

Deployment & Integration Guide

This guide covers the deployment models and integration options for Tekle Holographics solutions, with a focus on how to securely and effectively deploy our systems in various environments (cloud, on-premises, hybrid) and how to integrate Tekle’s hardware and software with your existing enterprise systems. This document is written for IT architects, system integrators, and project managers responsible for implementing Tekle’s holographic technology within their organizations. We will outline deployment prerequisites and steps, network topology considerations, and detail the integration points such as RealityBridge connectors and the Tekle Holographics SDK (THSDK). Following this guide will help ensure a smooth deployment and a well-integrated solution that leverages your current infrastructure and workflows.

Deployment Models Overview

Tekle supports three primary deployment models for our platform:

1. **Tekle-Hosted Cloud** – Tekle manages the entire infrastructure in our cloud environment, delivering the solution to you as a SaaS (Software-as-a-Service).
2. **On-Premises (Offline)** – The solution is deployed within your organization’s network with no dependency on external networks, ideal for secure or air-gapped environments.
3. **Hybrid / Self-Hosted** – A combination wherein you run Tekle server components in your own environment (data center or private cloud), possibly connected to external networks for remote access or updates, giving you control over data while still enabling collaboration across sites.

We will describe the deployment process and requirements for each model, including hardware/software needs, network setup, and maintenance.

1. Tekle-Hosted Cloud Deployment

In the Tekle-Hosted model, you essentially “plug and play.” Tekle handles the heavy lifting of server setup and management. Here’s how to implement:

- **Provisioning:** When you sign up for Tekle Cloud, our team creates a dedicated tenant for your organization in our multi-tenant cloud. We configure your initial administrator accounts and any region specifications (for example, if you chose EU data residency, we ensure your tenant’s data is in EU data centers).
- **Device Setup:** Your Tekle holographic devices (Holo-Tables, etc.) will need internet access to communicate with the cloud. Typically, this means connecting them to your network (wired or wireless) that has outbound internet connectivity on HTTPS (port 443). No inbound ports on your firewall need to be opened because devices initiate the connection out to our cloud. If your network uses a proxy or deep packet inspection, you may need to whitelist Tekle domains and allow WebSocket traffic. We provide a list of domains and IP ranges for our service.
- **Client Access:** Users can also access Tekle services via the web portal or plugins from their PCs. Ensure those PCs can reach our cloud endpoints as well. We support modern browsers (Chrome, Edge, Firefox) for the portal.
- **Authentication Integration:** By default, users will have Tekle Cloud credentials. If you prefer SSO with your corporate identity (Azure AD, etc.), our team will work with you to configure federation (SAML or OIDC). This involves exchanging metadata between your IdP and Tekle’s system. Once set, users can log in with your single sign-on and even use MFA enforced by your IdP. (Note: In cloud, we also support SCIM for user provisioning if you want to automate user management from your directory).
- **Data Migration:** If you have existing 3D models or data to load initially, you can upload via the portal or use our APIs to batch import. Some customers script the upload of hundreds of models – our integration team can assist with optimizing that.
- **Testing:** Once devices and users are set, perform a connectivity test. We recommend doing this in a pilot environment first: connect one device, have one or two users upload a test model, and go through the motions (viewing hologram, adding annotation, etc.) to confirm everything flows.

- **Security & Compliance:** For cloud, Tekle has ensured the environment is secure. However, you should still apply your organization’s usage policies. For instance, if some data is too sensitive to ever leave your network, even though our cloud is secure, you might decide to keep that particular data set only on an offline system. So, classify data and use the appropriate deployment accordingly (some clients use a mix: general data in Tekle Cloud, highly sensitive projects on an on-prem Tekle server).
- **Maintenance:** Tekle Cloud is maintained by Tekle – we handle updates (usually with minimal downtime windows communicated in advance), scaling, and backups. Your focus will be on managing user access and content. We provide usage reports and monitoring info via the admin portal so you can track your team’s usage.

The Tekle-Hosted deployment is the fastest to get started, often only requiring a day or two of network and device setup on your side. You benefit from our managed service and can be confident you’re always on the latest version.

