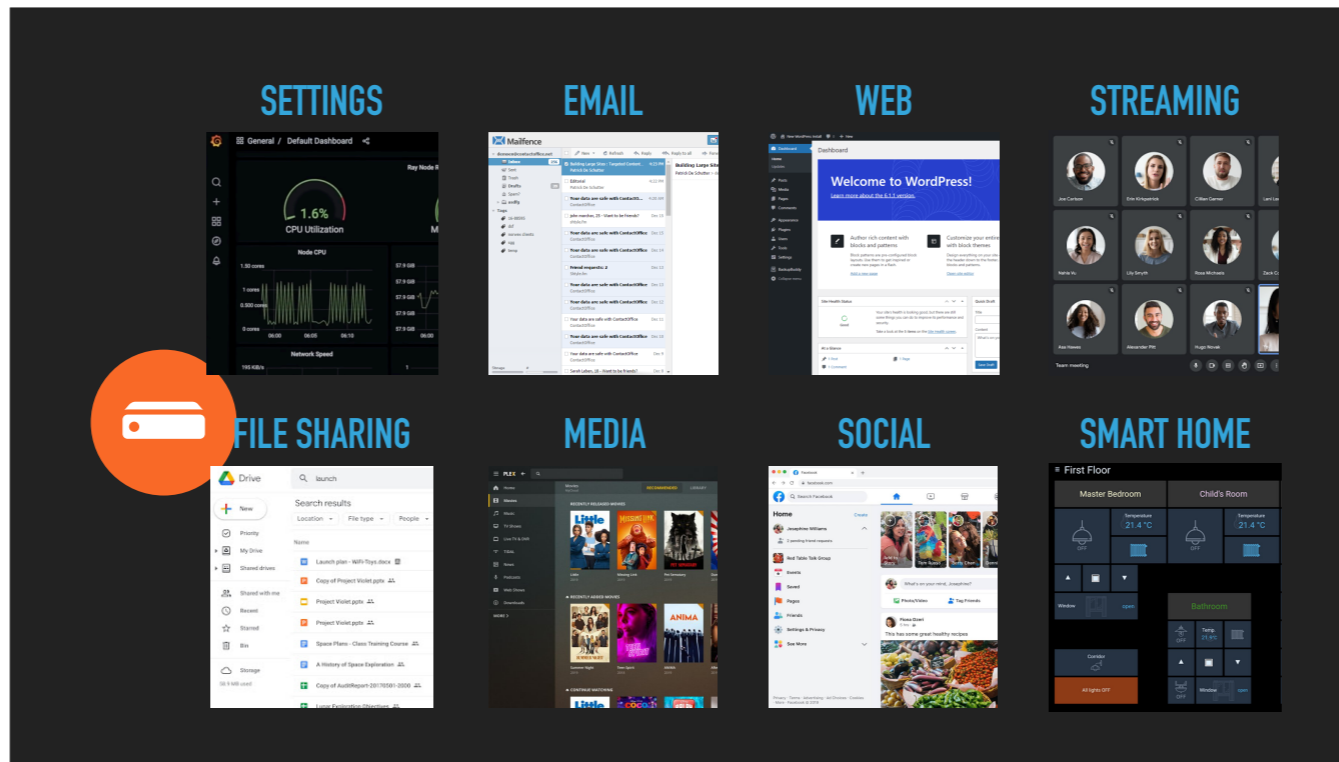
Also included would be a service for smart home management, so one can manage and automate one's smart home devices, such as turning off the lights, locking the doors, and enabling security notifications when the last person leaves the house.
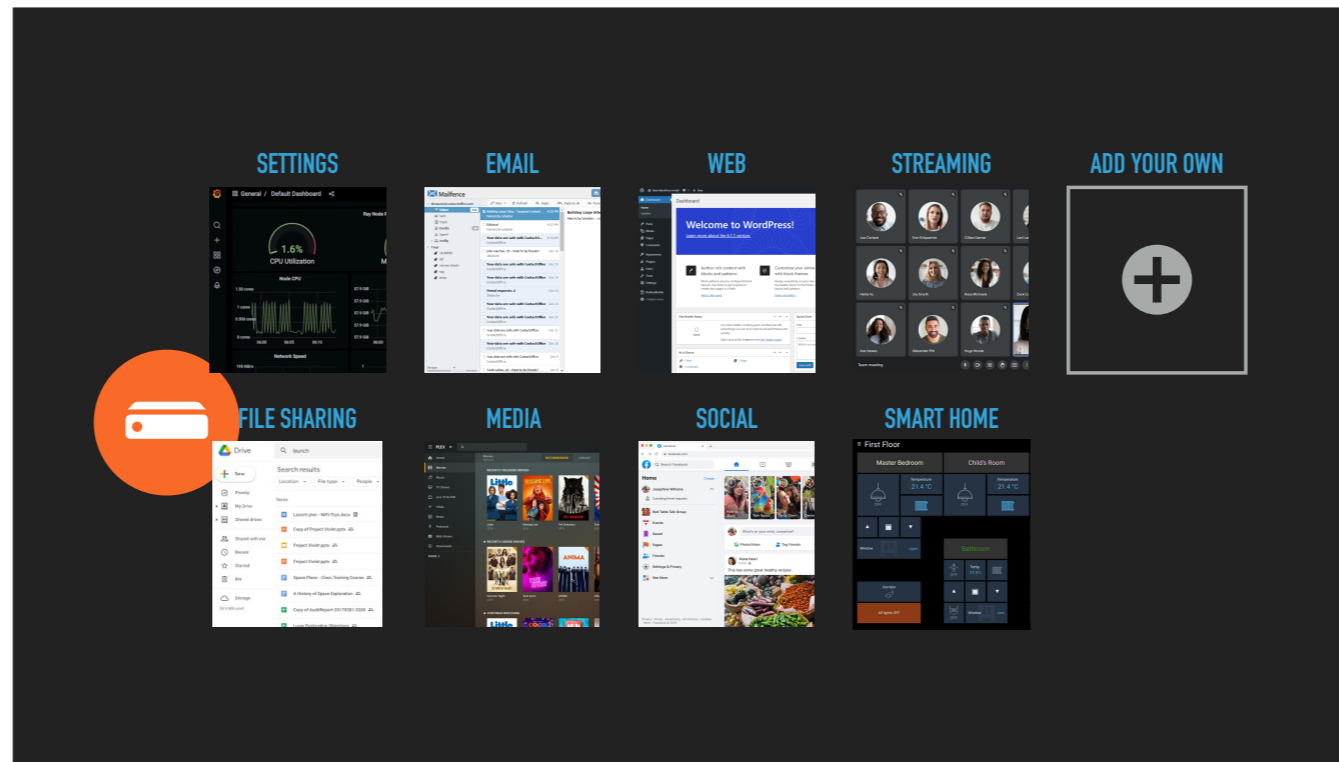
Other distributed services such as peer-to-peer file sharing and virtual online world presence may be optionally added via an App Store-like process. You should be able to install and run compiled and packaged third-party services as you see fit, but services should be vetted and authenticated by a managing organization to prevent malware and software viruses.

Note that having a modular architecture like this allows services to evolve quickly as standards change. Instead of waiting months or years for Web and other Internet servers to adopt new protocols as in the past, such widespread changes can now happen in hours.
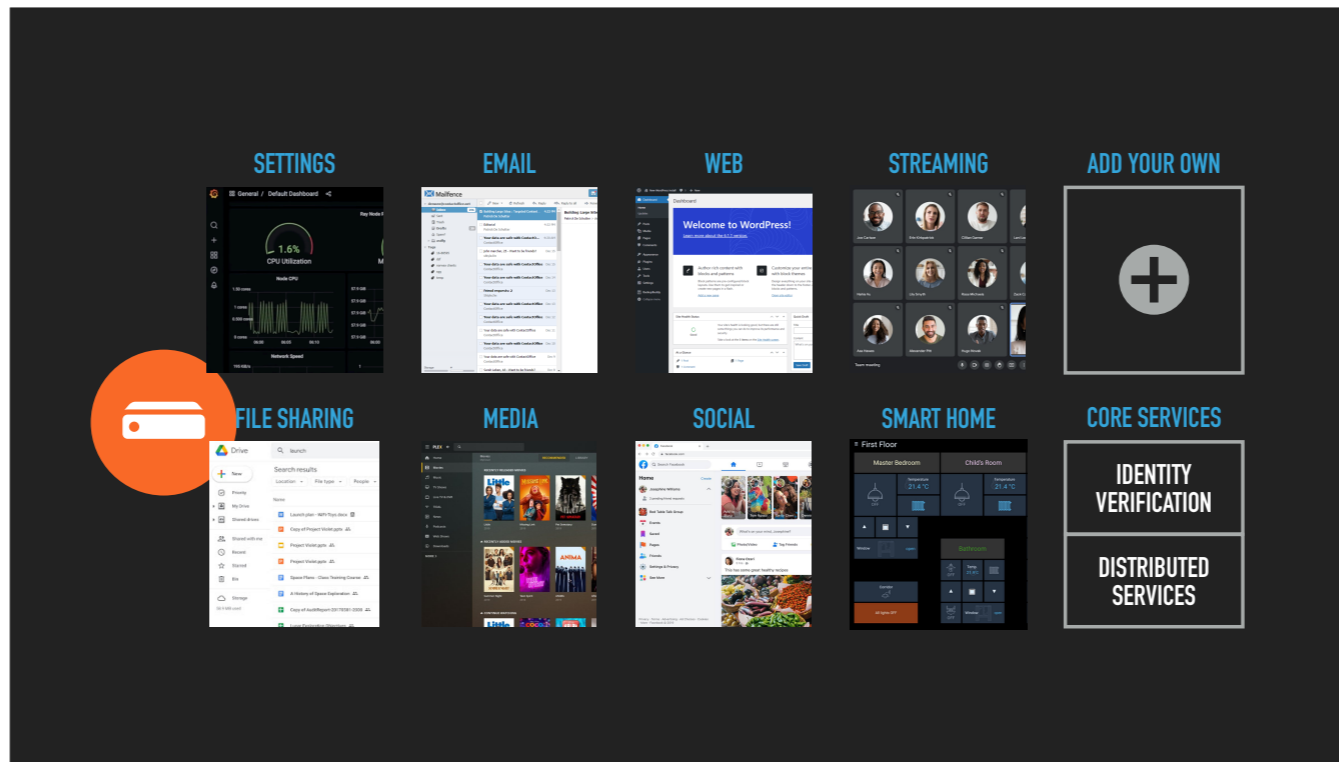
Finally, in addition to all that, this personal server has two unique core services to address issues of identity and inequality. The first is the **identity verification service**, and the second is the **distributed services manager**.

