

# System Architecture



Thinknx has launched Thinknx UP, a completely redesigned version of the Classic Thinknx software. To learn more about the migration, click [here](#).

- [Introduction](#)
- [Thinknx Solution Components](#)
  - [Thinknx Servers](#)
  - [Thinknx Clients](#)
  - [Thinknx Cloud](#)
  - [Thinknx Configurator](#)
- 3. [Communication Between Components](#)

## Introduction

ThinKnx is a supervision system for home and building automation based on client-server architecture.

- The **server** consists of one ThinKnx device (Compact, Micro, Rack or Touch) that represents the logical unit of the system and controls all the equipment connected to it.
- The **client** consists of any PC, touch screen, tablet or smartphone connected to the LAN via Ethernet or WiFi: all these devices, through the ThinKnx software, communicate with the server to control the plant.
- The **Thinknx cloud** is also an important part of the system architecture and is used for multiple purposes that serve both the integrator and the end user.
- The **Configurator** is provided as the indispensable tool to build complete Thinknx solutions. It is a software used by the integrator to create attractive and customizable graphical interfaces and deploy them on the servers and clients to control the different systems installed in the plant.

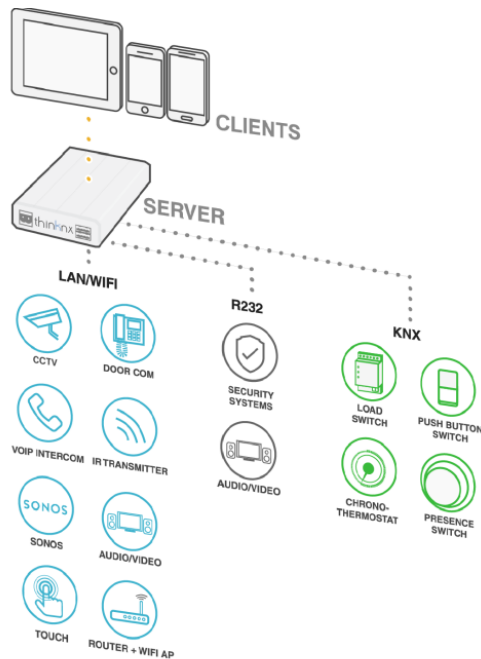
Figure 1: Scheme of a typical integrated installation

The picture to the left shows a standard installation of ThinKnx system in a KNX or Modbus plant.

The ThinKnx server interfaces with KNX plant through its dedicated **bus connector**. It is also possible to interface with Modbus or other buses through **RS232** or **TCP**.

Additional systems can be integrated with server through RS232 or TCP. For example, anti-theft systems, audio or video matrix, home theater systems, etc.

The ThinKnx server controls all the devices connected to it and, working as a gateway,



allows interactions between the connected devices even if they are using different technologies. The clients, on the other hand, send their commands to the server which forwards them to the remaining devices allowing the communication between both parties.

The Gateway feature inside the server allows to control and integrate different devices and systems:

- *For example, Sonos Player or IR transmitter can be controlled from KNX keypads by simply configuring server logic.*
- *Anti-theft system can also send telegrams on KNX or launch scenes on particular events.*
- *When an intercom door station makes a call to a Thinknx client, the server can perform actions on other devices or KNX.*

With a powerful tool such as the Thinknx server, the possibilities are endless.

## ThinKnx Solution Components

### ThinKnx Servers

During normal runtime, ThinKnx server is the logical unit of the supervision system and communicates directly with the ThinKnx clients.

The ThinKnx servers range was born to cover all the possible installation cases of a supervision system. All the devices are able to interface directly with the KNX plant and have several communication ports to connect to the Ethernet and to other buses (like Modbus etc.) or to other devices like anti-theft systems (through RS232 or TCP/IP). They are also accurately designed for 24/7 operation and optimized for high reliability. In respect to the basic functions - that are included on every server model - additional facility can be activated by means of licenses.

