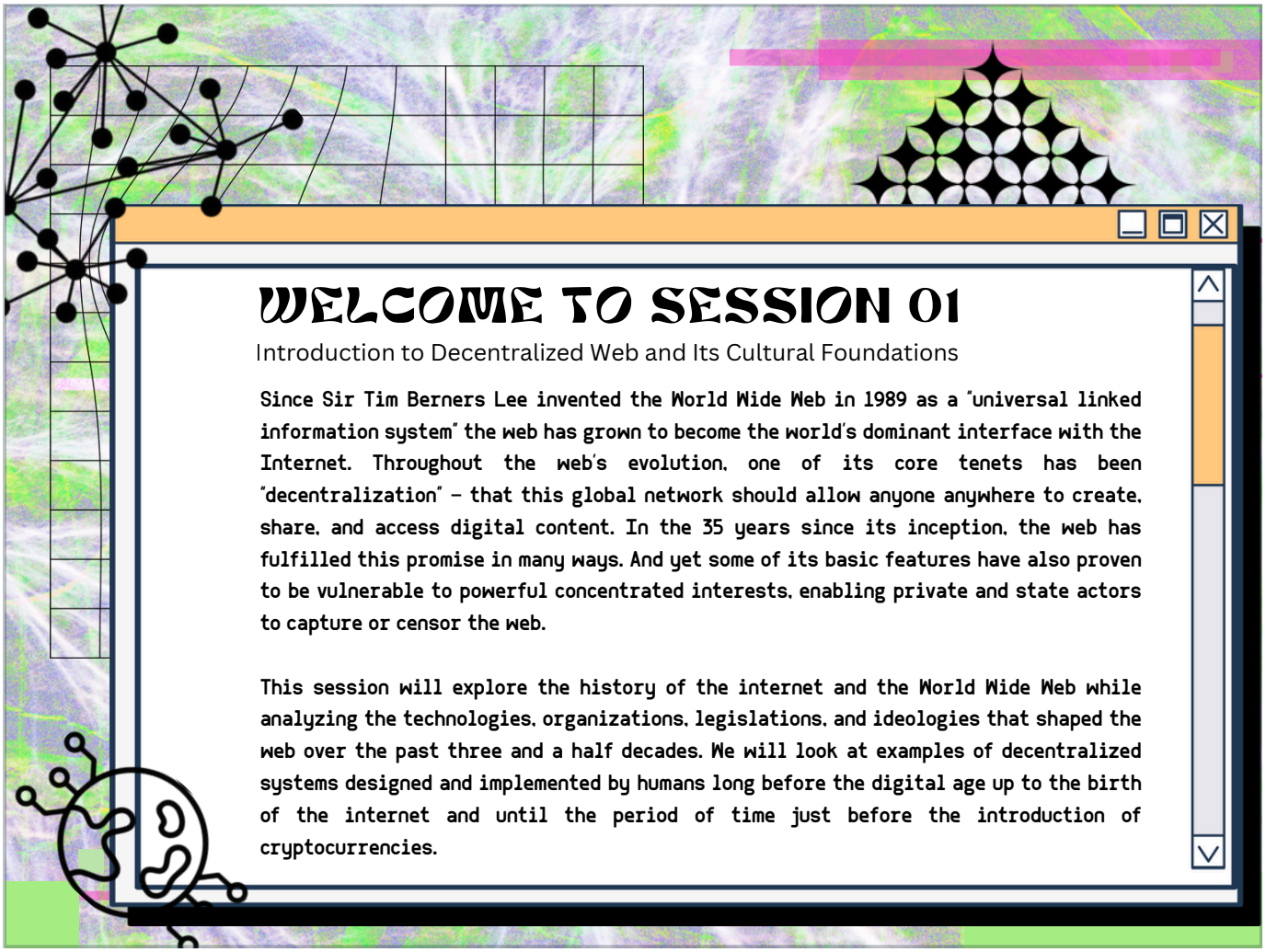
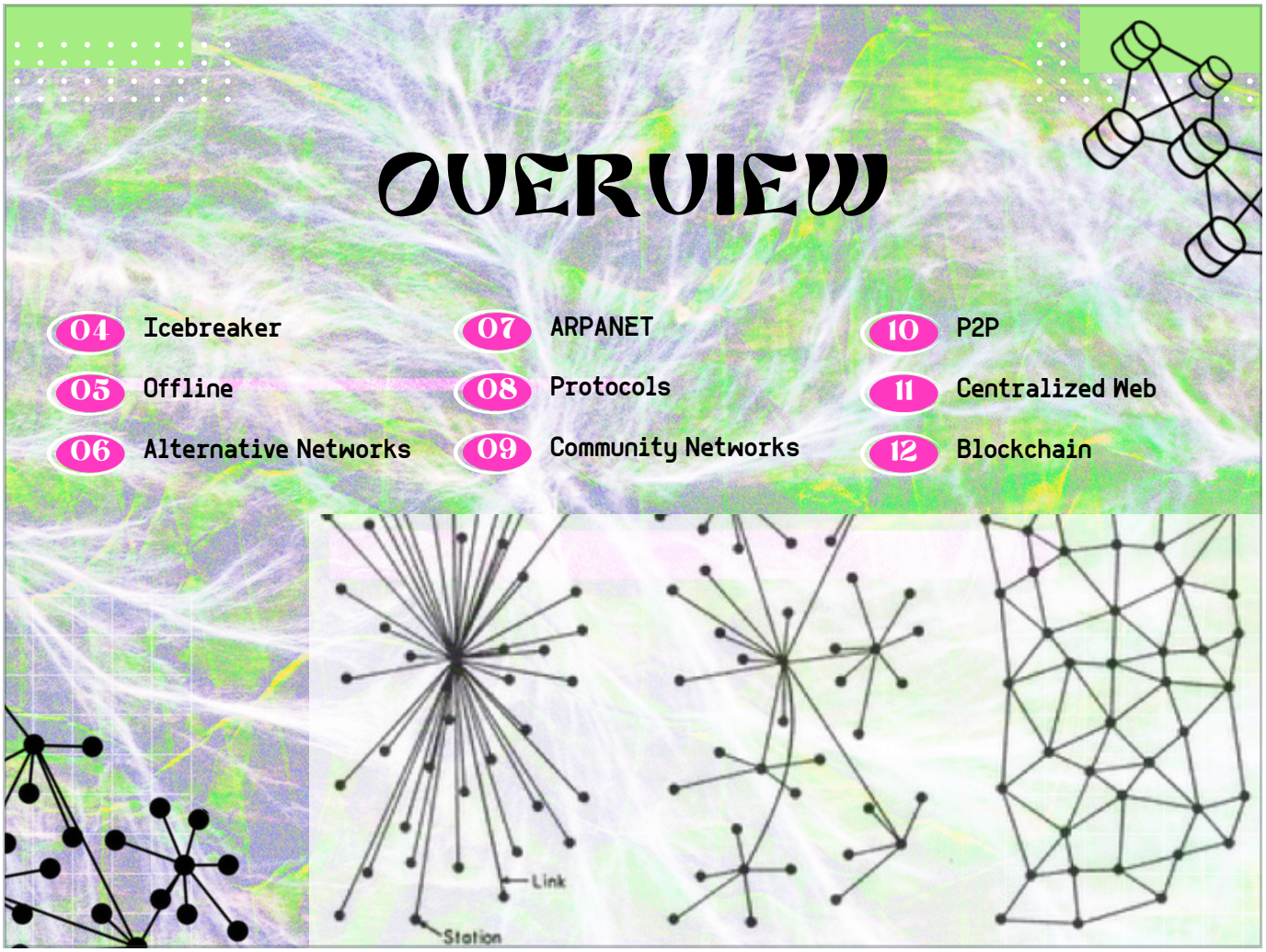


Sir Tim Berners-Lee is recognized for inventing the World Wide Web (WWW) in 1989 at CERN in Geneva, Switzerland. He introduced the first web page on August 6, 1991: https://info.cern.ch/hypertext/WWW/TheProject.html, which was hosted on a NeXT computer adorned with a sign that read, “This machine is a server. DO NOT POWER IT DOWN!!” This marked the beginning of the World Wide Web.

One of the foundational principles of the internet is decentralization, with the web acting as its graphical user interface. This concept of enabling individuals everywhere to create and share content has been essential to the web’s expansion. However, as we will discuss, while the web has largely delivered on this promise, it has also encountered challenges posed by powerful entities trying to control or censor it.

We will examine the key technologies, organizations, and ideologies that have influenced its development over the past 35 years, from early decentralized systems to the emergence of cryptocurrencies.



Before getting into the slides, we would like to provide an overview of the agenda, starting with an icebreaker to get to know a little bit about everyone who is attending the live session.

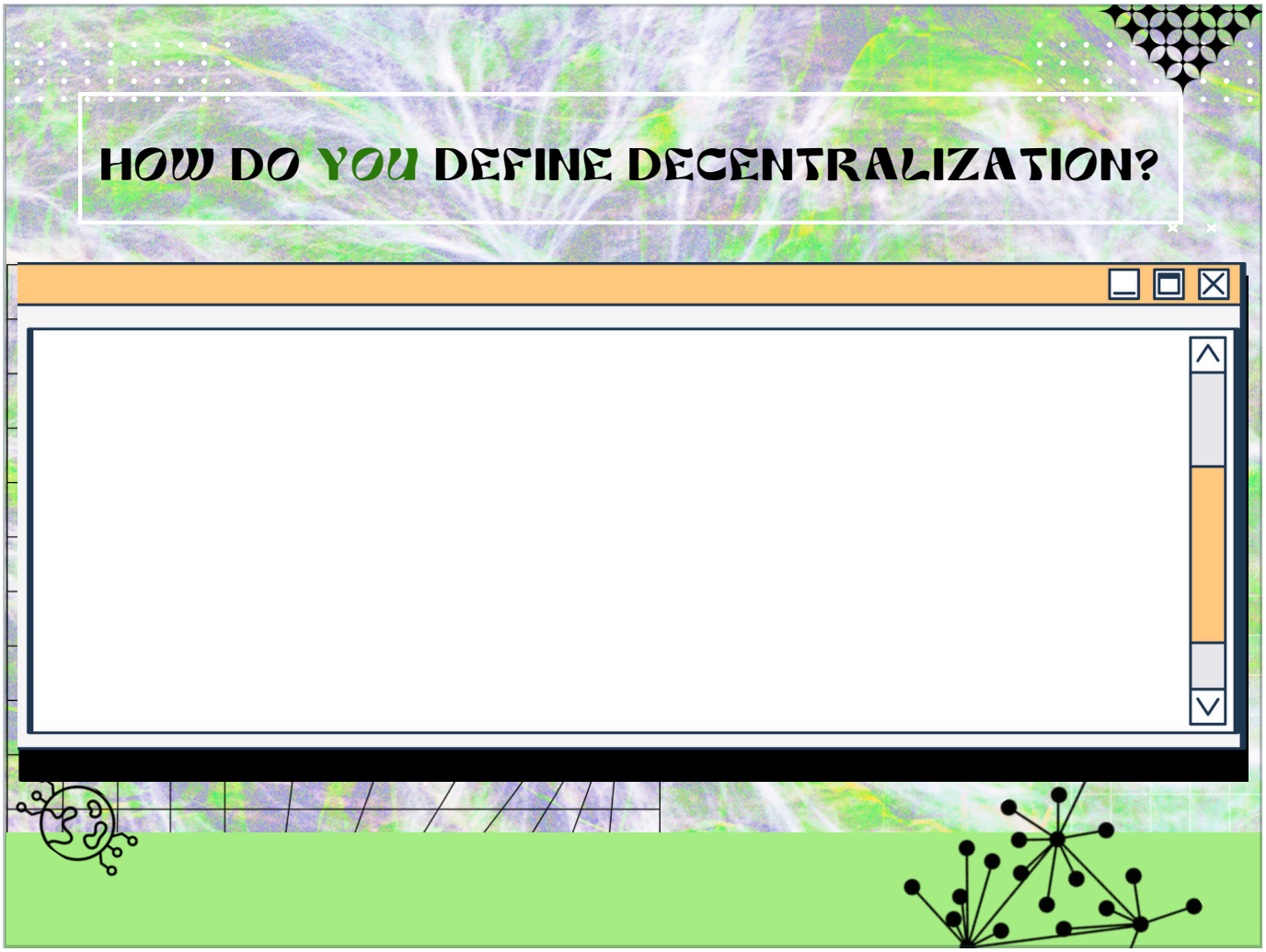
Next, we'll discuss "Offline" and "Alternative Networks," exploring how these concepts contribute to our understanding of decentralization.

Then we will review the early days of the internet with ARPANET and examine the critical role of protocols.

We will also discuss "Community Networks" and "Peer-to-Peer (P2P)" systems, showcasing how communities and individuals have shaped decentralized networks.

Finally, we'll contrast this with the "Centralized Web" and introduce "Blockchain," setting the stage for deeper discussions later.

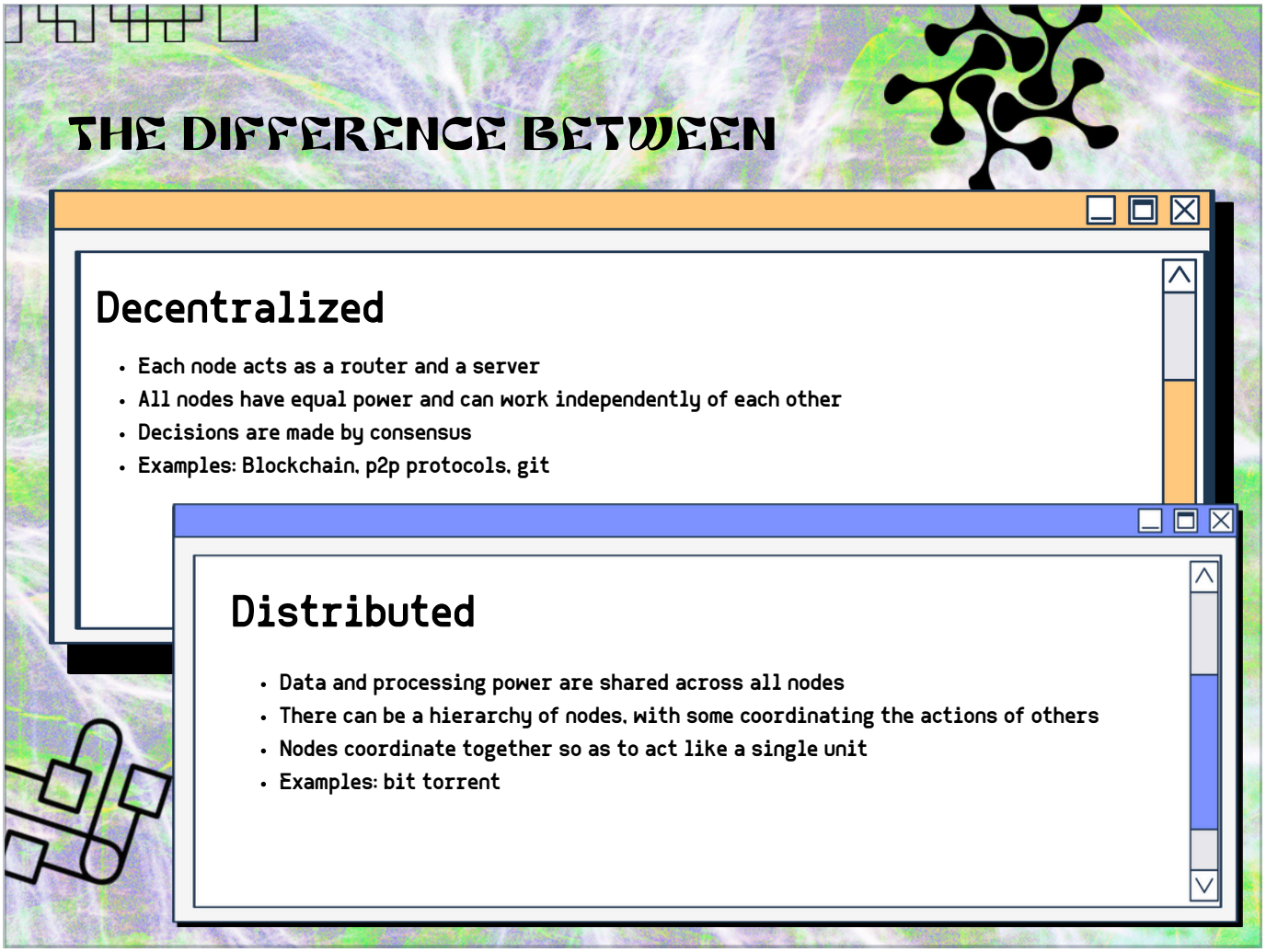
Each of these topics will help us build a comprehensive understanding of the decentralized web and its cultural foundations.



What does "decentralization" mean? At its core, decentralization is about distributing authority and decision-making across multiple points rather than having a single point of control.

This concept isn't just limited to technology. It’s about empowering individuals and creating resilient systems. Think about how communities or organizations where everyone has a say might function.

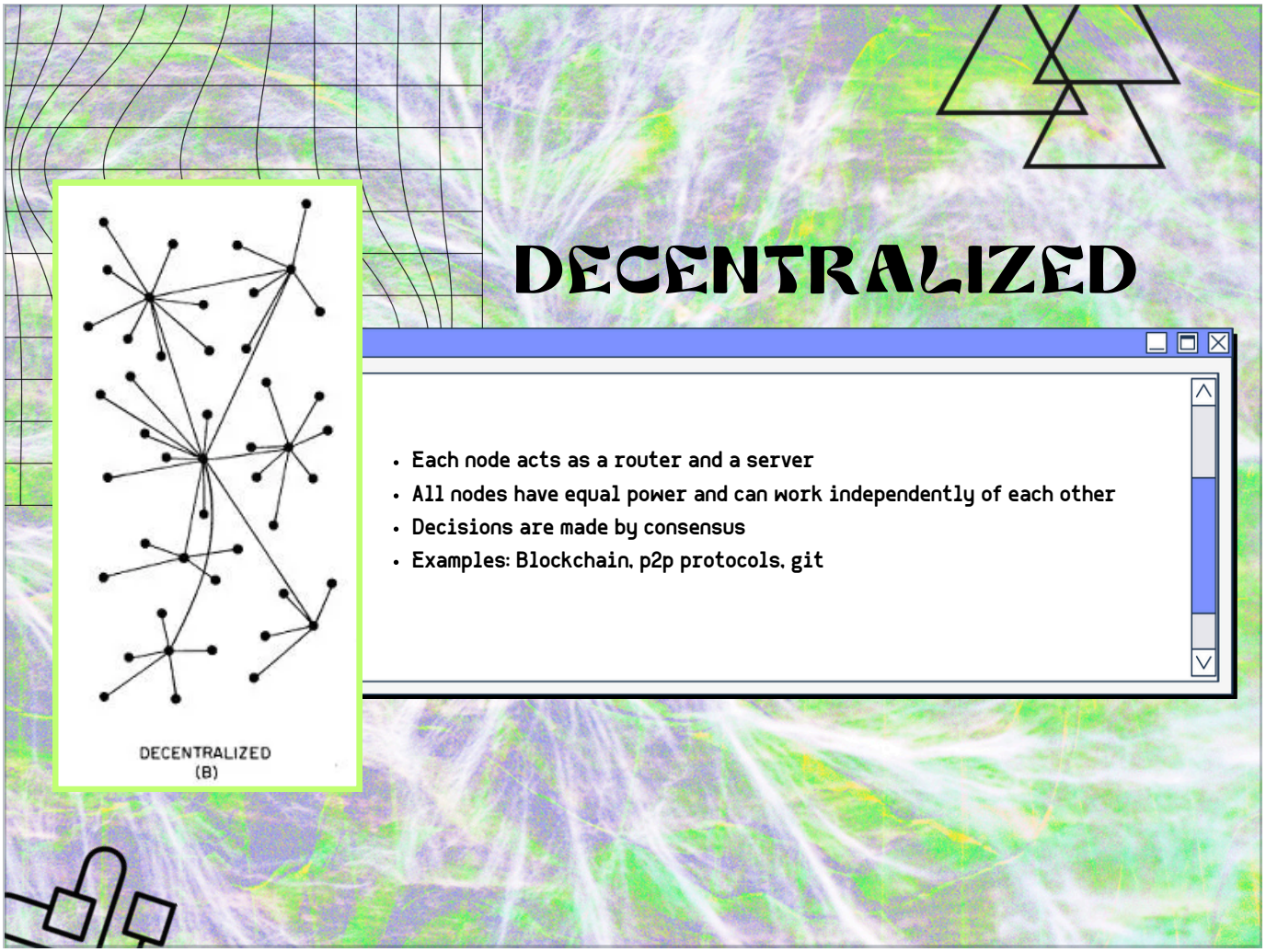
In the digital realm, decentralization allows for systems where each participant, or node, has equal power. This means that these nodes can operate independently but still contribute to the network's overall functionality.



In a decentralized system, each node operates both as a router and a server. Every node holds equal power and can function independently. Decisions in such a system are made by consensus, ensuring that no single node has overarching control. Think of examples like Blockchain, peer-to-peer protocols, and Git.

In distributed systems, the data and processing power are shared across all nodes. Although there's a collaboration, you might find a hierarchy where some nodes manage and coordinate others, working together as a cohesive unit. BitTorrent is an example of this.

Notice the key differences: the independence and equality in decentralized systems versus the possible hierarchy and coordination in distributed ones. These distinctions are critical in understanding how systems operate and collaborate.



Going deeper into decentralized networks -

First, each node acts as both a router and a server. This means every part of the network can transmit and receive data, enhancing resilience. For example, in BitTorrent file sharing, your computer both downloads pieces of a file from others and simultaneously uploads pieces to other users, making you both a client and a server.

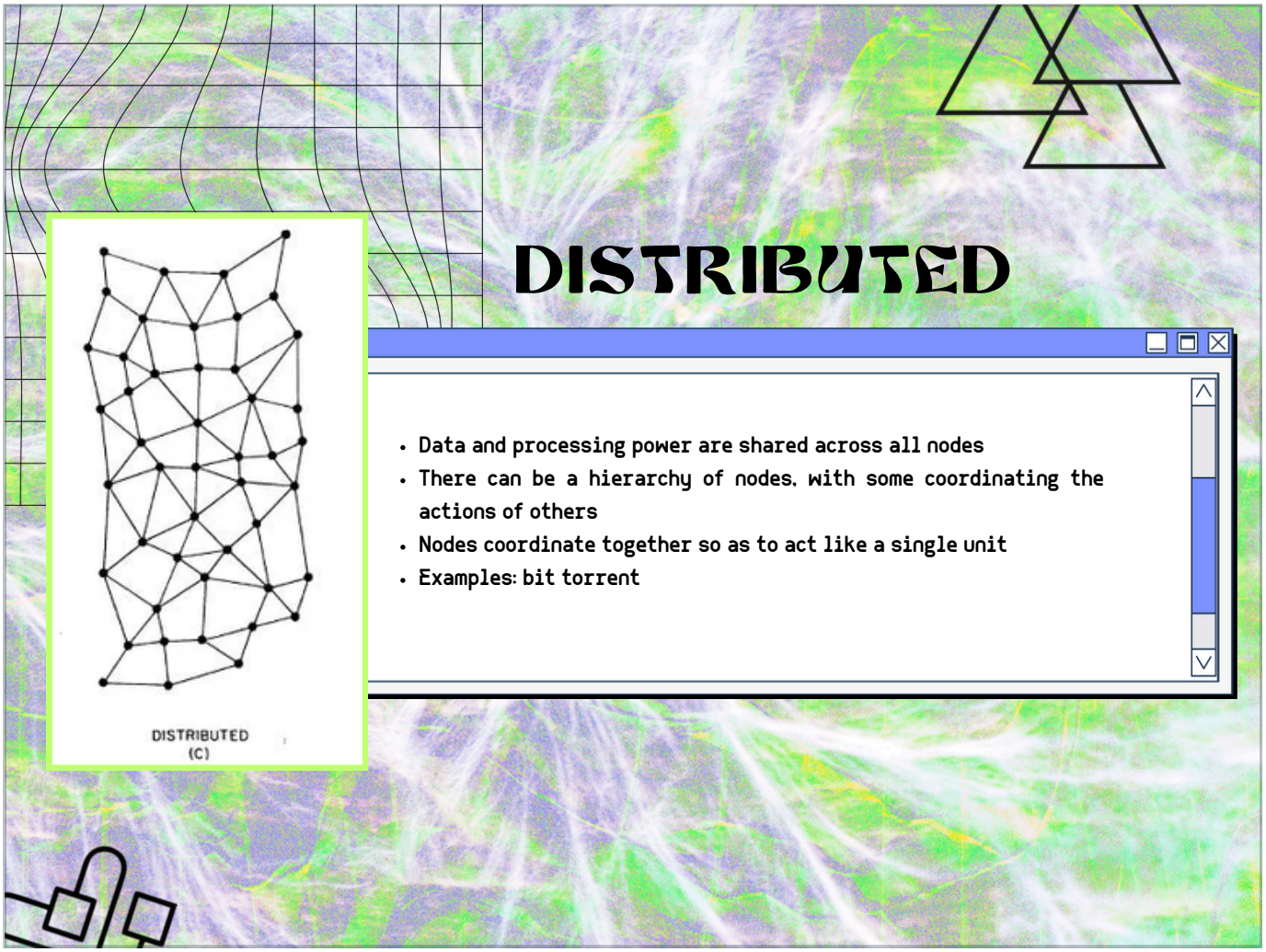
Secondly, all nodes possess equal power and the ability to operate independently. This ensures there's no single point of failure, making the system robust. Consider Bitcoin's network: if thousands of mining computers go offline, the remaining nodes continue validating transactions without disruption, unlike a traditional bank system where server failures can halt all operations.