THE IDENTITY VERIFICATION SERVICE

This service's sole job is to verify to the world that you are the owner of a globally unique human identity, and it should be impossible for any other process in the world to do so without your consent. Like the other services, it runs all day, every day.
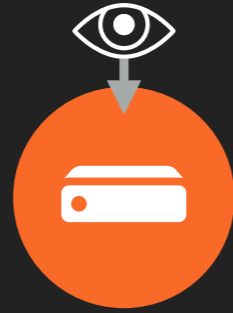
When you provided your biometric information to your server, it created a new identity.

THE IDENTITY VERIFICATION SERVICE
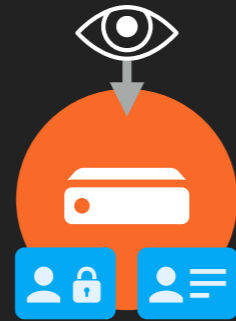
1 Biometric data is taken by the server

2 Public and private identity keys are made

This identity has two components: a private part which is stored securely within the server, and a public part which is then associated with your Internet services and all media and communications that you make available via the Internet. Here's how it works:

After you create your identity on your server, you can associate your other devices (like your phone or laptop) with that identity.

Now, when you upload an image onto a Web site from those devices, that media is associated with your identity. Your identity may be embedded in the media's metadata or may be invisibly watermarked into it.

THE IDENTITY VERIFICATION SERVICE
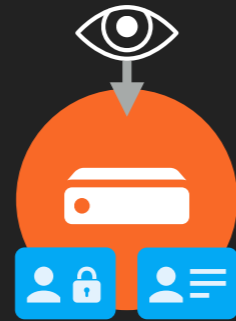
1 Biometric data is taken by the server

2 Public and private identity keys are made
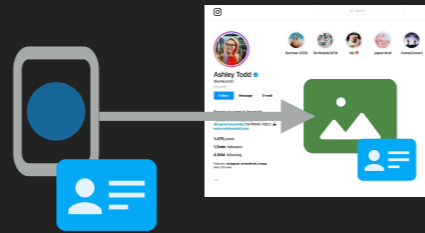
3 Your public key is put on your Web clients

4 Media uploaded to the Web includes your public key

5 Your server verifies that your identity is associated with that media

    Now anyone with access to that image can verify that you are a globally unique human being that owns that identity. Note that this does not mean that people can easily find out what you look like, where you live, or how you can be contacted, unless you choose to make that information public. Also note that anyone tampering with your identity information may be subject to identity theft laws.

    In this world, sites that accept user-generated content such as Facebook can force content to be associated with an identity and can verify it immediately at any time. Anyone viewing or copying such content can verify the identity associated with it and can face legal consequences if they alter that information.

    This system provides a basic foundation that allows media creators (such as photographers, musicians, writers, and actors) to open a dialogue with those who might wish to modify, purchase, sell, or resell their content.

    It provides a way for human-generated content to be decisively separated from AI-generated and AI-modified content, which can help establish trust and value and prevent AI training data from being polluted and self-referential.

    And it allows the formation of trusted, safe, and private communication spaces, from email threads and chat rooms to social media news feeds. You can verify that people are who they say they are while maintaining privacy.

## THE DISTRIBUTED SERVICES MANAGER

The next unique service addresses issues of inequality, and that is the distributed services manager.
This service performs the functions that a data center does, but in a distributed fashion. Let's examine how it works.

HOW DATA CENTERS WORK

SITE

We need to start by describing the things that data centers do that allow Web sites to "scale up", in other words, to handle massive amounts of users. In a typical data center, thousands of Web servers are placed in vertical stacks known as server racks.
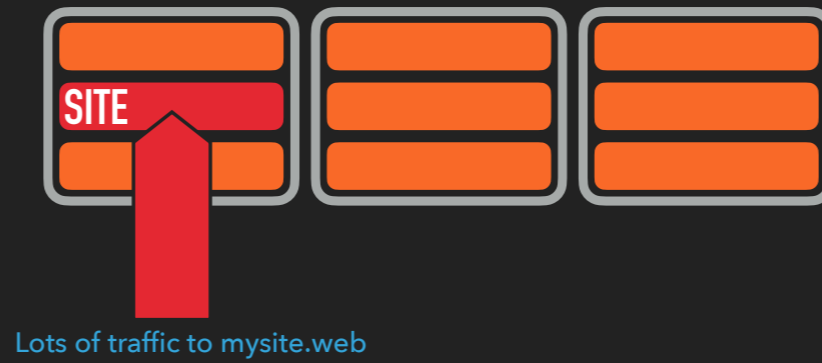
When a Web site receives too much network traffic than it can handle…
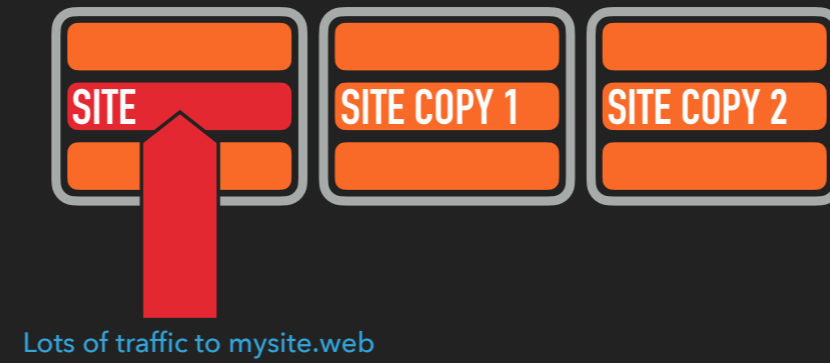
…exact copies of the site are created in a process called **mirroring**.
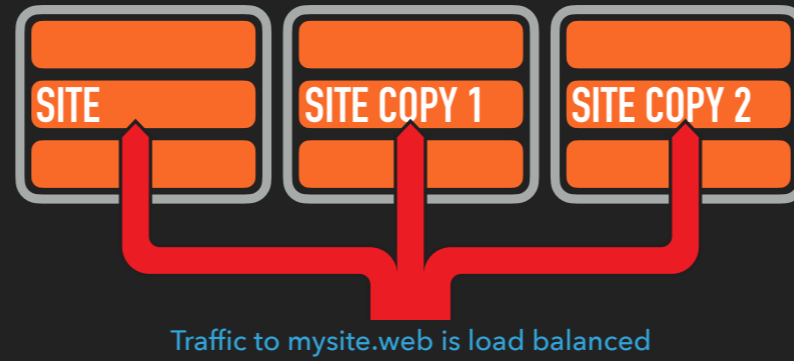
The network traffic is then split up and sent to these mirrored sites in a process called **load balancing**. Also, if a Web site requires too much computing power or storage space than it has available to it, it can then be transferred to a more powerful server that does.
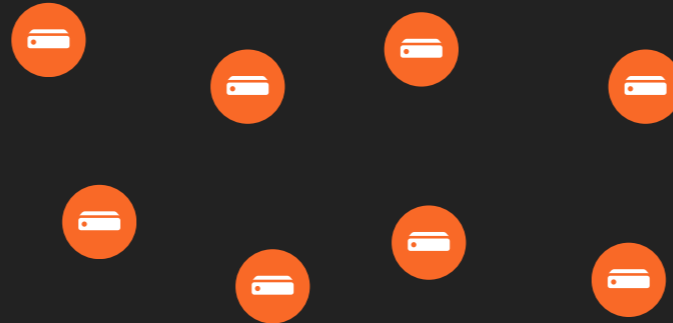
But in our new network of personal servers, how can we make this work? Instead of servers being placed in one location in a data center, they now exist in people's houses (and everywhere else), connected by the Internet, all over the world.

