

# Cloud Considerations

A non-comprehensive comparative guide for deciding between cloud and on-site services.

By Daniel Patterson - April 23, 2025

<tldr>

This short guide compares some of the most notable pros and cons of both cloud-based and on-site network systems. In the full article, you'll be able to find explanations for each of these points.

- Advantages of Cloud Operations
  - Consumer Perspective
    - Your Data is Always Backed Up
    - You can Access Your Data from Anywhere
    - Any Computer can be a Workstation
    - Your Service is Online when you are Offline
    - The Server Doesn't Crash
  - Software Publisher Perspective
    - Updates can be Published Instantly
    - Updates can be Rolled Back Instantly
    - Incremental Usage and Licensing
- Disadvantages of Cloud Operations
  - The Internet is not Always Available
  - Your Company's Trade Secrets are in the Hands of a Stranger
  - You Must Use What You Get
  - Your Success is Reliant on Another Party
  - Newer is not Always Better
  - Do YOU Have a Backup?
  - Product Lock-In
  - You Don't Have Any Vote on New Features
  - You Don't Have Any Vote on System Policies
  - Cloud Services can be Hacked
  - Service Creep - Paying for What You Don't Need
- Advantages of On-Site Operation
  - A Non-Internet Ecosystem
  - 100% Uptime is Possible

- A Working Version can be Retained
- Mix and Match is a Thing
- All of your Important Files are Inside the Firewall
- You Have Your Backup
- You Have Freedom to Change
- Your System is Capable of Nearly Anything
- No Limit on Services Used
- No Limit on Services Published
- Drastically Less Expensive than in the Past
- Disadvantages of On-Site Operation
  - IT Staff are Expensive
  - Maintenance and Troubleshooting can be Complicated
  - Security Configuration and Maintenance
  - Licensing is a Nuisance
  - Comparing the Real Costs

</tldr>

## Introduction

Sometimes, it seems that just about everyone has moved their operations entirely to the cloud, and after all, why not? There are numerous advantages to running a cloud-based system, not the least of which is that you need virtually no internal IT support at all, especially if you only have a few basic needs, as is the case with a small design agency whose software is mostly web-based and collaborative to begin with.

Top that off with the fact that some things only operate at their best in the cloud because of the enormous resources they require (OpenAI and its 30,000 graphics card PC, cough). For a lot of cases, we seem to have also reached a point where even the most self-sufficient organization could only attain a hybrid style of cloud vs. on-site operation at the very best.

But there are also major disadvantages to reliance on cloud services, especially to organizations that either have to assure ongoing real-time production or that have to guarantee secrecy and confidentiality, especially for regulatory reasons.

In this article, I'll go over many of the practical considerations an organization might face when deciding or reconsidering about which kind of cloud services to use.

# Advantages of Cloud Operations

As I mentioned, cloud operation carries with it several advantages that make it difficult to ignore - both for the end consumer and for the software publisher.

## Consumer Perspective

From the perspective of the customer, these are some of the most noticeable advantages of partnering with a cloud provider.

### Your Data is Always Backed Up

Many cloud-based services not only utilize redundant hardware to store your data on the server you log in to, but they synchronize any changes to your data, in real-time, at multiple other locations around the world.

Just the fact that your active data is off-site can provide immense relief if there is an emergency that disrupts operations at one or more of your physical locations. However, the added consideration that multiple active copies of your data exist simultaneously on multiple hard drives at one location and multiple locations around the world virtually guarantees that even the provider company can suffer complete disruptions at their own locations without any possibility of loss to your data.

### You can Access Your Data from Anywhere

The very nature of cloud services makes them accessible over the Internet, meaning that with only a secure Internet connection and the right permissions, you can get your work done equally well from the office downtown or from a beach-side table in Tahiti.

Even while traveling from location to location, it's easy to stop for a few minutes at a local coffee shop with WiFi access to catch up with the day's activities.

### Any Computer can be a Workstation

One of the most appealing benefits of cloud operations from the customer's perspective is the sheer simplicity of access. In most cases, any computer with a web browser and an internet connection can instantly become a functional workstation. There's no need to

allocate specialized hardware, install heavy software packages, or perform custom configurations. The power, processing, and data storage all happen at the cloud provider's location, leaving the end-user's device light, responsive, and easy to replace if needed.

This flexibility makes onboarding new users fast and cost-effective. Whether someone is working from a laptop at home, a shared office desktop, or even a thin client in a remote location, they can access the same tools and environment with minimal setup. It also greatly reduces the burden on IT support teams, since there's less that can go wrong on the user's machine, and updates or fixes can often be managed centrally.