2. On-Premises (Offline) Deployment

Deploying Tekle on-premises gives you full control and isolates the system within your secure network. This option involves setting up a Tekle server locally and connecting your devices to it.

- **Server Hardware Requirements:** Depending on the scale, you’ll need one or more servers. For a typical deployment (supporting up to, say, 10 devices and 50 concurrent users), a single server with 8+ CPU cores, 32+ GB RAM, and fast SSD storage (~1-2 TB, depending on how much model data you store) is recommended. It can run on Windows Server or Linux (we support both; Windows if you plan to integrate with AD easily, Linux for possibly better performance and cost). For larger deployments, we might use two servers (one for application, one for database) or a cluster. Our team will provide guidelines tailored to your expected load.

- **Server Software Installation:** Tekle will provide the server software package. This comes either as:
 - A pre-configured VM image (OVA or Hyper-V image) that you can import, which contains OS + Tekle software ready (just needs network configuration).
 - Or an installer script for your OS to set up the necessary components (application service, database, etc.). If using Windows, it might set up IIS/SQL Express; on Linux, Nginx/PostgreSQL, for example.
 - In either case, you'll run through a configuration wizard: set the admin account, organization name, initial network settings, license key activation (if offline, we provide an offline activation method).
- **Network Setup:** Decide if this server will be completely offline or accessible to remote sites. If fully offline/air-gapped, it should be placed in the secure network zone with no internet. Devices should be in the same network or have a route to it. If you want multi-site access (e.g., two offices connecting to one on-prem server), you might allow connections through a VPN or private WAN linking sites. In that case, ensure the proper routing – e.g., the server may need two NICs or specific firewall rules to allow devices from both subnets. If remote collaborators (not in network) need access, you'll have to punch a hole – typically via a reverse proxy in your DMZ that forwards to the Tekle server, or via a VPN solution for those users. We have documents describing how to publish the Tekle service through a reverse proxy with SSL.
- **Certificates:** By default, the on-prem server will generate a self-signed SSL certificate for the web services (so devices and browsers can connect via HTTPS). For production, you'll likely want to use your organization's CA or a trusted certificate. This avoids browser warnings and allows devices to trust the server. We support using custom certificates – simply replace the default cert with one issued by your CA (guides provided). If offline with no internal PKI, devices can be configured to trust the self-signed cert.
- **Active Directory Integration:** In offline mode, integration with AD is common for user management. Our on-prem server can integrate with AD LDAP – enabling users to log in with their Windows domain credentials. During setup, you would provide the LDAP server info and an account to bind for lookups (over LDAPS ideally). You then map AD groups to Tekle roles if desired (for example, members of "TekleAdmins" AD group get admin rights in Tekle automatically). This lets you avoid creating separate accounts and leverages your existing user authentication and password policies.

- **Deployment of Devices:** Once the server is up, you point devices to it. The device's configuration menu has a "Server URL" setting (on Tekle Cloud devices it's preset to cloud, but in on-prem you switch it to your server's address). For offline, you might set it to something like <https://tekle-server.local> (provided your DNS or hosts file resolves that). After pointing and restarting, the device will now talk to the local server for login and data. If the network is properly set, it should connect and prompt for the local credentials (or if AD integrated, domain login).
- **Testing & Calibration:** With one device connected, test uploading a model to the local server (via the admin portal or plugin configured to point to local server), then load it on the device. Ensure multi-user collaboration works if you have two devices (they should discover each other's presence via the server, if on same project). Adjust any firewall if something is blocked (the server uses a couple ports: typically, 443 for main API, maybe another for real-time signaling like 8080 unless we encapsulate in 443 – depends on config. This is documented in our network ports table).
- **Backup and Maintenance:** On-prem means you are responsible for backups. We provide tools: the server has a backup script or UI which can dump the database and content files. You should schedule regular backups to a secure location. Also plan for updates: Tekle periodically releases software updates. For on-prem, we will notify you and either provide a patch installer or new VM image. Apply updates in a timely manner (we recommend having a staging instance, if possible, to test updates before production). We understand in offline; updates might be less frequent due to change control – but note updates often include security improvements, so plan maintenance windows to update. Our support can assist remotely if you allow (some clients let us remote in under supervision to apply updates).
- **Scaling:** If usage grows, on-prem scaling could mean upgrading the server hardware (adding more RAM/CPU) or moving to a multi-server architecture. Tekle's on-prem can scale vertically easily; for horizontal scaling (multiple app servers), we'd consult with you – it's possible to have multiple application servers behind a load balancer with a shared database, but that gets complex. Many offline deployments are smaller scale, so one beefy server suffices.