It can actually work in a very similar way, with a few twists.
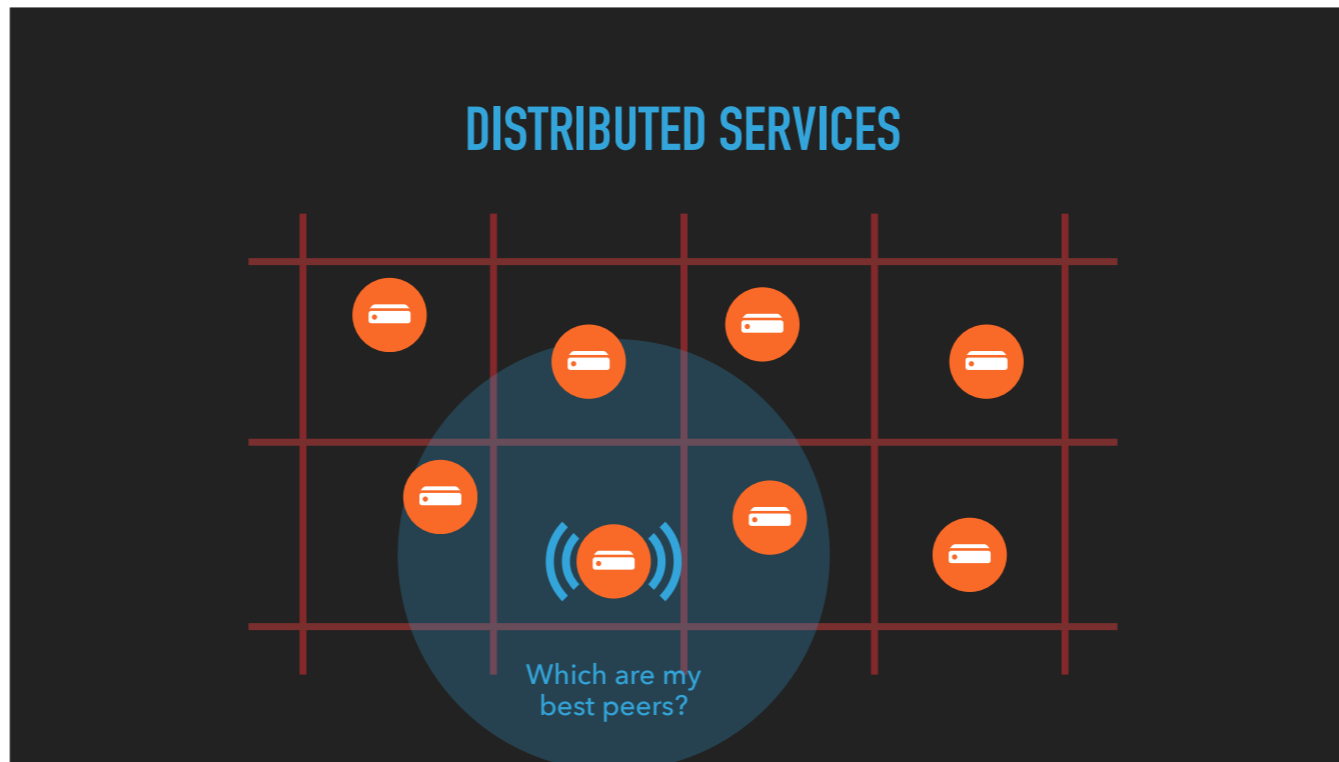
When a personal server starts up, and when needed, it sends a low power, high range signal to discover similar local devices within about a mile. Devices in the area respond with their Internet address, status, capabilities, and rough geographic location (which is kept to a resolution of about 1000 meters or half a mile for privacy).
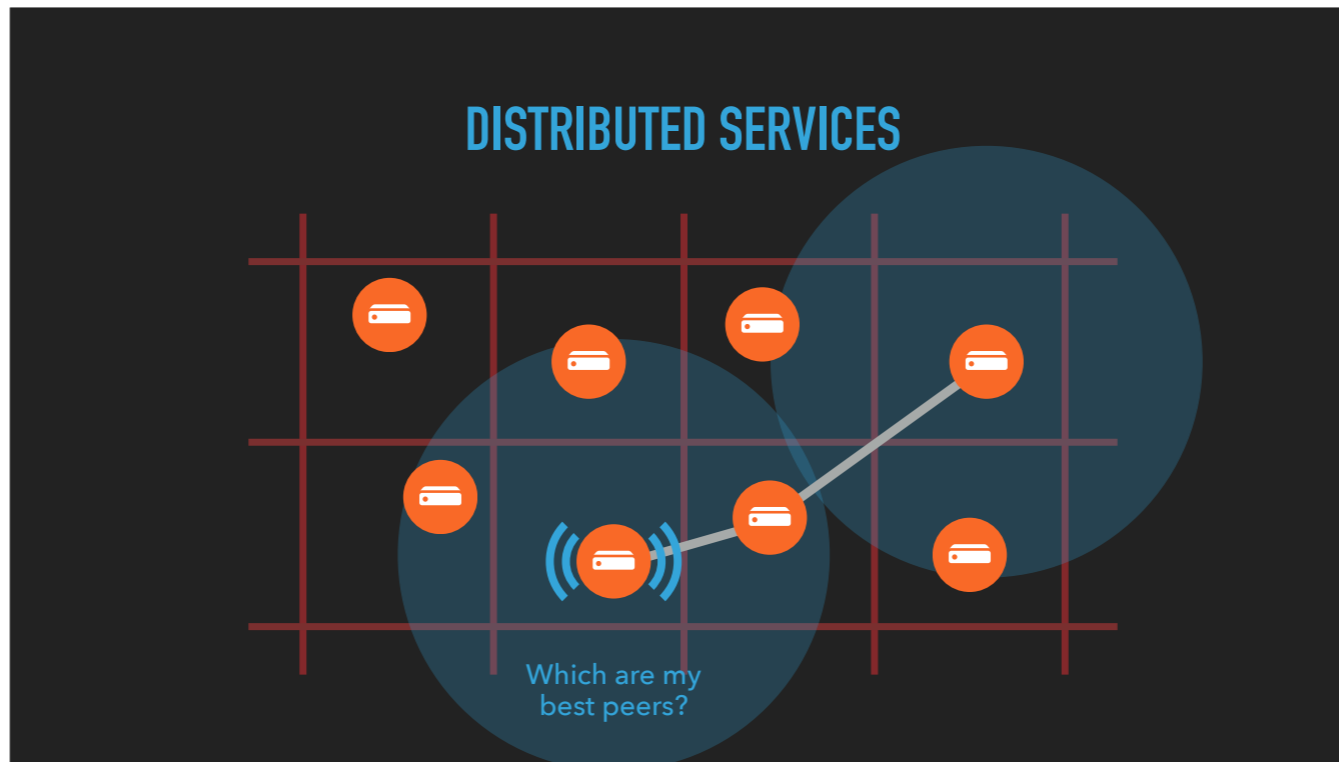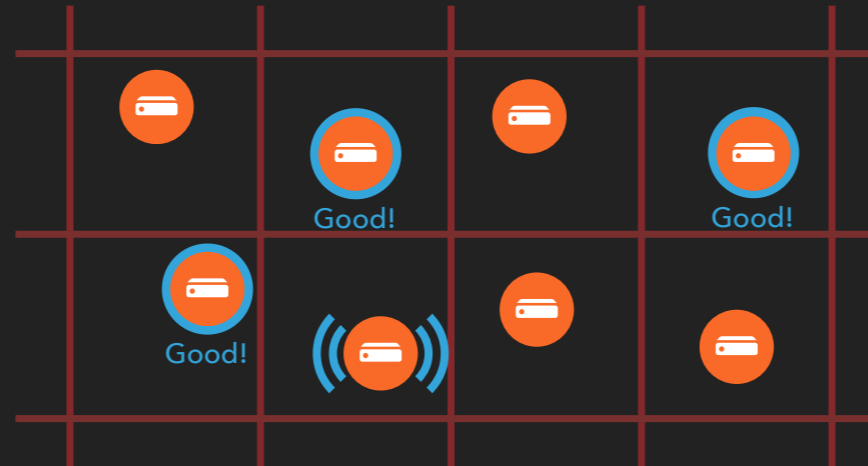
Lists of peers are shared, so servers further and further away can be discovered. One can think of this process as being similar to how slime molds branch out to search for nutrients.
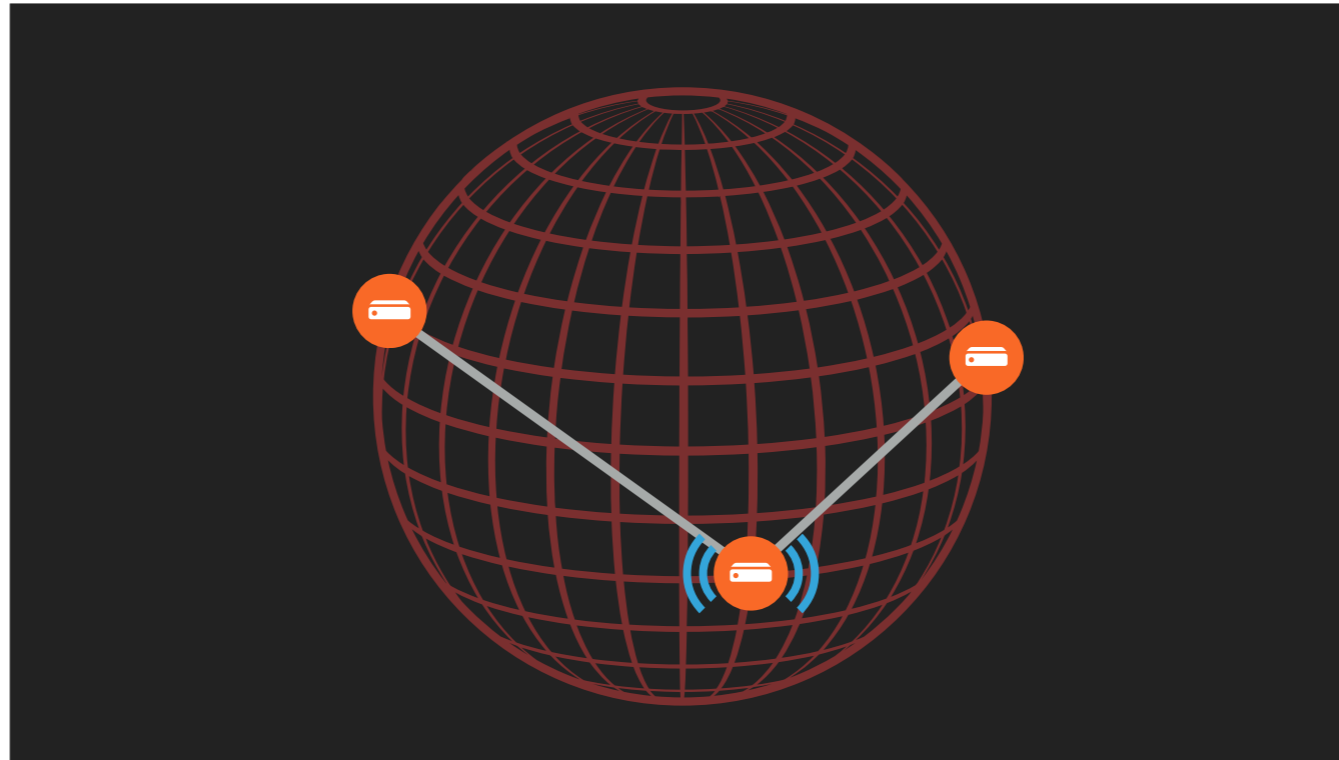
And by examining sources of traffic and broadcasting over regions of the Internet, peers that are globally distant can be discovered and selected as well.