With cloud platforms, the workstation is no longer a physical asset, but a kind of service, instantly ready to go from virtually anywhere.

## Your Service is Online when you are Offline

One of the most recognizable advantages of cloud-based operations is that your services remain available even when you're not. Whether it's an online storefront, a digital portal, or an automated client service tool, your customers can continue to interact with your organization 24/7/365, even if no one from your team is in the office, logged in, or even awake. The cloud infrastructure handles everything in the background, ensuring seamless access and continuity regardless of your local schedule or staff availability.

This always-on presence allows your organization to serve a global audience, accommodate off-hours inquiries, and generate value around the clock. Orders can be placed, support requests can be submitted, content can be accessed, and workflows can be initiated, all without needing someone to manually flip a switch.

In a world where responsiveness and availability are key to customer satisfaction, cloud operations empower your business to stay open and functional, even when you're completely offline.

## The Server Doesn't Crash

To a typical end-user, one of the most remarkable features of cloud-based services might be how stable everything feels. Applications are always available, transactions go through without a hitch, and downtime is virtually nonexistent.

Of course, in reality, servers do crash, even at the perfect provider's location, but thanks to sophisticated, multi-level resiliency strategies, a subscriber will rarely ever notice. Behind

the scenes, cloud platforms are built with automatic failover, redundancy, and rapid recovery systems that kick in the moment something goes wrong.

This means that if one server fails, the workload is seamlessly shifted to another one and if a configuration is corrupted, a known working version can be restored in seconds.

Commercial cloud infrastructure is designed not just for performance, but for graceful failure, so that your experience remains smooth, uninterrupted, and reliable. For the subscriber, this translates into what feels like a world of perfect uptime, without having to manage a single line of infrastructure definitions. No midnight calls, no scramble to reboot machines; just consistent service, delivered effortlessly.

## Software Publisher Perspective

From the perspective of a software publisher, it is much faster and easier to distribute functionality through web-based pages and sites than to compile and distribute desktop software. Here are some of the advantages of publishing cloud-based applications and functionality.

### Updates can be Published Instantly

When a new version of the software has been tested and is ready to roll out, it is simply taken live. In many cases, even in-session users will immediately notice whatever differences have been introduced.

### Updates can be Rolled Back Instantly

In the periodic case where a new feature didn't exactly work as planned, the system can often also instantly be rolled back to a previous version without causing any break in user workflow. Of course, this is dependent upon the specific differences between the two versions, and one not breaking the activity of another.

## Incremental Usage and Licensing

As you have probably noted, cloud services are nearly always subscription based. In many cases, the contractual agreement is monthly, and you can quit at any time. If you close your service, it is turned off and you no longer have access.

In a previous era, where nearly all software was locally run, licensing was a real hassle, not only for the users, but for software publishers as well. Several schemes were tried to assure that customers wouldn't make and distribute virtually infinite numbers of unauthorized copies. Some of those schemes worked well, and some didn't work out so well.

Because physical software had to be packaged, transported, and delivered, it had a physical cost associated with it. Additionally, because it was assumed you were only going to purchase one or two versions of that software at most, the entire price of research and development for that product had to be reclaimed from the total number of copies sold, which historically kept the single-license per-copy price of many brands unreasonably high (not that many of those same brands are any less expensive today).

## Disadvantages of Cloud Operations

As great as it sounds being a cloud subscriber isn't all sunshine and buttercups. The following sections help to outline many of the disadvantages you might notice either right away or soon after committing to a cloud system.

### The Internet is not Always Available

Virtually all cloud services at this time have only transient state, meaning that when your local cable modem or WiFi goes out for a few minutes or even longer, you aren't able to do any work at all. This consequentially means that when your customer calls to place a \$200,000 order for tennis shoes, the sales staff, at best, will have to write all of the details on a legal pad, hoping to remember all of the questions that need to be asked while taking the order, and at worst, will notify the customer that they won't be able to take the order at this time because "the system is down".

Regardless of whether or not that sale is going to be lost, an organization's system being down during working hours should not ever be an acceptable condition.

## Your Company's Trade Secrets are in the Hands of a Stranger

Even if a cloud provider promotes themselves as highly secure and trustworthy, the fundamental reality remains that you don't actually know who they are. You don't know the individuals who manage your data, their backgrounds, affiliations, or true motivations.

While companies often present a polished front with compliance certifications and reassuring messaging, the internal workings, or who has access to what, under what circumstances, and what their intentions might be, are beyond your visibility or control.

