FAID
FIGAF AI-DRIVEN

CUSTOMER INSTALLATION GUIDE

Install Figaf Manager and the FAID Apps in your SAP BTP subaccount

From an empty Cloud Foundry space to running FAID Apps, step by step, on SAP standard services in your own subaccount.

IN THIS GUIDE

- What you install, and where it runs
- What to prepare in SAP BTP before the session
- Deploy Figaf Manager from the BTP cockpit
- Five setup steps inside Figaf Manager
- Deploy a Figaf Tool, if you need a new one
- Install the FAID Apps
- Connect the Figaf Tool and your systems
- Troubleshooting

Figaf Manager and FAID · November 2026

Screens from the installation dry run of 13 November 2026. Names, e-mails and keys are replaced with placeholders.

OVERVIEW

What you install, and where it runs

Everything lives in one Cloud Foundry space of your SAP BTP subaccount and uses SAP standard services. Figaf Manager installs, updates and monitors the rest.

YOUR SAP BTP SUBACCOUNT · CLOUD FOUNDRY SPACE

`Figaf Manager`
You deploy it once from the cockpit. It runs the setup and manages the FAID Apps.

`Shared backend`
Deployed automatically with the first FAID App. Holds the connections.

`FAID Apps`
Installed, started, updated and removed from Figaf Manager, one or many at a time.

`figaf-db`
SAP PostgreSQL, hyperscaler option. FAID gets its own role and schema.

`figaf-faid-xsuaa`
Authorization and Trust Management. Roles of the manager and the apps.

`figaf-faid-credstore`
SAP Credential Store. Connections and the technical management user.

`figaf-connectivity`
Optional. Cloud Connector tunnel to on-premise SAP PI/PO.

`figaf-destination`
Optional. BTP destinations of on-premise SAP PI/PO.

Sign-in goes through SAP Identity Authentication once step 1 has run.

CONNECTED SYSTEMS

Figaf Tool

Your existing Figaf Tool, or a new one that Figaf Manager deploys for you. Its URL and API client are stored once, never typed into an app.

SAP Integration Suite

Your tenants, reached through service keys stored in the Credential Store.

AI provider optional

One provider for the whole installation, used by apps with AI features.

THE INSTALLATION AT A GLANCE

Four phases, most of them automated

You do two things by hand in the BTP cockpit. Everything else runs from Figaf Manager, which shows each `cf` and `btp` command live in its terminal drawer.

0

BTP ADMINISTRATOR · BEFORE THE DAY

Prepare SAP BTP

- Subaccount with Cloud Foundry
- Entitlements assigned
- A dedicated technical user

A

about 10 min

BTP COCKPIT

Deploy Figaf Manager

- Create a space
- Deploy the zip and manifest
- Claim the one-time setup token

B

steps 1–3

FIGAF MANAGER · SETUP

Set up the installation

- Sign in to Cloud Foundry
- 1 Prepare the space
- 2 Store the management user
- 3 Check the base services

C

steps 4–5

FIGAF MANAGER · FAID APPS

Install and connect

- Optional: deploy a new Figaf Tool
- 4 Install the first apps
- 5 Connect the Figaf Tool
- Add Integration Suite systems

> **Plan for a sandbox first**
> Install into a sandbox subaccount first and keep it for testing upgrades. Repeat the same steps in production once you are happy.

Accounts, roles and access

Have these ready before the installation session. The technical user takes the longest to arrange.

In SAP BTP

- A subaccount with the **Cloud Foundry environment** enabled. Subaccounts without it cannot be selected.
- Your own user as **Space Developer** and **Space Manager** in the space you create. You need this to deploy and to read the application log.
- Optional: a user who may **assign role collections** in the subaccount. Figaf Manager then assigns `FAID-Manager-Admin` for you. The same SAP BTP sign-in lets the manager deploy a new Figaf Tool.
- Outbound access from Cloud Foundry to `manager-download.figaf.com`, `faid-download.figaf.com`, `github.com` and `hub.docker.com`.

A dedicated technical user

- An e-mail account for a **technical account, never a person's**, for example `figaf-tech@acme.com`.
- Registered with SAP so it can sign in with a **password**, and with **two-factor authentication switched off**.
- Added to the subaccount and to the Cloud Foundry space as **Space Developer** and **Space Manager**.

Figaf Manager stores this user in the Credential Store and signs itself in after every restart. Without it you need a new one-time passcode each time.

Entitlements to assign to the subaccount

Figaf Manager reads the plans your subaccount offers. A plan that is not entitled cannot be created, and step 1 stops.