Finally, decisions are reached through consensus. This democratic approach is a cornerstone of decentralized systems, ensuring all participants have a voice. In Wikipedia, anyone can edit articles, but changes are reviewed and approved by the community rather than a single editorial authority, creating knowledge through collective agreement.

To illustrate, consider technologies like Blockchain (where cryptocurrency transactions require

network agreement), peer-to-peer protocols (like Spotify's old music sharing model), and Git (where software developers can work independently but merge changes through collaborative approval). These systems embody these principles by distributing power and decision-making across many participants rather than concentrating control in one entity.

Let's now explore how distributed systems differ, which we'll cover next.



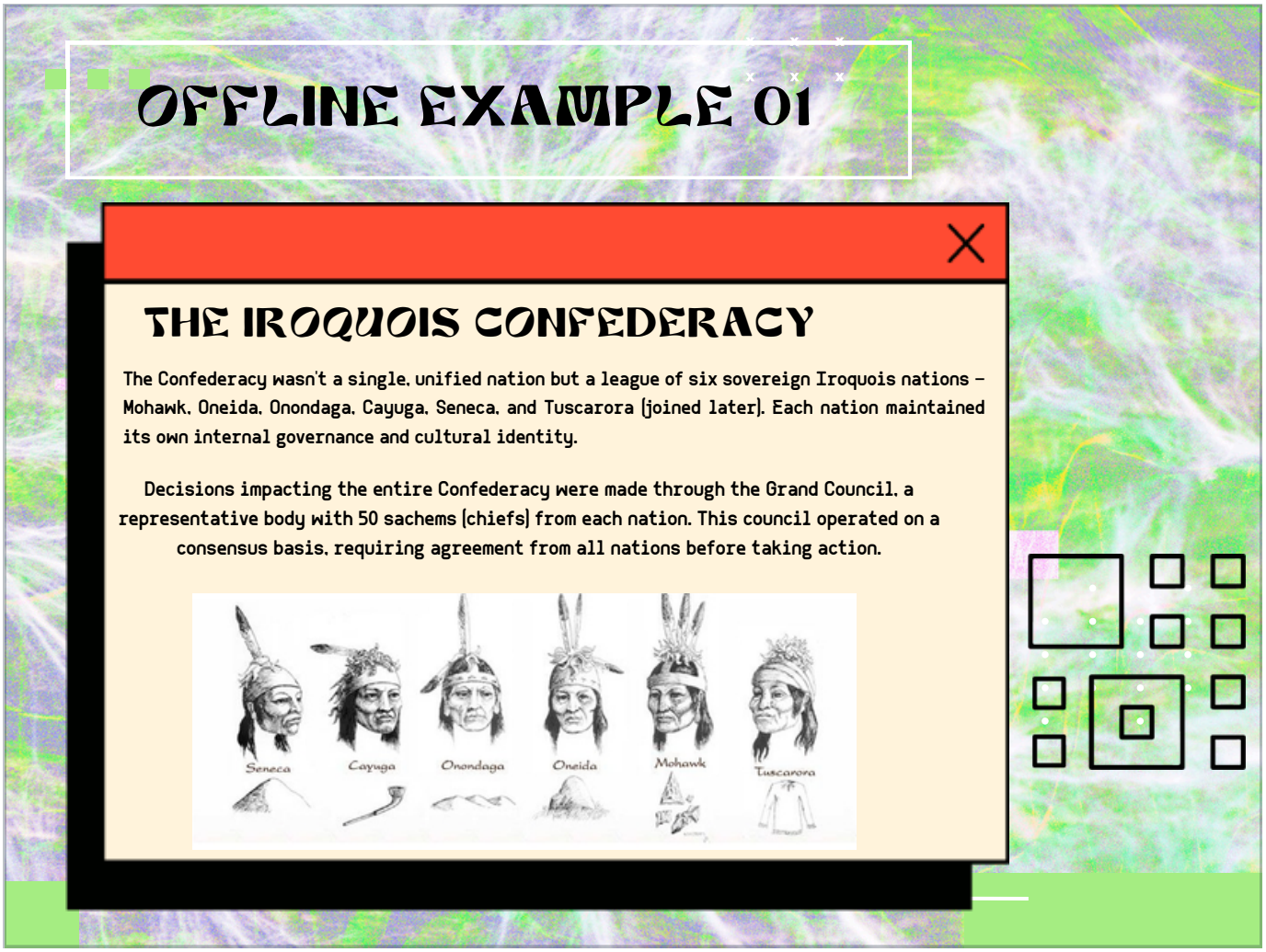
In a distributed system, both data and processing power are shared across all nodes. Unlike decentralized systems, distributed networks can feature a hierarchy. This means some nodes may take on coordination roles, directing and managing the actions of others. For example, in Netflix's content delivery network, regional servers act as coordinators that cache popular movies closer to users, while smaller edge servers follow their lead to ensure smooth streaming.

An important characteristic of distributed systems is how nodes work together to function as a single unit. This coordination allows for efficient processing and resource management. Think of Google's search engine: when you search for something, your query is instantly processed across thousands of servers worldwide, with some handling the search algorithm, others managing the index, and coordinators ensuring results appear as one seamless response. Similarly, cloud services like Dropbox store pieces of your files across multiple data centers, but present them to you as a single, unified storage system.

A familiar example of this is BitTorrent, where files are shared and downloaded through a network of interconnected nodes, but tracker servers coordinate which peers have which file pieces. Another relatable example is online gaming: when you play a multiplayer game, dedicated servers coordinate player actions and game state, while your device and other players' devices handle

rendering and input processing.

So, while decentralized and distributed systems share some similarities, the key distinction lies in the presence of hierarchies and coordinated actions in distributed systems. Keep this in mind as we look into real-world applications next.



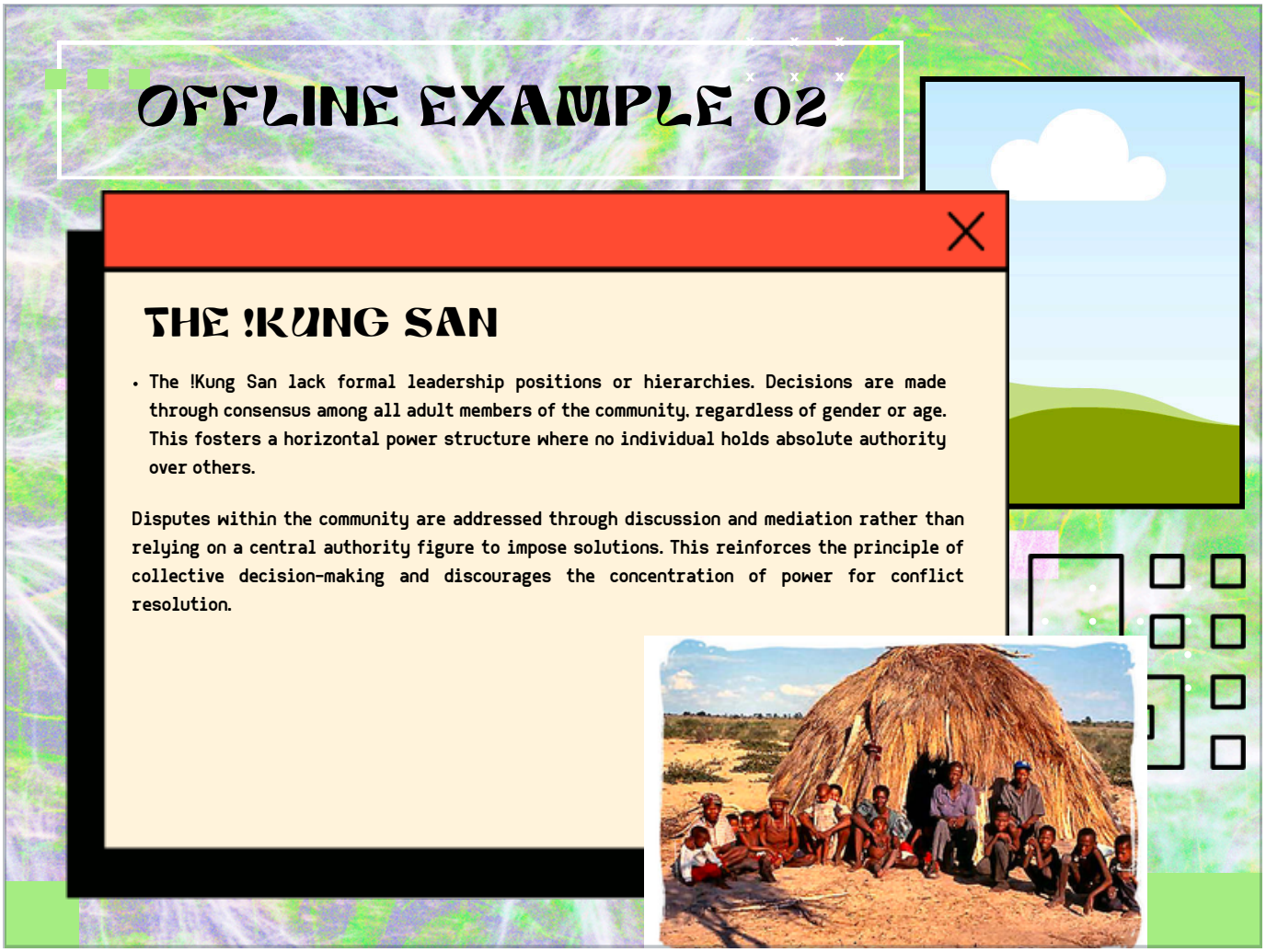
The Iroquois Confederacy is an example of a decentralized yet cohesive system.

The Confederacy was a league of six sovereign nations—each with its own governance and cultural identity.

A vital point here is the Grand Council, comprised of 50 sachems, or chiefs, representing each nation. This council made decisions through consensus, meaning all nations needed to agree before any action was taken.

This illustrates a powerful decentralized model where unity was achieved without sacrificing the autonomy of individual nations.

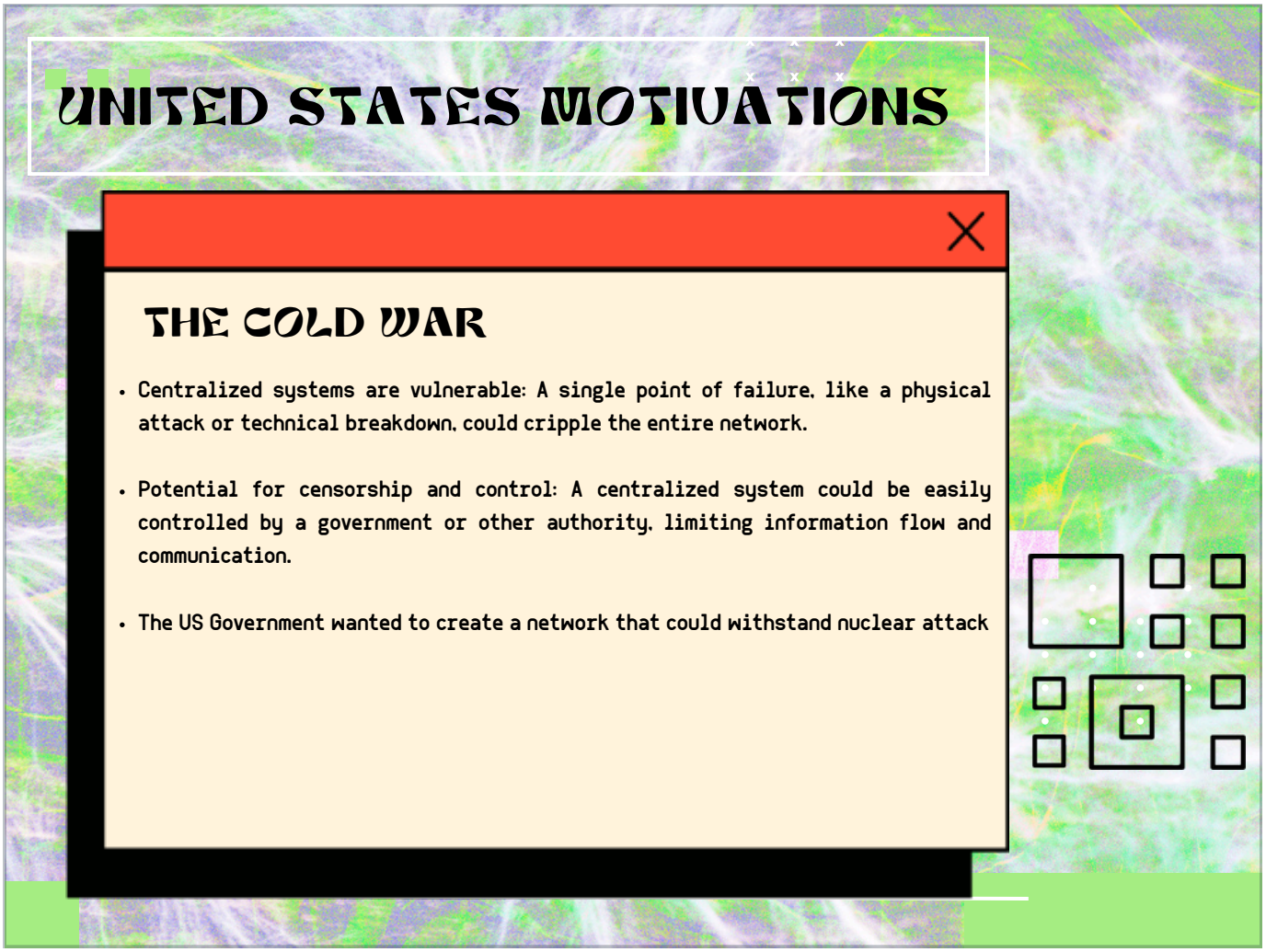
Keep in mind that this system allowed for both independence and cooperation, a balance that was fundamental to the Confederacy's enduring legacy.



Another example of decentralized decision-making is the !Kung San, a group that exemplifies the power of consensus. Unlike the hierarchical structures we often see, the !Kung San have no formal leaders. Every adult, regardless of age or gender, participates equally in decision-making.

This horizontal power structure ensures that no single person holds undue influence. Instead, decisions are made collectively. When disputes arise, they're resolved through open discussion and mediation, reinforcing their commitment to shared decision-making.

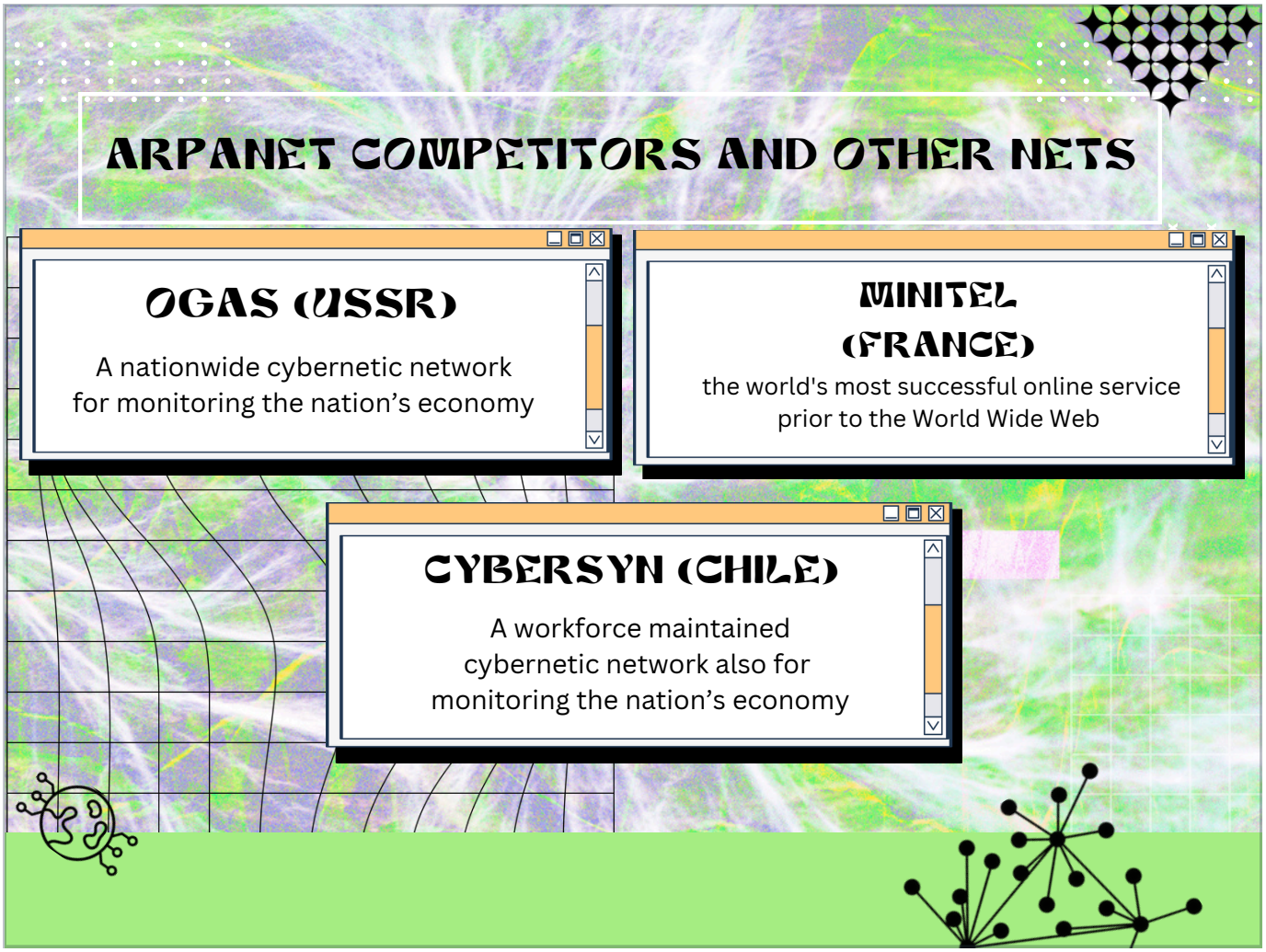
This approach discourages the concentration of power and highlights the strength of community-driven solutions. It's a striking example of how distributed systems can work in human societies, just as they do in technology.



Centralized systems come with significant vulnerabilities. Consider what happens if there's a single point of failure, like a physical attack or a technical breakdown—it could incapacitate an entire network.

Another concern is the potential for censorship and control. In a centralized system, a government or authority could easily restrict information flow, limiting communication. This was a real fear during the Cold War era, where control over information was crucial.

In response to these risks, the US Government was driven to develop a network resilient enough to withstand a nuclear attack. This motivation laid the groundwork for what would eventually become the distributed networks we rely on today.



While the Internet was being developed in the US, there are other parallel efforts happening in other countries.

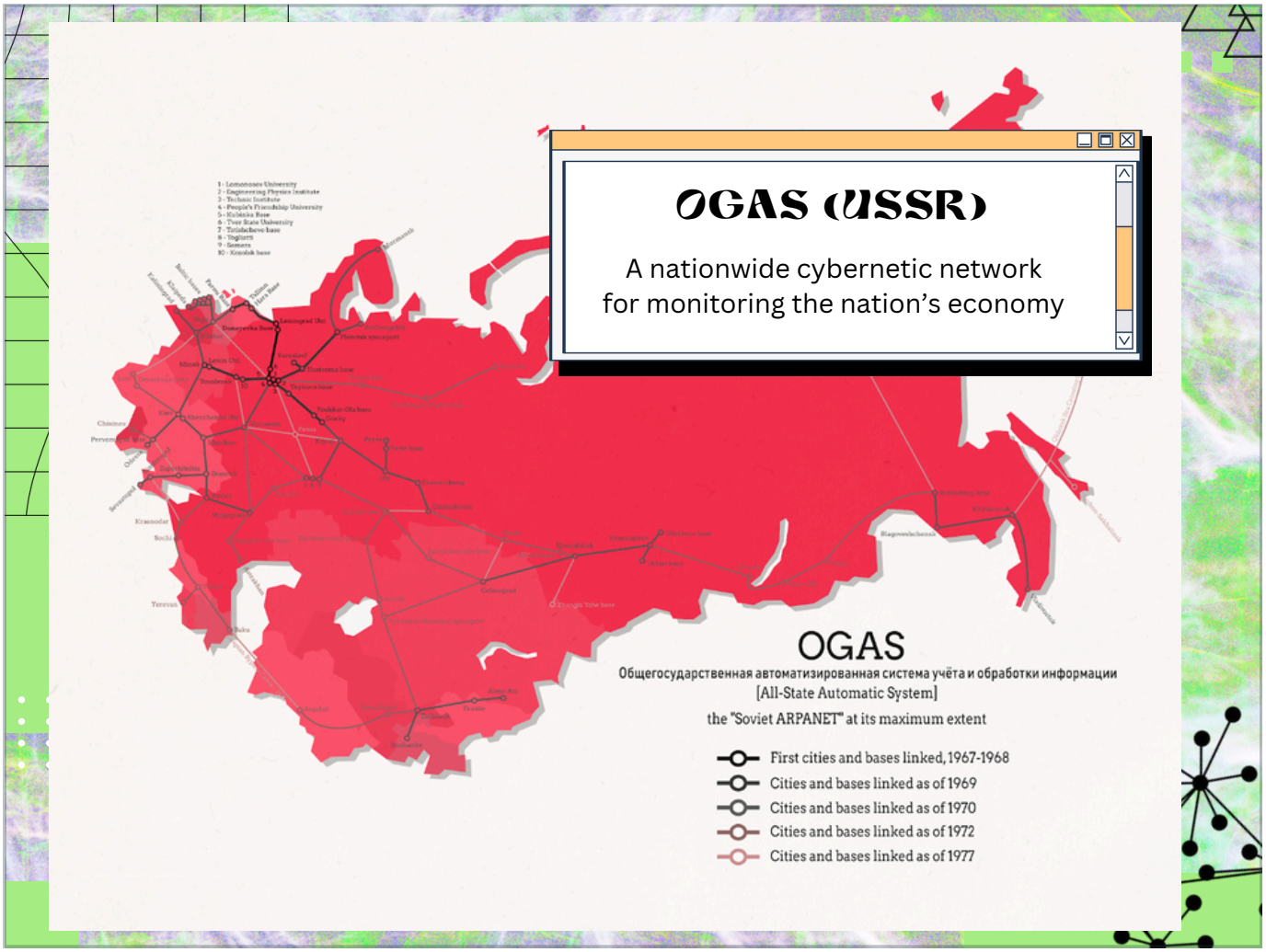
From 1962-1970 the Soviet Union was developing the Общегосударственная автоматизированная система учёта и обработки информации, "ОГАС" (OGAS) or translated to English: "National Automated System for Computation and Information Processing".

From 1971-1973, Chile developed the CYBERSYN project. This was an effort to create a cybernetic network to efficiently manage and monitor the nation's economy. It was about harnessing technology to keep track of economic activities and make informed decisions.

Like OGAS, CYBERSYN was designed to be a nationwide network focused on economic monitoring. It represented a big step towards integrating technology with governance, showcasing the USSR's ambition in digital oversight.

Finally, we have Minitel from France which went live in 1982 and ran until 2012. Before the rise of the World Wide Web, Minitel was the most successful online service globally. It offered a glimpse into the potential of digital networks for everyday use, paving the way for modern internet services.

As we compare these networks, think about the different motivations and challenges that shaped them. We will take a more detailed look at each of these projects over the next few slides.



OGAS was the Soviet Union's attempt at creating a cybernetic network.

OGAS aimed to manage and monitor the nation's economy. This was a massive undertaking, especially for its time.

The network's goal was to improve economic efficiency and decision-making by centralizing data and communications. This ambition reflects the era's drive towards harnessing technology for centralized control.

Despite its potential, OGAS faced numerous challenges, including political resistance and technological limitations. These hurdles ultimately prevented it from reaching full implementation. The project lost political support and eventually funding in 1970, which halted its development.