If your organization has developed proprietary algorithms, unique workflows, or even market-disrupting strategies, storing them in the cloud introduces an intangible but real risk. Once uploaded, those assets are no longer physically contained within your walls.

Even with encryption and contractual safeguards, you're placing critical intellectual property into systems that are managed by people you've never met and cannot verify. It only takes one disgruntled employee, one compromised insider, or one policy loophole for novel ideas to be unintentionally exposed, or even worse, deliberately hoisted.

The value of innovation often lies in its exclusivity. If you're pioneering a new approach or product, the moment those concepts are accessible, even indirectly, to external eyes, their competitive edge can begin to evaporate. In cloud environments, where multi-tenancy and shared infrastructure are standard, there's always a shadow of uncertainty.

Trust, in this case, is not just about whether a cloud provider keeps your data safe; it's about whether you can afford to risk someone else even knowing what you're working on.

## You Must Use What You Get

When you adopt a cloud service, you're automatically confined to a predefined set of tools, interfaces, and capabilities. These services are designed for broad capability, which means they're optimized for general use rather than your specific needs. If the platform lacks a feature your business requires or implements it in a way that doesn't align at all with your workflow, you are forced to adapt or find a workaround.

Unlike on-site solutions where you can modify, extend, or fully rebuild components to suit your exact specifications, cloud offerings tend to be rigid and non-negotiable.

Additionally, cloud ecosystems can be surprisingly insular. While they make it easy to build within their walls, they often purposely make it harder to extend beyond those same walls. You might find yourself locked into proprietary formats, limited APIs, or non-portable and inaccessible data structures. This creates a silo effect, where valuable outputs, like

models, insights, or automation logic, can't easily be reused across other platforms or shared horizontally within your own technology portfolio.

Ultimately, you are boxed in, using what is available rather than what is possible, which limits your potential innovation and agility.

## Your Success is Reliant on Another Party

One of the more subtle but deeply consequential risks of operating in the cloud is the transfer of control, not just over infrastructure or features, but over your ability to succeed at all.

When you commit to a cloud ecosystem, you're effectively stepping into someone else's domain, where the rules, the pathways, and the priorities are not yours to set. Whether it's infrastructure-as-a-service, platform tools, or distribution channels, your ability to grow, scale, or even survive often hinges on whether the provider allows you to do so.

This power imbalance is particularly stark in environments where the provider is also the gatekeeper of traffic, discoverability, or monetization.

Consider the case of Google. It is well-established that your website's visibility in search results, the cost of advertising, and even user access can all be influenced by how much you pay and how well you play within their evolving algorithms. Similarly, web hosting platforms may prioritize traffic to their premium-tier customers, creating an uneven playing field even among users of the same platform. If you're a YouTube creator or an Android app developer, your visibility, reach, and revenue can all rise or fall at any moment based on opaque criteria that are entirely out of your hands. The success you earn may not be the success you're allowed to keep, and you will never be able to explain why.

The core issue is that when your operations are dependent on a cloud provider's good favor, on how they define engagement, enforce policy, or allocate exposure, you are no longer operating in a neutral environment. You're subject to a shifting landscape where decisions are made behind closed doors, often based on internal incentives that have nothing to do with your mission. Flourishing in that space is possible, but when the gatekeepers leave the door open for you.

## Newer is not Always Better

In today's tech environment, including the elements of the cloud, the dominant narrative is that change is synonymous with progress. Cloud providers and software vendors



constantly push updates, rebrand features, and introduce entirely new paradigms - all under the banner of innovation. But beneath the glossy marketing and breathless announcements lies a difficult truth: change, in and of itself, is not inherently valuable.

If a new version of a tool brings more complexity, fewer customization options, or inconsistent behavior, then what's being offered isn't progress - it's disruption without direction.

Too often, the newest tools prioritize trendiness over trustworthiness. Flashy interfaces replace dependable ones. Integration gets sacrificed for automation. In chasing what's next, developers and users alike are increasingly forced to leave behind what actually worked. There are countless examples where legacy systems or older versions of software delivered faster performance, clearer workflows, or greater control than their modern counterparts. Yet in the cloud, backward compatibility is rarely a priority.

Once a provider obsoletes a feature or shifts its roadmap, you're left with no recourse, other than the consideration that you're always free to leave whenever you want to. Meanwhile, the older, often better, ways of doing things vanish into obsolescence, not because they failed, but because they didn't fit the new vision.

This pressure to constantly adapt can erode productivity and stability. Businesses find themselves spending time re-learning awkward interfaces, re-validating processes, or re-engineering integrations, all in response to forced upgrades that don't necessarily bring any new value at all. It's a costly cycle.