- **ThinKnx Compact** It is the most versatile device of all, and supports all kind of functions (Security, VoIP, Audio/Video). IT is the ideal device for the majority of plants.
  - Fanless processing device optimized for continuous operation
  - Mass memory on industrial Compact-Flash
  - Power: 12-18 VDC - 1A Max
  - 1x EIB/KNX port with direct connection through clips
  - 1x network port
  - 1x standard RS232 serial port with DE9P connector

- 2x USB port
- KNX telegrams led

*For more info about ThinKnx Compact server please check out the related [page](#) on our website.*

- **ThinKnx Micro** It is a very small device with the ability to completely control KNX systems. Can be enhanced with VoIP function for intercom communication.
  - Fanless processing device optimized for continuous operation
  - Mass memory on MicroSD
  - Power: 12-24 VDC - 1A Max
  - 1x EIB/KNX port with direct connection through clips
  - 1x network port
  - KNX telegrams led
  - Consumption 1 Watt

*For more info about ThinKnx Micro server please check the related [page](#) on our website.*

- **ThinKnx Micro ZWave** It is a very small device with the ability to completely control KNX and Zwave systems. Can be enhanced with VoIP function for intercom communication.
  - Fanless processing device optimized for continuous operation
  - Mass memory on MicroSD
  - Power: 12-24 VDC - 1A Max
  - Z-Wave transceiver
  - 1x EIB/KNX port with direct connection through clips
  - 1x network port
  - KNX telegrams led
  - Consumption 1 Watt

*For more info about ThinKnx Micro ZWave server please check the related [page](#) on our website.*

- **ThinKnx Envision** All-in-one Touch Server which allows control of the home or building using a high resolution touch screen.
  - Quad-core ARM Cortex A9 CPU
  - 1 GB DDR3 SDRAM
  - Capacitive touch screen
  - 7" screen 158x93mm visible area, 1024x600 resolution or 10" screen 217x136mm visible area, 1280x800 resolution
  - Linux embedded operating system
  - Power: 12-24 VDC - 1.5A Max
  - 1x EIB/KNX port with direct connection through clips
  - 1x network port
  - Consumption: 10 W Max (for 7" screen) or 12 W Max (for 10" screen)

*For more info about ThinKnx Envision server please check the related [page](#) on our website.*

- **ThinKnx Rack** It is the most powerful and expandable device,

designed for large buildings or demanding applications.

- Fanless processing device optimized for continuous operation
- Mass memory on industrial Compact-Flash
- External power supply 220V AC
- 1x EIB/KNX port with direct connection through clips
- 2x serial ports configurable for RS232 or RS485
- 1x serial port RS232 9 pins
- 4x USB ports
- 2x Ethernet ports

*For more info about ThinkKnx Rack server please check the related [page](#) on our website.*

## ThinkKnx Clients

Thinknx clients are all the devices connected to the network and using the Thinknx application to connect to a Thinknx server. ThinkKnx offers a wide range of native applications to allow tablets, smartphones, touch screens and PCs to interface with the supervision system. The choice to create native applications comes from the necessity to obtain the best possible performances during the Wi-Fi or 3G connection to ThinkKnx servers, providing the user with an unprecedented experience.

- **ThinkKnx Player for Windows** This application allows to control the building from any Windows device. Suitable even for the Windows 10 operating system, it includes settings that allows to disable the full-screen mode, resize the application window and visualize the mouse cursor, adapting the application to the device which the application is installed on and improving the user's experience.
- **ThinkKnx Player for Mac** This application allows to control the building from any computer running Mac operating system. It offers the same functions of the Player for Windows application.



ThinkKnx Player for Mac is an Air application, thus to be installed on Mac computers it requires Adobe Air software installed. Adobe Air software can be found at <https://get.adobe.com/air/>.

- **ThinkKnx for iPad** This is the app for iPad that, connecting to ThinkKnx servers, allows to manage automation directly from iOS tablets. It allows to control several buildings using a quick and intuitive project selector.
- **ThinkKnx Pocket for iPhone** This is the app for iPhone that, connecting to ThinkKnx servers, allows to manage

automation directly from iOS phones. It allows to control several buildings using a quick and intuitive project selector.

- **ThinKnx for Android** This is the app for Android tablet and smartphone that coordinates the intelligent building through a Wi-Fi or 3G connection to ThinKnx servers. The interface is designed with attention to the smallest detail to adapt the graphics to the display size of the employed device.