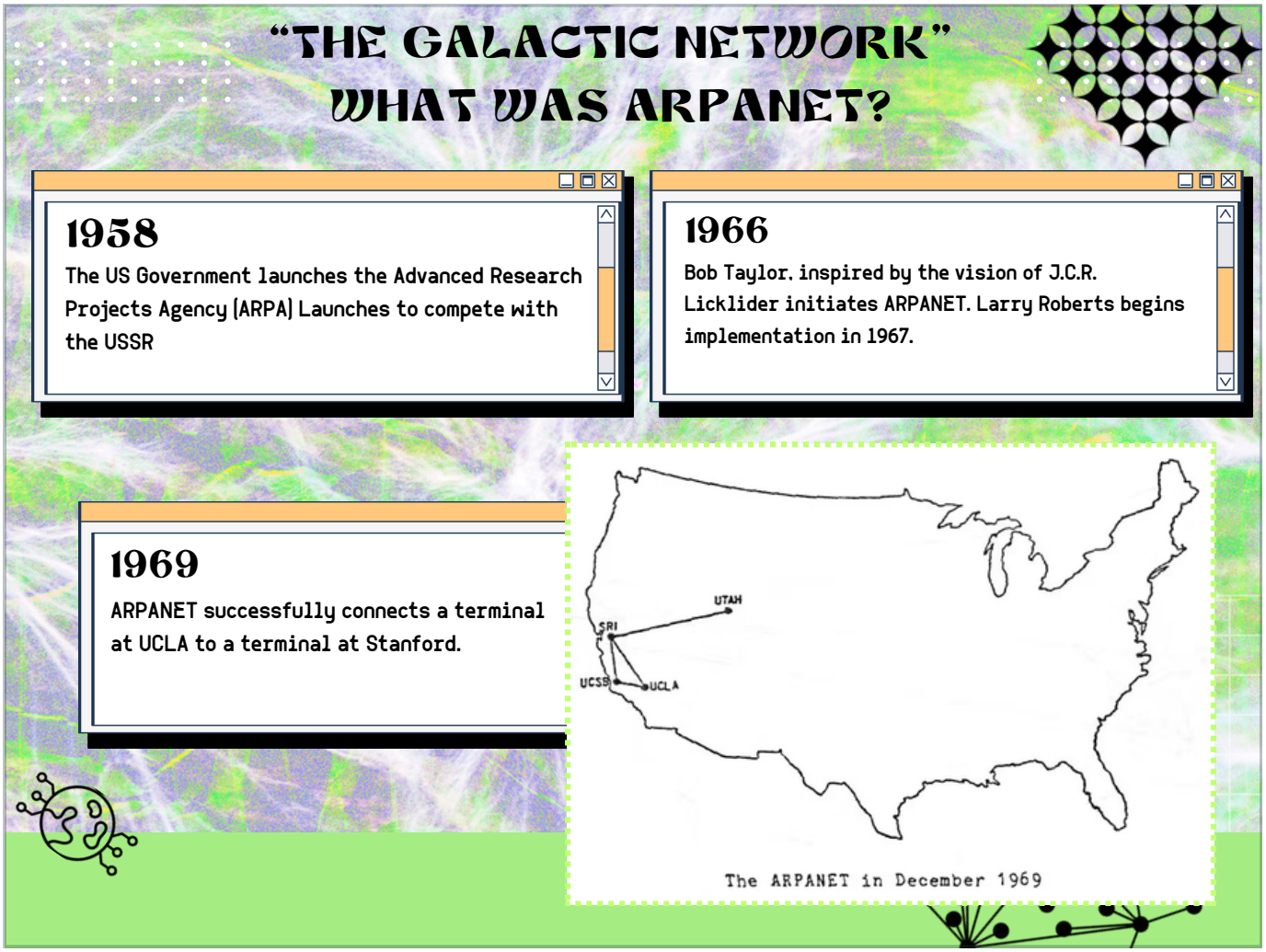
In Chile, CYBERSYN was designed to maintain a real-time overview of the nation's economy. This system wasn't just about data—it was a proactive tool for economic management. Imagine it as a dynamic interface where economic fluctuations could be monitored and addressed swiftly.

One of the key aspects of CYBERSYN was its workforce-driven nature. Unlike some other networks, it emphasized human involvement alongside technology. This collaboration was crucial in adapting quickly to economic changes.

However, CYBERSYN's ambitious vision came to an abrupt end in 1973 when a military coup led by General Augusto Pinochet overthrew President Salvador Allende's government. The new regime dismantled the system, viewing it as a symbol of socialist central planning that conflicted with their free-market ideology. The control room was destroyed, and the project's leaders were forced into exile or silence, marking the end of one of history's most innovative attempts at cybernetic economic management.

France's Minitel network was the world’s most successful online service before the World Wide Web came into being. It enabled users to access a variety of services, such as online directories, banking, and shopping.

What made Minitel stand out was its accessibility; it was available to millions of users across France, demonstrating the potential of online networks well before the internet became mainstream.

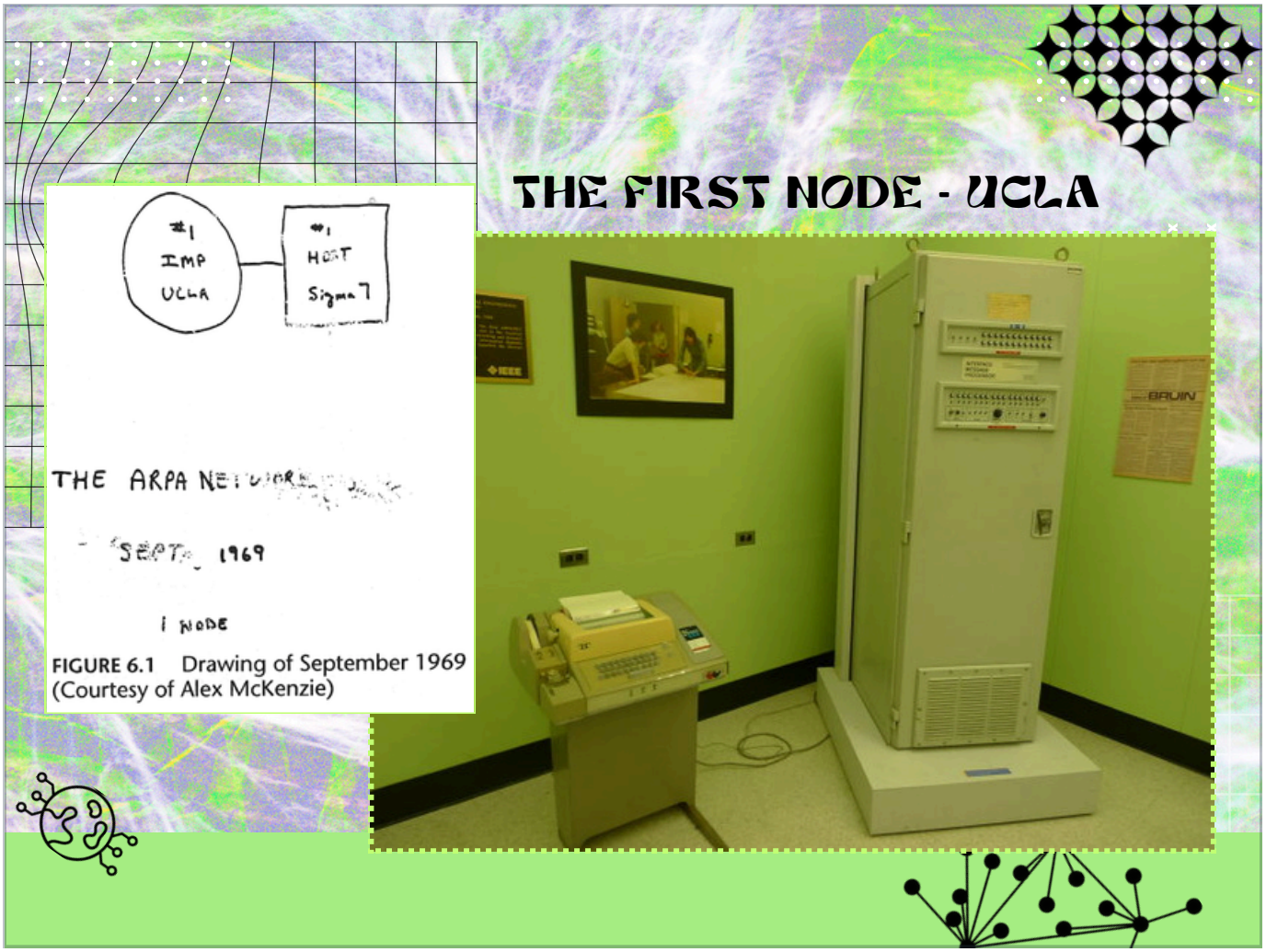


In the US, the early internet was referred to as the ARPANET, so-called after the Advanced Research Projects Agency or ARPA, which developed the project.

The term "Galactic Network" was coined by J.C.R. Licklider, envisioning a globally interconnected set of computers. This vision inspired the creation of ARPANET.

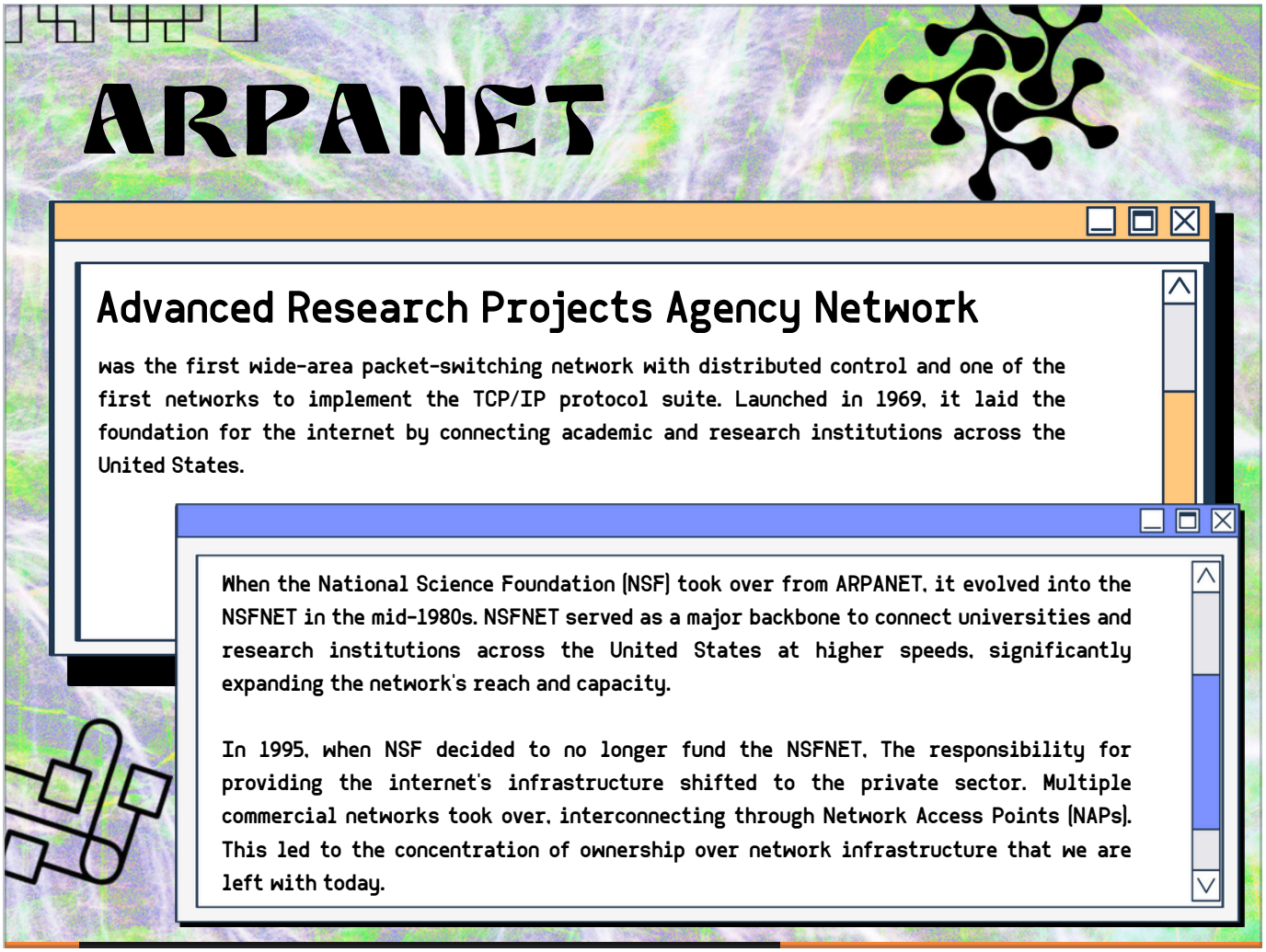
In 1958, amidst the tension of the Cold War, the United States government established the Advanced Research Projects Agency, or ARPA, to compete with the USSR in technological advancements. In 1966, Bob Taylor, driven by Licklider's vision, initiated ARPANET, with Larry Roberts beginning its implementation the following year.

By 1969, ARPANET connected the first two terminals between UCLA and Stanford. This event marked a historical moment in the development of networked communication.



The initial node of the ARPANET was established at the University of California, Los Angeles (UCLA) on September 2, 1969.

Later on October 29, 1969, this node would connect to the second established node at Stanford University. The first message ever sent was an attempt to "login". However, the system crashed while the message was being typed into the terminal, so that only "lo" was transmitted.

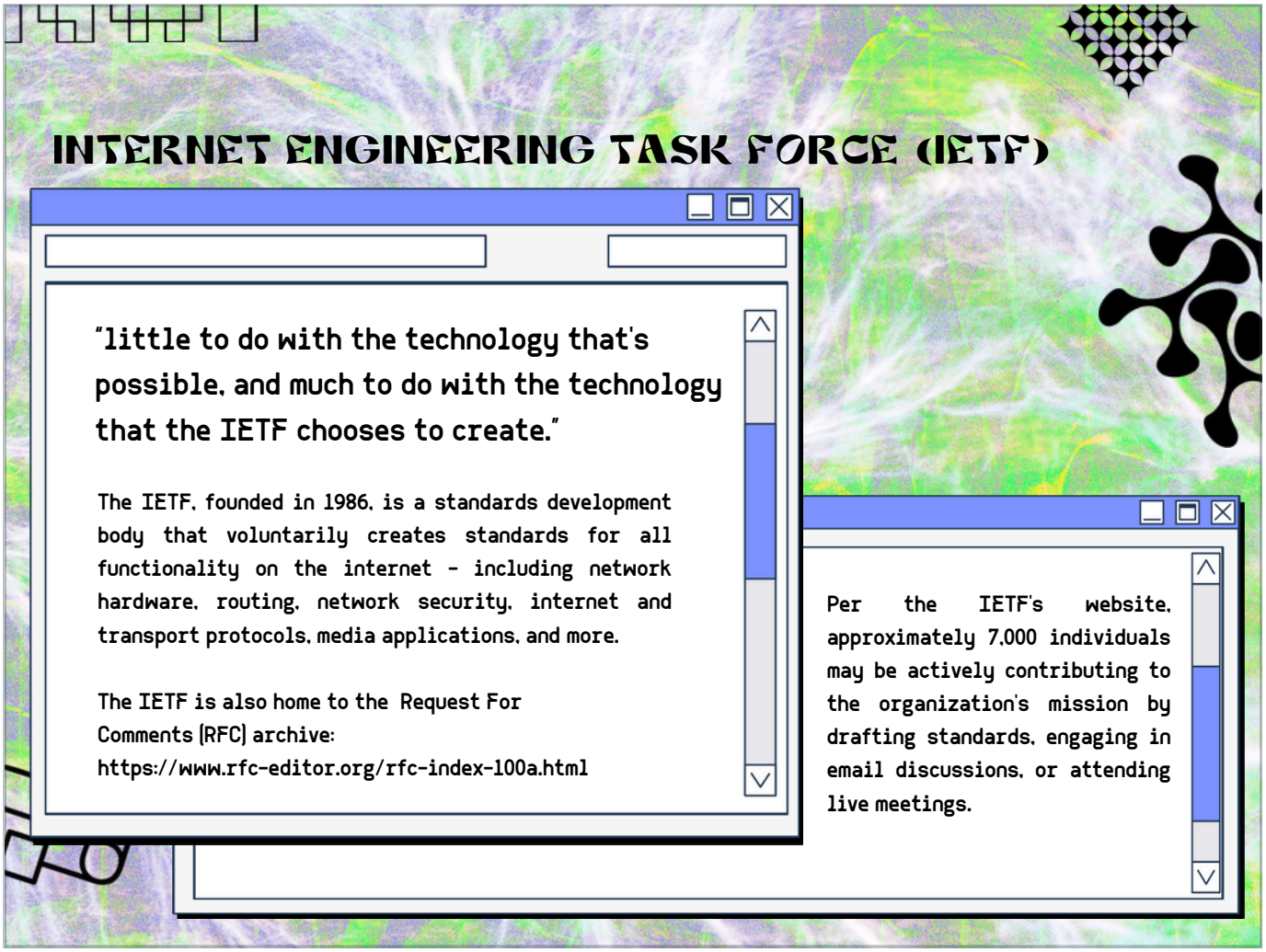


ARPANET wasn't just the first wide-area packet-switching network; it was also the first implementation of the TCP/IP protocol suite, which is one of the key protocols we use for data transmission over the internet, still in use today and largely unchanged.

Launched in 1969, ARPANET was a network connecting academic and research institutions across the United States. The modern internet grew out of this initial network.

Coming into the mid-1980s, the National Science Foundation took over the project, transforming ARPANET into the NSFNET. This allowed for a significant expansion, connecting universities and research institutions at higher speeds and greater capacity.

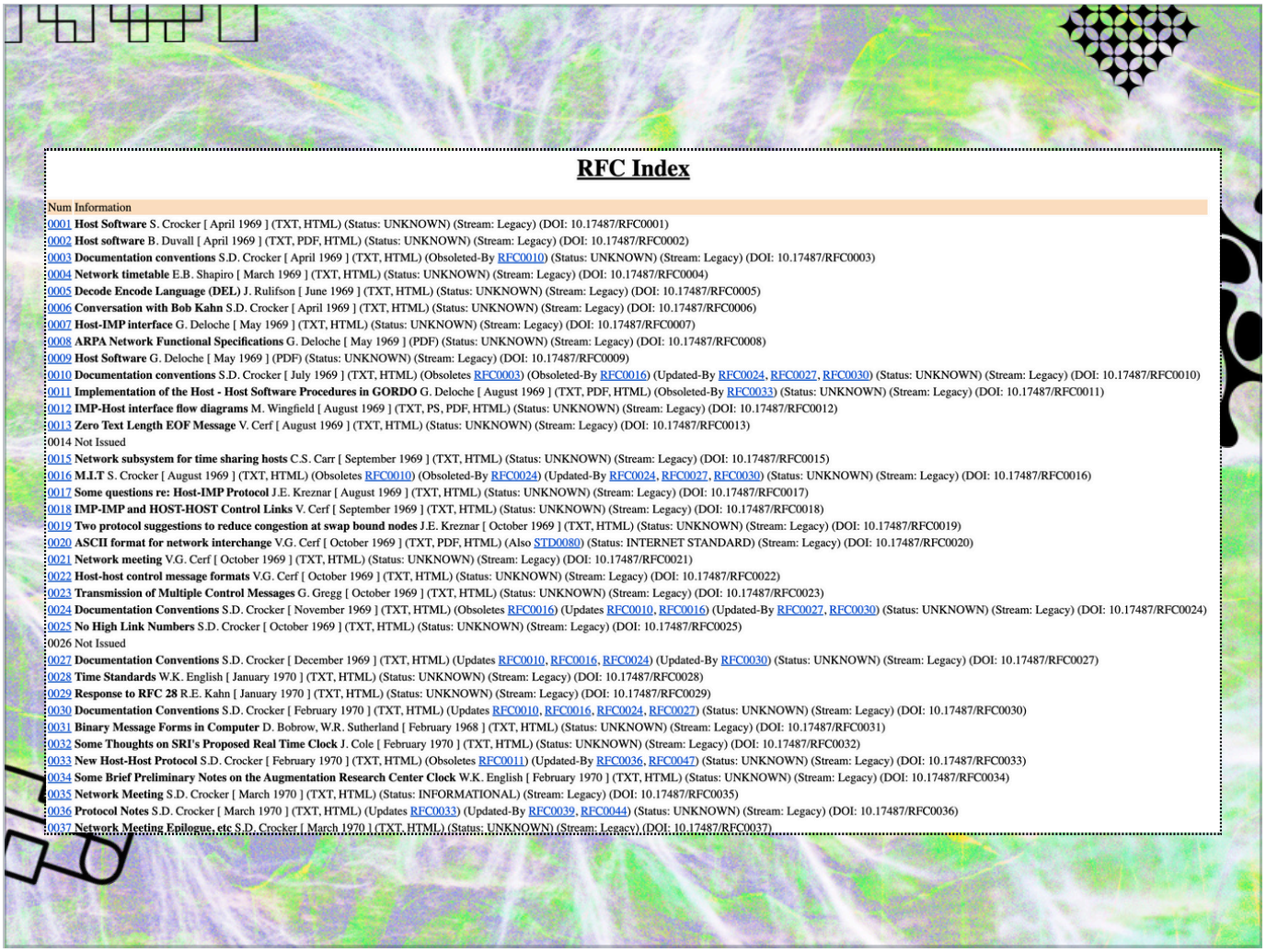
By 1995, the NSFNET era ended as the private sector assumed control, leading to the commercial networks we know today. This shift marked a pivotal point in the internet's history, influencing the structure and ownership of network infrastructure.



The Internet Engineering Task Force, or the IETF, founded in 1986, plays an important role in shaping the internet as it is today. It's a collaborative body that voluntarily establishes standards for everything from network hardware to internet and transport protocols.

An interesting aspect of the IETF is its Request For Comments, or RFC archive. This serves as a repository of all technical and organizational notes, providing a rich history of internet standards development.

According to their website, around 7,000 individuals might be actively participating in the IETF's mission at any given time. They are drafting standards, engaging in discussions, or attending meetings. It is a global effort to keep the internet running smoothly.