As alluded to above, change that doesn't lead to improvement is not innovation, it is noise. For a subscriber of a cloud system, who has no influence over product direction, resisting that noise is not an option.

## Do YOU Have a Backup?

Cloud providers often offer robust backup strategies as a selling point, noting their capabilities for redundancy across regions, multiple failover systems, and built-in disaster recovery. But there's a catch: those backups are for the vendor's safety from litigation, not necessarily for your safety as a customer.

While your data might be safe from accidental loss or hardware failure, actually extracting a complete, usable copy of it for your own purposes is a completely different story. In many cases, cloud platforms intentionally make it difficult or outright impossible for customers to create a full, independent backup of everything they've stored, built, or configured.

This imbalance serves the purpose of establishing lock-in. If you can't easily export your data, or if your backup is fragmented, proprietary, or missing critical metadata and configurations, then moving to another provider, or even just setting up a local contingency, becomes an infeasible task.

The provider gains leverage by ensuring that your continued use of their platform isn't just a choice, it's the only feasible option. Whether you're planning for disaster recovery, a vendor transition, or just safeguarding your intellectual property, the question isn't whether *they* have your data backed up, but whether *you* do.

## Product Lock-In

Speaking of lock-in, one of the more insidious challenges of cloud operations is product lock-in. This is a deliberate strategy where cloud providers slowly, but persistently, shape your workflows around their own exclusive tools, formats, and methodologies. Over time, you find yourself doing things in subtly different ways than industry standards suggest, not because it's better, but because that's how the platform requires it. Whether it's through proprietary APIs, unusual data formats, or custom configurations, your operations start to conform not to best practices, but to the whims of a specific provider's own ecosystem.

This divergence creates a long-term dependency that's hard to detect until it's too late. Once your data is stored in unique formats, and your processes are tied to exclusive tools, moving to another service isn't just inconvenient, it can be virtually impossible without manual re-entry of literally everything in your company's history because other vendors won't recognize the formats, can't replicate the components, or simply don't support the way things are done. The end result is a quiet but powerful form of captivity, since switching away would mean either losing critical information or transferring it manually across unfamiliar systems.

In this environment, you're no longer choosing what you think could be the service for the job, you're tolerating the only one that can read your data. That's not partnership; that's dependency, and it can severely limit your agility, bargaining power, and your company's own agility.

## You Don't Have Any Vote on New Features

When you subscribe to a cloud-based platform, it's easy to assume that your role as a paying customer gives you some level of influence over the direction of the product. In

reality, it doesn't. Feature development is typically driven not by the needs of existing users, but by the publisher's internal strategy to attract new customers.

Rather than building what would deepen value for those already invested in the platform, companies focus on shiny gimmicks, surface-level enhancements, or marketing-friendly tools aimed at expanding the user base, usually at the expense of long-term usability or refinement.

Even when you take the time to submit thoughtful, specific feature requests, for solutions that could meaningfully enhance your organization's use of the platform, those ideas are rarely heard in any meaningful way. The volume of incoming suggestions from the public is overwhelming, and within that noise, even smart, high-impact ideas can be indistinguishable from offhand comments or unrealistic demands.

Worse still, the initial triage of requests is often handled by support personnel or product managers who aren't fully immersed in the nuances of what the system is actually for. Instead, they follow well-documented procedures to categorize feedback and move on.

Even if your request does make it to the ears of a sympathetic developer, the decision to act on it rarely lies with them, since product direction is shaped at the higher levels of marketing, executive strategy, and competitive positioning, not in the engineering trenches where the usefulness of your idea might be obvious.

This disconnect leaves customers in a frustrating position where they know what they need, they know it would improve the system, but they have no real seat at the table. In that case, you're left waiting, watching new features roll out, often unrelated to your own needs, and realizing that your voice, however experienced or invested, doesn't carry any weight where it counts.

## You Don't Have Any Vote on System Policies

As a cloud subscriber, you're not just using someone else's infrastructure, you're operating within their rules. Every system policy, permission model, and architectural limitation is defined by the vendor, and you have little or no say in how those rules are applied.

In many cases, critical activities you need to perform are either restricted outright or made unnecessarily complex due to blanket security models, especially those built to ensure multi-tenant separation. While these constraints may serve the provider's broader operational goals, they often ignore the practical realities faced by your own organization.

One common friction point we experience is the lack of nuanced role management. Your team might include individuals who need to perform a mixture of administrative and operational tasks, but without being granted full admin rights. Unfortunately, many cloud platforms fail to support this kind of role granularity. Their access controls are too coarse, forcing you into an all-or-nothing approach that introduces internal risk or hinders productivity.