In this way every server builds an internal list of its best local and global peers for any given task at any given moment, taking into account their historical and predictive availability, responsiveness, available computing power, available storage capacity, and location.
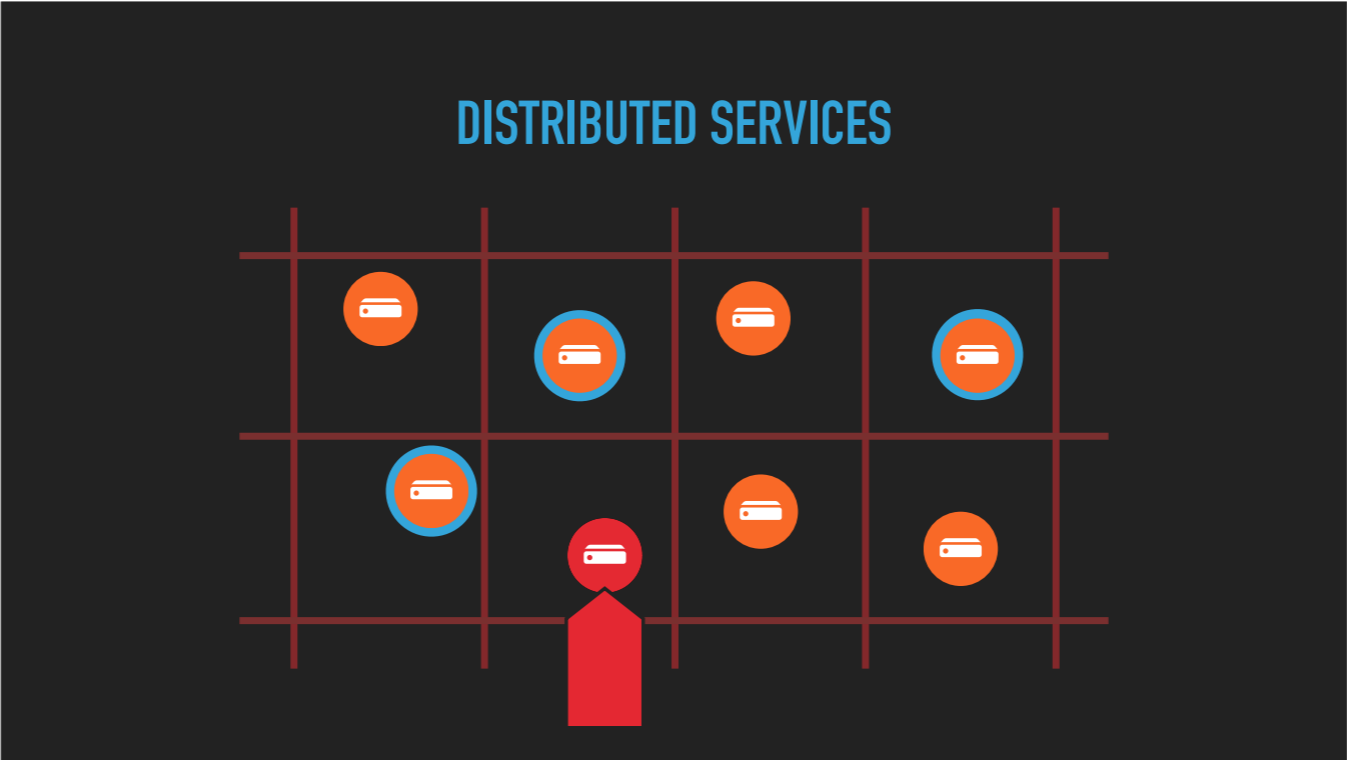
Now, when a Web site on your personal server receives more Internet traffic than it can handle…
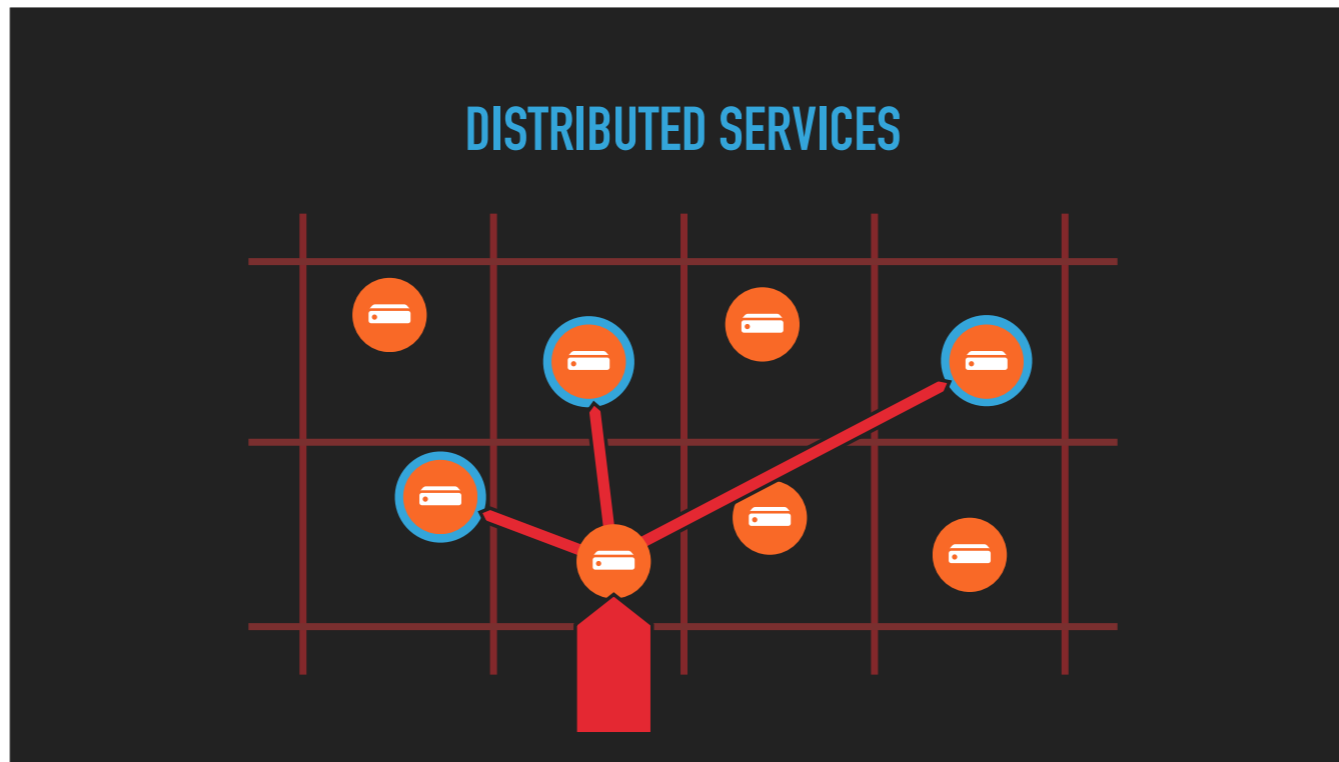
...it mirrors itself on the server's best peers chosen for the situation, both locally and globally, and splits up and redirects that traffic to them. This process is repeated until the traffic is handled adequately and reversed when the traffic dies down. In this way Web applications **expand** and **contract** across the public Internet to adjust to current demand.
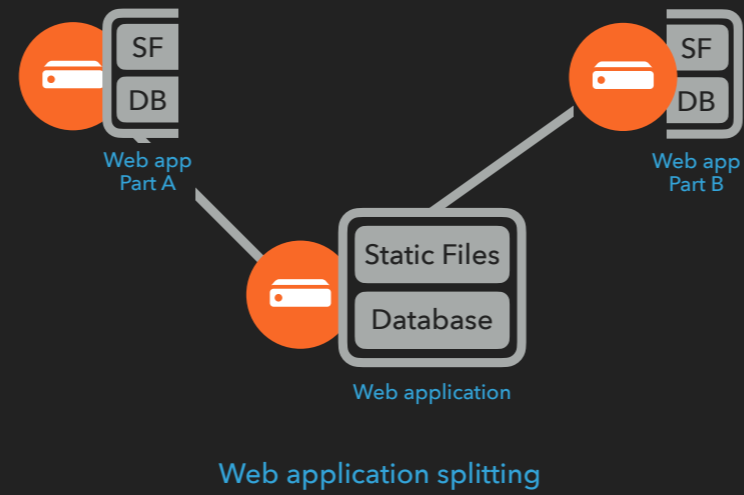
DISTRIBUTED SERVICES

Static Files

Database
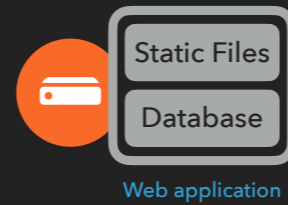
Web application

Web application splitting

Here's another method similar to those used in data centers. If your Web site takes up more computing power or storage space than one server can handle…

…it divides its code and its database among its best peers so its resource needs can be met. Think of it like cell division.