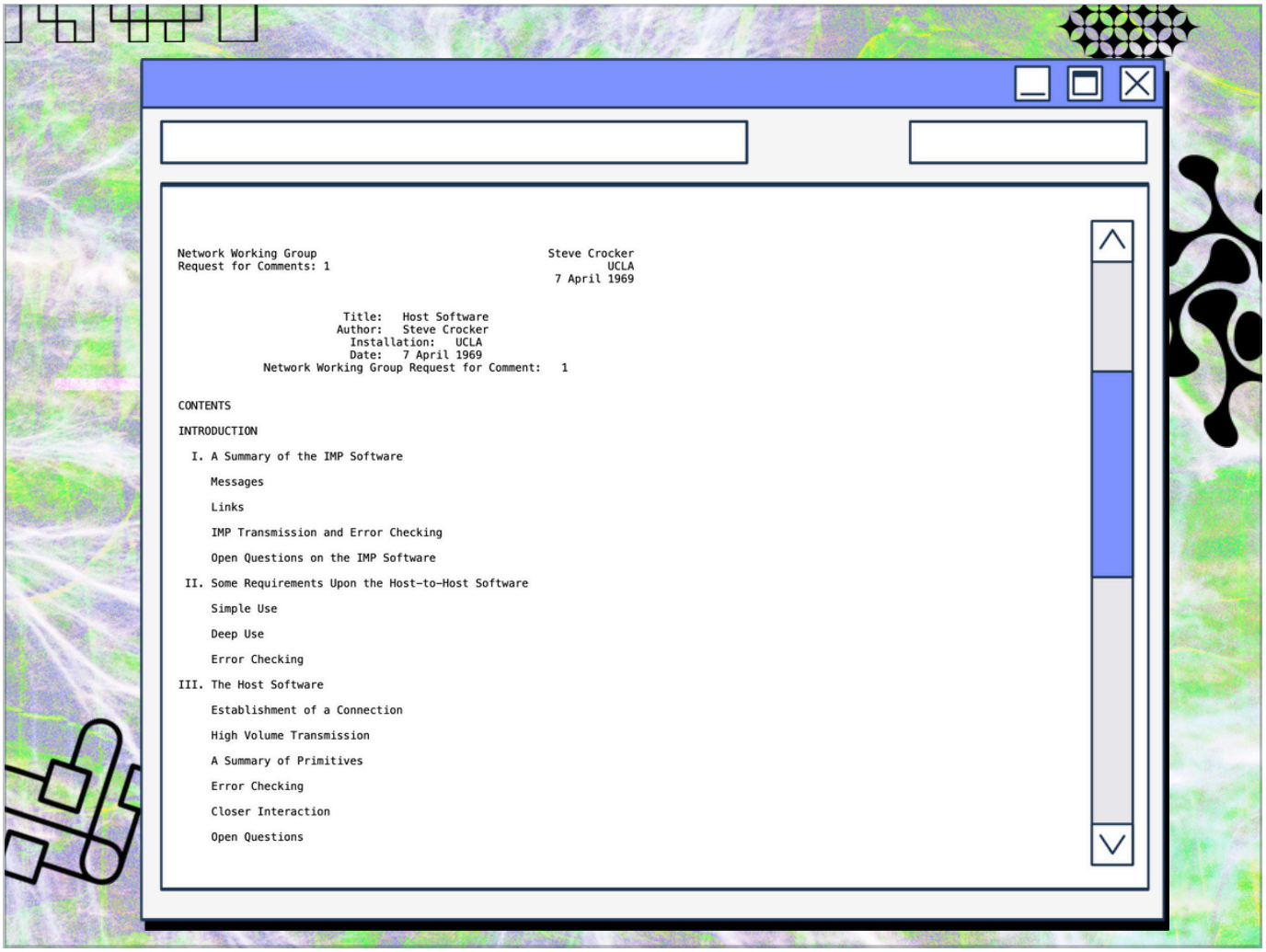


RFC Index

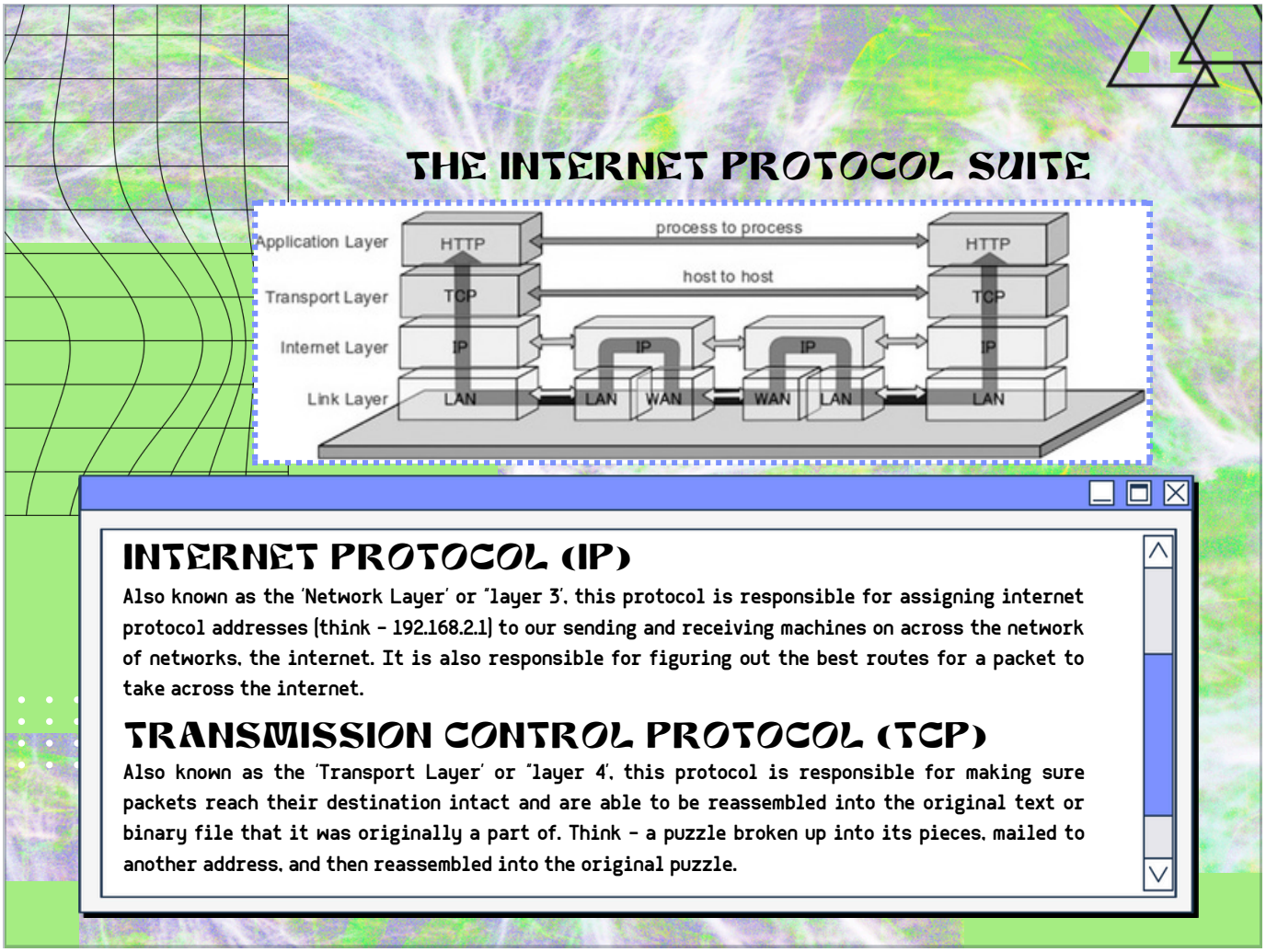
Num Information

0001 **Host Software** S. Crocker [April 1969] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0001)
0002 **Host software** B. Duvall [April 1969] (TXT, PDF, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0002)
0003 **Documentation conventions** S.D. Crocker [April 1969] (TXT, HTML) (Obsoleted-By RFC0010) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0003)
0004 **Network timetable** E.B. Shapiro [March 1969] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0004)
0005 **Decode Encode Language (DEL)** J. Rulifson [June 1969] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0005)
0006 **Conversation with Bob Kahn** S.D. Crocker [April 1969] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0006)
0007 **Host-IMP interface** G. Deloche [May 1969] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0007)
0008 **ARPA Network Functional Specifications** G. Deloche [May 1969] (PDF) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0008)
0009 **Host Software** G. Deloche [May 1969] (PDF) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0009)
0010 **Documentation conventions** S.D. Crocker [July 1969] (TXT, HTML) (Obsoletes RFC0003) (Obsoleted-By RFC0016) (Updated-By RFC0024, RFC0027, RFC0030) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0010)
0011 **Implementation of the Host - Host Software Procedures in GORDO** G. Deloche [August 1969] (TXT, PDF, HTML) (Obsoleted-By RFC0033) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0011)
0012 **IMP-Host interface flow diagrams** M. Wingfield [August 1969] (TXT, PS, PDF, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0012)
0013 **Zero Text Length EOF Message** V. Cerf [August 1969] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0013)
0014 Not Issued
0015 **Network subsystem for time sharing hosts** C.S. Carr [September 1969] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0015)
0016 **M.I.T** S. Crocker [August 1969] (TXT, HTML) (Obsoletes RFC0010) (Obsoleted-By RFC0024) (Updated-By RFC0024, RFC0027, RFC0030) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0016)
0017 **Some questions re: Host-IMP Protocol** J.E. Kreznar [August 1969] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0017)
0018 **IMP-IMP and HOST-HOST Control Links** V. Cerf [September 1969] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0018)
0019 **Two protocol suggestions to reduce congestion at swap bound nodes** J.E. Kreznar [October 1969] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0019)
0020 **ASCII format for network interchange** V.G. Cerf [October 1969] (TXT, PDF, HTML) (Also STD0080) (Status: INTERNET STANDARD) (Stream: Legacy) (DOI: 10.17487/RFC0020)
0021 **Network meeting** V.G. Cerf [October 1969] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0021)
0022 **Host-host control message formats** V.G. Cerf [October 1969] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0022)
0023 **Transmission of Multiple Control Messages** G. Gregg [October 1969] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0023)
0024 **Documentation Conventions** S.D. Crocker [November 1969] (TXT, HTML) (Obsoletes RFC0016) (Updates RFC0010, RFC0016) (Updated-By RFC0027, RFC0030) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0024)
0025 **No High Link Numbers** S.D. Crocker [October 1969] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0025)
0026 Not Issued
0027 **Documentation Conventions** S.D. Crocker [December 1969] (TXT, HTML) (Updates RFC0010, RFC0016, RFC0024) (Updated-By RFC0030) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0027)
0028 **Time Standards** W.K. English [January 1970] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0028)
0029 **Response to RFC 28** R.E. Kahn [January 1970] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0029)
0030 **Documentation Conventions** S.D. Crocker [February 1970] (TXT, HTML) (Updates RFC0010, RFC0016, RFC0024, RFC0027) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0030)
0031 **Binary Message Forms in Computer** D. Bobrow, W.R. Sutherland [February 1968] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0031)
0032 **Some Thoughts on SRI's Proposed Real Time Clock** J. Cole [February 1970] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0032)
0033 **New Host-Host Protocol** S.D. Crocker [February 1970] (TXT, HTML) (Obsoletes RFC0011) (Updated-By RFC0036, RFC0047) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0033)
0034 **Some Brief Preliminary Notes on the Augmentation Research Center Clock** W.K. English [February 1970] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0034)
0035 **Network Meeting** S.D. Crocker [March 1970] (TXT, HTML) (Status: INFORMATIONAL) (Stream: Legacy) (DOI: 10.17487/RFC0035)
0036 **Protocol Notes** S.D. Crocker [March 1970] (TXT, HTML) (Updates RFC0033) (Updated-By RFC0039, RFC0044) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0036)
0037 **Network Meeting Epilogue, etc** S.D. Crocker [March 1970] (TXT, HTML) (Status: UNKNOWN) (Stream: Legacy) (DOI: 10.17487/RFC0037)

This is a screenshot from the front of the Request for Comments (RFC) index webpage. https://www.rfc-editor.org/rfc-index.html



This is a screenshot of the first document in the Request for Comments (RFC) series, titled "Host Software," published in 1969 by Steve Crocker to define the "IMP software" used for host-to-host communication on the early ARPANET. You can view this document yourself at https://datatracker.ietf.org/doc/html/rfc1



On this slide, we look at the Internet Protocol Suite, which is foundational to our digital communication.

On the bottom layer we see the "Link Layer". This encompasses Layers 1 and 2, the "Physical Layer" which refer to the medium over which the data signal is transmitted. This could be an ethernet cable, a wireless radio or light.

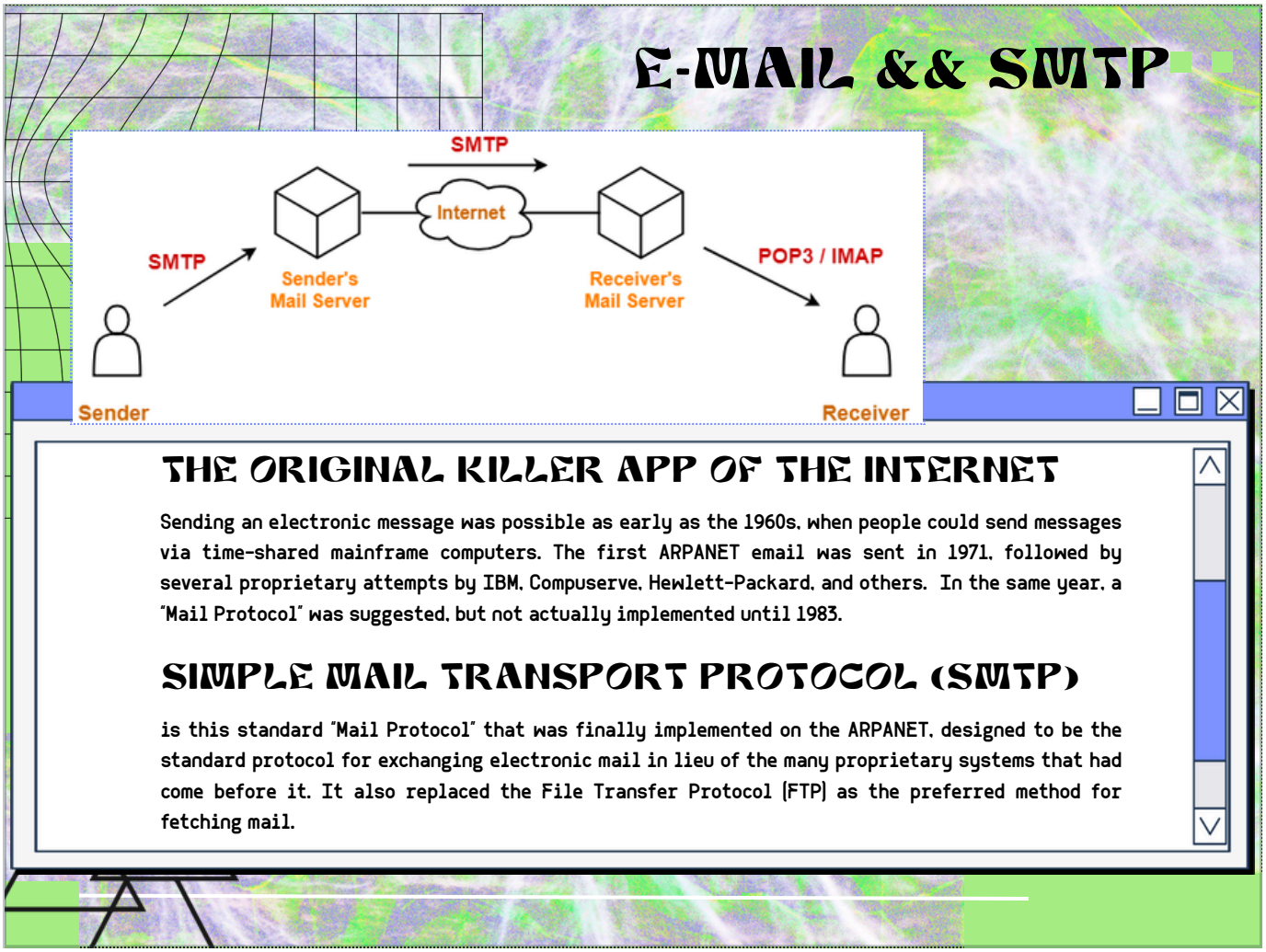
Layer 2 is the "Media Access Layer" and this refers to the hardware address of your networked device. Every piece of hardware capable of connecting to the network has a unique identifier burned into it known as the "Media Access Control" or MAC address. This address is a 12-digit hexadecimal "name tag" that allows devices to be identified on a local network (LAN) and helps in sending data to the correct device

Next is the Internet Protocol, or IP, often referred to as the "Network Layer" or "Layer 3". This protocol is critical to the functioning of the internet as it assigns IP addresses, like 192.168.2.1, to devices, making sure each one can send and receive information across the internet. A key role of IP is to determine the most efficient routes for data packets to travel, ensuring they reach their destination effectively.

Next in the stack is the Transmission Control Protocol, or TCP, also called the "Transport Layer" or "Layer 4". This protocol takes on the essential job of ensuring that data packets not only reach their destination but do so intact and in the correct order. Imagine it like mailing a puzzle, piece by piece, and having it arrive at its destination to be reassembled perfectly.

Together, IP and TCP form the backbone of the Internet Protocol Suite, enabling seamless communication over the internet.

Beyond Layer 4 are Layers 5, 6, and 7, which together make up the "Application Layer" which is where we group the actual human-facing applications we use, like email clients, file transfer programs like Filezilla or web browsers like Chrome, Safari, Firefox or Edge.

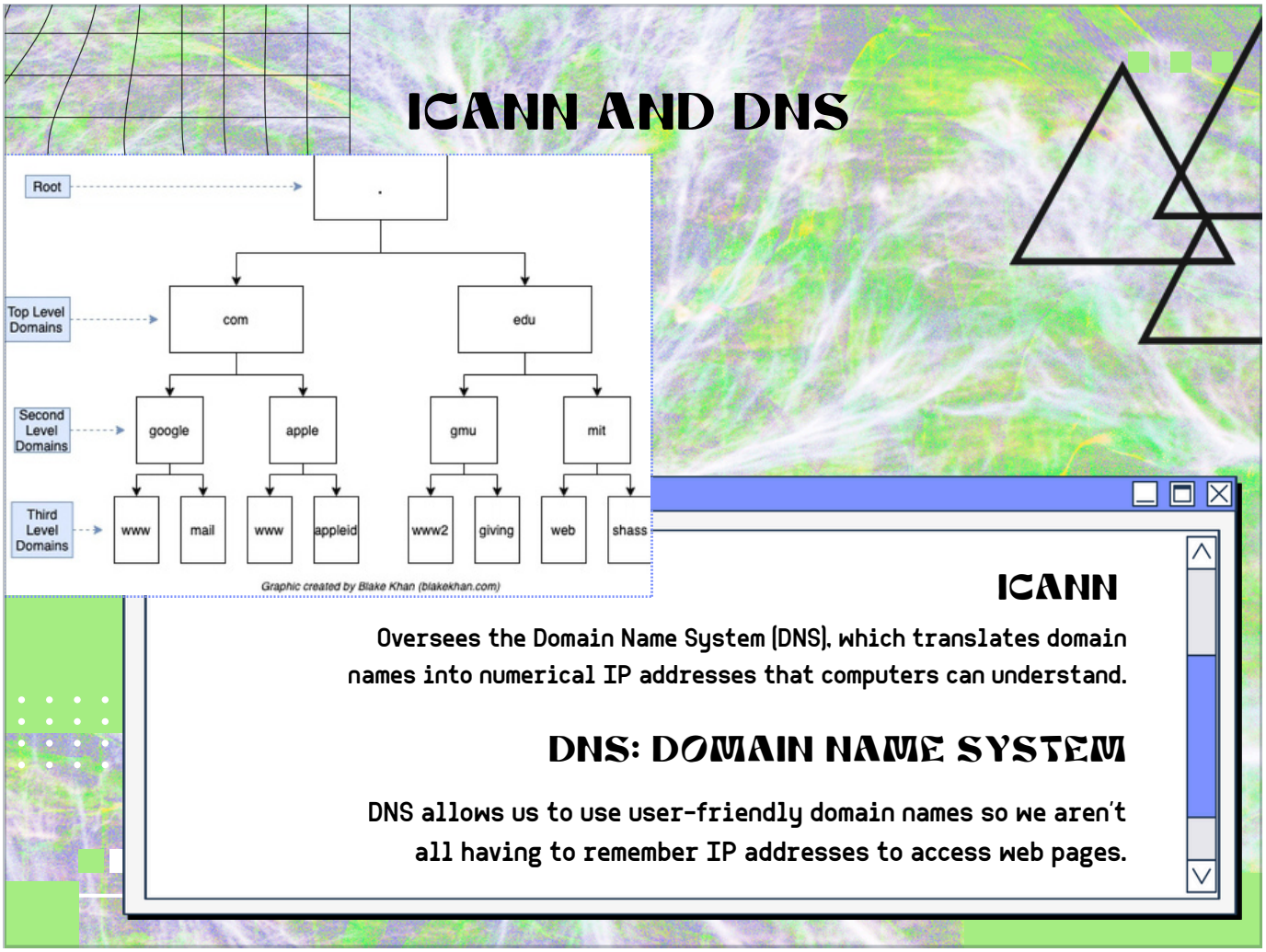


Here we take a look at what was the most widely used application deployed to the internet: Email and the
Simple Mail Transport Protocol (SMTP).

Sending electronic messages dates back to the 1960s, using time-shared mainframe computers. The very first ARPANET email was sent in 1971.

In 1983, the standard "Mail Protocol" was finally implemented on the ARPANET. This protocol was crucial because it unified various proprietary systems from tech giants like IBM and Hewlett-Packard, making electronic mail more accessible and consistent.

SMTP was developed in 1982, replacing the File Transfer Protocol (FTP) as the primary method for fetching mail. Its implementation marked a significant milestone, transforming email into the "killer app" of the internet, a tool so essential that it drove widespread adoption of network technology.



ICANN, which stands for the Internet Corporation for Assigned Names and Numbers, plays a crucial role in internet infrastructure. It oversees the Domain Name System, or DNS, which is the system that translates those memorable, user-friendly domain names, like example.com, into numerical IP addresses that computers use to communicate.

We can think of DNS as the internet's phonebook. Without it, we'd have to remember complex strings of numbers just to visit a website.

This diagram illustrates the hierarchical structure of the Domain Name System (DNS), showing how internet domain names are organized in a tree-like hierarchy from most general to most specif ic.

Root Level:
At the top is the "Root" - represented by a dot (.) - which is the highest level of the DNS hierarchy. All domain name lookups ultimately trace back to this root level.

Top Level Domains (TLDs):
Below the root are Top Level Domains like ".com" and ".edu". These are managed by specific

organizations - .com for commercial entities and .edu for educational institutions.

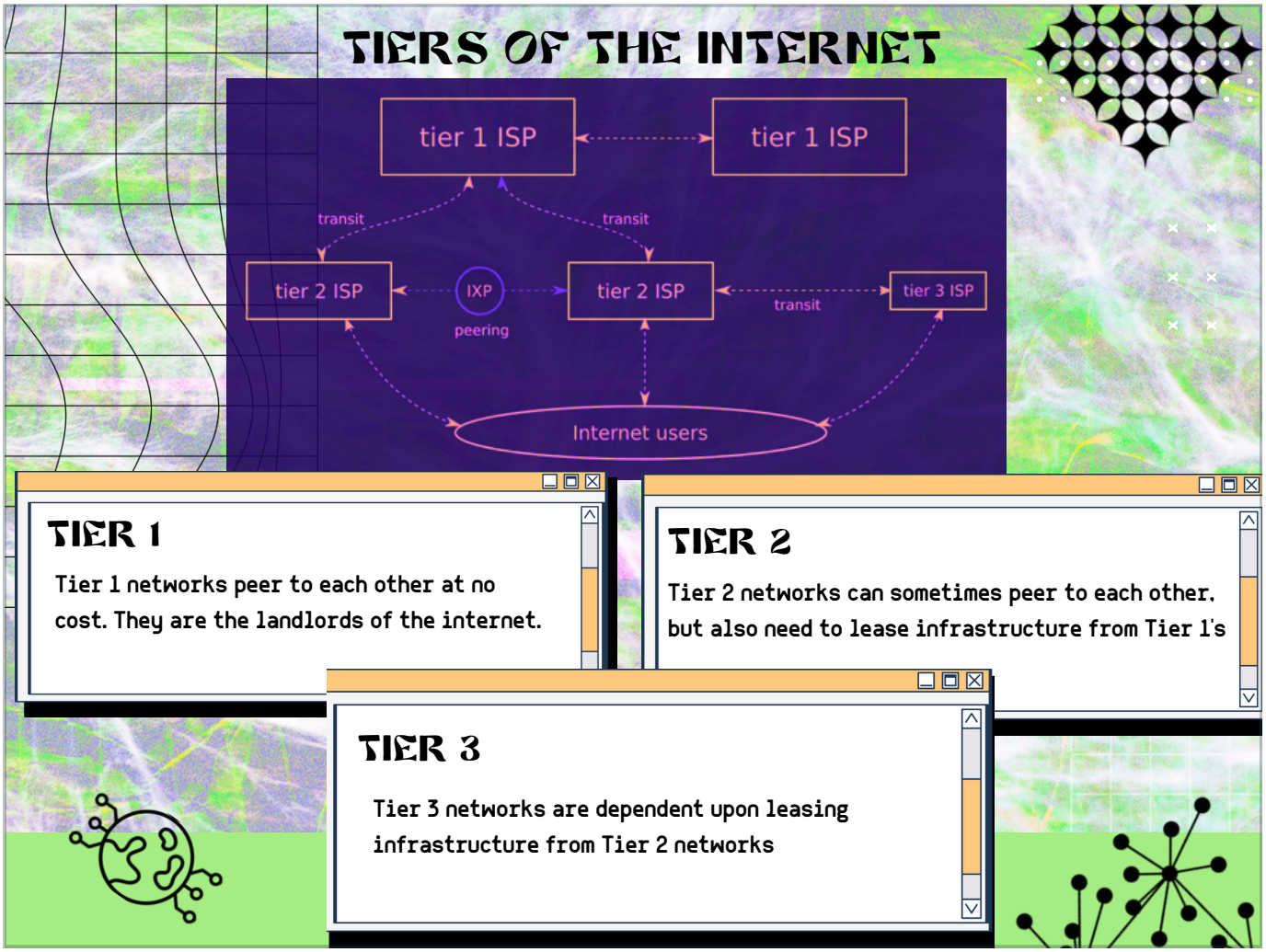
Second Level Domains:
Under each TLD are Second Level Domains that organizations register:

Under .com: "google" and "apple"
Under .edu: "gmu" (George Mason University) and "mit" (Massachusetts Institute of Technology)

Third Level Domains (Subdomains):
At the bottom level are subdomains that organizations create for specific services:

Under google.com: "www" and "mail"
Under apple.com: "www" and "appleid"
Under gmu.edu: "www2" and "giving"
Under mit.edu: "web" and "shass"

How It Works:
When you type "www.google.com" into your browser, DNS resolvers traverse this hierarchy from right to left: starting with the root, then .com, then google, then www - ultimately resolving to the IP address of Google's web server. This hierarchical system allows for distributed management while maintaining global consistency in how domain names are resolved to IP addresses.RetryClaude can make mistakes. Please double-check responses.