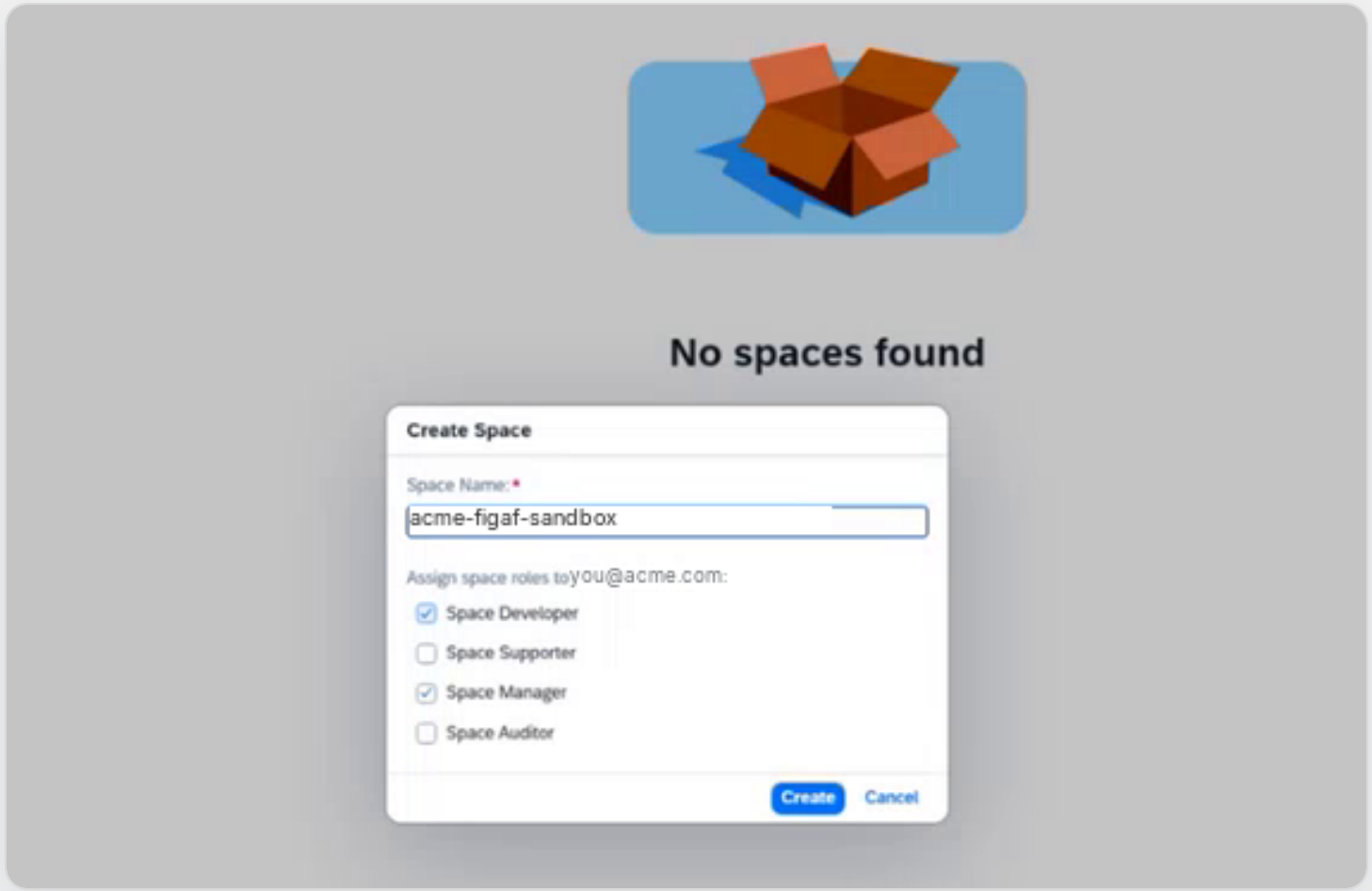
SERVICE	PLAN		USED FOR
PostgreSQL, hyperscaler option	`standard`	required	`figaf-db`. The dry run used 2 GB memory and 20 GB storage. High availability is not needed, and Google Cloud landscapes do not offer it.
SAP Credential Store	`standard`	required	`figaf-faid-credstore`. The free plan allows one instance per subaccount and very few entries, so use standard.
Authorization and Trust Management	`application`	required	`figaf-faid-xsuaa`, the roles of Figaf Manager and the FAID Apps.
Cloud Foundry runtime	memory quota	required	Figaf Manager uses 512 MB. Add its approuter, the shared backend and the apps you plan to install. A new Figaf Tool uses 3700 MB by default.
Connectivity	`lite`	optional	Only with on-premise SAP PI/PO reached through SAP Cloud Connector.
Destination	`lite`	optional	Same purpose. Both instances are shared with the Figaf Tool.

> **Cost**
>
> PostgreSQL standard and Credential Store standard are paid plans. What they cost depends on your SAP contract. Removing Figaf Manager does not delete these instances.

Create a space for Figaf

1. Open the **SAP BTP cockpit** and go to your subaccount.
2. Choose **Cloud Foundry → Spaces → Create Space**.
3. Name it after your company and the environment, for example `acme-figaf-sandbox`.
4. Give yourself **Space Developer** and **Space Manager**, then choose **Create**.

> **One space per installation**
> Use a separate space, and ideally a separate subaccount, for sandbox and production.