In this way, every personal server can provide its computing and storage resources to every other on the Internet, and they can work together in concert to do the kinds of things that data centers do. This is the promise of a more equitable, decentralized World Wide Web. Instead of the infrastructure owning you, **you** become the infrastructure.
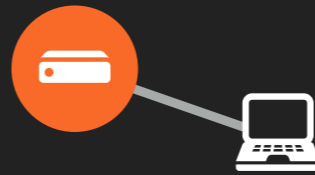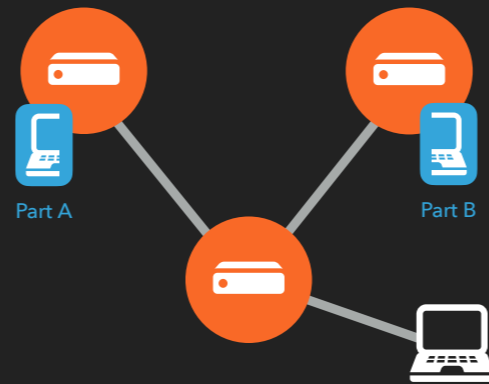
Here's another example - your personal server also runs a distributed backup service. When you create a new Web site or edit it…
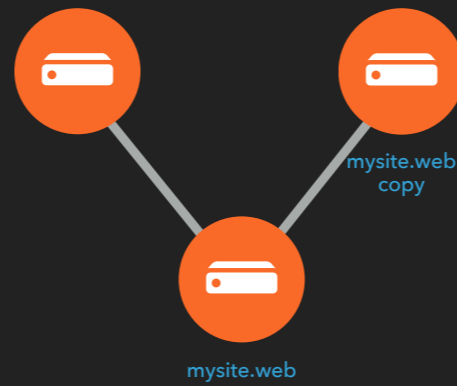
…its data is split and stored among other servers on the network, so you can retrieve different versions of it at any time. It can do the same for the data on your personal computer, phone, and all other devices, and you should be able to backup as much data as the amount of storage you make available for public use.

A decentralized Web is also fault tolerant - when you create a Web site, or modify any service's data, an up-to-date mirror in an optimal yet stable physical location is always created, so that if your server loses power or is unavailable for any reason, all of your Internet services (such as email) can still be accessed and used through other communications channels until normal operations are restored.
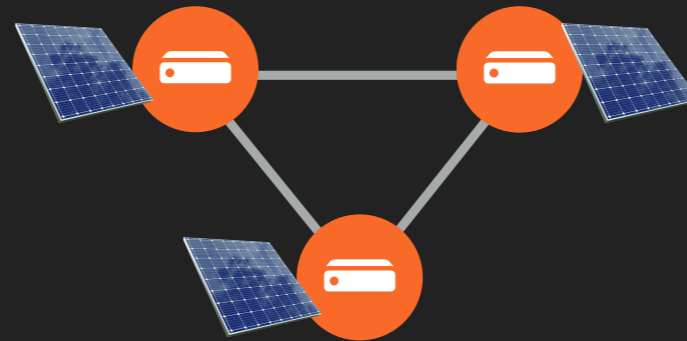
Because personal servers are designed to have a low power draw, a network of solar-powered personal servers can be quickly deployed to handle regional communications in times of emergency such as severe weather events, or can be used to run robust communications networks in hostile environments or places without Internet access or even electricity.

One day such networks may even provide municipal services such as emergency communications, meter reading, weather and earthquake sensors, or power outage reporting.

WORKING WITH DATA CENTERS
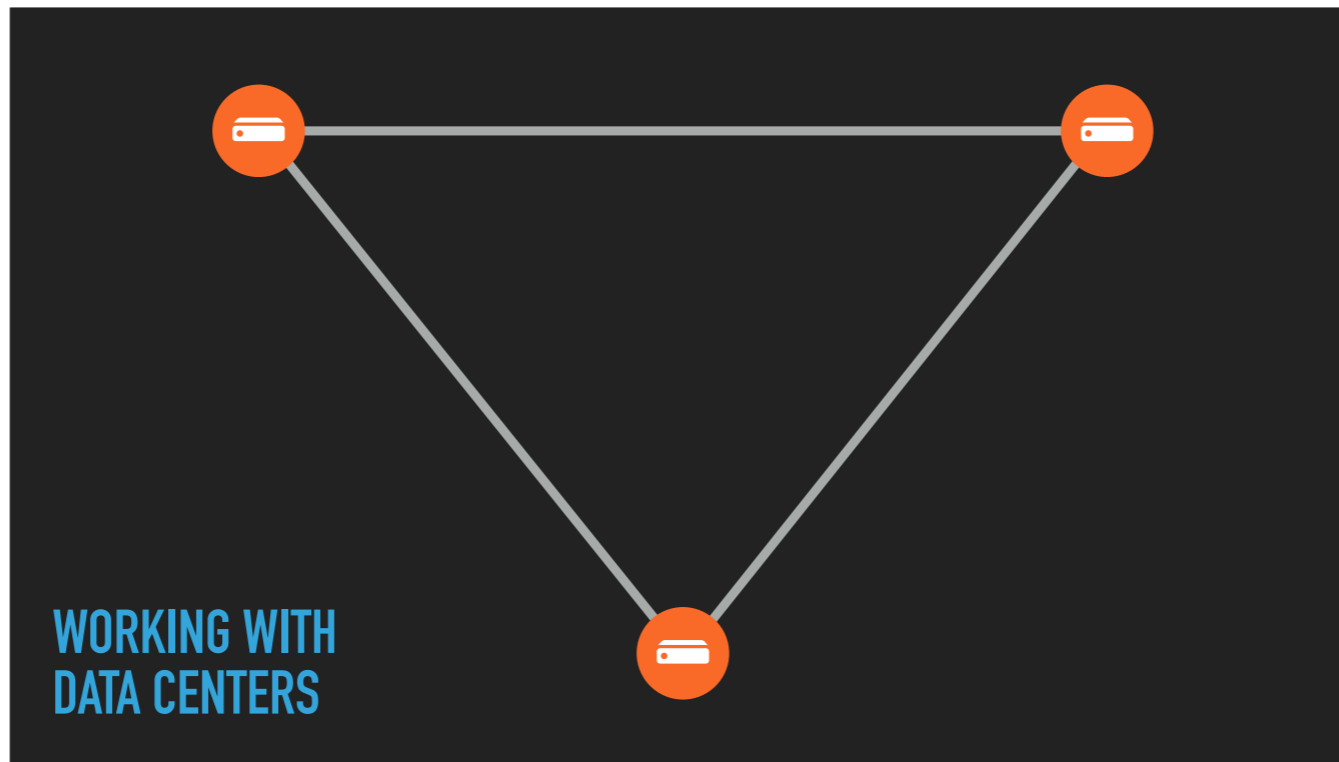
Having a decentralized Web does not mean that centralized services will go away. If your Web site ends up requiring more resources than the network of personal servers can handle…