On-prem deployment is more involved but grants independence from internet and full data sovereignty. Just ensure your IT provides the necessary environment (cool, powered server rack, network configuration, etc.), and follow security best practices (e.g., keep that server patched, even offline, apply OS patches periodically from a WSUS or equivalent offline repo).

3. Hybrid / Self-Hosted Deployment

A hybrid deployment can take various forms, but essentially, it's you hosting the Tekle server in an environment that is still connected or partially connected. One common scenario is hosting Tekle in your private cloud (e.g., Azure, AWS, or a private data center) to serve multiple sites.

- **Infrastructure Setup:** If using public cloud (Azure/AWS), you can use our provided templates (ARM template for Azure or CloudFormation for AWS) that create the needed VMs, storage, networking for Tekle server. This automates much of the install. You'll need to supply parameters like VM sizes, region, admin credentials, etc. Once deployed, it's similar to the on-prem server: it runs in your cloud subscription.
- **Networking in Hybrid:** Because it's in, say Azure, it has internet connectivity by default, meaning remote offices and users can reach it over the internet (we recommend restricting that via firewall rules or virtual network settings – e.g., only allow known office IPs or require VPN). Often, hybrid is used to connect multiple company sites: you might have HQ and branch offices all connecting to the Tekle server in Azure. In that case, you ensure each office network can reach the Azure endpoint (over site-to-site VPN or just open internet if you secure it enough).
- **Offline & Online Mix:** Some hybrid scenarios include an offline component – e.g., run a Tekle server on-prem that occasionally connects to Tekle Cloud to sync certain data or get updates. While not common, we can support scheduled synchronization or one-way data push. For instance, maybe you run offline normally, but once a week you connect it to internet to fetch software updates or to sync anonymized usage stats for global analysis. These setups are custom – Tekle would work with you to define how and ensure security (like you only connect through a controlled gateway).
- **Maintenance in Hybrid:** If you're hosting, you manage it (with our support as needed). Keep your cloud VMs secure (apply security groups, NSGs, etc.). Monitor resource usage in your cloud portal – if user count grows, you might need to increase VM size. But also, you have auto-scale options if set up right (maybe auto-scale the app service based on CPU usage, etc., if stateless frontends – advanced config).
- **Data Migration to Cloud:** If moving from on-prem to cloud (or vice versa), we provide migration tools: you can back up from one and restore to another environment, as long as versions match. This means you can start on-prem and later move to a private cloud as needs change, or the opposite, with minimal fuss.

- **Integration with Company Services:** In self-hosted, you have more flexibility to integrate with other internal systems. For example, hooking Tekle logs into your enterprise SIEM (we can output logs to a syslog server, or you just retrieve from the server). Or connecting Tekle to your data lake – some customers periodically export model usage data from Tekle DB to analyze how often certain designs are reviewed, etc. Being self-hosted means direct DB access is possible (we caution to treat it read-only to not corrupt app, but it's your instance, you have DB creds).
- **Cloud-Specific Features:** If hosting on Azure, you can take advantage of Azure's services – e.g., use Azure SQL as the database instead of managing SQL on VM, or use Azure Monitor for health. The same for AWS with RDS, CloudWatch, etc. Tekle's architecture can fit into those managed services, often requiring just slight config changes (like pointing our app to the managed DB connection string). This can give you the benefits of cloud (redundancy, managed DB backups, etc.) with the control of your own environment.