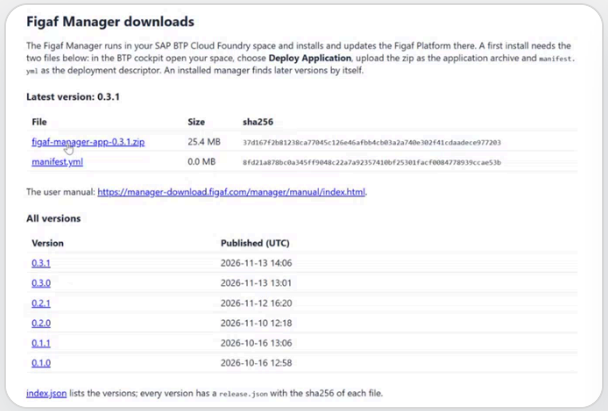


Subaccount → Cloud Foundry → Spaces → Create Space.

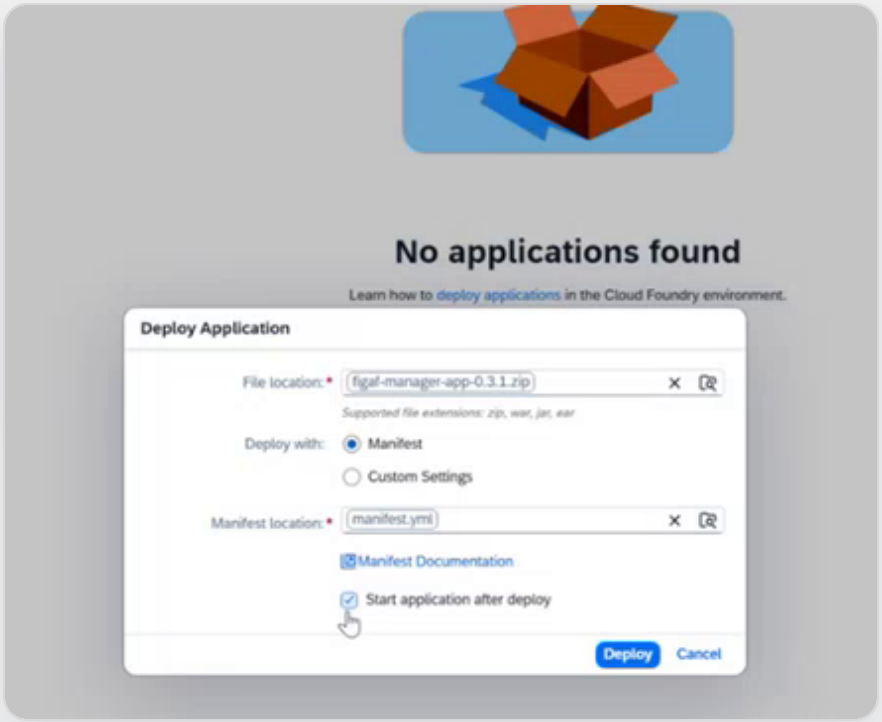
Download and deploy Figaf Manager

1. Open **manager-download.figaf.com** and download both files of the latest version: the application zip and `manifest.yml`.
2. In your space, open **Applications → Deploy Application**.
3. Choose the zip as **File location** and `manifest.yml` as **Manifest location**. Keep **Start application after deploy** ticked.
4. Choose **Deploy**. The app `figaf-manager` starts in about a minute.

Use the manifest as it is. The download page lists a sha256 checksum for every file.



The public download page.

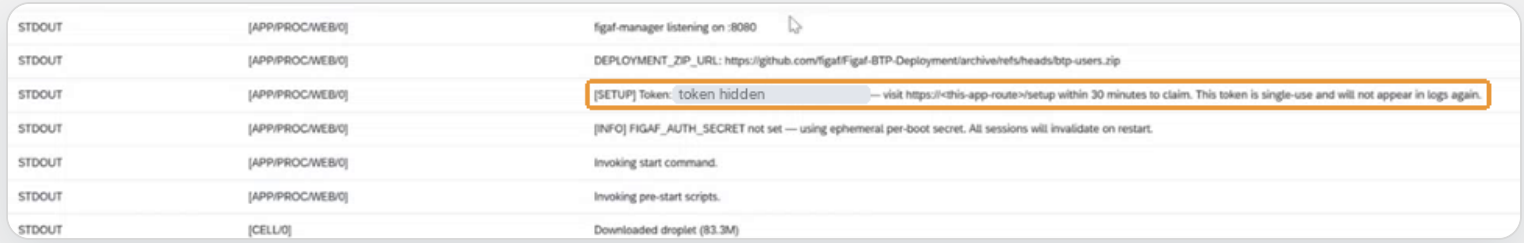


Deploy Application with the zip and the manifest.

