

# *Thin client*

In [computer networking](#), a **thin client** is a simple (low-performance) [computer](#) that has been [optimized](#) for [establishing a remote connection](#) with a [server](#)-based computing environment. The server does most of the work, which can include launching [software](#) programs, performing [calculations](#), and [storing data](#). This contrasts with a [rich client](#) or a conventional [personal computer](#); the former is also intended for working in a [client–server model](#) but has significant local processing power, while the latter aims to perform its function mostly locally. <sup>[1]</sup>



*Thin clients connected to their server via a [computer network](#)*



A public thin-client computer terminal inside a [public library](#)



An [Aleutia E3](#) thin client, with flash memory

Thin clients occur as components of a broader computing infrastructure, where many clients share their computations with a server or [server farm](#). The server-side infrastructure uses [cloud computing](#) software such as [application virtualization](#), hosted shared desktop (HSD) or [desktop virtualization](#) (VDI). This combination forms what is known as a cloud-based system, where desktop resources are centralized at one or more [data centers](#). The benefits of centralization are hardware resource optimization, reduced [software maintenance](#), and improved [security](#).

- Example of hardware resource optimization: [Cabling](#), [bussing](#) and [I/O](#) can be minimized while idle memory and processing power can be applied to user sessions that most need it.
- Example of reduced software maintenance: [Software patching](#) and [operating system](#) (OS) migrations can be applied, tested and activated for all users in one instance to accelerate roll-out and improve administrative efficiency.

- Example of improved security: Software assets are centralized and easily [fire-walled](#), monitored and protected. Sensitive data is uncompromised in cases of desktop loss or theft.

Thin client hardware generally supports common [peripherals](#), such as keyboards, mice, [monitors](#), [jacks](#) for sound peripherals, and open [ports](#) for [USB](#) devices (e.g., printer, flash drive, webcam). Some thin clients include ([legacy](#)) [serial](#) or [parallel ports](#) to support older devices, such as receipt printers, scales or time clocks. Thin client software typically consists of a [graphical user interface](#) (GUI), cloud access agents (e.g., [RDP](#), [ICA](#), [PCoIP](#)), a local [web browser](#), [terminal emulators](#) (in some cases), and a basic set of local [utilities](#).

## Characteristics

---

### Architecture

In using cloud-based architecture, the server takes on the processing load of several client sessions, acting as a host for each endpoint device. The client software is narrowly purposed and lightweight; therefore, only the host server or server farm needs to be secured, rather than securing software installed on every endpoint device (although thin clients may still require basic security and strong authentication to prevent unauthorized access). One of the combined benefits of using cloud architecture with thin client desktops is that critical IT assets are centralized for better utilization of resources. Unused memory, bussing lanes, and processor cores within an individual user session, for example, can be leveraged for other active user sessions.

The simplicity of thin client hardware and software results in a very low [total cost of ownership](#), but some of these initial savings can be offset by the need for a more robust cloud infrastructure required on the server side.

An alternative to traditional server deployment which spreads out infrastructure costs over time is a cloud-based subscription model known as [desktop as a service](#), which allows IT organizations to outsource the cloud infrastructure to a third party.

### Simplicity



TA7 thin client by [Gigabyte](#)

Thin client computing is known to simplify the desktop endpoints by reducing the client-side software footprint. With a lightweight, read-only [operating system](#) (OS), client-side setup and administration is greatly reduced. Cloud access is the primary role of a thin client which eliminates the need for a large suite of local user applications, data storage, and utilities. This architecture shifts most of the software execution burden from the endpoint to the data center. User assets are centralized for greater visibility. Data recovery and desktop repurposing tasks are also centralized for faster service and greater scalability.

## Hardware

While the server must be robust enough to handle several client sessions at once, thin client hardware requirements are minimal compared to that of a traditional PC laptop or desktop. Most thin clients have low-energy processors, [flash storage](#), memory, and no moving parts. This reduces the cost and power consumption, making them affordable to own and easy to replace or deploy. Numerous thin clients also use Raspberry Pis.<sup>[2]</sup> Since thin clients consist of fewer hardware components than a traditional desktop PC, they can operate in more hostile environments. And because they typically don't store critical data locally, risk of theft is minimized because there is little or no user data to be compromised.

