

Thin client

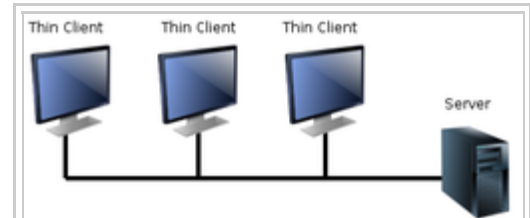
From Wikipedia, the free encyclopedia

A **thin client** is a lightweight computer that is purpose-built for remoting into a server (typically cloud or desktop virtualization environments). It depends heavily on another computer (its *server*) to fulfill its computational roles. This is different from a conventional desktop PC (fat client), which is a computer designed to take on these roles by itself. The specific roles assumed by the server may vary, from hosting a shared set of virtualized applications, a shared desktop stack or virtual desktop, to data processing and file storage on the client's or user's behalf.

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 makes use of cloud computing software such as application virtualization, hosted shared desktop (HSD) or desktop virtualization (VDI). This combination forms what is known today as a cloud based system where desktop resources are centralized into 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 users sessions that most need it.
- Example of reduced software maintenance: Software patching, security updates, application/OS updates, and 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 consists of a computer terminal which provides



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

I/O for a keyboard, mouse, monitor, jacks for sound peripherals, and open ports for USB devices, e.g. printer, flash drive, web cam, card reader, smartphone, etc. Some thin clients include legacy serial and/or parallel ports to support older devices such as receipt printers, scales, time clocks, etc. Thin client software typically consists of a GUI (graphical user interface), cloud access agents (e.g. RDP, ICA, PCoIP), a local web browser, terminal emulations (in some cases), and a basic set of local utilities.

Contents

- 1 Characteristics
 - 1.1 Architecture
 - 1.2 Simplicity
 - 1.3 Hardware
 - 1.4 Graphics
 - 1.5 Limitations
- 2 History
- 3 Thin client variants
 - 3.1 Zero client
 - 3.2 Web client
 - 3.3 Repurposing a PC as a thin client
- 4 List of protocols used by thin clients
- 5 See also
- 6 References

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 trade-offs of using cloud architecture and thin clients is that the host becomes a single point of failure. If the hosting server crashes, for example, all users lose their session unless a fail-over process is predefined. This risk can be mitigated by building redundancies into the host environment. Redundancy provides reliable host availability but adds cost to the cloud computing model. Redundancy can be provided in the form of networked servers, clustered file systems, load balancing, etc.

The simplicity of thin client hardware and software results in a very low TCO (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. *NOTE:*

infrastructure costs can be spread out by choosing a subscription based cloud model known as Desktop as a Service or DaaS. DaaS allows IT organizations to outsource the cloud infrastructure to a third party.

Simplicity

Thin client computing is known to simplify the desktop endpoints by reducing the client-side software footprint. With a lightweight, read-only OS (operating system), 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 software execution burden from the endpoint to the data center. User assets are centralized for greater visibility and control. Data recovery and desktop repurposing tasks are also centralized for faster service and greater scalability.



Gigabyte
TA7 thin
client

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 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. 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 UDP (User Datagram Protocol) 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, however, remain a challenge for thin clients. These use cases might include the 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 use case 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.

History

Thin clients have their roots in multi-user systems, traditionally mainframes accessed by some sort of terminal computer. 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 multiuser 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 NT 3.5.1 as the multi-user operating system WinFrame in 1995. 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 term *thin client* was coined in 1993 by Tim Negriz, VP of Server Marketing at Oracle Corp., 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 Negriz' 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.^[1]

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, however, 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 and 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.



A connected Samsung Chromebox as seen from above.