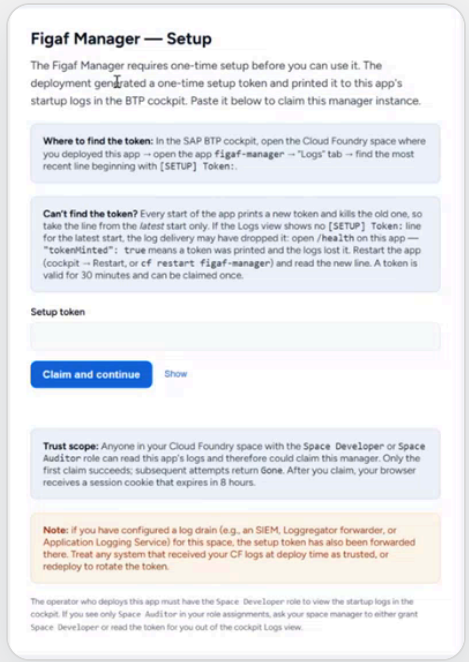
Claim the one-time setup token

1. Open `figaf-manager` and copy its address under **Mapped Routes**.
2. Open **Logs** and find the newest line starting with `[SETUP] Token:`. Copy only the token.
3. Open the manager's address, paste the token and choose **Claim and continue**.

> **Valid for 30 minutes, once**
>
> No token in the log, or too late? Restart `figaf-manager` in the cockpit and use the new line.



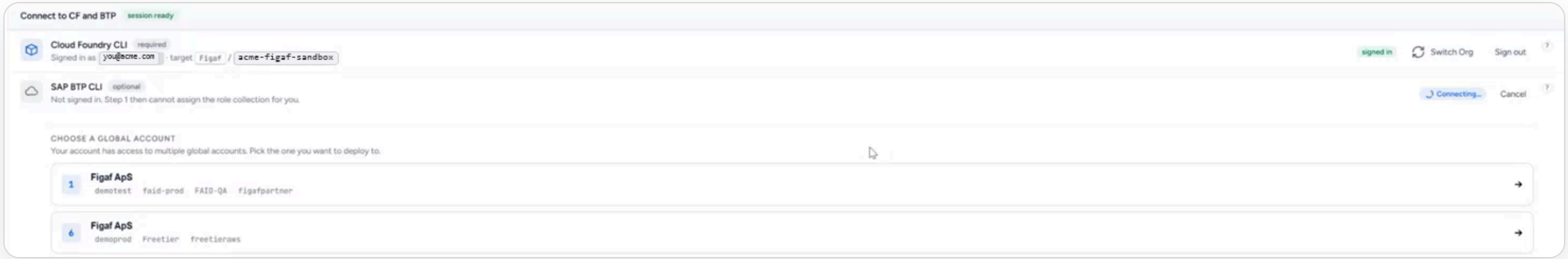
figaf-manager → Logs. The token value is hidden here.



The manager's setup page. It also explains where the token is and who can read it: anyone with Space Developer or Space Auditor in the space. After the claim your browser session lasts 8 hours.

Connect to Cloud Foundry and SAP BTP

1. **Cloud Foundry CLI** required: choose **Get passcode**, copy the temporary authentication code from the SAP page and paste it back. Pick the org and the space you created.
2. **SAP BTP CLI** optional: sign in, choose the global account, then the subaccount. Only then can step 1 assign `FAID-Manager-Admin` to you.
3. Check the summary line: signed in as you, with the right org and space as target.