Similarly, you might find that two tools within the same platform, tools that seem like they should work together, simply don't. Whether it's an unsupported integration or a policy conflict that prevents a seemingly obvious workflow, the system is bound by limitations you can't change, and that the provider may not even acknowledge as a problem.

The core issue is that these policies are often shaped by internal priorities, not customer needs. Security, scalability, and cost-efficiency for the vendor frequently take precedence over flexibility or usability for the user. And while there may be forums, tickets, or feedback portals, the reality is that you have no real influence on how those policies evolve. You're bound by what has already been decided and those decisions are made by people who may never have seen your use case, and who also aren't required to accommodate it.

## Cloud Services can be Hacked

No matter how secure a cloud provider claims to be, the risk of compromise is always present, and when it happens, there's nothing you can do to stop it. Unlike managing your own network, where you can control responses, patching, access, and containment strategies, a cloud breach unfolds entirely outside your visibility and authority. If attackers gain access to the provider's infrastructure, your data, along with that of countless others', can be swept up in the damage.

The consequences of such a breach are far-reaching. Confidential information can be exposed, intellectual property and trade secrets can be stolen, and severe financial losses can quickly follow.

Perhaps even more damaging is the loss of trust. Once your stakeholders learn that sensitive data has been compromised, the reputational hit can take years to recover from.

While it's easy to become frustrated with the challenges of maintaining your own security, it's critical to remember that when your data is in the cloud, the ability to respond or defend against a cyberattack isn't yours to wield. The risk is real, and the fallout is completely beyond your control.

## Service Creep - Paying for What You Don't Need

In the pursuit of solving complex organizational challenges, IT teams often start by assembling a mix of cloud services, grabbing a little of this, a little of that, to quickly address immediate needs.

At first, this modular approach seems practical and agile. But over time, it often leads to a kind of *service creep*, where small, targeted additions evolve into sprawling, interdependent systems. Eventually, the solution to one problem becomes tied to a much larger, more expensive platform that does far more than what's actually needed.

Later, when it's time to rein in spending, these dependencies become painfully clear. That one dashboard tool is hooked into a data pipeline managed by another service, which feeds a reporting engine that relies on a compute instance elsewhere, and so on. What once seemed like a set of flexible options has quietly hardened into a costly web of interconnected services that can't be easily disentangled, resulting in a bloated cloud bill filled with components you no longer actively use but can't afford to turn off without disrupting critical functionality.

Attempts to reduce costs are met with technical debt, brittle integrations, and the realization that simplifying isn't simple anymore.

Service creep doesn't just waste money, it undermines your ability to make clean, strategic decisions about your technical portfolio. In cloud environments where everything is designed to scale *up* effortlessly, scaling *down* often turns out to be the far greater challenge.

## Advantages of On-Site Operation

Now that you might be reconsidering whether or not cloud services are worth it, I'd like to turn your attention to some of the truly amazing and awe-inspiring effects and by-products of managing your own on-site systems.

### A Non-Internet Ecosystem

A typical casual first act of setting up on-site operations is probably to configure some internal servers or workstations and a primary network. There are many cases where entire departmental networks have no connection to the Internet at all, making those particular

sections completely immune to outside attacks. Imagine a world where the existence of cybercrime doesn't even matter!

In this type of network, every type of functionality and every type of service must be provided by physical on-site devices. Rack-based, desktop, and laptop computers each have all of the software they need to perform every operation that would ever be required of them. It might seem too obvious, but if a server is hosting virtual operations for multiple desktop computers in the department using a client/server type of configuration, appropriate parts of that software must be running directly on the server at all times.

A department's first connection to the Internet has often been to provide internal intranet services, which connect two of the organization's departments or locations, but only on an encrypted level, and only for use with specific activities. This specific use, typically referred to as a Virtual Private Network (VPN), allows the organization to avoid purchasing dedicated high-speed lines from the telecommunications company, and to use cheap, fast, existing infrastructure that can be set up in a day. Upon reaching this level, there might be no need to increase the scope of operation, other than in size and productivity.

## 100% Uptime is Possible

Although we are talking about a considerable expense to achieve true 100% uptime, it is both possible and practical within an on-site network to achieve a system that nearly can't fail.

This state is achieved by having redundant-everything, all of it online at all times, coupled with a power system that has an instant fail-over to batteries, followed by only a brief delay to integrated, regulated clean generator power.