## ThinKnx Cloud

Thinknx Cloud is used for several services in the Thinknx system:

1. **Modifying and updating the project remotely on server and clients.** Using the Thinknx Configurator, the integrator or installer can deploy the modified project to the Cloud. The server will automatically be alerted and will reboot once the new version is downloaded, and all the clients will get a notification that a newer version of the project is available to download on their devices. For more information regarding Thinknx Cloud, check out [this section](#).
2. **Automatic client connection.** In the absence of a fixed external IP address at the automated location, it is possible to use Thinknx Cloud to allow remote access to the project for all clients. They will be able to monitor their entire house from anywhere using only the Internet at no additional cost. However, some services might not be available for this option, such as answering a SIP call remotely or viewing a local IP camera from outside the house. For more information regarding Thinknx Cloud, check out [this section](#).
3. **Database for live data storage.** Every project will require to save some data for continuous monitoring, chart visualization, or monthly reporting. This data will be saved for each single server in a dedicated space on the cloud. Every time a Thinknx application is launched, the server will contact the cloud to retrieve this data, and update them as they change. Only a small amount of values is saved locally on the server. For more information regarding Thinknx Cloud, check out [this section](#).
4. **Dynamic DNS.** This option is used in the absence of a *fixed* external IP address at the automated location. Instead, the public IP address is accessible but changes after each connection renewal or on router reboot. Thinknx offers a free dynamic DNS service configurable from the server's web page. With this service, a host name (such as *myhome.my.thinknx.net*) is associated to the public IP address of the router and the server will always be reachable through this host name no matter what the public IP address is. For more information regarding Thinknx Cloud, check out [this section](#).

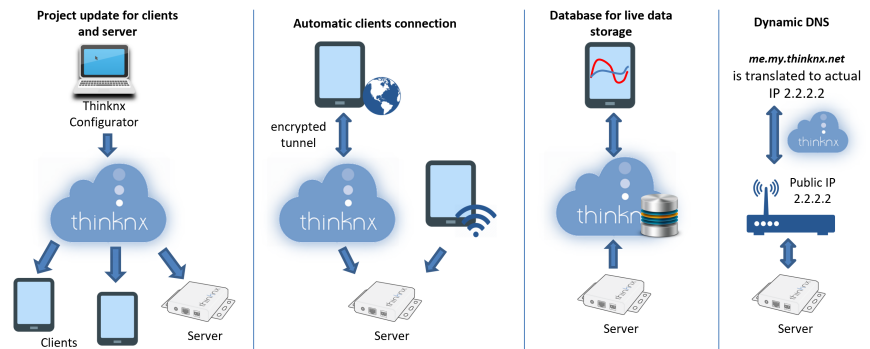


Figure 2: Thinknx Cloud Services

## ThinKnx Configurator

The ThinKnx Configurator software is the indispensable tool for the creation and development of a supervision project. It makes possible to create all the connections between the graphical user interface and all the devices integrated into the plant.

Following some simple steps and intuitive parameters, graphic interfaces can be deeply customized and are accessible from every client and device. Just as efficiently, it is possible to create logics and to set up all the system elements to obtain deep integrations between all the existing devices.

Finally, the same tool allows to upload the project on client devices and ThinKnx servers with differentiated exports based on the specific user.

For a detailed explanation of the Configurator, please refer to [Thinknx Configurator](#) chapter, or visit our [quick guide](#) to learn how to create the first project.

## Communication Between Components

All the clients can connect to the server using two IP addresses, one local address to use inside the LAN and one host name or external address to use outside the LAN for remote access. It is highly recommended to configure the local IP address of the server to be fixed and not use DHCP service. Otherwise, it is possible to run into client-server connection problems when the DHCP lease expires.

The fixed local address can be configured by accessing the server's web interface using port 5051 TCP. This can be done by inputting `http://server_ip_address:5051` into the web browser. In addition to **port 5051** TCP used for the web interface, the client-server connection is possible through **port 7550** TCP. **Port 5050** is used to deploy a Thinknx project from Configurator

to a client's device when using the local network instead of Thinknx Cloud.

To allow clients a remote access to the server, multiple scenarios can be considered depending on the actual situation:

- The existence of a **static external IP address** which is directly accessible from the Internet. This option is usually sold at an additional cost by the Internet provider. In that case, the external IP address can be set inside the Configurator, and clients will try to connect directly to this IP. Since the server is usually behind a router, the router has to be configured to forward connections to the server.