Web apps seamlessly scale up to more capable networks of peers

WORKING WITH DATA CENTERS
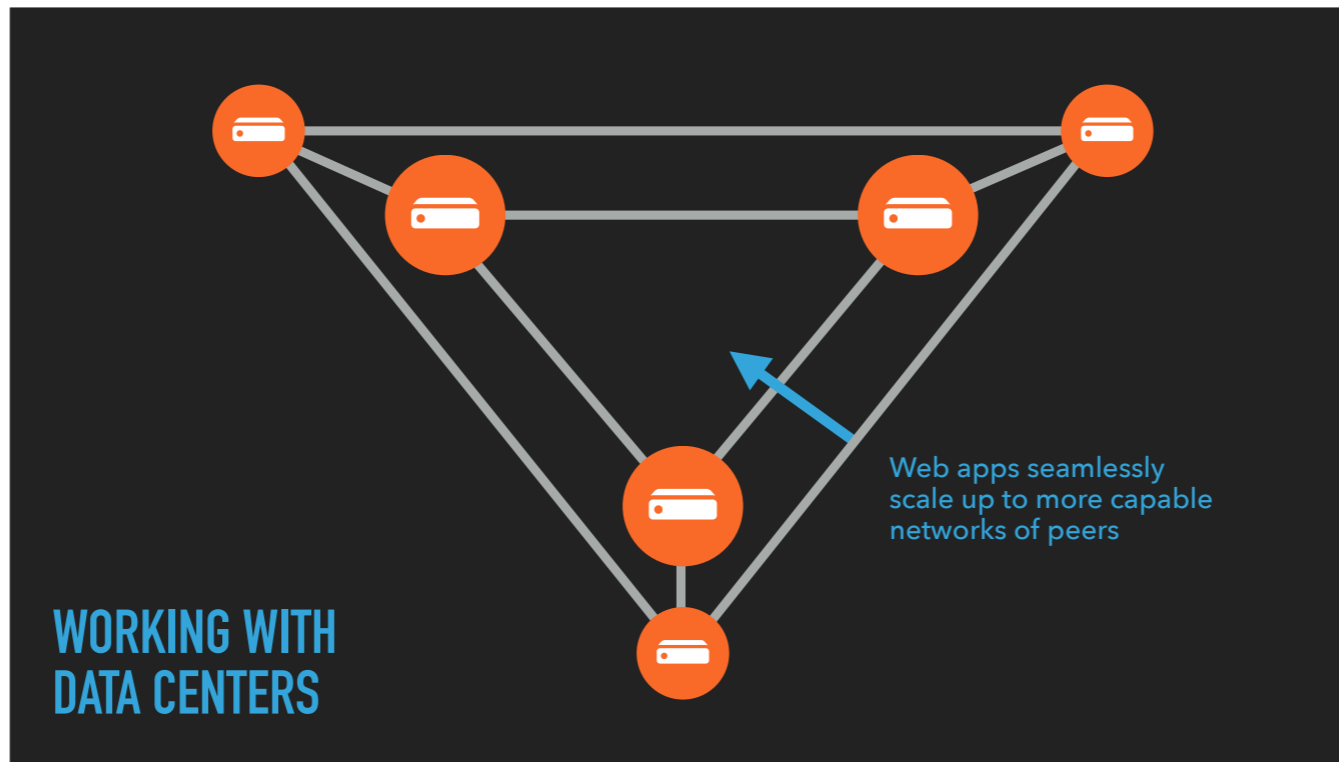
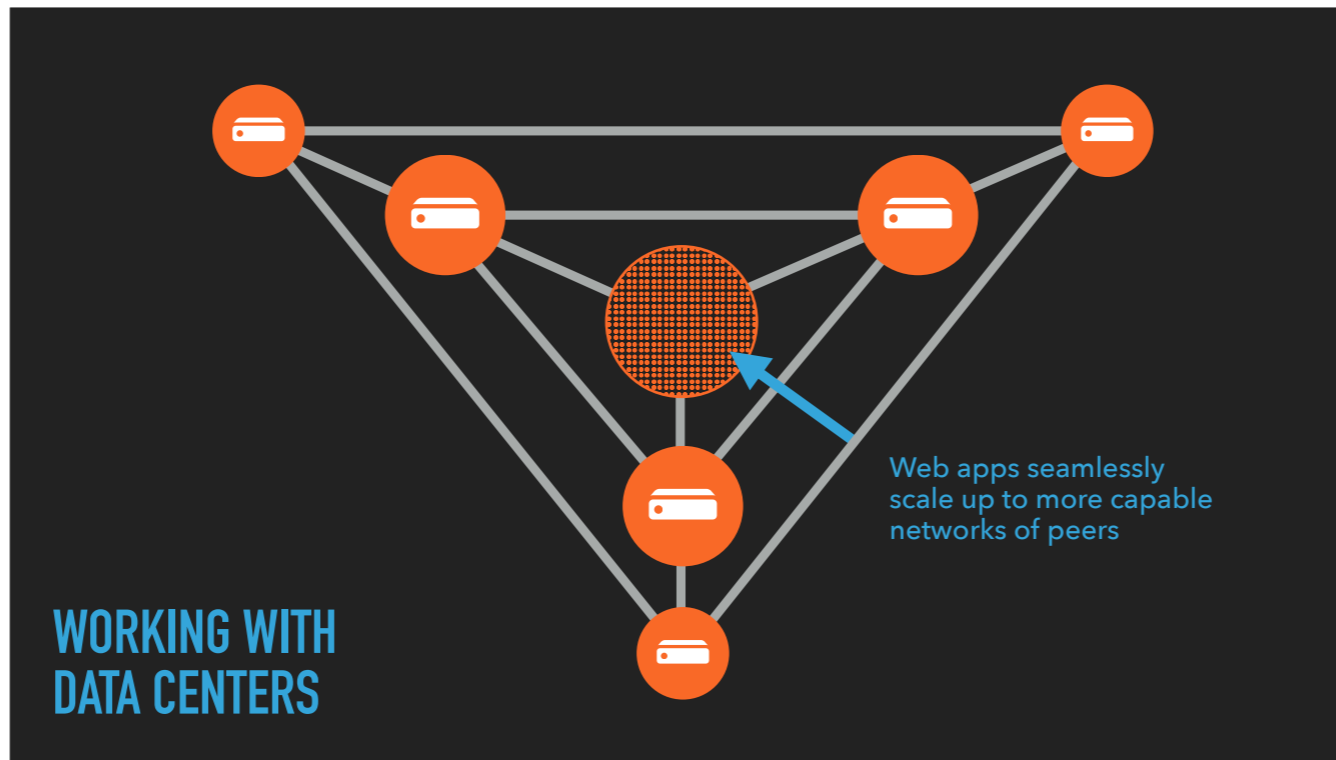…ideally it would make a seamless transition to a publicly owned network comprised of much more powerful servers, all completely powered by renewable energy sources.

WORKING WITH
DATA CENTERS

Web apps seamlessly
scale up to more capable
networks of peers

And from there it could scale up seamlessly to a commercial semi-centralized network of data centers.

In this way the burden of having to host small to medium-sized Web sites and applications is removed from the largest data centers, which can then focus solely on the largest clients and applications.

This makes sense considering that at least 40% of all Web sites, including their associated code and data, are only about 1 gigabyte in size.

The entire infrastructure thus becomes more energy-efficient, cost-effective, and equitable.
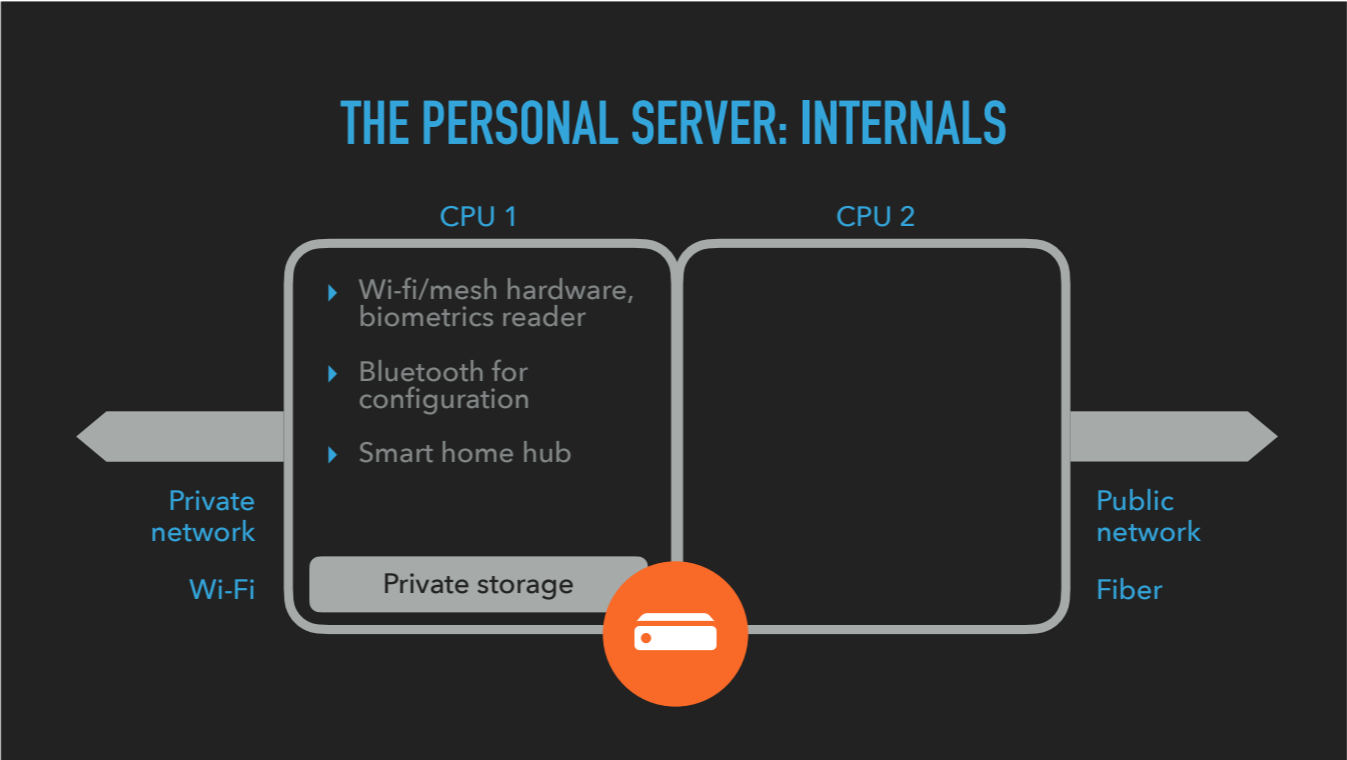
THE PERSONAL SERVER: INTERNALS

So now you've seen what it can do, I'm briefly going to go over the parts a personal server device needs in order to make this all work.