In a non-failure operation, inter-site communications are also doubled or even tripled with different transmission types, like landline trunk, fiberoptic, and even broadband cellular hubs specifically configured to act as business Internet connections. Let's also not forget sneakernet, where all data packets are delivered by hand.

## A Working Version can be Retained

Contrary to popular current sentiment, software doesn't ever age. If it was working well in 1992, it will probably work even better now, just due to the fact that PCs are roughly thirty times more powerful now than they were then. Until somewhat recently, the main problem

with older software wasn't so much that it didn't run faster on newer PCs as it was that a lot of older software was just plain terrible compared to newer advancements.

However, although there are immense improvements available, extreme care should be exercised because we now live in an age where the leader of one of the world's largest corporations would utter something like "move fast and break things" - a naive misquote of the engineering sentiment to "fail quickly" - and that younger developers everywhere would immediately begin to adopt the misquoted sentiment over the legitimate one.

Numerous products that were beginning to shine by the dawn of the 21<sup>st</sup> century have now been turned to trash, some of them only remaining barely usable. Anyone who really knew what Microsoft Outlook was actually about in the early 2000s, compared with the limited number of unreliable features it offers today, will immediately understand that the current era of software is dominated by constant transition, and it sometimes would be better right now to hold onto an older copy that was actually known to work - even if without some of the newer integrations for the time-being.

## Mix and Match is a Thing

When running software locally on physical hardware, you can often take the output of one application, then run it through another application to enhance it or convert it entirely to something else.

Often, a popular software application will play host to a number of other third-party services and applications that can use the primary application's data for creating workarounds to the software's own shortcomings, building detailed reports not available in the primary software, or outputting formats directly compatible either with the primary publisher's competition or a party operating in a completely different sector of industry.

Every organization is at least slightly unique so there is probably no overall suite that is ever able to take into account all of the different activities any organization needs to perform. The ability to use one element from one vendor, another element from a second vendor, and to even create the third element yourself to bridge the other two are infinitely more powerful than simply taking what an online vendor would give you as a supposed "solution".

## All of your Important Files are Inside the Firewall

After you decide that some Internet access is not only warranted but needed for research, business to business activities, business to customer activities, and general real-time

communication of all sorts, a strictly internal network can quickly be connected to the Internet with firewall/router.

Just as with one of the most important traits of Cloud-based data backup, you can and should implement a multiple hard-drive system on your active file server and synchronize that data in real-time over multiple locations in a process known as a Distributed File System (DFS). To be even more protected, you could consider implementing a Version Control System (VCS) for at least some of the file sets in your structure, which allows you to retrieve any previous version of a file alternatively to its current version.

When your files are synchronizing between your physical locations, their data packets might physically be transported across the physical lines constituting the Internet, but since they are well-encrypted by your VPN configuration, making the data impervious to external inspection, and since they are routed directly from one site to the other in as few steps as possible for the communications provider, there is very little chance anyone outside of your organization would be able to read any of your data.

## You Have Your Backup

One of the clearest advantages of managing your operations on-site is the freedom to back up your data your way. You're not limited to a single backup strategy dictated by a third-party provider. Instead, you can create multiple layers of protection that include live copies on active servers, cold storage for long-term archives, encrypted portable backups on external devices, and synchronized snapshots across multiple geographical locations.

With the right setup, your data can exist in any number of formats, relational, flat file, object-based, compressed, or raw, or whatever best suits your technical and compliance needs.

This level of control empowers you to build a true business continuity plan (BCP) that rivals, or even exceeds, the safeguards offered by commercial cloud providers. But beyond disaster recovery, owning your backup process means you also gain the ability to revert your entire system to any previous operational state in history.

Whether you need to audit a prior configuration, investigate a historic data point, or completely roll back after a failed deployment, you're not just recovering files, you're recovering time, intent, and context. That's a depth of operational agility most third-party platforms simply don't and can't provide.

Maybe best of all, your backups exist entirely on your terms. You choose where they're stored, how they're protected, how long they're retained, and how quickly they can be



restored. This autonomy not only improves organizational resilience but puts your organization in control of its own timeline, whether past, present, or future.

## You Have Freedom to Change

One of the most powerful advantages of on-site operation is the freedom it gives you to shape your infrastructure around your organization's evolving needs. When you're running systems in-house, you're not locked into someone else's roadmap or constrained by a cloud provider's assumptions. Instead, you and your team get to decide exactly which components, tools, and configurations will deliver the most value, and you can adapt those choices at any time, on your own terms.