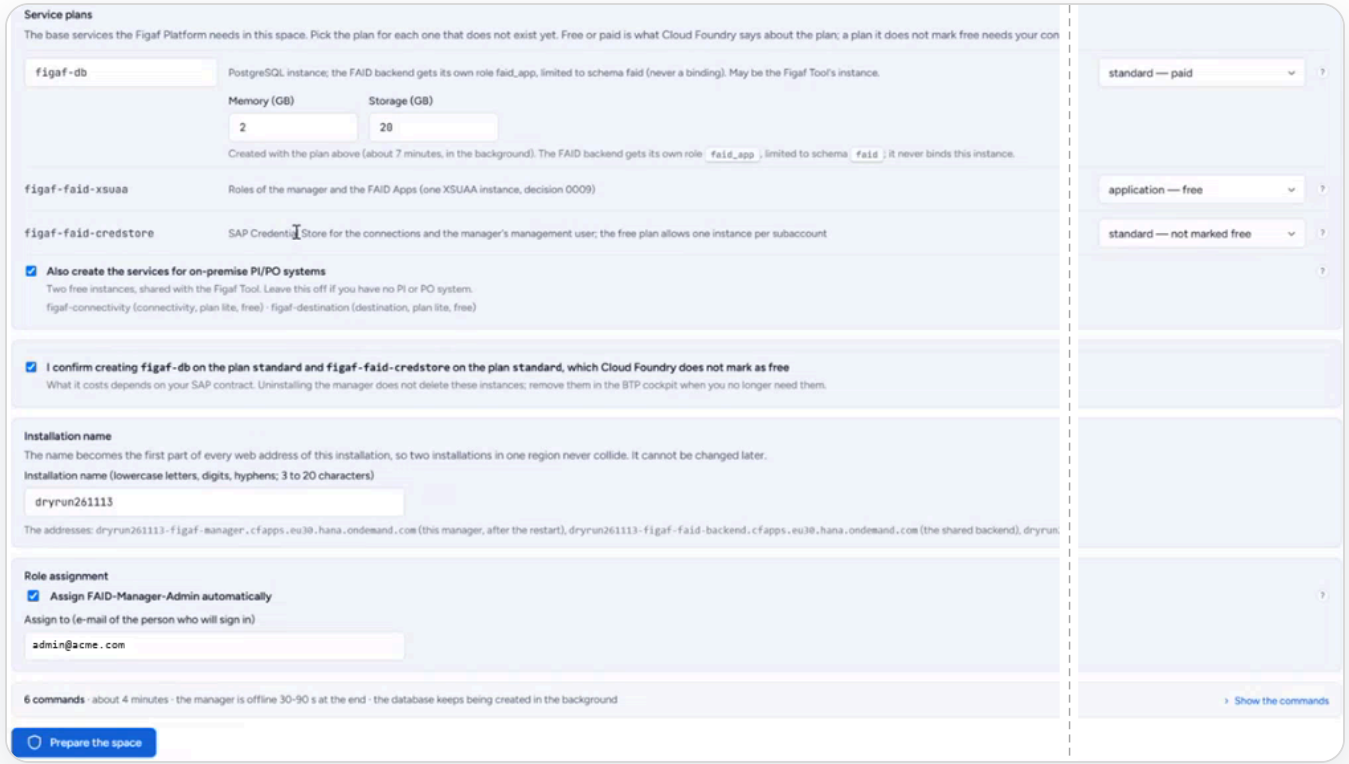


Cloud Foundry is signed in. The SAP BTP CLI asks for the global account.

> **Pick the right subaccount**
> The list shows every global account and subaccount you can reach. Read the names before you choose.

Prepare the space

figaf-db	`standard`. The dry run used 2 GB memory and 20 GB storage.
figaf-faid-xsuaa	`application`.
figaf-faid-credstore	`standard`.
PI/PO services	Tick only if you connect on-premise SAP PI/PO.
Confirm	Tick the box that confirms the paid plans.
Installation name	3–20 lowercase letters, digits and hyphens, for example `acme-sbx`. It starts every web address and **cannot be changed later**.
Role assignment	Assign `FAID-Manager-Admin` to the e-mail of the person who will sign in.



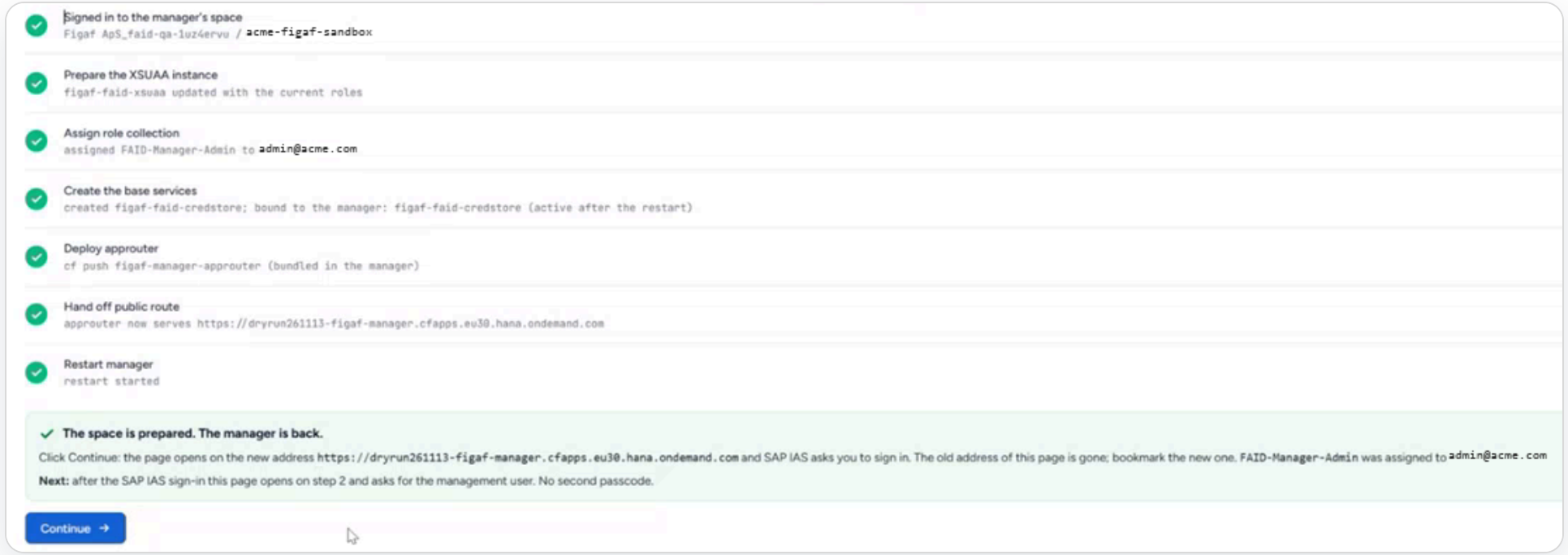
Choose **Prepare the space**. It takes about 4 minutes. The manager is offline for 30–90 seconds at the end, and the database keeps being created in the background.