## Graphics

Modern thin clients have come a long way to meet the demands of today's graphical computing needs. New generations of low energy chipset and CPU ([Central Processing Unit](#)) combinations improve processing power and graphical capabilities. To minimize latency of high resolution video sent across the network, some host software stacks leverage multimedia redirection (MMR) techniques to offload video rendering to the desktop device. Video codecs are often embedded on the thin client to support these various multimedia formats. Other host software stacks makes use of [User Datagram Protocol](#) (UDP) in order to accelerate fast changing pixel updates required by modern video content. Thin clients typically support local software agents capable of accepting and decoding UDP.

Some of the more graphically intense use cases remain a challenge for thin clients. These use cases might include applications like photo editors, 3D drawing programs, and animation tools. This can be addressed at the host server using dedicated [GPU](#) cards, allocation of [vGPUs](#) (virtual GPU), workstation cards, and [hardware acceleration](#) cards. These solutions allow IT administrators to provide power-user performance where it is needed to a relatively generic endpoint device such as a thin client.

## Limitations

To achieve such simplicity, thin clients sometimes lag behind desktop PCs in terms of extensibility. For example, if a local software utility or set of device drivers are needed in order to support a locally attached peripheral device (e.g. printer, scanner, [biometric security device](#)), the thin client operating system may lack the resources needed to fully integrate the required dependencies (although dependencies can sometimes be added if they can be identified). Modern thin clients address this limitation via port mapping or USB redirection software. However, these methods cannot address all scenarios. Therefore, it is good practice to perform validation tests of locally attached peripherals in advance to ensure compatibility. Further, in large distributed desktop environments, printers are often networked, negating the need for device drivers on every desktop.

While running local productivity applications goes beyond the normal scope of a thin client, it is sometimes needed in rare use cases. License restrictions that apply to thin clients can sometimes prevent them from supporting these applications. Local storage constraints may also limit the space required to install large applications or application suites.

It is also important to acknowledge that network bandwidth and performance is more critical in any type of cloud-based computing model. IT organizations must ensure that their network can accommodate the number of users that they need to serve. If demand for bandwidth exceeds network limits, it could result in a major loss of end user productivity.

A similar risk exists inside the data center. Servers must be sized correctly in order to deliver adequate performance to end users. In a cloud-based computing model, the servers can also represent a single point of failure risk. If a server fails, end users lose access to all of the resources supported by that server. This risk can be mitigated by building redundancies, fail-over processes, backups, and load balancing utilities into the system. Redundancy provides reliable host availability but it can add cost to smaller user populations that lack scale.

## Providers

Popular providers of thin clients include [Chip PC Technologies](https://www.chippc.com/), [Dell](#) (acquired [Wyse Technology](#) in 2012), [HP](#), [ClearCube](#), [IGEL Technology](#), [LG](#), [NComputing](#), [Stratodesk](https://www.stratodesk.com/) , [Samsung Electronics](#), and [ZeeTim](https://www.zetim.com/) .

## History

---



A connected [Samsung Chromebox](#) as seen from above

Thin clients have their roots in [multi-user systems](#), traditionally [mainframes](#) accessed by some sort of [computer terminal](#). As computer graphics matured, these terminals transitioned from providing a [command-line interface](#) to a full [graphical user interface](#), as is common on modern advanced thin clients. The prototypical multi-user environment along these lines, [Unix](#), began to support fully graphical [X terminals](#), i.e., devices running [display server](#) software, from about 1984. X terminals remained relatively popular even after the arrival of other thin clients in the mid-late 1990s. Modern Unix derivatives like [BSD](#) and [Linux](#) continue the tradition of the multi-user, remote display/input session. Typically, X software is not made available on non-X-based thin clients, although no technical reason for this exclusion would prevent it.

[Windows NT](#) became capable of multi-user operations primarily through the efforts of [Citrix Systems](#), which repackaged [Windows NT 3.51](#) as the multi-user operating system [WinFrame](#) in 1995, launched in coordination with Wyse Technology's Winterm thin client. [Microsoft](#) licensed this technology back from Citrix and implemented it into [Windows NT 4.0 Terminal Server Edition](#), under a project codenamed 'Hydra'. Windows NT then became the basis of Windows 2000 and Windows XP. As of 2011 Microsoft Windows systems support graphical terminals via the [Remote Desktop Services](#) component. The Wyse Winterm was the first Windows-display-focused thin client (AKA Windows Terminal) to access this environment.