Whether it's a new strategic goal, a shift in process, or a technological advancement, on-site systems give you the flexibility to pivot without any external friction. If your committee determines that a different approach would better serve the mission, whether that means replacing a database engine, introducing new middleware, modifying network architecture, or retiring outdated modules, you can take action immediately. Internal systems can be rearranged, expanded, or simplified to support new workflows and priorities without needing approval, workarounds to integrations, or feature requests to a third party.

With on-site operation, your IT infrastructure becomes a living part of your organization, able to grow, adapt, and evolve as needed. You're not bound by another company's vision of what your system will be limited to. Instead, you're free to build exactly what it needs to be to properly serve your goals.

## Your System is Capable of Nearly Anything

When you operate your systems on-site, you're stepping into a world of possibility. Whether you purchase software from major enterprise publishers or leverage powerful, well-supported open-source tools, you'll often find that your core needs can be met right out of the box.

But the real strength of on-site operation lies in how extendable and adaptable it becomes. Third-party developers and vendors actively build around popular platforms, offering add-ons, extensions, and plug-ins that address known limitations or serve niche industries the main providers don't explicitly target.

This thriving ecosystem of support transforms your infrastructure into something greater than the sum of its parts. Instead of waiting for a cloud vendor to release a long-promised

feature, or worse, deciding that feature will never arrive, you can go out and get exactly what you need, often today.

Your freedom truly shines in finding that what you need doesn't exist yet. With access to skilled developers familiar with your chosen stack or platform, you can create virtually anything, from lightweight utility scripts that bridge gaps between applications to full-scale, bespoke systems built from the ground up to match your vision.

In an on-site environment, you're not limited by someone else's roadmap, you're empowered by your own. With the right tools and the right team, there are no artificial ceilings. You can scale, reshape, integrate, and innovate without waiting for outside permission. Your system isn't just serving your goals, it can evolve with them.

## No Limit on Services Used

One of the most liberating aspects of running your own on-site infrastructure is complete freedom from artificial service restrictions.

In most cloud platforms, unless you're paying for a highest-tier or enterprise-level subscription, you're subject to tightly defined quotas that define how many services you can use, how often you can call them, how many requests you can make per second, per minute, or per month. These caps aren't based on your needs, they're based on the provider's pricing model.

With on-site operation, those ceilings vanish. Once you've deployed your chosen software on your own hardware, the only real limits are the ones defined by the capabilities of your machines, and even those can be expanded as needed or as budget allows. You're free to run one service, ten services, or your entire suite of tools simultaneously. Whether a particular function is triggered once per week or millions of times per second, you don't need to worry about cost spikes, throttling penalties, or delayed execution due to hidden metering.

This freedom to operate without any restraint means you can scale your internal processes in whatever direction best suits your mission, whether vertically, horizontally, reactively, or proactively, without having to request more credits, raise limits, or justify your growth to an outside provider.

In an on-site environment, your infrastructure works at the pace of your ambition, not at the pace of your subscription tier.

## No Limit on Services Published

One of the often-overlooked strengths of maintaining your own on-site infrastructure is the complete freedom to become a provider, not just a consumer, of digital services.

When you control the system, you're not just running tools for internal use, you're free to build and publish full-featured electronic services for your own customers, partners, or communities. Whether it comes in the form of APIs, data feeds, portals, or complex applications, you decide what gets delivered, how it's delivered, and to whom.

Crucially, this freedom comes with no external royalties, licensing hurdles, or per-customer pricing models imposed by a third-party vendor. You're not sharing profits or constrained by tiers, usage caps, or distribution limits. You can serve one customer or one million, offer a handful of features, or a full suite of interconnected services. The only constraints are the physical capabilities of your infrastructure, and those are entirely within your control to expand as demand grows.

This level of independence opens the door to creative service models, niche markets, and rapid experimentation. You can innovate without having to ask permission. Your roadmap is yours alone. And in a world increasingly dominated by platforms that place tolls at every gateway, owning your infrastructure means owning your future and delivering limitless value to the people who rely on you.

## Drastically Less Expensive than in the Past

One of the most compelling reasons to revisit on-site operation today is how affordable it has become. The cost of hardware has plummeted compared to just a few decades ago. What once required entire rooms of specialized equipment in the 1980s can now be handled by compact, energy-efficient machines costing a fraction of the price. Modern PCs and servers are not only vastly more powerful than their historical counterparts, but also significantly cheaper, quieter, and easier to maintain.

Couple this with the incredible momentum of the open-source movement, and the economics of self-hosted systems become even more attractive. Powerful, community-supported software stacks are now available for everything from databases and operating systems to automation, security, and analytics. These free or low-cost options continue to place downward pressure on commercial software pricing, giving organizations access to professional-grade capabilities without the recurring fees typically associated with premium, cloud-based platforms.