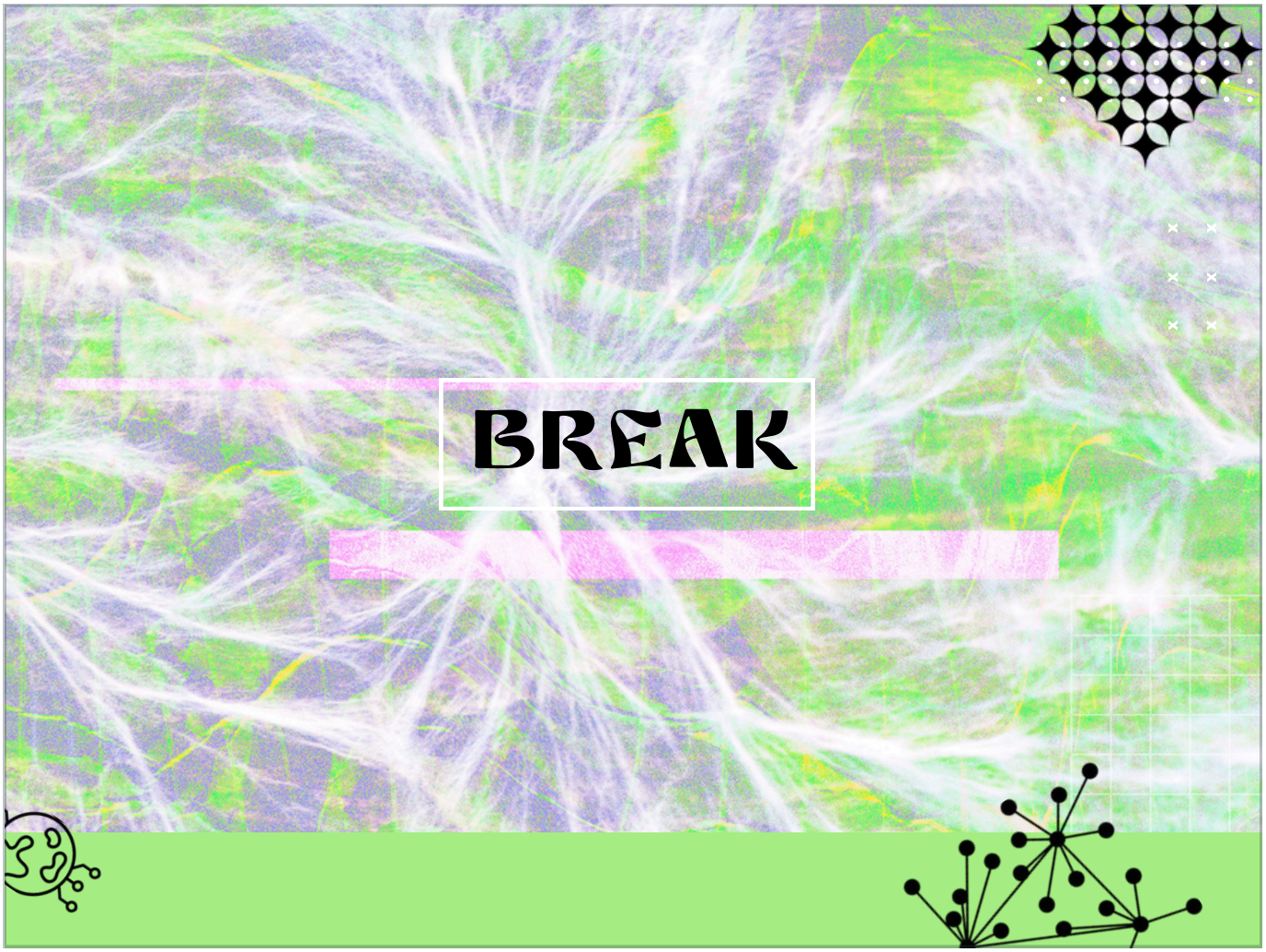
The internet is not entirely a horizontal structure - there exist hierarchies that reflect ownership of the network and therefore control and power structures that manage how data flows across the glo be.

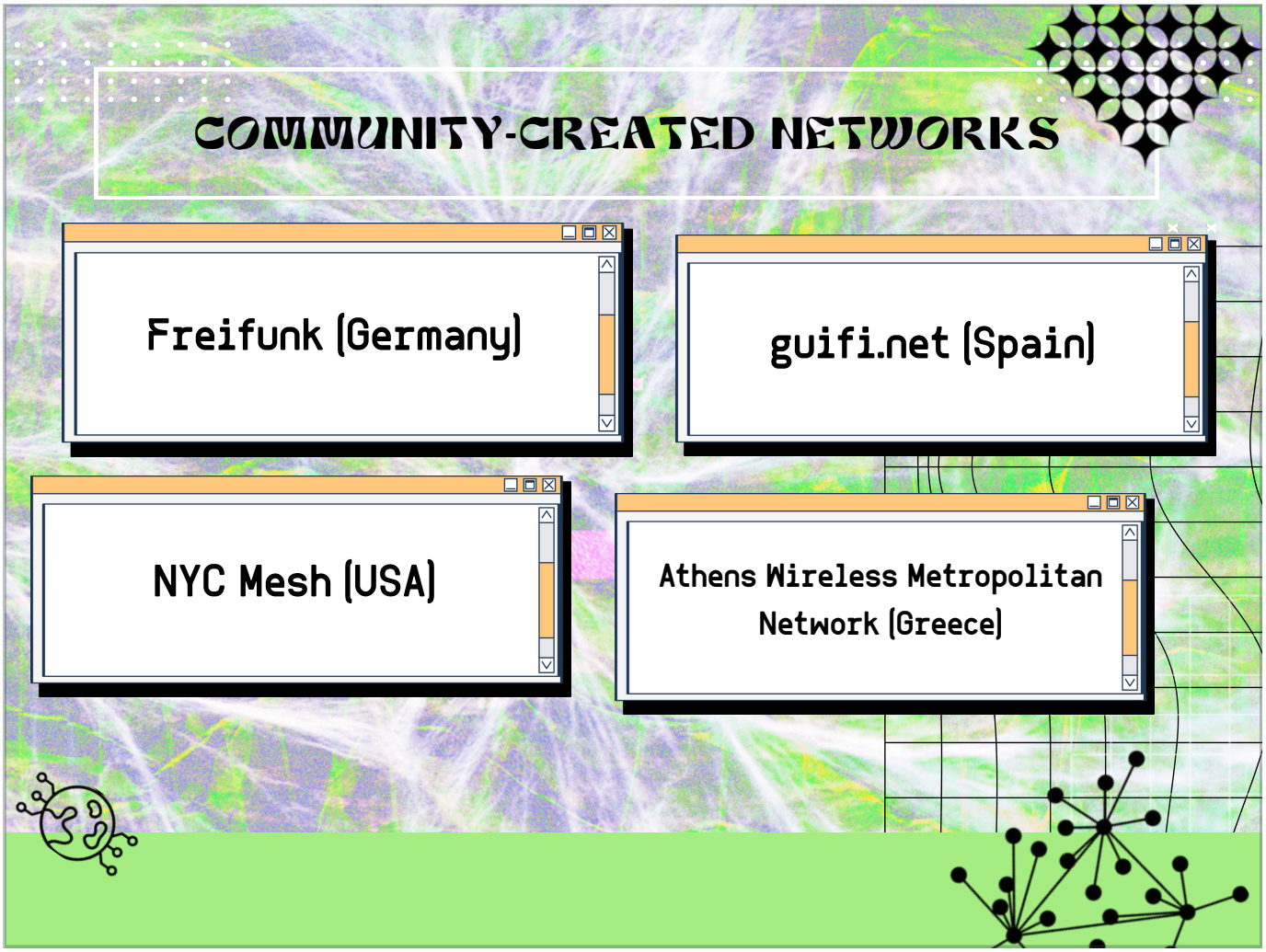
First are Tier 1 networks. Think of them as the landlords of the internet. They peer with each other at no cost, owning the infrastructure that forms the backbone of global connectivity. Major international telecommunications companies like AT&T, Verizon, Xfinity, China Telecom, and Deutsche Telekom are examples of Tier 1 Internet Service Providers (ISPs).

Next, Tier 2 networks. These are a bit more complex. They can sometimes peer with each other, but they also need to lease infrastructure from Tier 1 networks. So, they serve as both landlords and tenants. Examples of Tier 2 providers are national and regional telecommunications companies like Comcast, Cox, Charter Communications, Vodafone, British Telecom, and multinational carriers such as China Telecom and Softbank.

Finally, Tier 3 networks. These are entirely dependent on leasing infrastructure from Tier 2 networks. They're like tenants who rely on others for connectivity. These are local network providers that are focused on selling consumer services. Examples of Tier 3 ISPs are MetroPCS,

Spectrum, PŸUR, or Brisanet.


BREAK



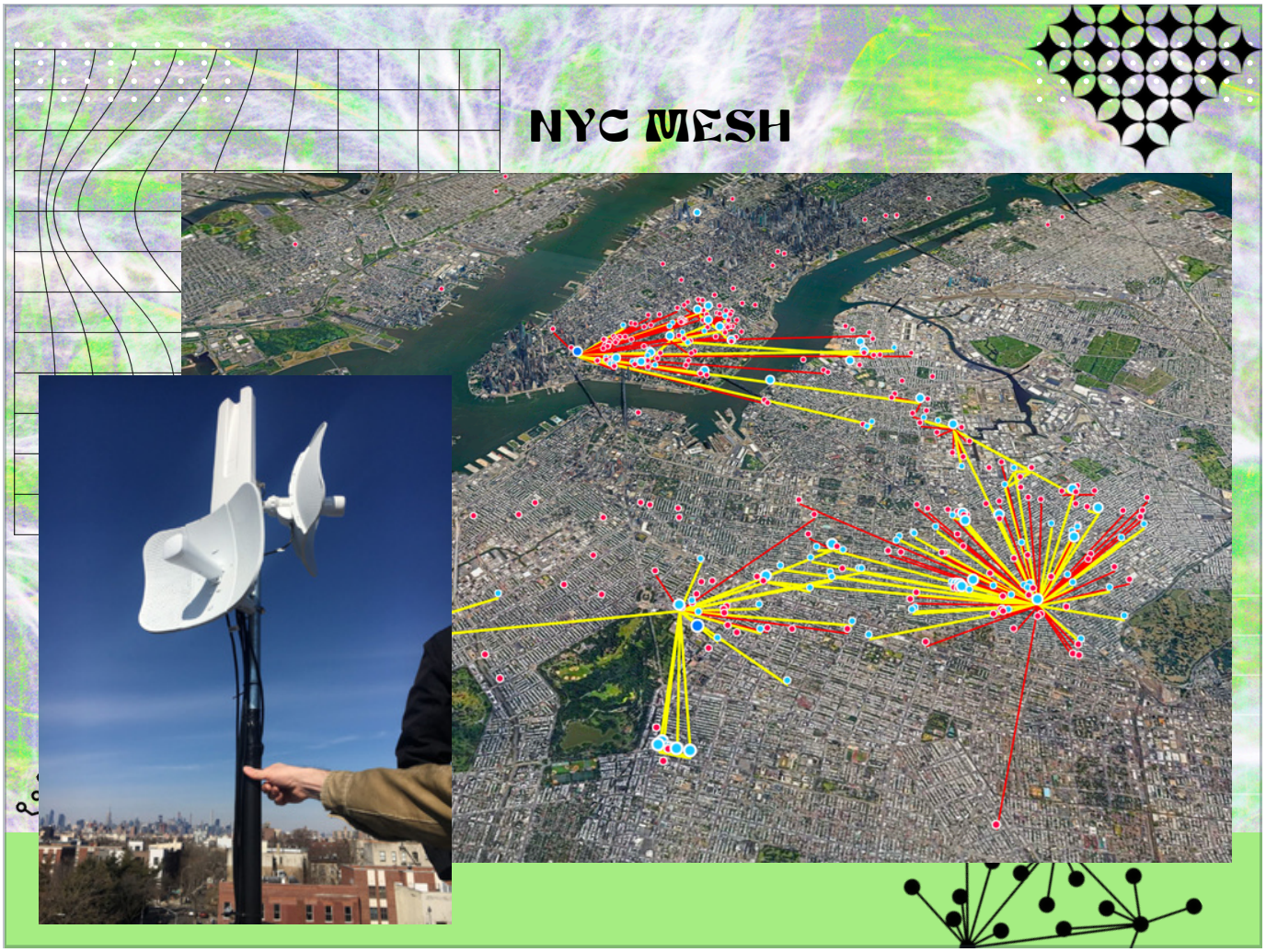
Additionally, there are several community-created networks, which offer an alternative to traditional internet infrastructure. These networks are built and maintained by communities, often to address local needs or to provide more equitable access to the internet. They also are primarily based on mesh networking protocols, which is different from the the WiFi protocols we use to connect to our home routers.

guifi.net is based in Spain. This is one of the largest community networks in the world, known for its open, free, and neutral network.

Then we have Freifunk in Germany. It emphasizes free and open communication, promoting local connectivity and community access. It operates over the BATMAN ADV mesh protocol.

In Greece, the Athens Wireless Metropolitan Network is a grassroots initiative that connects individuals and organizations across the city. It is also one of the oldest and largest active community mesh networks.

In the US, NYC Mesh is working to build a community-owned network that anyone can join, with the goal of making the internet more accessible.



This diagram shows NYC Mesh, a community-owned wireless network that provides internet connectivity across New York City using a mesh topology. Here's what's happening:

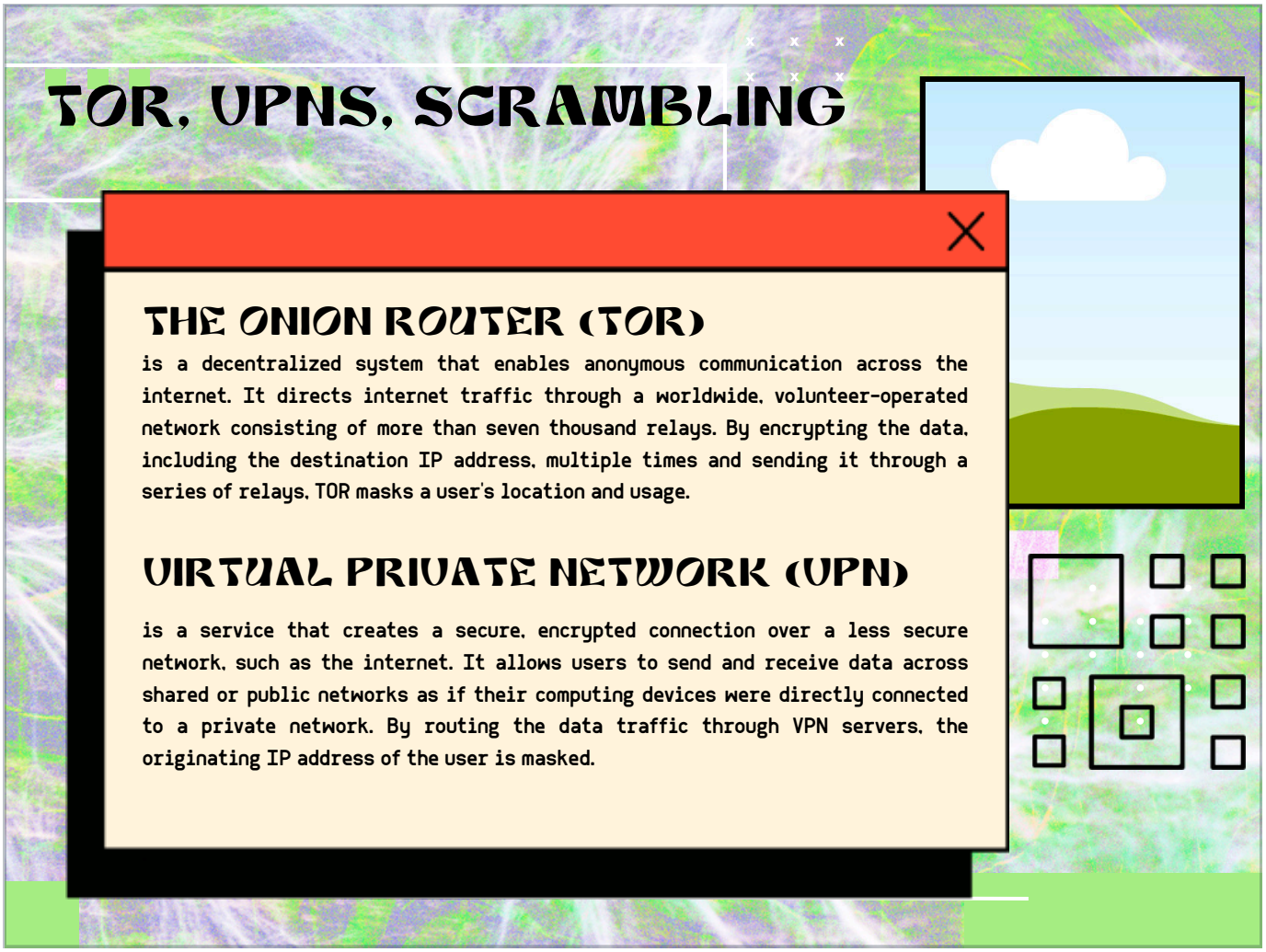
Network Structure:
The map displays hundreds of interconnected nodes (represented by colored dots) spread across Manhattan, Brooklyn, Queens, and other boroughs. The colored lines show wireless connections between nodes, with different colors likely indicating connection types or signal strengths (red, yellow, blue lines).

Hub and Spoke Pattern:
There's a major concentration of connections in what appears to be Manhattan, with several high-traffic hub nodes that have many connections radiating outward like spokes. This creates redundant pathways for data to travel between different parts of the city.

Physical Infrastructure:
The left side shows the actual hardware - rooftop-mounted directional antennas and wireless equipment that community members install on buildings. These create point-to-point wireless links between buildings, often spanning several blocks or even across rivers.

Mesh Topology Benefits:
Unlike traditional internet service that relies on centralized ISP infrastructure, this mesh network creates multiple pathways between any two points. If one node fails, traffic can automatically reroute through alternative paths, making the network more resilient.

Community-Driven:
Each node is typically installed and maintained by volunteers, creating a decentralized internet infrastructure that doesn't depend on major telecommunications companies. This provides an alternative to traditional broadband services while demonstrating how communities can build their own digital infrastructure.

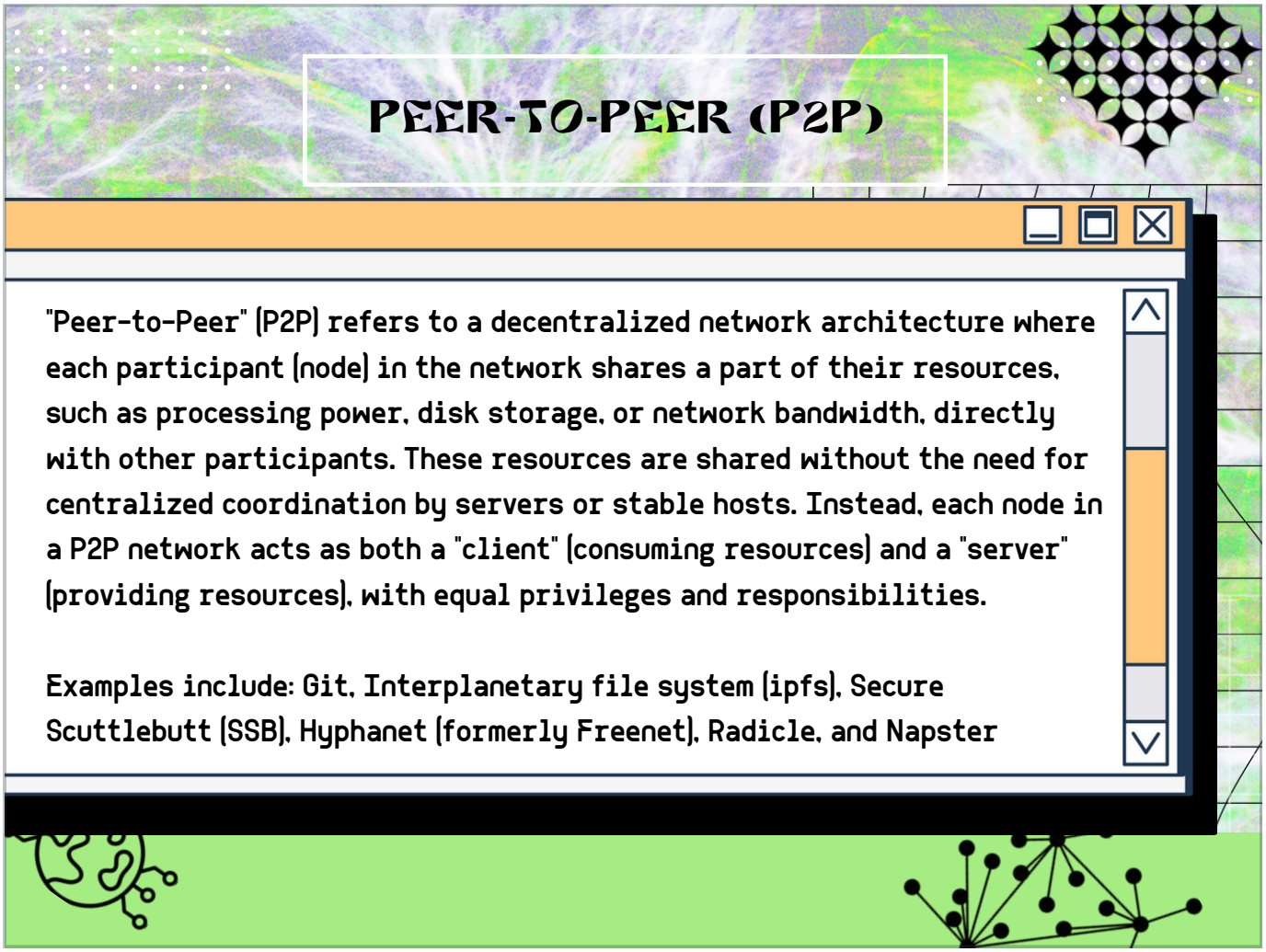


In this slide, we look at privacy and security with a focus on TOR and VPNs.

First is TOR, or The Onion Router. As a decentralized system, it enables anonymous communication by routing internet traffic through a network of over seven thousand volunteer-operated relays. This approach, combined with multi-layered encryption, effectively masks a user's location and internet usage. It's like navigating a maze that keeps changing, making it difficult for anyone to track your path.

Virtual Private Networks, or VPNs, create a secure, encrypted connection over less secure networks, like the internet. By routing your data through VPN servers, your original IP address is concealed, providing an added layer of privacy.

Both TOR and VPNs offer unique ways to protect our online footprint, giving us cover to navigate the digital world with slightly more confidence.



"Peer-to-Peer" or P2P networks are fascinating because they operate without a central authority. Imagine a network where each participant, or node, shares resources directly with others. This could be processing power, storage, or network bandwidth. Essentially, every node is both a "client" and a "server." This shared responsibility makes the network more resilient and egalitarian.

Some well-known examples are Git, IPFS, Secure Scuttlebutt, Hyphanet, Radicle, and Napster. These platforms demonstrate the potential of collaboration without relying on centralized servers.

Git is a version control system used for managing changes made to a codebase. It is particularly useful for managing changes happening within a team working on the same codebase.

IPFS or the InterPlanetary File System is a peer-to-peer, content addressable decentralized file system. When we say "content addressable," we mean that data is identified, located, and retrieved by unique address that is based on the file's contents rather than something like an IP ad dress.