The manager moves to its own address

- Step 1 puts an approuter in front of Figaf Manager and turns on sign-in through **SAP Identity Authentication**.
- The manager now lives at `https://<installation-name>-figaf-manager.cfapps.<region>.hana.ondemand.com`. The old address is gone, so **bookmark the new one**.
- Choose **Continue** and sign in. The page opens on step 2.

Sign-in loops or "forbidden"?

Your browser may still hold the old session. Sign out and in again, or open the new address in a private window.

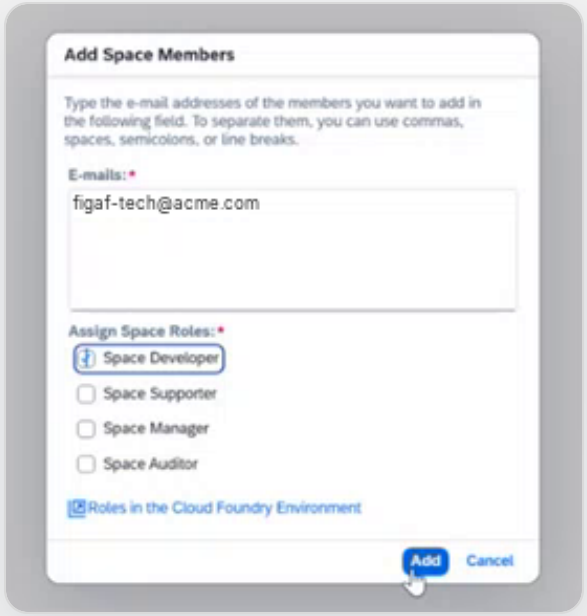


All seven actions done. The panel names the new address and the role assignment.

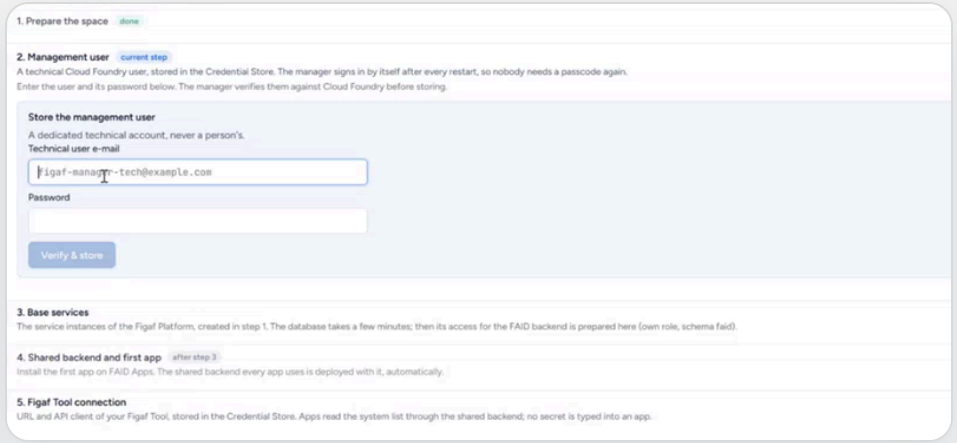
Store the management user

1. In the cockpit, add the technical user under **Security → Users** with the default identity provider.
2. Add it to the space under **Space Members** as **Space Developer** and **Space Manager**.
3. In Figaf Manager, enter its e-mail and password and choose **Verify & store**.

The manager checks the user against Cloud Foundry before storing it in the Credential Store. From then on it signs itself in, so nobody needs a passcode again.



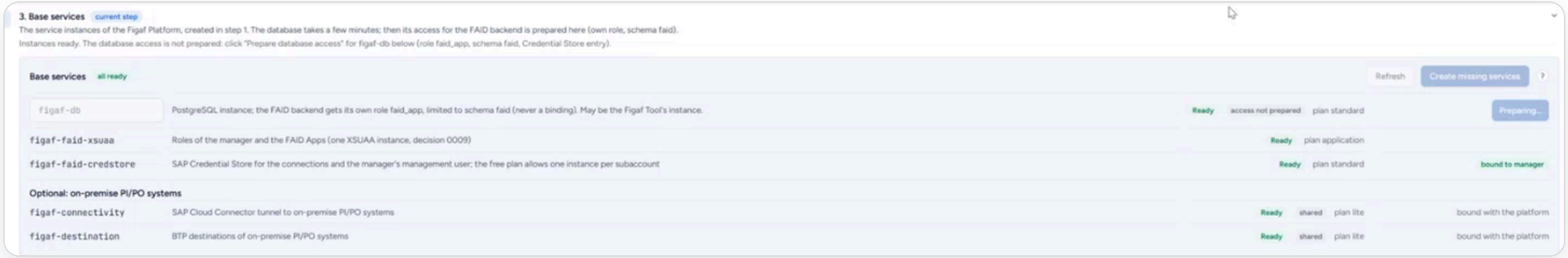
Space Members → Add Members.