THE PERSONAL SERVER: INTERNALS

CPU 1   CPU 2

Private network

Wi-Fi

Public network

Fiber
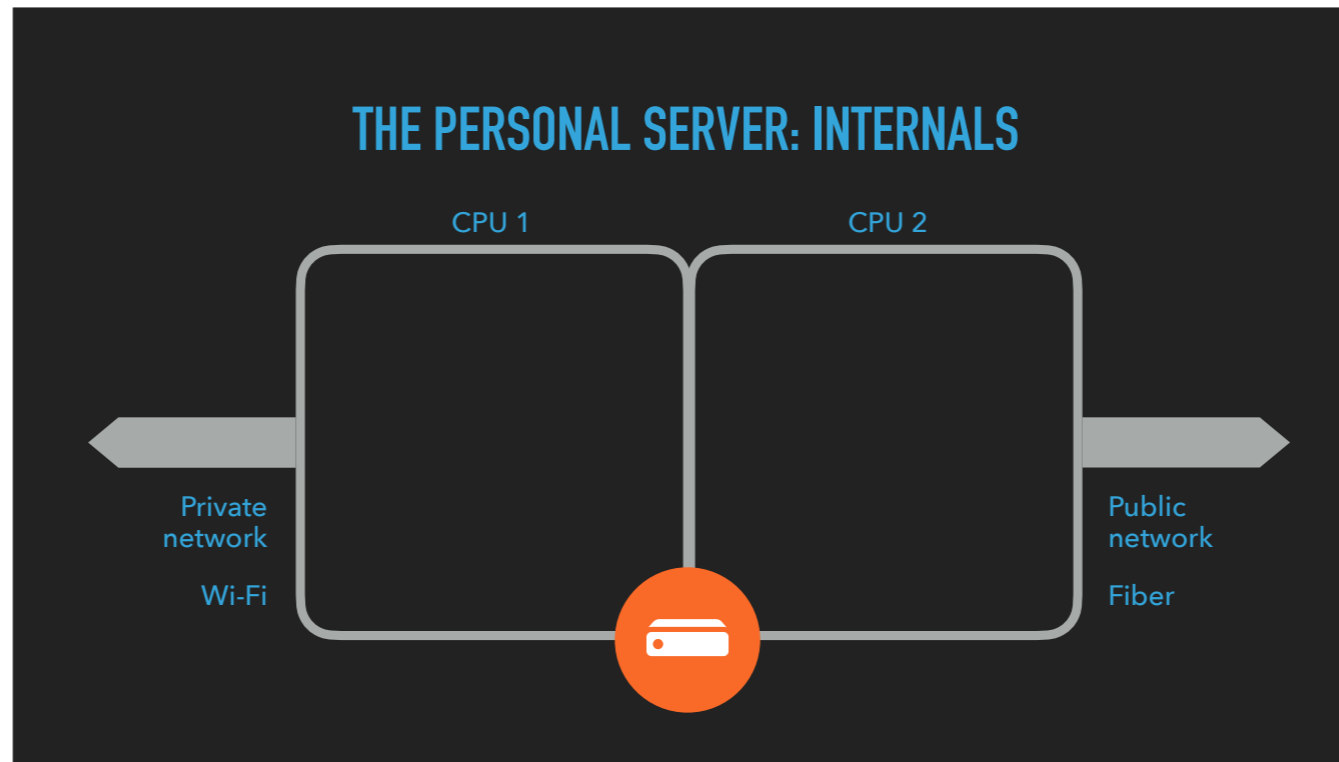
The personal server is actually two computers in one: One manages your private network and data, and the other manages data and services exposed to the public Internet. They are separated by hardware and software barriers where appropriate to maintain a high level of security and responsiveness.

Although wi-fi routers and home servers are currently considered separate devices, they will eventually converge into one device so that it will be impossible to imagine one without the other, in the same way that we cannot imagine having a smart phone without a digital camera today, for the same reasons: combining them enables unprecedented ease of use and radically new social and technical capabilities that were previously impossible.

THE PERSONAL SERVER: INTERNALS

CPU 1

CPU 2

- Wi-fi/mesh hardware, biometrics reader
- Bluetooth for configuration
- Smart home hub

Private network

Wi-Fi

Private storage

Public network

Fiber

The private side contains computing and storage resources to manage your internal network, encode and decode streaming data, and perform smart home tasks, among other things.

THE PERSONAL SERVER: INTERNALS

CPU 1

CPU 2

- Wi-fi/mesh hardware, biometrics reader
- Bluetooth for configuration
- Smart home hub

Cryptoprocessor

Private storage

Private network

Wi-Fi

Public network

Fiber
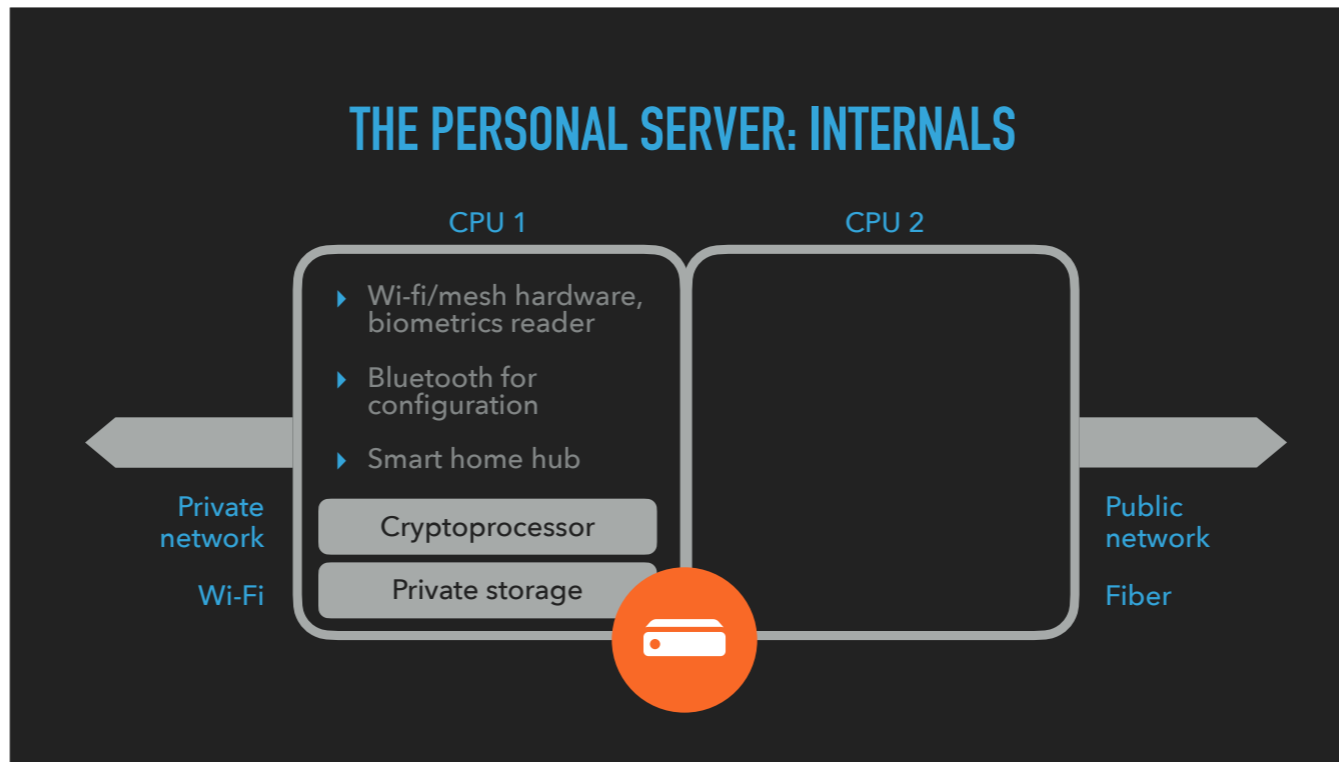
It also contains a chip called a **cryptoprocessor**, which securely stores private data relating to your identity and other confidential information.

THE PERSONAL SERVER: INTERNALS

CPU 1
- Wi-fi/mesh hardware, biometrics reader
- Bluetooth for configuration
- Smart home hub

Cryptoprocessor

Private storage

CPU 2
- Routing & services
- Low-power peer discovery/networking
- Distributed services management

Public storage

Private network
Wi-Fi

Public network
Fiber
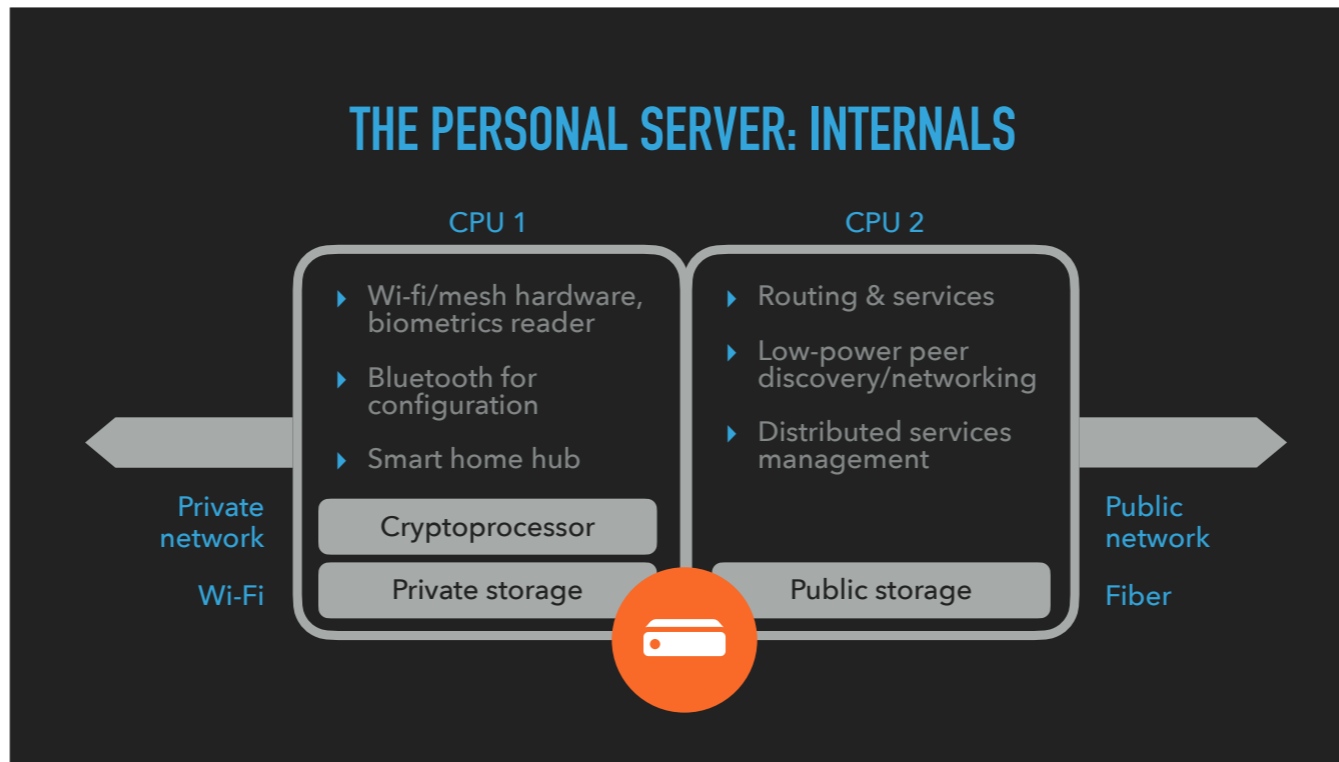
The public side hosts your own Web sites and applications, but it also contains public computing and storage resources - data from the distributed network is placed here as needed so your device can help mirror, redirect, backup, or partially host other people's services and applications. All data is securely encrypted so that you cannot determine what it is or where it comes from - it is part of the public infrastructure.