The term *thin client* was coined in 1993<sup>[3]</sup> by Tim Negris, VP of Server Marketing at [Oracle Corporation](#), while working with company founder [Larry Ellison](#) on the launch of [Oracle 7](#). At the time, Oracle wished to differentiate their server-oriented software from Microsoft's desktop-oriented products. Ellison subsequently popularized Negris' [buzzword](#) with frequent use in his speeches and interviews about Oracle products. Ellison would go on to be a founding board member of thin client maker Network Computer, Inc (NCI), later renamed Liberate.<sup>[4]</sup>



*Size comparison – traditional desktop PC vs Clientron U700*

The term stuck for several reasons. The earlier term 'graphical terminal' had been chosen to distinguish such terminals from text-based terminals, and thus put the emphasis heavily on *graphics* – which became obsolete as a distinguishing characteristic in the 1990s as text-only physical terminals themselves became obsolete, and text-only computer systems (a few of which existed in the 1980s) were no longer manufactured. The term 'thin client' also conveys better what was then viewed as the fundamental difference: thin clients can be designed with less expensive hardware, because they have reduced computational workloads.

By the 2010s, thin clients were not the only desktop devices for general purpose computing that were 'thin' – in the sense of having a small form factor and being relatively inexpensive. The [nettop](#) form factor for desktop PCs was introduced, and nettops could run full feature Windows or Linux; [tablets](#), [tablet-laptop hybrids](#) had also entered the market. However, while there was now little size difference, thin clients retained some key advantages over these competitors, such as not needing a local drive. However, 'thin client' can be a [misnomer](#) for slim form-factor computers using [flash memory](#) such as [compactflash](#), [SD card](#), or permanent flash memory as a [hard disk](#) substitute. In 2013, a Citrix employee experimented with a [Raspberry Pi](#) as a thin client.<sup>[5]</sup> <sup>[6]</sup> Since then, several manufacturers have introduced their version of Raspberry Pi thin clients.<sup>[2]</sup>

## See also

---

### Other client types



- **Dumb terminal**: Like thin clients, but have zero local processing power and support no peripherals
- **Rich client**: Have ample local processing power, although they are heavily network-dependent
- **Diskless node**: It has no local storage (e.g. no hard disk drives) but may have anything else that a full workstation has

## Related concepts

- [Centralized computing](#)
- [Desktop virtualization](#)
- [Multiseat configuration](#)
- [Time-sharing](#)

## Others

- [AOL TV](#)
- [Blade PC](#)
- [Sun Ray](#)

## References

---

1. Bengfort; Jacquelyn. "Thin vs. Thick vs. Zero Client: What's the Right Fit for Your Business?" (<https://biztechmagazine.com/article/2018/10/thin-vs-thick-vs-zero-client-whats-right-fit-your-business-perfcon>) . Technology Solutions That Drive Business. Retrieved 2021-10-10.
2. Brown, Eric. "Thin Client Market Embraces Raspberry Pi" (<https://www.linux.com/news/thin-client-market-embraces-raspberry-pi/>) . Linux.com. Retrieved 25 July 2022.
3. (thin client term was coined 1993) Richard Waters (June 2, 2009). "Is this, finally, the thin" (<https://www.ft.com/content/dc70f841-54b7-3ef1-abf0-d6f32b270f76>) .
4. "Liberate Technologies: Taking Strange to New Levels" (<https://nerdtwilight.wordpress.com/2009/09/17/iberate-technologies-taking-strange-to-new-levels/>) . 17 September 2009.
5. Dawood, Muhammad. "Citrix Receiver on Raspberry Pi delivering 1080p XenDesktop 7.1 experience" (<https://www.youtube.com/watch?v=8ns5fJuQUH4>) . Youtube.com.
6. Dawood, Muhammad. "How good is the new Raspberry Pi 4 as a thin client?" (<https://www.citrix.com/blogs/2019/07/08/how-good-is-the-new-raspberry-pi-4-as-a-thin-client/>) . Citrix.com.



Wikimedia Commons has media related to *Thin clients*.

Retrieved from

["https://en.wikipedia.org/w/index.php?title=Thin\\_client&oldid=1100543288"](https://en.wikipedia.org/w/index.php?title=Thin_client&oldid=1100543288)

---

Last edited 4 days ago by Fairwin99

WIKIPEDIA

---