Hybrid gives a nice balance: you maintain control over data (your cloud, your rules) and often can integrate easier with corporate IT standards, but you still allow widespread accessibility (since your cloud is globally reachable to your org). Just be sure to apply strong security on that externally reachable part (VPNs, IP restrictions, web app firewall, etc.) since it's exposed to internet if not locked down.

Integration with Enterprise Systems (RealityBridge Connectors)

Tekle's RealityBridge allows direct integration with various enterprise systems to streamline data flow. Here are common integrations and how to set them up:

- **BIM Software (Autodesk Revit, Navisworks):** We provide plugins for Revit and Navisworks (as well as support for IFC files for other BIM tools). To integrate:
 - Install the Tekle plugin into Revit/Navis (we supply an installer for these plugins).
 - The plugin will add a "Tekle Holographics" menu in those applications. The first time, you'll configure it with the server address (Tekle Cloud or your on-prem server) and your credentials.
 - Usage: An architect can open a Revit model and click "Publish to Tekle" which will package the model (filtering according to any settings like certain categories or levels if needed) and send it to Tekle. The plugin communicates via Tekle API behind the scenes.

- Security: The plugin uses the same user access rights, so a user can only publish to projects they have permission for. If on-prem, ensure their PC can reach the Tekle server on the necessary port.
- Once published, the model is available in the Tekle system for holographic viewing. You can also update models – republishing can create a new version or update the existing (depending on settings; Tekle can version models so you have v1, v2, etc.).
- Integration tip: Some customers script the plugin via Revit’s API to auto-publish at certain milestones (e.g., nightly publish of latest model to Tekle for morning review).
- **CAD and Other 3D Formats:** For generic CAD (AutoCAD, SolidWorks, etc.), Tekle supports import of formats like STEP, OBJ, glTF, etc. These aren’t direct “live” integrations, but you can export from those tools and then upload to Tekle manually or via API. If you need a more automated pipeline (e.g., whenever a new design is checked in to PLM, push to Tekle), you can script using our REST API to upload that file.
- **GIS (Geospatial):** Integration with GIS like Esri ArcGIS can be done by connecting Tekle to GIS data streams:
 - If on Tekle Cloud, we have built-in support to fetch ArcGIS map tiles or layers given credentials to an ArcGIS Online account. You’d input your ArcGIS API key or token in our portal, and then you can overlay GIS content (terrain, maps) in holograms.
 - If on-prem, you might connect to your internal GIS servers similarly; Tekle can consume WMS/WMTS or REST endpoints to pull maps or features.
 - A typical use is city planning: Tekle can integrate with a city’s GIS to pull in terrain and building footprints around a project site, giving context to a BIM model.
 - Ensure network access for Tekle to reach the GIS API (if Tekle Cloud to ArcGIS Online, it’s fine; if on-prem Tekle to on-prem GIS server, same LAN or open firewall).

- **IoT & Live Data:** Some deployments want real-time sensor data or simulation results integrated (like showing live construction progress or IoT sensor readings in a factory layout). Tekle’s platform allows such integration via our API or SDK:
 - For simple cases, Tekle has an API endpoint where an external system can push annotation or status updates attached to model elements. For example, you have sensors on building floors, an IoT platform could send an update to Tekle marking floor 5 as “hot” with a red highlight in the hologram.
 - For more complex visuals or interactions, using the THSDK (see next section) might be more appropriate, as you can fully program custom logic in Unity/Unreal.
- **Single Sign-On (SSO):** Integration with corporate SSO was touched on, but to reiterate: Tekle supports SAML 2.0 and OpenID Connect for SSO. Setting this up requires coordination with your IAM team:
 - Provide Tekle (or configure in our admin portal if available) with your IdP metadata (issuer, SSO URL, x.509 cert).
 - In your IdP, add Tekle as an app, configure the ACS (Assertion Consumer Service) URL we give, and map attributes (we typically need username, email, maybe group if doing auto-role mapping).
 - Test with a couple users. After SSO, Tekle can either auto-provision accounts (if user exists in IdP and allowed, we create a Tekle user entry on first login) or you pre-create them and just link by email/ID.
 - If using OIDC (like OAuth with Azure AD), a similar config with client ID/secret in Tekle and a enterprise app registration in Azure AD.
 - SSO makes life easier for users and aligns with your security (e.g., if someone leaves your company, disabling them in AD will also cut off Tekle access).
- **REST API Usage:** Tekle offers RESTful APIs for many functions (upload model, list projects, invite user, etc.). If you have a portal or system that you want to integrate with Tekle (for instance, a project management system where you click a button to view in hologram), you can use these APIs.
 - Authentication is via token (we support OAuth flows or using an API key for service accounts).