The result is that you can now build and maintain an on-site system capable of supporting hundreds of users for a small fraction of what similarly featured cloud services would cost, both today and compared to what they would have taken just 10 or 20 years ago.

What was once seen as a luxury reserved for enterprise giants is now within reach for nonprofits, small businesses, educational institutions, and agile teams with vision. With smart planning and modest investment, self-hosted infrastructure is not just feasible, it's surprisingly economical.

## Disadvantages of On-Site Operation

### IT Staff are Expensive

While salaries within IT vary, the overall position of an IT professional is generally towards the higher end of the salary spectrum within the white-collar workforce, especially for those with specialized skills and experience.

Depending upon the amount of equipment you operate, the daily challenges of handling data, the size of your network, and number of users needing on-site support, you might require anywhere from 1 to 10 expert-level professionals to keep everything running perfectly at all times.

### Maintenance and Troubleshooting can be Complicated

Changing things can also break things. As a result, I have seen a lot of organizations who, once they had something running, never touched it again. This might be why it still isn't that unusual to see a tape drive doing a backup somewhere.

The complicated nature of growing and maintaining a system can also lead to a less active adoption schedule of tools and techniques that could otherwise help the organization to improve its productivity or effectiveness.

Whoever said that IT was a subject that required constant study wasn't kidding, and it's nearly definite that one person isn't able to contain all of the details making up the field. However, as Albert Einstein responded when asked about a specific detail he didn't remember "[I do not] carry such information in my mind since it is readily available in books."

## Security Configuration and Maintenance

Even if you have no connection at all to the Internet, you will still need to protect certain files from prying eyes. People in various roles require not only different levels of permissions, but often, different kinds of permissions altogether.

Although the security on a clean system doesn't necessarily take a lot of time to maintain, a system without skilled administrators often gradually works its way toward a condition where everyone has permission to access everything to avoid frustrations that have periodically arisen. In a case like that, your entire system can become vulnerable.

## Licensing is a Nuisance

Relegated to being one of the necessary evils of running your own on-site computer system, there is always the constant reminder of what pieces have been licensed for which users, and under which time frames for the agreed-upon budgets?

There are databases and applications that can be used to manage this information if you decide not to put it in a spreadsheet, some of which have licensing requirements themselves.

## Comparing the Real Costs

When it comes to operational infrastructure, cost is always a key consideration, but the true expense of cloud versus on-site solutions isn't just about monthly invoices or sticker prices.

In a small organization, you might be surprised by how far a modest budget can go. With today's affordable yet powerful desktop PCs and a wide range of robust, free, open-source software, it's entirely possible to build a full-fledged, secure client/server network capable of supporting engineering, marketing, sales, operations, and even public-facing services for a tiny fraction of what comparable cloud platforms might charge. These systems can offer flexibility, customization, and control that's impossible to match in a prepackaged cloud offering.

As organizations grow, however, both approaches become more expensive; just in different ways. On-site operation requires investment in skilled IT staff, local infrastructure, and ongoing maintenance. However, funds spent on internal systems often supports your own local economy, creates jobs internally, and empowers your team with deep knowledge of your own systems.

Cloud services, by contrast, externalize much of that responsibility. With a cloud subscription, you are paying for convenience, scale, and hands-off operation, while also sending a consistent stream of money out of your organization, typically to providers headquartered in different cities or even countries.

In the end, the real financial question isn't just about what it costs, but who you would rather invest in. Both models carry tangible and intangible value. What makes the most sense depends on your goals, your resources, as well as where you want the control and the responsibility to live.

## Summary

At last, there's no one-size-fits-all answer in the debate between cloud and on-site operations, only a spectrum of choices based on the unique priorities of your organization.

Cloud services offer incredible convenience, ubiquitous access, and reliable stability with minimal internal infrastructure. For organizations focused on fast deployment, geographic flexibility, and always-on services, the cloud often shines.

Those same strengths can become weaknesses for businesses that value control, customization, and confidentiality though. On-site systems, once seen as outdated or overly complex, have evolved into highly efficient, cost-effective, and incredibly adaptable platforms. They offer the freedom to architect your environment exactly the way you need it and to retain full ownership of your data, your strategy, and your future.

Every advantage comes with a tradeoff, and every risk presents an opportunity. The goal isn't to blindly follow trends, but to understand what's truly best for your people, your processes, and your long-term vision. Whether you lean into the cloud, stay grounded on-site, or adopt a hybrid of both, the most important decision is the one that's made with all of the available information.