Secure Scuttlebutt, or SSB, is also a decentralized, peer-to-peer networking protocol, but focused

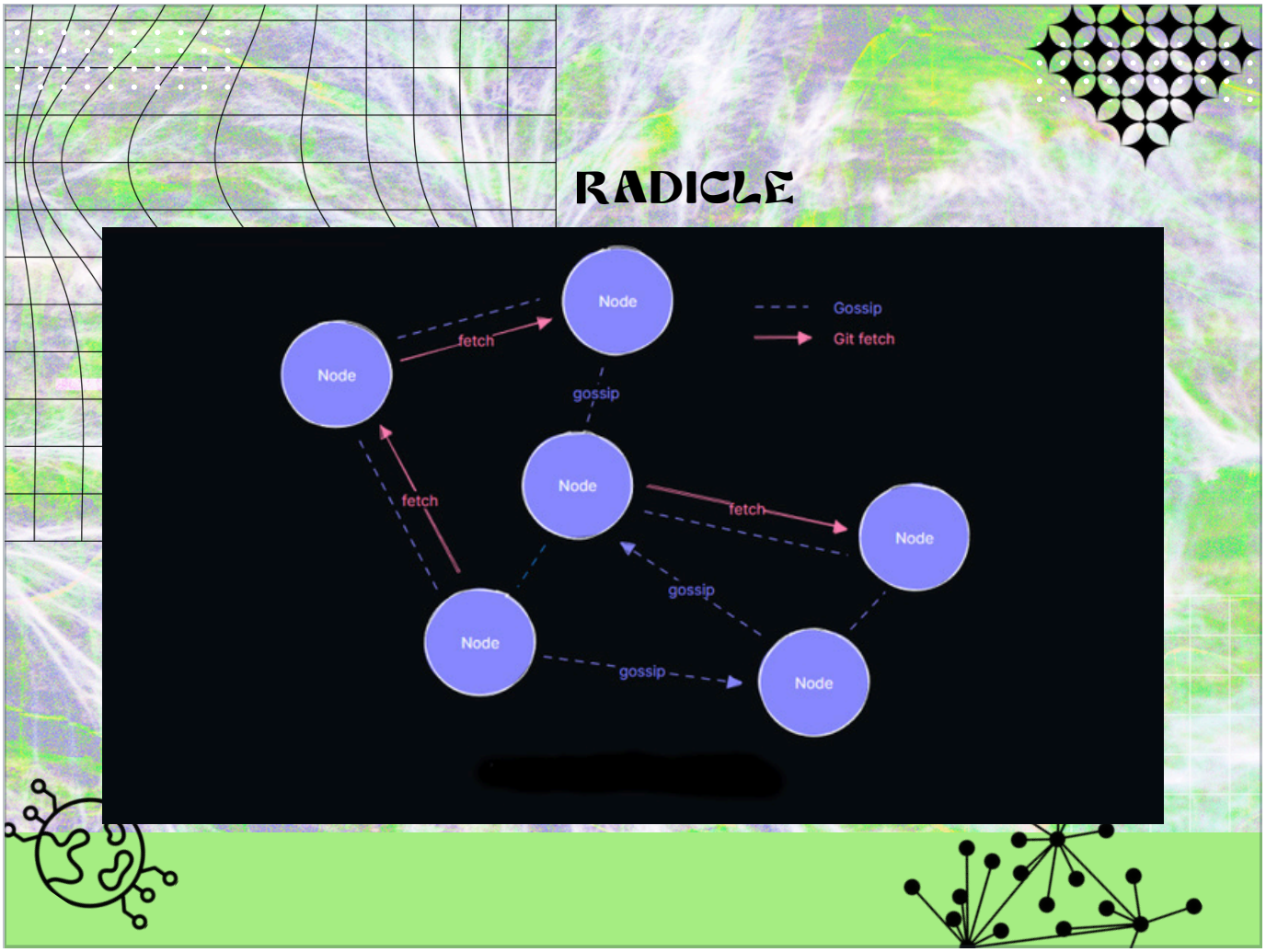
on social networks instead of file sharing.

Hyphanet is a peer-to-peer (P2P) anonymous network and decentralized data store designed for censorship-resistant publishing and privacy-preserving communication

Radicle is a decentralized, open-source code collaboration platform, which uses Git and a gossip protocol for data sharing among users' nodes rather than relying on centralized servers like GitHub. We will see a diagram that illustrates how the gossip protocol Radicle uses works.

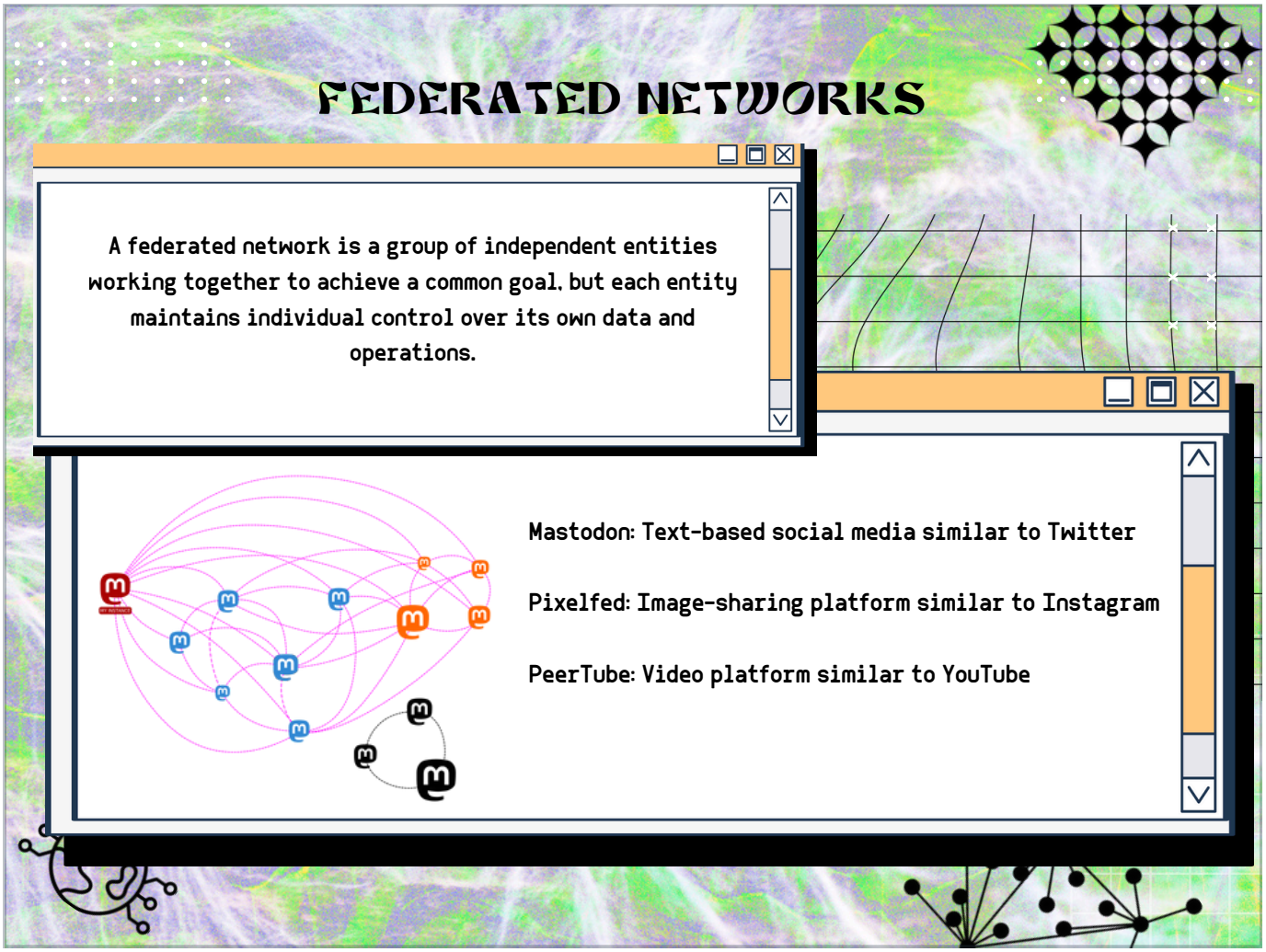
Finally Napster was a peer-to-peer file sharing protocol launched in 1999 that enabled the free mass-sharing of music files.

Consider how these networks empower individuals by decentralizing control. This is a key takeaway from the concept of P2P - it's about shared power and responsibility.



This image demonstrates the functioning of the peer-to-peer "gossip" protocol that Radicle is built upon.

It works by nodes periodically and randomly exchanging information - like repository updates or membership status - with a few peers. This process, similar to how gossip spreads through a crowd, efficiently and reliably distributes information across the decentralized network, ensuring all nodes eventually have the latest data without a central server. The protocol is used for repository discovery, replication, and to build robust routing tables, allowing Radicle to offer GitHub/GitLab-like functionality in a fully sovereign, peer-to-peer manner.



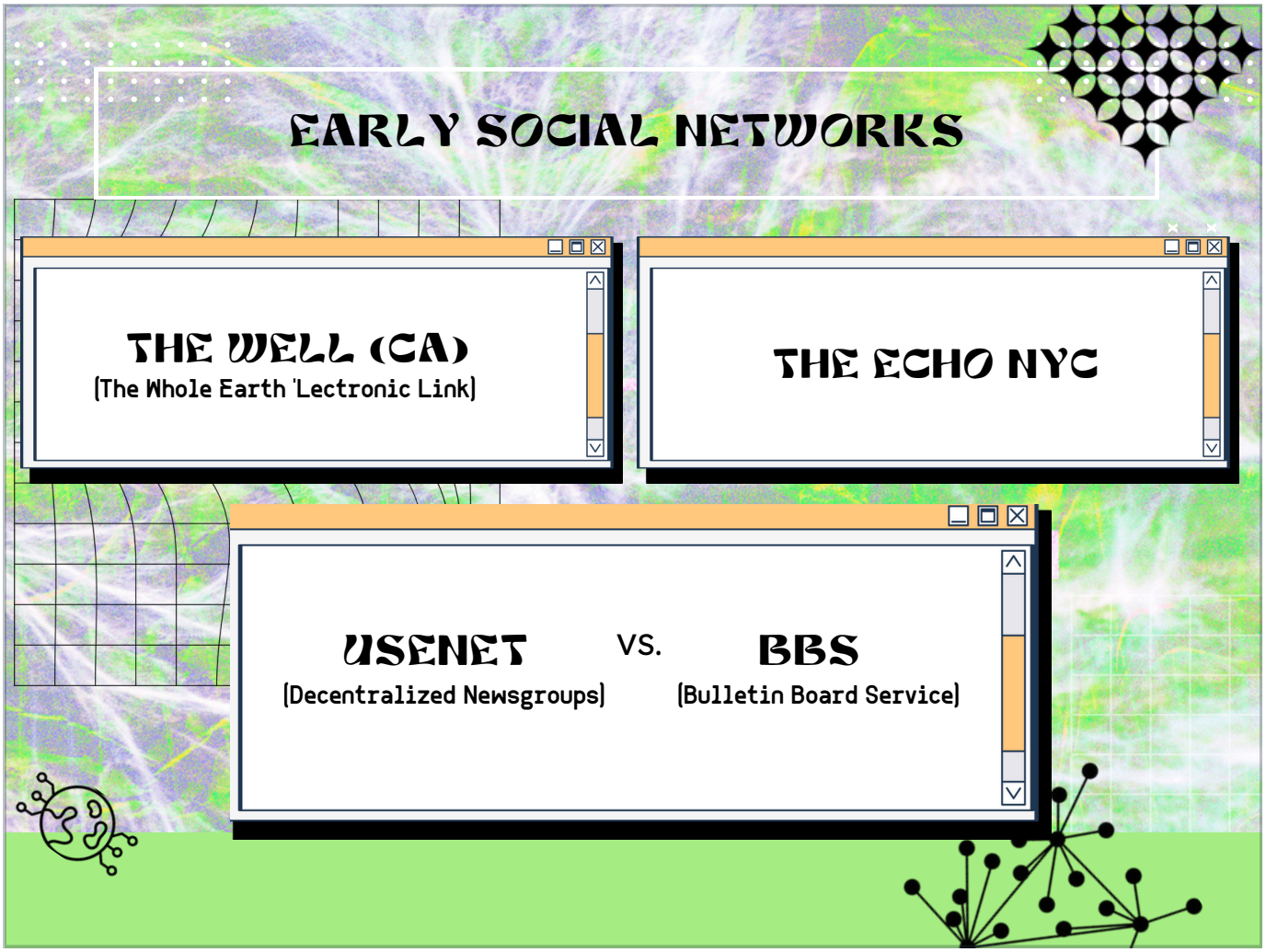
Federated networks are all about collaboration and independence. Each network works towards a shared goal, but maintains control over its own data and operations. This means users can have a sense of community without sacrificing privacy or control.

Consider Mastodon, a text-based platform similar to Twitter, where users engage in conversations across a network of independent servers.

Then there's Pixelfed, an image-sharing platform akin to Instagram, offering a decentralized approach to visual storytelling.

PeerTube provides a video-sharing experience reminiscent of YouTube, allowing creators to host their own content.

These platforms show how federated networks offer flexibility and empower users.



On this slide, we look at some of the earliest platforms. These early networks laid the groundwork for modern digital communities.

First, we have "The Echo NYC," known as a place for vibrant discussions among a diverse group of users in New York City. It was a hub for creative minds and emerging ideas.

"The Well," or "The Whole Earth 'Lectronic Link," which was a pioneering platform in California. It encouraged open conversations and knowledge sharing, becoming a model for online communitie s.

"Usenet," is a decentralized collection of newsgroups. It allowed users to post and read messages on various topics, promoting a free exchange of ideas across the globe.

Lastly, the "BBS," or "Bulletin Board Service," was a text-based system where users could connect via modems to share files and messages. It played a crucial role in the early days of online interaction.

These platforms were the precursors to what we now consider social media, each contributing

uniquely to the evolution of online communication.

These are some images from BBS's in the 90s, to give you an idea of what they looked like. Their interfaces were all text-based with colored ANSI graphics.

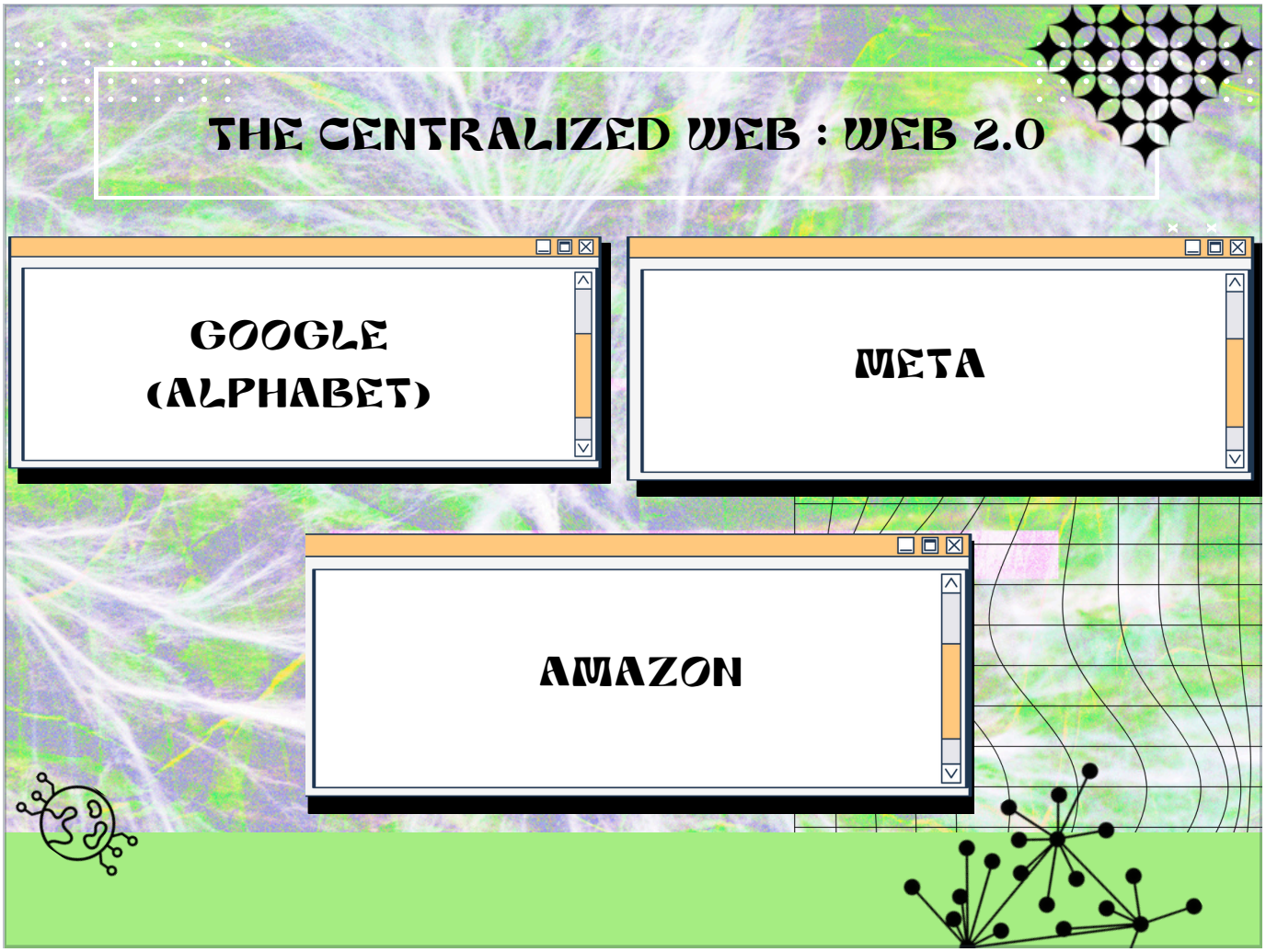
Here we look at some early Internet Service Providers, or ISPs.

AOL Instant Messenger was a near ubiquitous popular chat application. It made real-time communication widely accessible.

MCI Mail brought email into the business world, offering a reliable service that many companies came to depend on. It was part of the foundation that supported modern email systems.

CompuServe was a pioneer in offering a comprehensive range of services, from email to forums, paving the way for future online service providers.

These platforms laid the groundwork for the centralized web we use today, leading into the era of Web 2.0 with companies like META, Amazon, and Alphabet.

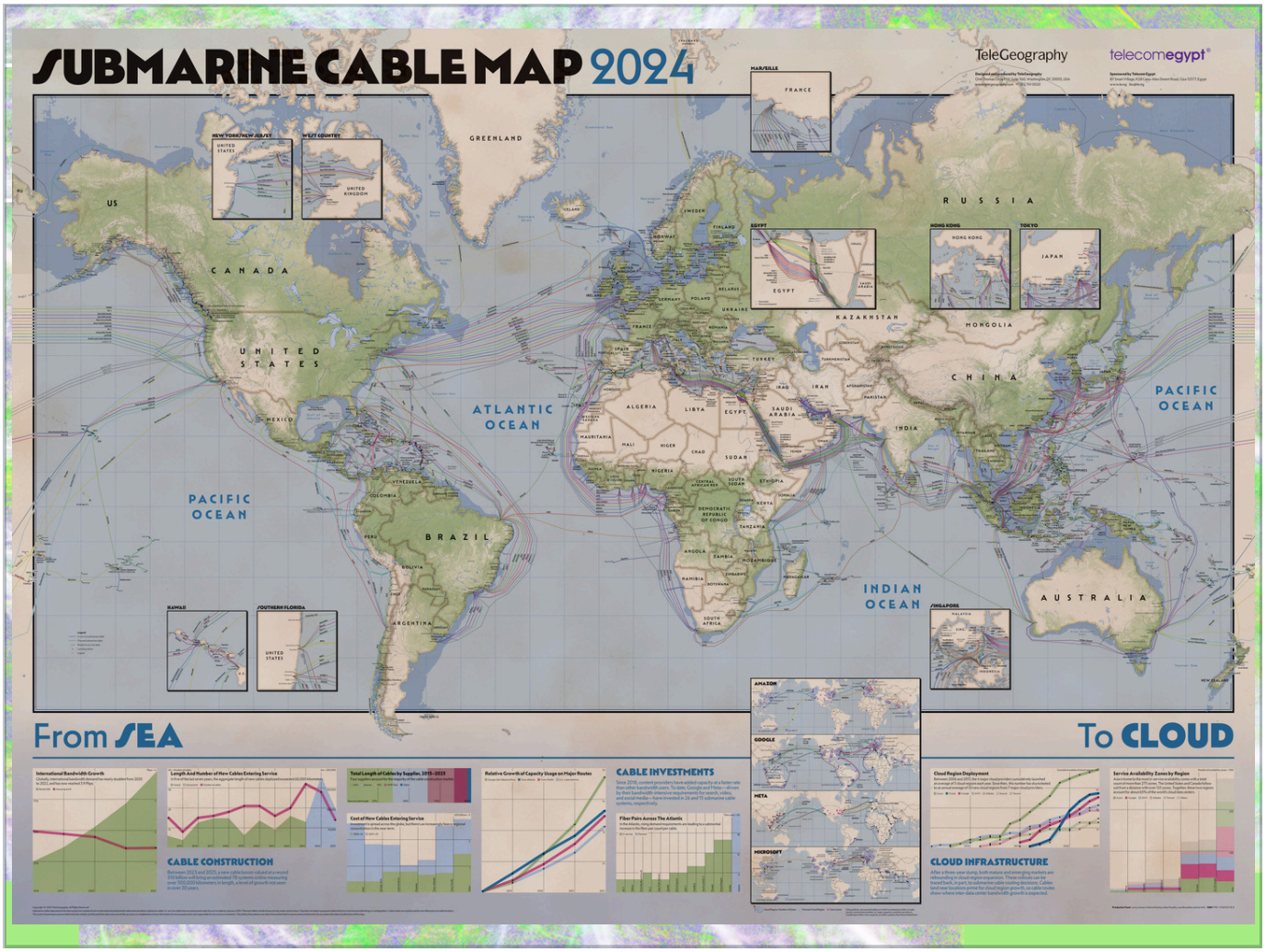


Google's position as Alphabet's primary subsidiary demonstrates how control over search translates into significant market power. As the dominant search engine handling over 90% of global search queries, Google generates approximately $200 billion annually, which it uses to secure exclusive default search placement agreements with device manufacturers and browser companies. For example, Google pays Apple an estimated $18 billion per year to remain the default search engine on iPhones and Safari. These payment arrangements create substantial barriers for competing search engines, as rivals cannot match Google's financial offers regardless of their search technology quality. This creates a reinforcing cycle: Google's search dominance generates advertising revenue, which funds exclusive placement deals, which maintains search dominance. The company's integration across multiple services, like search, email, maps, mobile operating systems, and web browsers, further strengthens its position by creating user data advantages and ecosystem lock-in effects.

Meta's evolution from Facebook into a multi-platform company illustrates how social connectivity has become foundational to the modern internet economy. The company operates the world's largest social networks. Facebook has nearly 3 billion users and Instagram with over 2 billion, together generating over $100 billion in annual revenue through targeted advertising based on extensive user data collection. Meta's strategy of acquiring potential competitors, notably

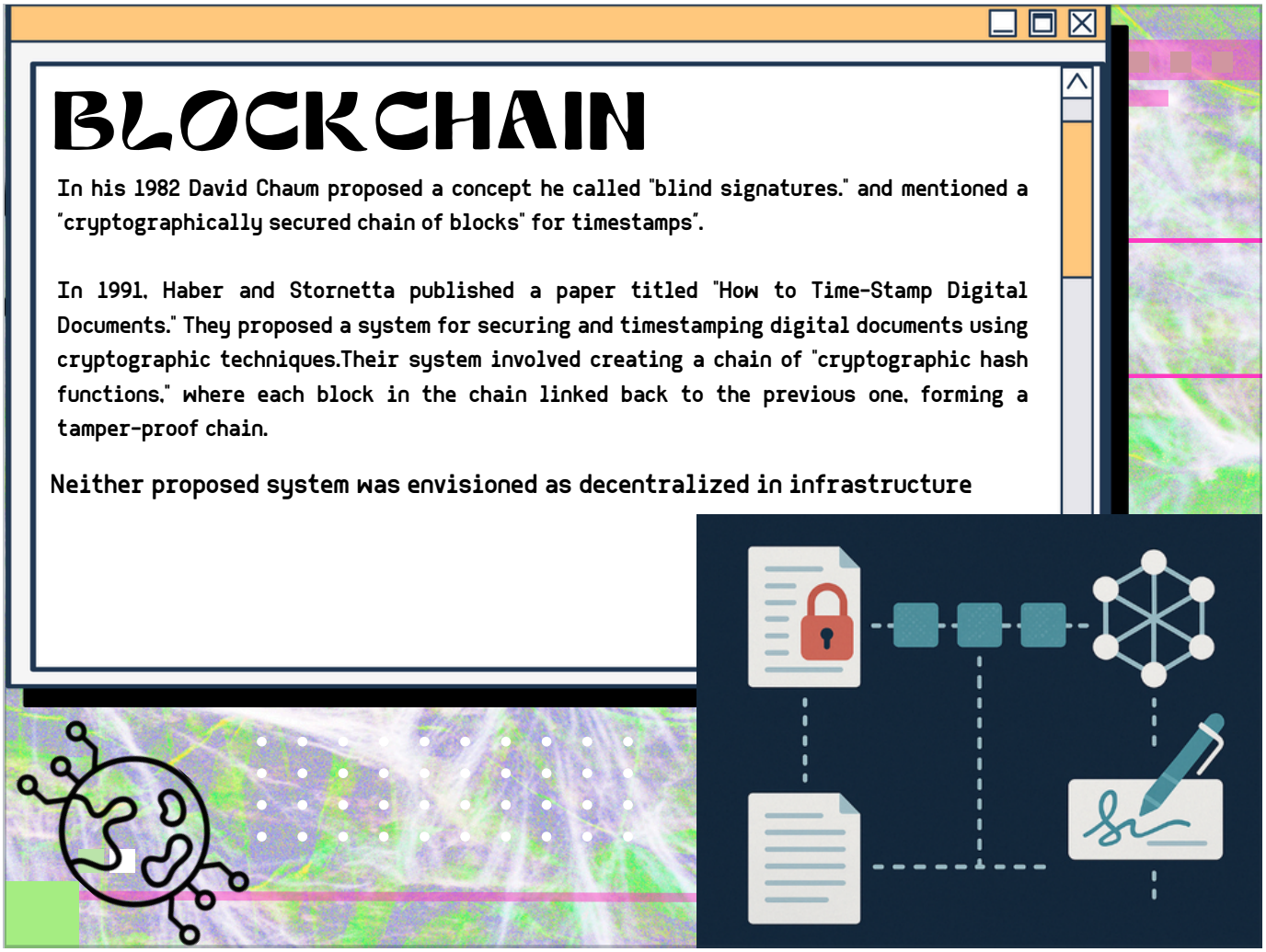
Instagram for $1 billion in 2012 and WhatsApp for $19 billion in 2014, has consolidated social media market share and eliminated emerging rivals. The company's control over multiple communication platforms creates significant network effects and switching costs that discourage users from migrating to alternatives, while its algorithmic content curation shapes what information billions of users see daily, extending Meta's influence beyond social networking into news distribution, commerce, and political discourse.