- Our API documentation (OpenAPI/Swagger spec) is provided so your developers can code against it.
- Example: A construction management app could, upon user clicking “View in Tekle”, call Tekle API to upload the relevant model or retrieve an existing Tekle model ID, then possibly trigger a notification to a device to load that model (the latter can be done via our SDK or scheduled session).
- Many integration use-cases can be addressed with a combination of API calls. And since you can self-host Tekle, you can even extend our API by adding a microservice that interacts with both Tekle and another system to orchestrate complex flows.
- **Webhook/Notifications:** Tekle can emit certain events (if enabled) as webhooks to your systems. For instance, when a model is updated or when a session ends. If you want to track usage or trigger follow-up actions (like log a meeting summary whenever a holographic review session ends), these webhooks can be useful. You’d provide a URL in your environment for Tekle to POST events to (with a secret for verification). On-prem, this stays internal; on Tekle Cloud, your webhook endpoint must be reachable from our cloud.

Tekle Holographics SDK (THSDK) – Custom Application Development

For deeper integration or to build custom holographic applications, THSDK is the tool. It allows you to use Unity or Unreal Engine to create tailored experiences on Tekle hardware.

- **Obtaining THSDK:** It comes as a Unity Package (for Unity) or a Plugin for Unreal, along with documentation and sample projects. You’ll need to have a Tekle developer license to use it (the license key is installed on the target device to allow custom apps to run).
- **Development Setup:**
 - Unity: Import the THSDK Unity package into your Unity project. It provides prefabs and scripts for connecting to Tekle devices. You can develop on a standard PC – in editor, you might simulate or connect to a device on the network for testing.

- Unreal: Similar, you add the Tekle plugin, then you can use Blueprints or C++ to interface.
- You'll have a library to call functions like "Connect to device", "Display this model", "Get user input", etc.
- **Creating Apps:** With THSDK, you can do things like create a training simulation with interactive holographic characters, or an application that merges live IoT feeds with 3D.
 - Example: A company wants a custom UI on the hologram for maintenance procedures – they use Unity to overlay step-by-step instructions next to equipment holograms. Using THSDK, they show text, handle user gestures, and control the device's content beyond what the standard Tekle interface offers.
- **Deployment of Custom Apps:** Once you build your Unity/Unreal app, you deploy it to the Tekle device. This typically means generating a build (for Windows, as our devices run Windows), and installing it on the device. The device will require the THSDK runtime (which is usually pre-installed on developer-enabled devices) and a valid license for third-party apps.
 - We provide a mechanism to launch custom apps on the device (either replacing the default Tekle interface or via a launcher menu).
 - Security: All custom apps must be signed by Tekle or an authorized developer certificate to run – this is to prevent unverified code on the device (we can authorize your developer cert).
- **Hybrid Use with Tekle Platform:** You don't have to go fully custom; some apps use a hybrid approach. For example, use Tekle Cloud for data management and multi-user sync, but a custom Unity app for specialized rendering on the device. THSDK supports connecting to Tekle services – for instance, your custom app can still log into Tekle server and retrieve models, but then present them with a different UI or additional effects.
- **Development Support:** Tekle offers developer support for THSDK. If you run into issues integrating or need guidance on performance optimizations (like how to handle very large models in Unity for holographic display), our engineers can provide best practices.