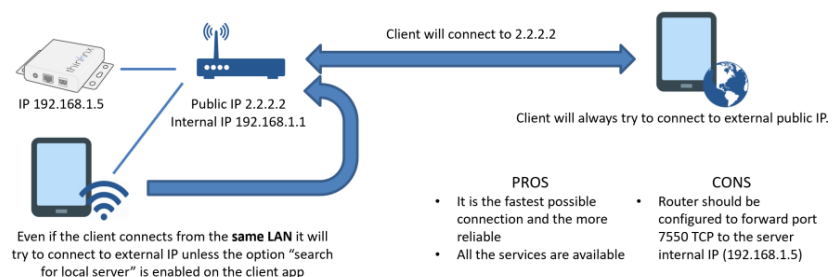


Figure 3: Static External IP Address

- The existence of a **dynamic external IP address** which is directly accessible from the Internet but changes at router reboot or after a connection renewal. In that case, and to overcome the problems tied to not having a static IP, Thinknx offers a free Dynamic DNS service configurable from the server's web interface. With this service, a hostname will be associated to the public IP address of the router and will always be translated to the actual IP address.

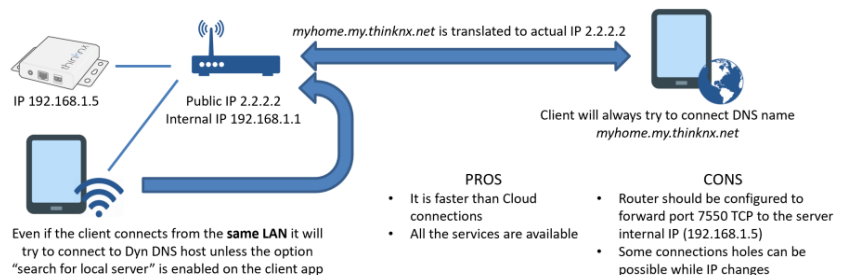


Figure 4: Dynamic External IP Address

- The **absence of a public IP address**. In that case, the IP

given from the Internet provider is not directly reachable from the Internet because it is part of a sub-network or behind a NAT. By enabling and using Thinknx Cloud, remote clients can establish a connection to the server using a tunnel created by means of the Cloud.

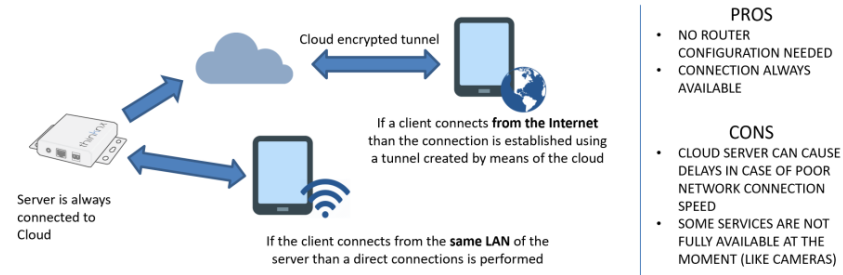


Figure 5: No Public IP address

## Network requirements for Thinknx services

To run properly, our services need to access some internet services and optionally needs some port forwarding. Hereafter all the involved ports and hosts with specifications of connection direction and purpose.

Service	Host port and protocol	Comments	Note
NTP	123 (TCP and UDP) outbound/inbound	Used to sync clock with NTP servers.	1
Thinknx cloud	cloud.thinknx.eu 80, 443, 24532 (TCP) outbound	Used to connect to Thinknx cloud service	2
Thinknx projects sync	data.thinknx.eu 80, 443 (TCP) outbound	Used to download project updates from cloud	2
Thinknx clients	7550 (TCP) inbound	Used by client to connect to server for control	3



Service	Host port and protocol	Comments	Note
Internal webpage	5051 (TCP) inbound	Used to connect to server webpage for maintenance	4
Remote assistance	<a href="http://www.pulsareng.it">www.pulsareng.it</a> 80, 443 (TCP) outbound	Used to sync with remote assistance service	5

**Notes:**

1. This service is mandatory to sync clock with internet services. Server can act as NTP server as well (inbound connection) and it is advised to use just for local network (not exposed to the internet)
2. These services are required when cloud connection is used.
3. Port 7550 inbound is only required when direct client connection is adopted. It usually requires to create port forwarding rules on router/firewall depending on network infrastructure
4. Access to webpages is not required and it is advised not to expose to the internet.
5. Remote assistance service used on request to obtain factory support

From:  
<http://www.thinknx.com/wiki/> - Learning Thinknx

Permanent link:  
[http://www.thinknx.com/wiki/doku.php?id=sys\\_arch&rev=1557827741](http://www.thinknx.com/wiki/doku.php?id=sys_arch&rev=1557827741)

Last update: **2019/05/14 11:55**