The PERSONAL SERVER: INTERNALS
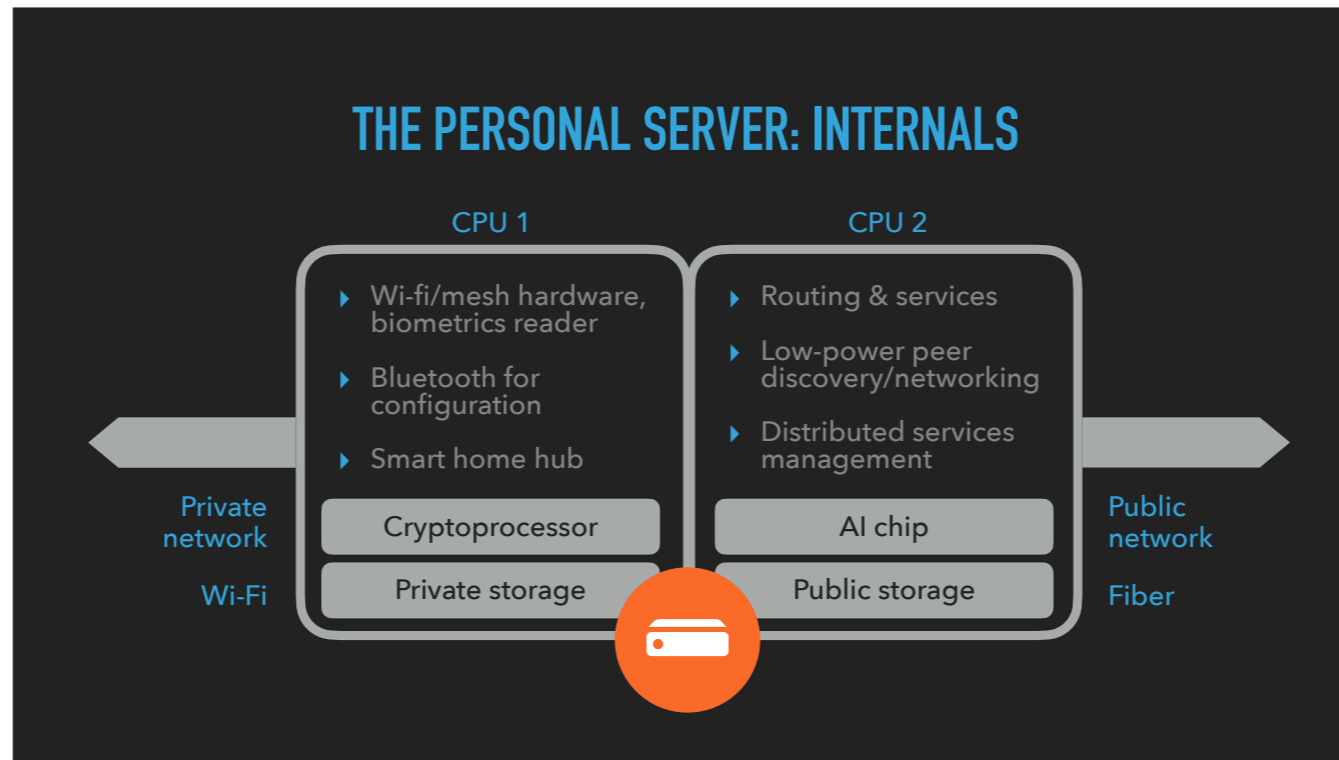
The public side also contains an **AI chip**, typically a GPU (graphics processing unit) today, which uses machine learning and other techniques to predict and optimize resource use and network communications. It determines how and when to optimally choreograph, copy, divide, and distribute Web applications across the network.

This device runs an operating system written in a low-level memory-safe language such as Rust. All services use a common **service API** and **database API**. This allows any Web application to be optimally split apart so it can be scaled up in a federated fashion and run on the distributed Web.

Because these parts make up critical infrastructure, the system and all core services should be open source and signed and verified by a managing organization.

THE PERSONAL SERVER: ANCESTORS

We've seen devices similar to this before that could be considered ancestors. 25 years ago, thousands of companies ran one of these:

COBALT QUBE, 1998

19cm (7.5")

150 MHz, 256MB RAM,
10GB drive, 25W, $1,000

This little server was called the Cobalt Qube, and it ran Web, email, and file sharing services. It drew 25 watts of power and cost one thousand dollars. Twenty-five years later, this is what it became:

COBALT QUBE, 1998

19cm (7.5")

150 MHz, 256MB RAM,
10GB drive, 25W, $1,000

RASPBERRY PI, 2023

8.5cm (3.3")

1.8 GHz, 2GB RAM,
no storage, 1W, $35

…the Raspberry Pi. Managed by the Raspberry Pi Foundation, it is ten times faster than the Qube, with ten times the memory, drawing one tenth the power, and costs $35. Looking ahead 25 years from now, what might the personal server's specs look like?

It may very well be over ten times faster, with over ten times the memory, while staying affordable and energy-efficient enough to be a staple of the home in 2050. But will we really have to wait another 25 years for it?

It is inevitable that this device will be built at some point. Mainstream computing devices already incorporate energy-efficient CPUs, GPUs, biometric readers, and cryptoprocessors.

It could be built by any well-known company, but it would be far better to have it shepherded by a non-commercial open source organization. What's the alternative?

This device represents a unique mix of hardware and software that must work together closely, making it a very challenging goal for an open source organization to achieve. Any commercial organization able to build it would inevitably want to leverage it to encourage ecosystem lock-in as we see today with residential technology built by Apple, Google, and Amazon. It would be more likely to be expensive and proprietary, thus reintroducing the kind of inequity this system was created to prevent.

This is also why standards, protocols, and policies must be designed so the system naturally encourages a virtuous cycle of equity, power efficiency, and decentralization, versus runaway accumulations of wasteful centralized power and resources.

WHAT DO WE NEED TO MOVE FORWARD?

What do we need to move forward? To begin with, we're going to have to do a lot to address fundamental Web inequality.

WHAT DO WE NEED TO MOVE FORWARD?

▸ Policies that treat the Internet as a universal utility

We need policies that treat Internet access as a universal utility, not a commercial service that only the privileged can access.

WHAT DO WE NEED TO MOVE FORWARD?

▸ Policies that treat the Internet as a universal utility

▸ Policies that define and provide minimum levels of symmetrical Internet access to everyone

We need policies that define and provide minimum levels of symmetrical Internet access to everyone. We currently have a 25-year-old global digital highway where one lane goes **6%** the speed of the other. Again, **this is no way to build critical infrastructure.** It is impossible to make a usable, equitable, decentralized World Wide Web without **symmetrical Internet access.** This is one of the key blockers to this vision.

In the US in 2022 the FCC floated a goal of 1Gbps connections for all citizens. This is a moonshot-level challenge, and it can be accomplished in our lifetimes if we make our voices heard.

WHAT DO WE NEED TO MOVE FORWARD?

▸ Policies that treat the Internet as a universal utility

▸ Policies that define and provide minimum levels of symmetrical Internet access to everyone

▸ Policies that encourage and fund community-driven and municipal networks

We need policies that encourage and fund community-driven and municipal networks, particularly in monopolized markets. This can help bring affordable, high-quality access to rural and low-income areas.

WHAT DO WE NEED TO MOVE FORWARD?

▸ Policies that treat the Internet as a universal utility

▸ Policies that define and provide minimum levels of symmetrical Internet access to everyone

▸ Policies that encourage and fund community-driven and municipal networks

▸ A non-commercial organization to research, develop, test, harden, and standardize decentralized Web server platforms, architectures, and APIs

Finally, we need a non-commercial organization to research, develop, test, harden, and standardize decentralized Web server platforms, architectures, and APIs. This is a new world of research open to all manner of possibilities, and touches on biometrics, decentralized cryptography, service choreography, machine learning, new container technology, and new database architectures, to mention just a few areas that we'll need progress in to enable this vision.

THE WEB IS FOR
EVERYONE

The Web is for everyone. Let's build the next 25 years of it using what we've learned from the last 25. Thank you.

WEB 2050

KEVIN HUGHES

KEV@KEVCOM.COM

kevcom.com/web2050

Kevin Hughes is an internationally-recognized pioneer in Web design and software. He is one of six members of the World Wide Web Hall of Fame and created "imagemap", which enabled interactive images on the Web. He contributed to the Web's CSS (Cascading Style Sheets) specification, designed the first shopping site on the Web (the Internet Shopping Network), and made the icons for the Apache Web server. He lives in Honolulu, Hawaii.
Feel free to share the link to this talk publicly to foster the discussion of these concepts:

kevcom.com/web2050