Amazon's transformation from an online bookstore into a comprehensive e-commerce and cloud computing giant demonstrates how digital platforms can reshape entire industries while concentrating market power. The company controls approximately 40% of U.S. e-commerce through its marketplace platform, which hosts millions of third-party sellers while simultaneously competing against them with Amazon's own private-label products, creating inherent conflicts of interest. Amazon's integrated ecosystem, combining retail, logistics, cloud services (AWS), advertising, and entertainment, creates significant barriers for competitors who cannot match its scale economies and cross-subsidization capabilities. The company's control over essential e-commerce infrastructure, from warehousing to last-mile delivery, allows it to set terms that smaller retailers struggle to meet, while its access to marketplace seller data provides competitive advantages in product development and pricing. Amazon's dominance extends beyond retail through AWS, which powers roughly one-third of the internet's cloud infrastructure, giving the company influence over both how people shop and how businesses operate online, raising concerns about market concentration and the company's ability to favor its own services across multiple industries.



This map illustrates an overview of where the submarine cables that carry internet traffic lie on the bottom of the ocean floor.

The centralized web concentrates immense power in the hands of a few technology giants where Google controls information discovery, Meta dominates social communication, and Amazon shapes commerce, creating monopolistic conditions that stifle competition and innovation. This concentration extends to physical internet infrastructure, as these same companies increasingly own the submarine cables that carry global internet traffic, with Google, Meta, Amazon, and Microsoft now controlling or co-owning a significant portion of transoceanic data transmission capacity. The result is reduced consumer choice, higher costs passed through the economy, and and outsized amount of control over both digital platforms and the underlying infrastructure that enables global internet connectivity, shutting out market competition, data sovereignty, and democratic discourse.



Here we look atearly foundational concepts leading to blockchain technology.

In 1982, David Chaum introduced an idea known as "blind signatures." He also talked about a “cryptographically secured chain of blocks” for timestamps. This was an important first step in using cryptography to enhance security.

Then in 1991, Haber and Stornetta published a pivotal paper titled "How to Time-Stamp Digital Documents." They proposed a system employing cryptographic techniques to secure and timestamp documents. Their method involved creating a chain of "cryptographic hash functions," where each block linked back to the previous one, forming what we now recognize as a tamper-proof chain.

It's important to note that neither Chaum's nor Haber and Stornetta's proposals initially envisioned a decentralized infrastructure. This marks a significant distinction from what we now understand as blockchain technology.

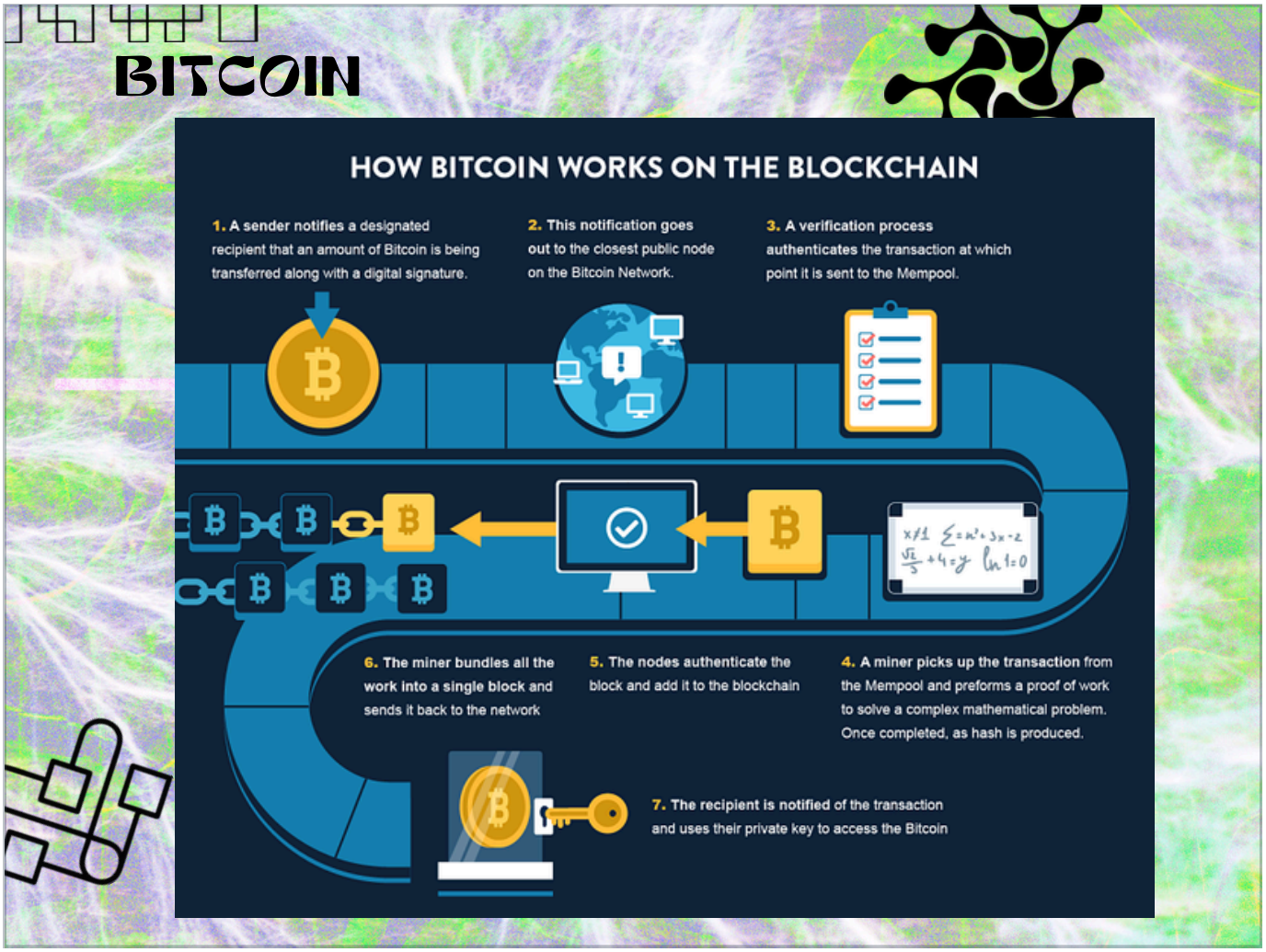
Bitcoin's transaction data isn't stored on a single server. Instead, it's maintained on a distributed ledger, or a network of computers, or nodes, worldwide. This decentralization is key to its security and transparency.

There is also the concept of Bitcoin mining. High-powered, specialized mining rigs are essential for solving proof-of-work problems efficiently. These rigs are far more effective than personal computers, but they are very expensive to run.

Electricity is another critical factor. The computational power required for proof-of-work consumes significant electricity. Costs can vary greatly depending on location and rates, making this a major consideration for miners.

Lastly, mining difficulty is crucial. As more miners join the network, the difficulty of solving these puzzles increases. This adjustment ensures a consistent rate of block creation, but it also demands more computing power and higher electricity costs.

In essence, nodes in this network are constantly verifying transactions using proof-of-work. This process eliminates the need for a central authority, ensuring a secure and transparent system.



This graphic describes how Bitcoin works on the blockchain:

1. A sender notifies a designated recipient that an amount of Bitcoin is being transferred along with a digital signature.

2. This notification goes out to the closest public node on the Bitcoin network.

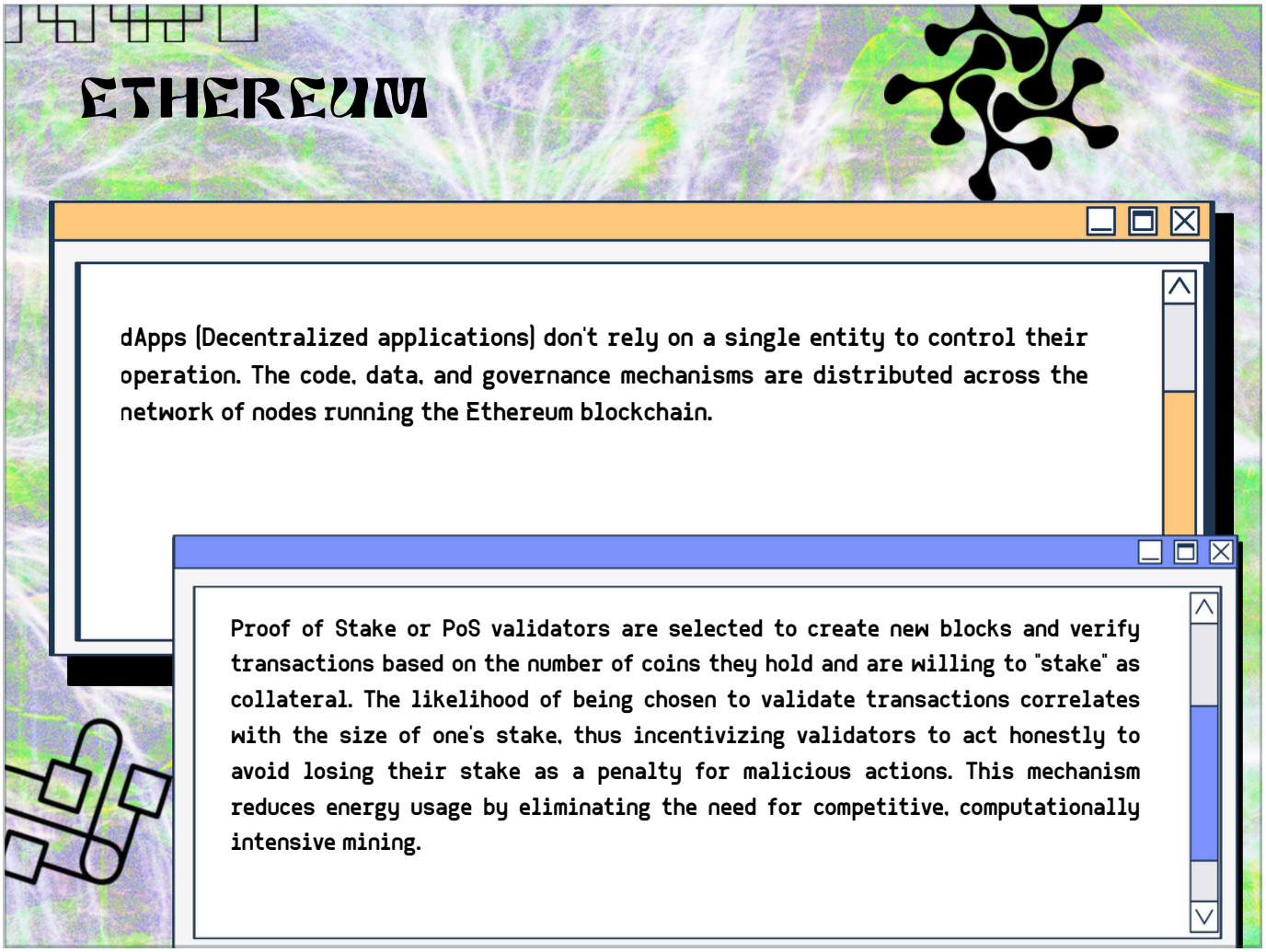
3. A verification process authenticates the transaction at which point it is sent to the Mempool.

4. A miner picks up the transaction from the Mempool and performs a proof of work to solve a complex mathematical problem. Once completed, a hash is produced.

5. The nodes authenticate the block and add it to the blockchain.

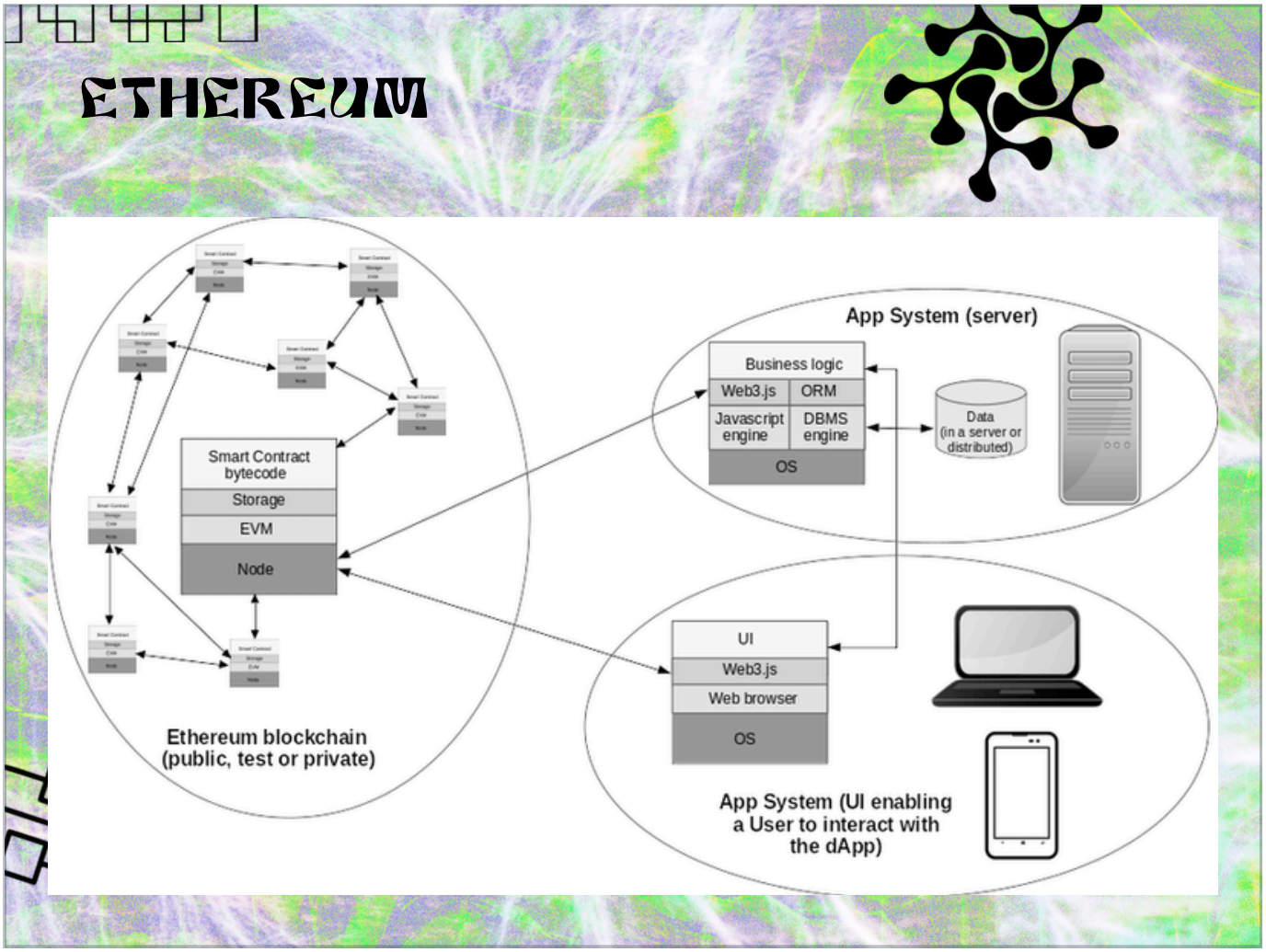
6. The miner bundles all the work into a single block and sends it back to the network.

7. The recipient is notified of the transaction and uses their private key to access the Bitcoin.



Ethereum revolutionizes how applications are developed and managed by introducing dApps, or decentralized applications. These applications operate without a single control point. Instead, they're powered by the Ethereum blockchain, distributing the code, data, and governance across numerous nodes.

Now, let's talk about the consensus mechanism Proof of Stake, or PoS. Unlike Bitcoin's energy-intensive proof-of-work, PoS selects validators to create new blocks based on the number of coins they’re willing to stake as collateral. This process not only ensures honesty among validators, since they risk losing their stake for any malicious actions, but it also significantly reduces energy consumption.

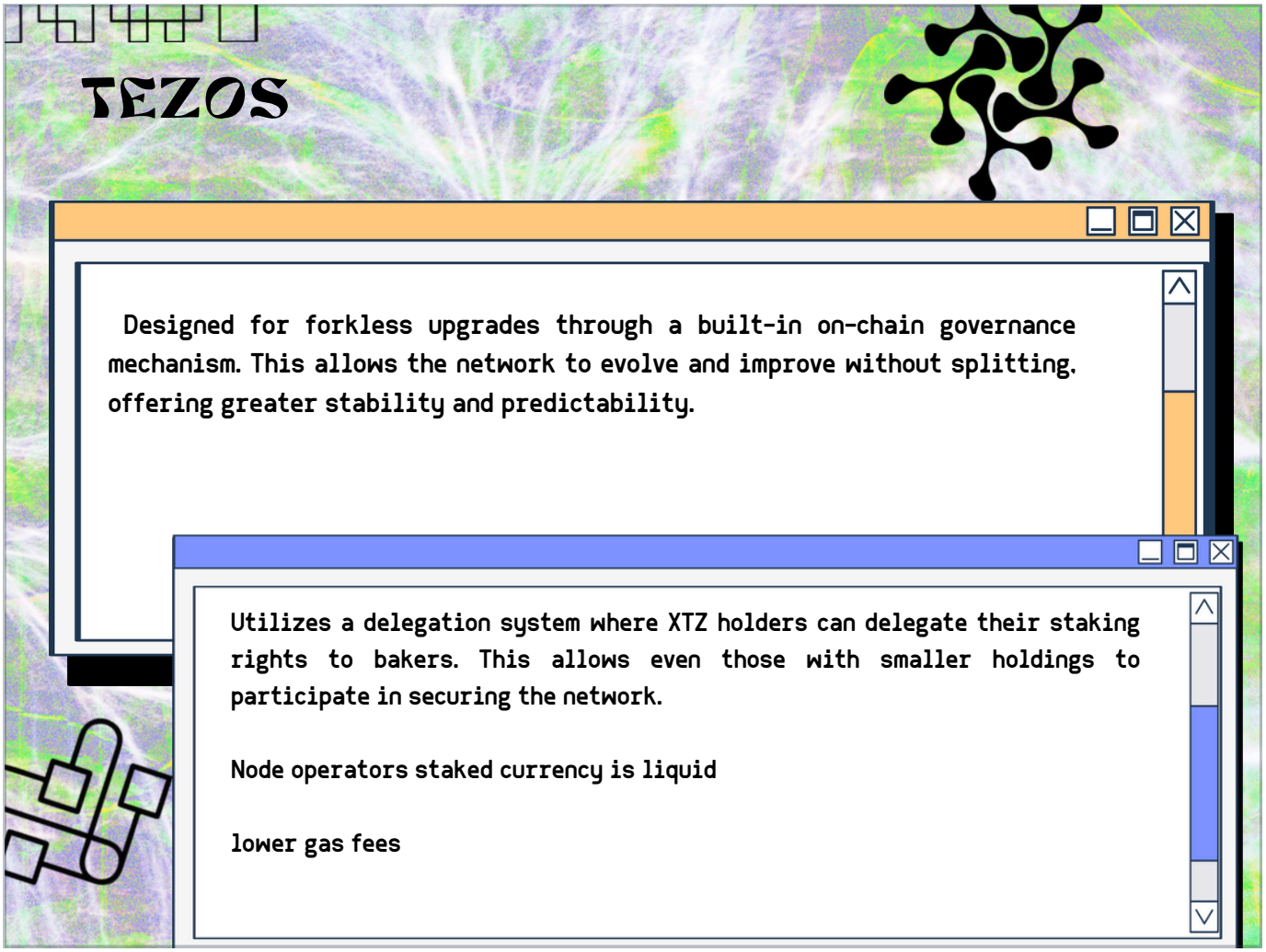


This diagram illustrates the architecture of a decentralized application (dApp) built on the Ethereum blockchain, showing how it differs from traditional centralized web applications. On the left side, you see the Ethereum blockchain network - a distributed system where multiple nodes (computers) communicate with each other to maintain a shared ledger. At the center is a smart contract containing bytecode that runs on the Ethereum Virtual Machine (EVM), with its own storage layer. This smart contract serves as the backend logic that would traditionally run on a centralized server.
On the right side, there are two App Systems that interact with the blockchain:

Server-side system - Contains business logic, Web3.js integration, JavaScript engine, ORM, DBMS engine, and connects to distributed data storage
Client-side system - The user interface layer with Web3.js, web browser, and operating system that users interact with through devices (laptop and mobile phone)

The key difference from traditional web architecture is that instead of all the backend logic running on centralized servers owned by one company, the core application logic runs as smart contracts on the decentralized Ethereum network. The Web3.js libraries enable both the server and client systems to communicate with the blockchain, while still allowing for some centralized components

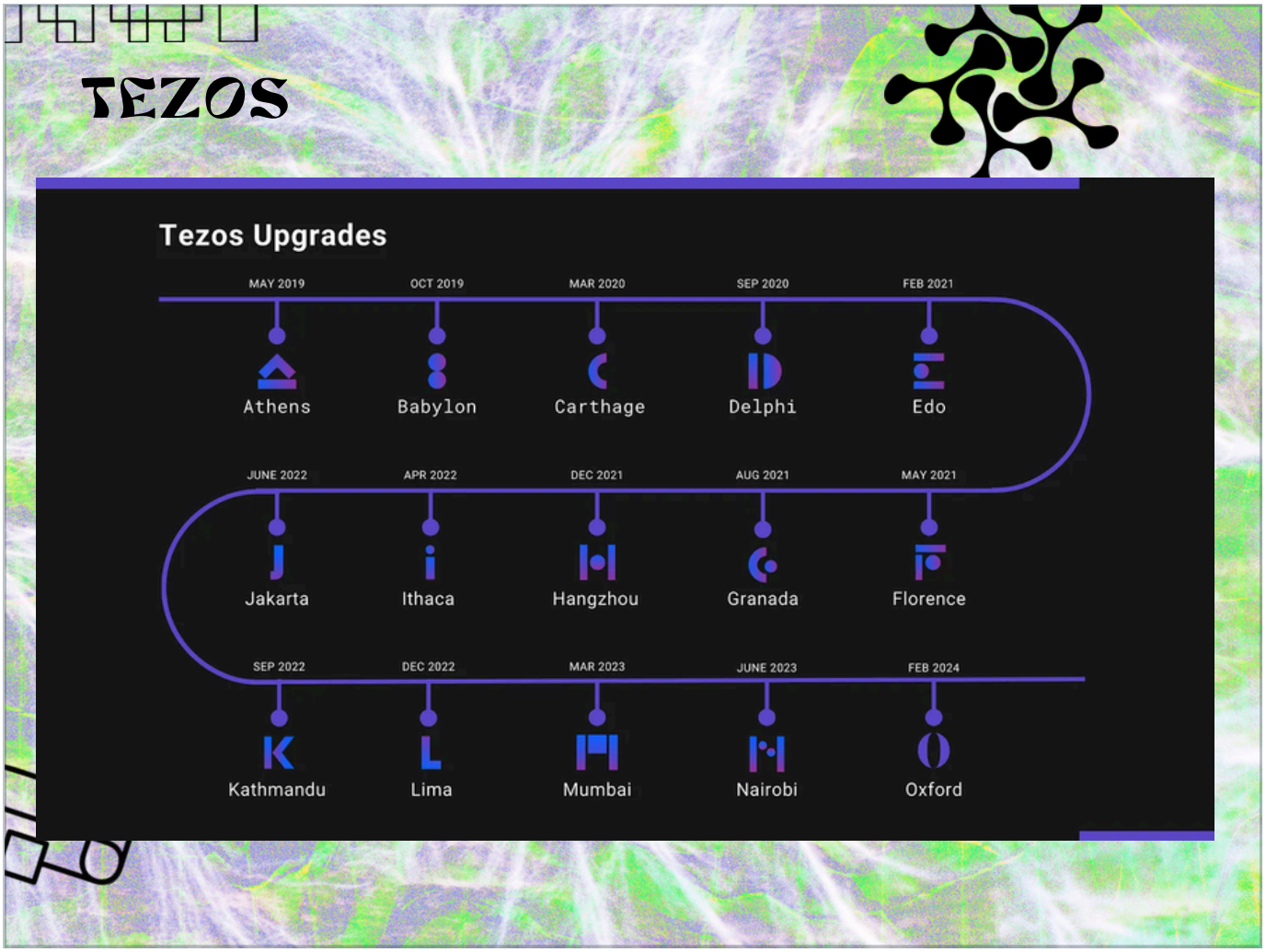
like databases and user interfaces when needed.
This hybrid approach decentralizes critical functions while maintaining usability through familiar web interfaces.