Step 2 in Figaf Manager.

Check the base services

1. Wait until every service shows **Ready**. The database takes several minutes. Use **Refresh**.
2. If a service is missing, choose **Create missing services**.
3. Choose **Prepare database access** on `figaf-db`. FAID gets its own role, limited to its own schema.



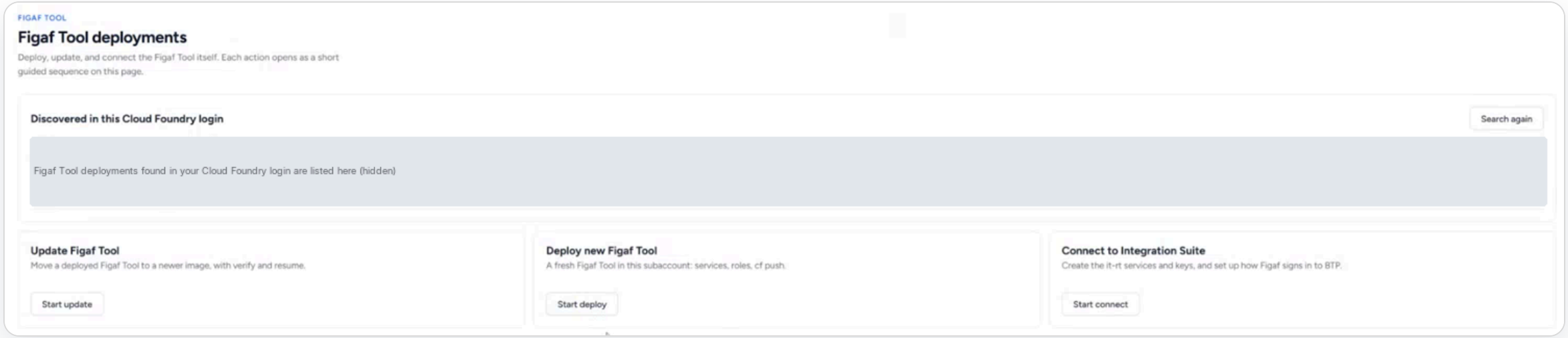
All base services ready. Database access is being prepared.

> **Repair later**
> This step stays available. If an instance goes missing after the installation, you repair it here.

Deploy a Figaf Tool from Figaf Manager

Deploy the Figaf Tool before you install the FAID Apps: the apps work once the Figaf Tool is connected. Already run one? Skip to step 4. If not, Figaf Manager deploys a fresh one into your space and reuses its database.

1. Open **Figaf Tool** in the Figaf Manager menu. It lists the Figaf Tool apps it finds in your Cloud Foundry login.
2. Under **Deploy new Figaf Tool**, choose **Start deploy**. It creates the services and roles and pushes the app.
3. The deployment needs an **SAP BTP CLI sign-in**. If you see *BTP login required*, sign in first, as shown on the next page.



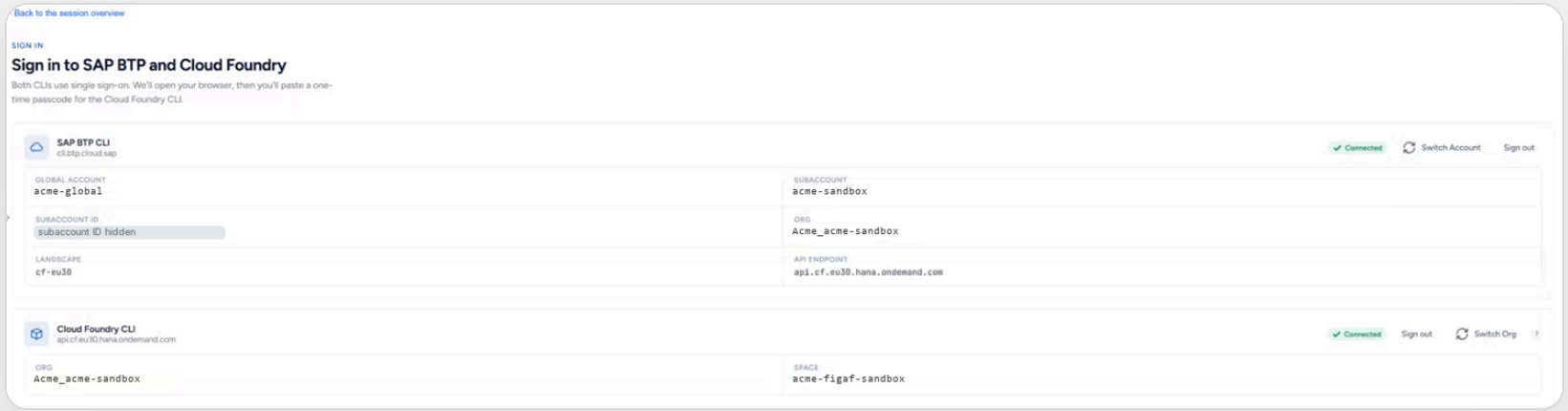
Figaf Manager → Figaf Tool. **Update Figaf Tool** later moves a deployed Figaf Tool to a newer image. **Connect to Integration Suite** creates the service keys the Figaf Tool uses.