Size comparison - traditional desktop PC vs Clientron U700

Thin client variants

Zero client

While a traditional thin client is streamlined for multi-protocol client-server communication, it still contains a full operating system. A zero client makes use of very lightweight firmware that merely initializes network communication through a basic GUI (Graphical User Interface), decodes display information received from the server, and sends local input back to the host. A device with such simple functionality has less demand for complex hardware or silicon, and therefore becomes less prone to obsolescence. Another key benefit of the zero client model is that its lightweight firmware represents an ultra-small attack surface making it more secure than a thin client. Further, the local firmware is so simple that it requires very little setup or ongoing administration. It's the ultimate in desktop simplification but the trade-off is flexibility. Most mainstream zero clients are optimized for one communication protocol only. This limits the number of host environments that a zero client can provide its users with access to. ^[2]



Web client

Web clients only provide a web browser, and rely on web applications to provide general-purpose computing functionality. However, note that web applications may use web storage to store some data locally, e.g. for "offline mode", and they can perform significant processing tasks as well. Rich Internet Applications for instance may cross the boundary, and HTML5 Web Applications can leverage browsers as run-time environments through the use of a cache manifest or so called "packaged apps" (in Firefox OS and Chrome).

Examples of web thin clients include Chromebooks and Chromeboxes (which run Chrome OS) and phones running Firefox OS.

Chromebooks and Chromeboxes also have the capability of remote desktop using the free Chrome Remote Desktop browser extension, which means, other than being a web thin client, they can also be used as an ultra-thin client (see above) to access PC or Mac applications that do not run on the Chromebook directly. Indeed, they can be used as a web thin client and an ultra-thin-client simultaneously, with the user switching between web browser and PC or Mac application windows with a click.

Chromebooks are also able to store user documents locally – though, with the exception of media files (which have a dedicated player application to play them), all such files can only be opened and processed with web applications, since traditional desktop applications cannot be installed in Chrome OS.

Repurposing a PC as a thin client

The following tools/solutions can allow a PC to be used as a thin client – in some cases, even if it has no working hard drive:

- Software Thin Client
- Linux Terminal Server Project
- Thinstation
- WTware^[3]
- OpenThinClient
- Puppy Linux
- AnywhereTS
- Remote desktop software
- V2 Cloud Client^[4]
- IGEL Technology UDC2 or Universal Desktop Converter 2
- VDI Blaster^[5]

List of protocols used by thin clients

- Appliance Link Protocol
- Citrix ICA or Independent Computing Architecture
- Citrix HDX
- HP RGS or Remote Graphics Software
- LTSP or Linux Terminal Server Project
- NFS
- PCoIP or PC over IP
- various video codecs
- Microsoft RDP or Remote Desktop Protocol
- Microsoft RemoteFX or RFX
- Secure Shell or SSH, an encrypted replacement for telnet.
- Virtual Network Computing
- NComputing UXP (User eXperience Protocol)
- X11, central to traditional Unix graphical windowing systems
- XML, HTML, or JSON over HTTP (Ajax)
- XDMCP
- NoMachine NX Software (NX)
- SPICE (protocol)

See also

- Blade PC
- Centralized computing
- Desktop Virtualization
- Dumb terminal
- Fat client
- Hybrid client
- Multiseat configuration
- Nettop
- Smart client
- Terminal services
- Time-sharing

- Univention Corporate Client (UCC), an Open Source solution for the central management of PCs, notebooks and thin clients
- Useful
- Windows PE
- AOL TV
- NComputing
- X11, central to Unix windowing

References

1. <https://nerdtwilight.wordpress.com/2009/09/17/liberate-technologies-taking-strange-to-new-levels/>
2. →§
3. "WTware" (<http://www.wtware.com/>). *http://www.wtware.com/*. External link in |work= (help)
4. <https://v2cloud.com>
5. <http://www.devonit.com/software/vdi-blaster>

Retrieved from "https://en.wikipedia.org/w/index.php?title=Thin_client&oldid=780475342"



Wikimedia Commons
has media related to
Thin clients.

Categories: Thin clients

-
- This page was last edited on 15 May 2017, at 10:34.
 - Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.