Tezos is designed for forkless upgrades, due to its built-in on-chain governance mechanism. This means the network can evolve without splitting, ensuring more stability and predictability in its growth.

Another interesting feature of Tezos is its delegation system. Here, XTZ holders can delegate their staking rights to bakers. This setup is inclusive; it allows even those with smaller holdings to participate in network security, fostering a sense of community involvement.

Moreover, the staked currency in Tezos remains liquid, adding flexibility for node operators. Plus, with lower gas fees, Tezos provides a cost-effective environment for transactions.



This diagram shows the timeline of Tezos blockchain network upgrades from May 2019 to February 2024. Tezos uses a unique on-chain governance system that allows the protocol to upgrade itself through voting by stakeholders, avoiding the need for hard forks that split the networ k.
Each upgrade is named after a city and represented by a distinctive logo:
First generation (2019-2021):

Athens (May 2019) - The first major upgrade
Babylon (Oct 2019)
Carthage (Mar 2020)
Delphi (Sep 2020)
Edo (Feb 2021)

Second generation (2021-2022):

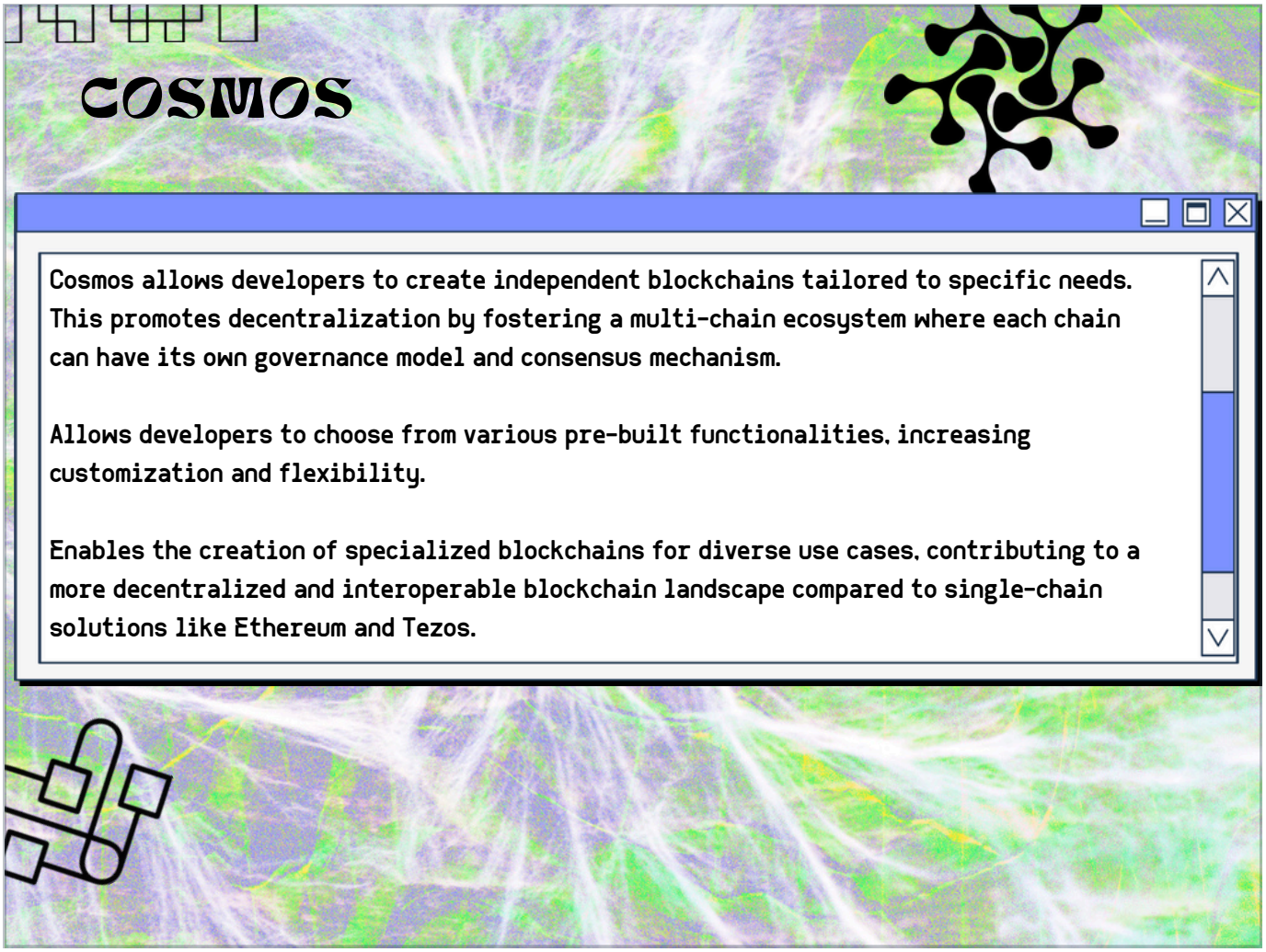
Florence (May 2021)
Granada (Aug 2021)
Hangzhou (Dec 2021)

Ithaca (Apr 2022)
Jakarta (June 2022)

Third generation (2022-2024):

Kathmandu (Sep 2022)
Lima (Dec 2022)
Mumbai (Mar 2023)
Nairobi (June 2023)
Oxford (Feb 2024)

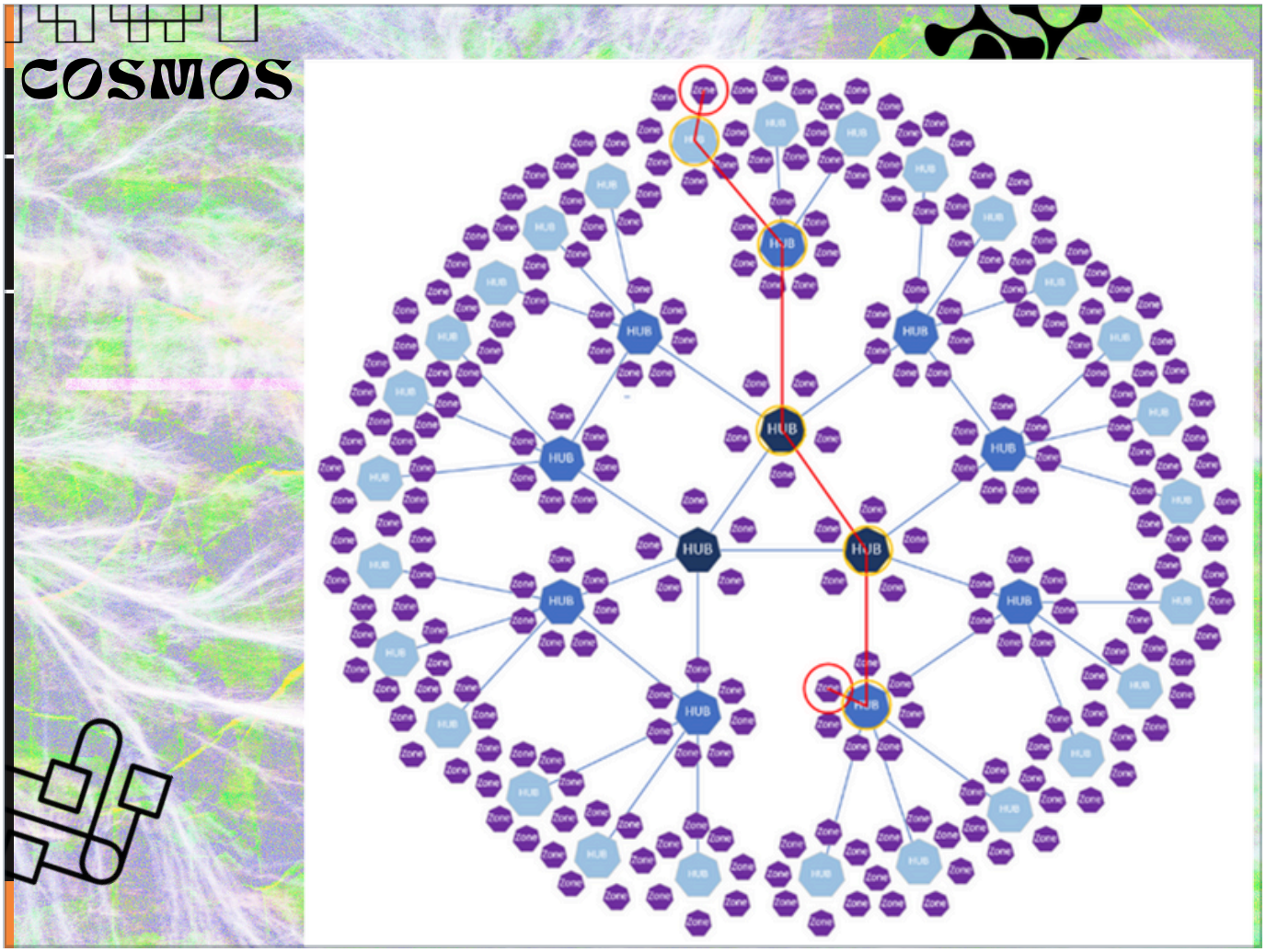
The curved timeline design emphasizes the continuous, evolutionary nature of Tezos development, with upgrades occurring roughly every 3-6 months. Each upgrade typically introduces new features, optimizations, or protocol improvements while maintaining network continuity. This approach contrasts with other blockchains that require contentious hard forks to implement major changes.



Cosmos offers a unique approach to blockchain development by allowing developers to create independent blockchains, each tailored to specific needs. This significantly promotes decentralization, as it fosters a multi-chain ecosystem. In this ecosystem, each chain can have its own governance model and consensus mechanism, unlike the single-chain solutions of Ethereum and Tezos.

Another aspect of Cosmos is the ability for developers to choose from various pre-built functionalities. This choice increases both customization and flexibility, enabling developers to build specialized blockchains for diverse use cases. This approach contributes to a more decentralized and interoperable blockchain landscape.

Cosmos stands out by addressing the challenges of interoperability and customization in the blockchain space. This makes it an attractive option for developers looking for alternatives to more rigid single-chain solutions.



This diagram illustrates the Cosmos ecosystem's "Internet of Blockchains" architecture, showing how multiple independent blockchains can interconnect through a hub-and-spoke model.

Key Components:

Hub Networks - The larger blue circles labeled "HUB" represent Cosmos Hub and other major hub blockchains that serve as central connection points. These hubs facilitate inter-blockchain communication and asset transfers.

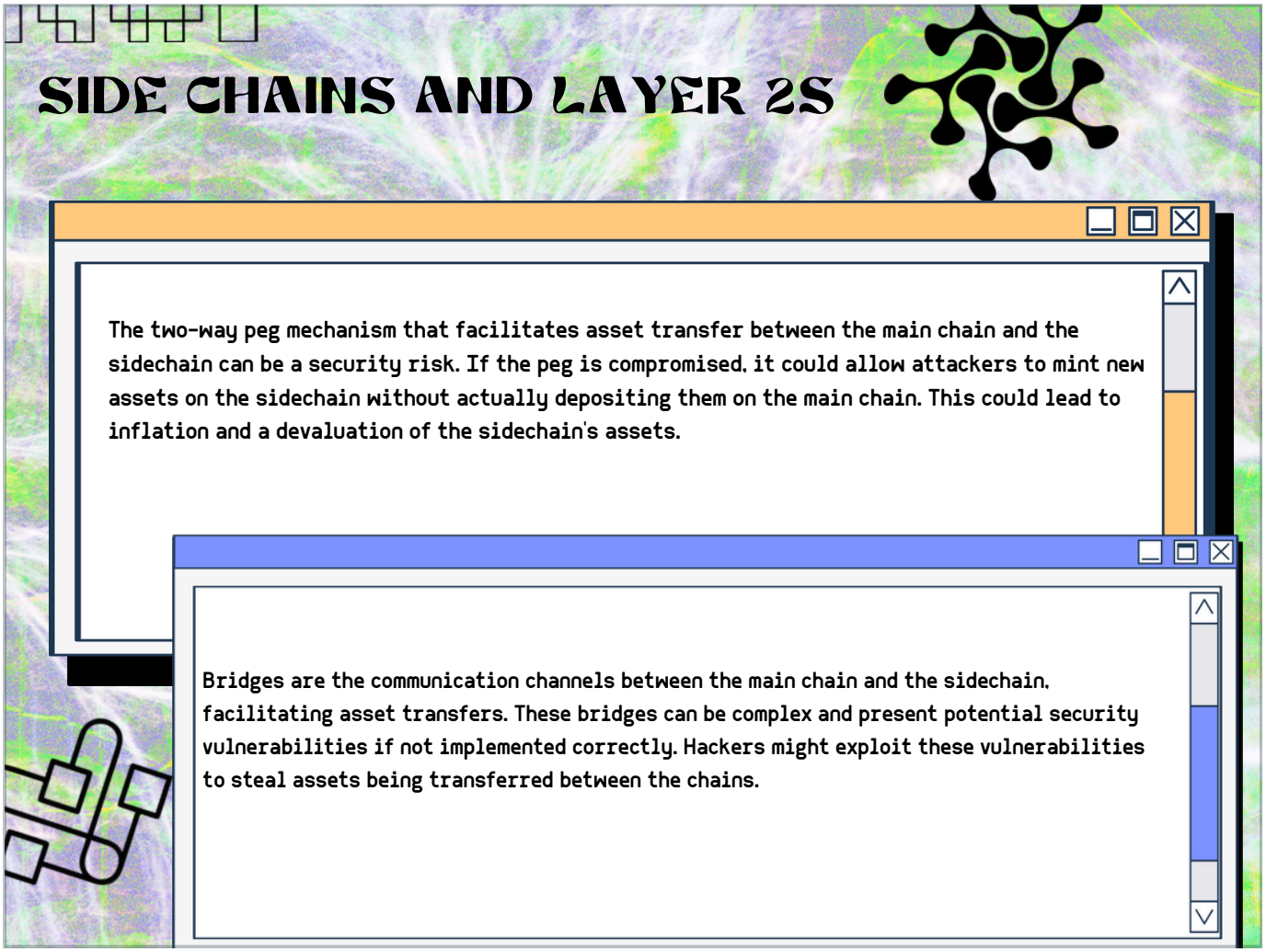
Zone Blockchains - The smaller purple circles represent individual blockchain "zones" - independent blockchains built using the Cosmos SDK that can have their own governance, validators, and tokens.

Inter-Blockchain Communication (IBC) - The blue lines connecting hubs to zones and zones to each other represent IBC protocols, which enable secure communication and asset transfers between different blockchains.

Network Topology:

The central hub appears to be the main Cosmos Hub, with several secondary hubs positioned around it
Each hub connects to multiple zones, creating clusters of interconnected blockchains
The red highlighted path shows a specific route for cross-chain transactions
Some zones (light blue circles) appear to be larger or more significant networks

This architecture allows each blockchain to maintain sovereignty and specialize in specific use cases while remaining interoperable with the broader Cosmos ecosystem. Unlike monolithic blockchains, this modular approach enables scalability and customization while preserving the ability to transfer assets and data across chains.



Side Chains and Layer 2 solutions. These technologies are crucial for scaling and enhancing blockchain systems.

First, let's talk about the two-way peg mechanism. This is what helps assets move back and forth between the main chain and the sidechain. However, if this peg is compromised, it poses a significant security risk. Imagine if attackers could mint new assets on the sidechain without depositing them on the main chain - it could lead to inflation and decrease the value of those assets.

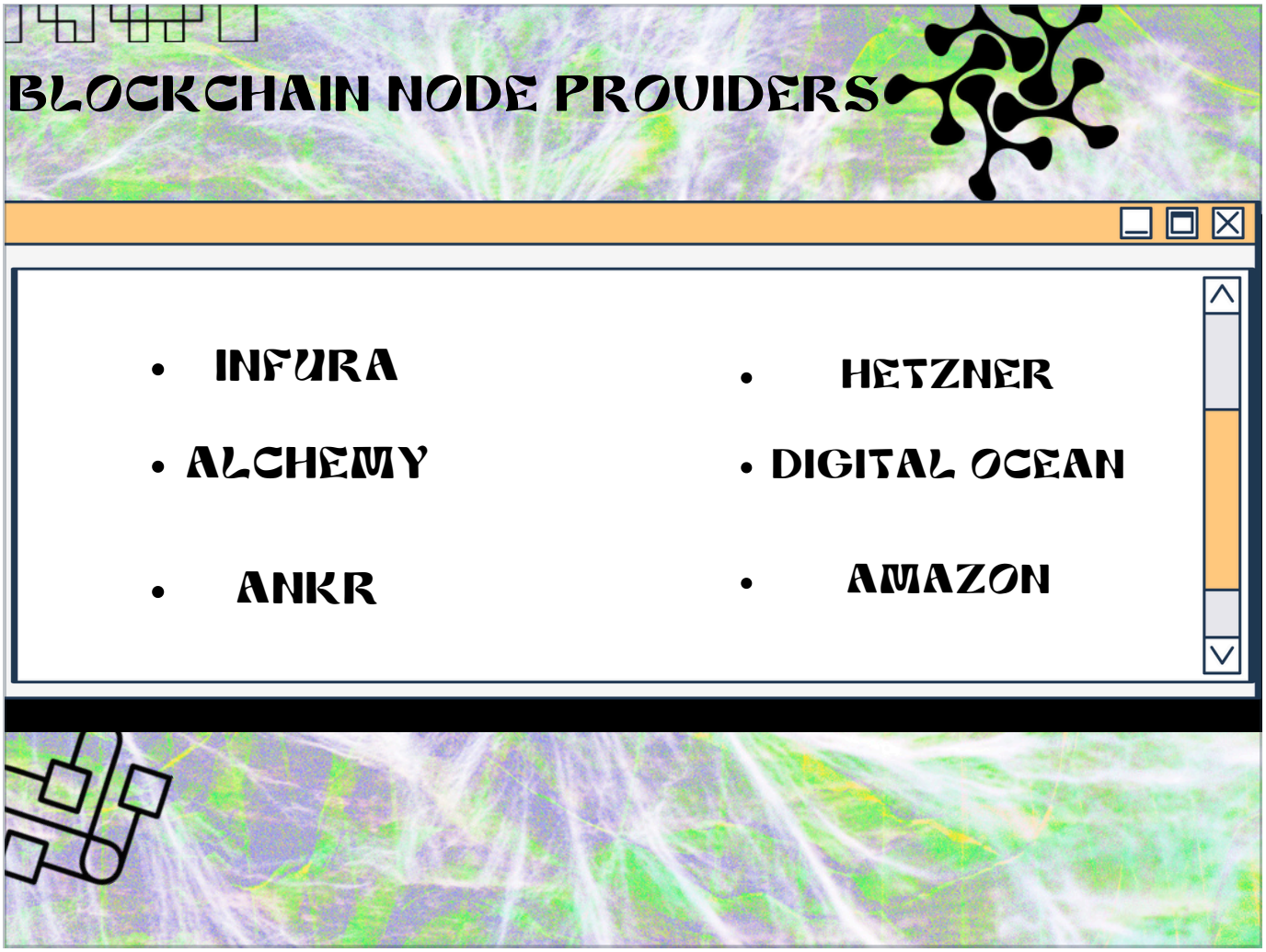
Next, consider the bridges. These are like communication channels that make asset transfer possible between chains. While they are essential, their complexity can introduce security vulnerabilities. If not implemented carefully, hackers might find ways to exploit these vulnerabilities to steal assets.

So, while Side Chains and Layer 2s offer great scalability and efficiency, it’s important to be aware of these potential risks and work towards robust security solutions.

Custodial chains like Flow and Solana take a unique path compared to major players like Bitcoin, Ethereum, and Tezos. While they still use blockchain technology for key features like transparency and immutability, they include centralized aspects in their governance and consensus processes.

The key difference here is the limited number of validators, which are often under the control of platform operators. This raises important questions about centralization and the risk of censorship.

However, there's a trade-off: this structure can lead to significantly faster transaction processing times and lower fees.



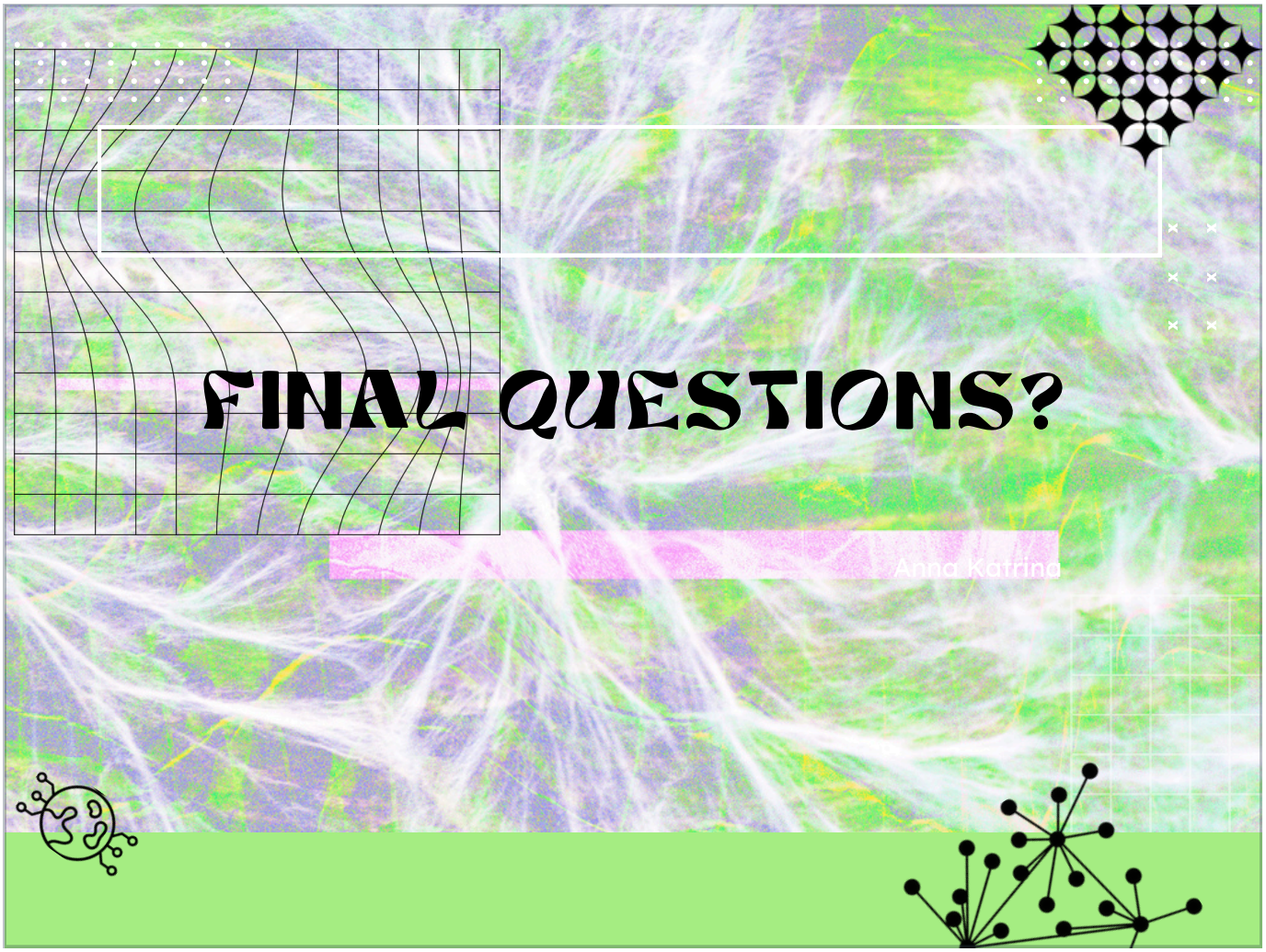
Here we look at blockchain node providers.

Known for its reliability, Infura offers a robust platform for developers to connect to the Ethereum network without running their own nodes. This makes developing decentralized applications more accessible.

Alchemy is celebrated for its powerful suite of developer tools and analytics, helping streamline the building and monitoring of blockchain applications. It's an excellent choice for developers seeking an all-in-one solution.

Ankr is another key player, offering a decentralized infrastructure that lowers the barrier to entry for accessing blockchain nodes. Ankr focuses on providing affordable pricing and ease of use.

We also have Hetzner, Digital Ocean, and Amazon. These providers are not blockchain-specific but offer cloud services that can host blockchain nodes. Their established infrastructures provide scalability and reliability, though they might require more technical know-how to set up.


FINAL QUESTIONS?
Anna Katrina