- **Use Cases Enabled:** RealityBridge covers many pre-built integrations, but THSDK unleashes creativity. Past examples from clients:
 - A defense client integrated Tekle’s display with a simulation engine: using the SDK, they fed real-time battlefield simulation data into a holographic map in Unity, allowing custom avatars and icons beyond standard Tekle features.
 - A manufacturing client built a custom quality assurance app where a Tekle Holo-Wall would show a holographic overlay on a car part and the app guided inspectors through checking points, recording their input in an external system. THSDK let them create the guided workflow UI on the hologram.
 - Because Unity and Unreal are so powerful, essentially any 3D interactive application you can imagine can be adapted to run on Tekle’s volumetric displays, leveraging the THSDK to interface with the hardware (like head-tracking, multi-user perspective management, etc., which the SDK handles so you don’t have to reinvent that).

Deployment Considerations: Security, Updates, and Performance

When deploying Tekle solutions and integrating, keep in mind:

- **Security:** Always align with your IT security. For cloud, restrict access as needed. For on-prem/hybrid, keep servers patched, use firewalls to limit ports, and regularly review user access in Tekle (remove users who left projects/company).
- **User Training:** Ensure end-users (designers, engineers, etc.) know how to properly use the plugins and devices. We often provide training sessions or materials. A well-trained user base gets the most out of the integration (e.g., they’ll know to keep BIM models organized so the holographic output is optimal, or how to use the collaborative features effectively).
- **Device Placement and Environment:** Holographic displays often function best in certain lighting and space conditions. Deployment includes the physical setup: ensure a stable power source, appropriate room (low direct sunlight for clear projection, enough space for users to walk around the device safely). We include guidelines for device environment. Also consider network connectivity for devices – wired is generally more stable than Wi-Fi for streaming large model data, though Wi-Fi (especially Wi-Fi 6) is feasible.

- **Updates and Compatibility:** Tekle frequently updates software – when updating server or cloud, update device software accordingly (they usually come in tandem). Also, when updating integrated software (like if Autodesk releases a new Revit version, you might need to update Tekle’s plugin to ensure compatibility; we strive to release plugin updates quickly for new versions of major software).
- **Performance Tuning:** Large models can be heavy. Tekle’s software and devices handle a lot, but it’s good practice to optimize models – e.g., remove unnecessary detail not needed for visualization, use our decimation tools for point clouds, or segment very large models into sections. Integration can help here: for example, our Revit plugin lets you choose to export only specific 3D views or categories, so you can exclude non-visible elements to improve performance.
- **Scaling Users:** If planning a big deployment (dozens of devices, hundreds of users), consult with Tekle for architecture – maybe you’ll need multiple Tekle servers (regional servers per geography), or a load balancer in front of two application nodes for redundancy. We can help design such.
- **Support Setup:** As part of deployment, decide how support will work. If Tekle Cloud, you contact us directly. If on-prem, sometimes first-line support is your IT who then contacts us if needed. We provide admin guides for your IT to handle basic tasks (user management, resetting devices, etc.) so they can resolve minor issues quickly.

By considering these factors, you’ll ensure that your Tekle holographic solution is deployed smoothly, integrates seamlessly with your existing systems, and is primed to deliver value immediately.

Conclusion and Next Steps

Deploying Tekle Holographics solutions can transform the way your teams collaborate and visualize complex data. This guide has outlined the steps and considerations for various deployment modes and highlighted how to integrate Tekle into your enterprise ecosystem.

The next steps would be:

- Review internally which deployment model suits your security and operational needs best.
- Engage your IT networking and security teams early to align on requirements (ports, server placement, SSO, etc.).

- Plan a pilot project: choose a specific project or department to start with, deploy a device or two and test the end-to-end integration. This will yield valuable insights and user feedback before scaling up.
- Leverage Tekle’s support and services: our team can assist with installation, do a network assessment, conduct training sessions, and even help customize integrations or develop THSDK applications if needed.
- Establish a support and update schedule with Tekle to keep the system up-to-date and secure over time.

With careful planning and execution following this guide, you will set a strong foundation for your Tekle Holographics deployment. We are here to help every step of the way, ensuring that your organization gains the maximum benefit from holographic technology with minimum disruption. Welcome to the future of immersive collaboration and data visualization!