Give the manager an SAP BTP sign-in

1. Open **Session & access** at the bottom of the menu and sign in to the **SAP BTP CLI**.
2. A browser tab opens. Confirm **Yes, log in to SAP BTP** and sign in with your SAP user.
3. Choose the **subaccount**. Subaccounts without Cloud Foundry are shown but cannot be selected.
4. When both CLIs show **Connected**, go back to **Figaf Tool → Start deploy**.



The SAP page that confirms the single sign-on for the BTP CLI.



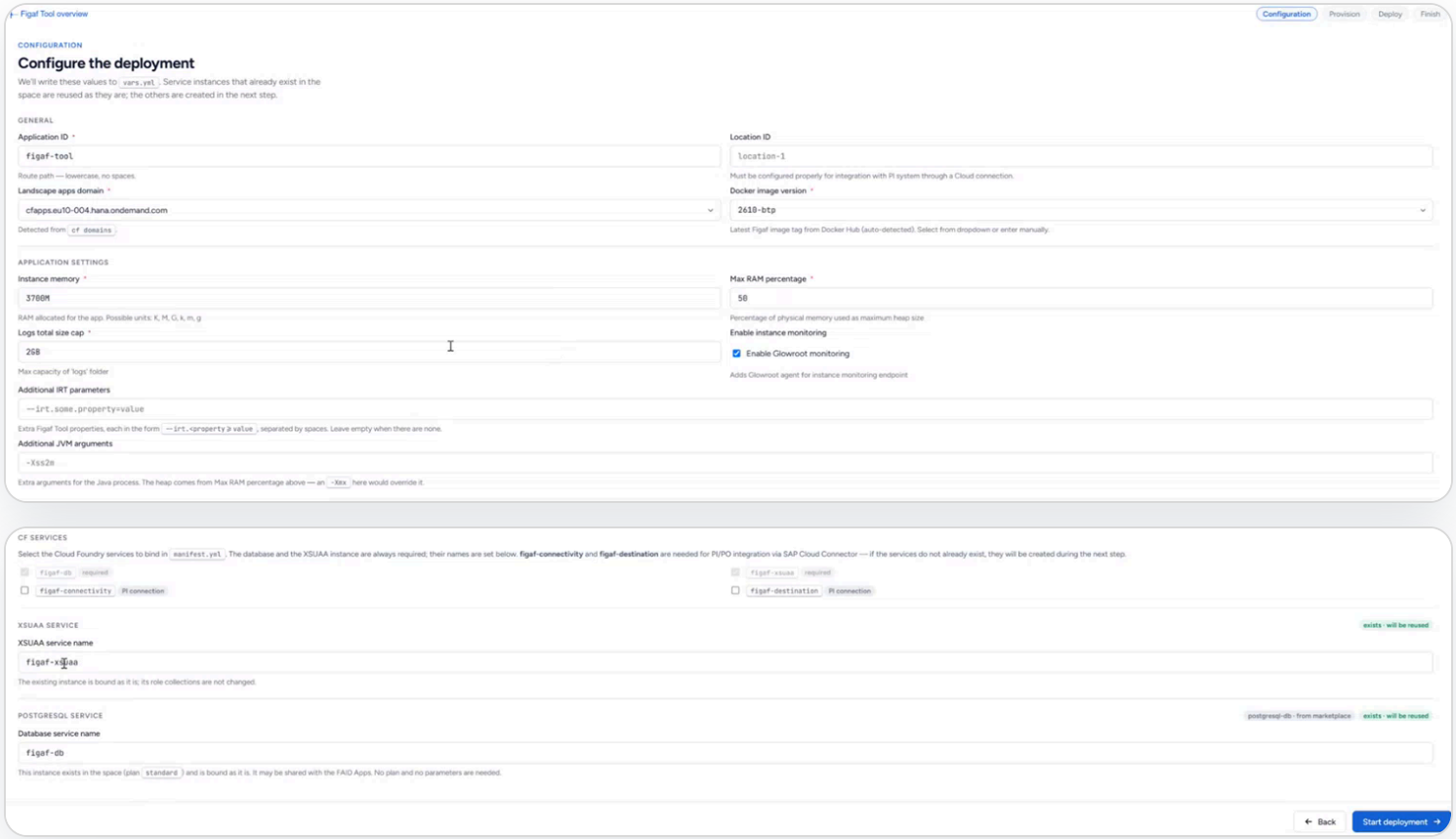
Session & access: global account, subaccount, org and space. Names are placeholders.

Configure the deployment

Application ID	`figaf-tool`, the route path. Lowercase, no spaces.
Apps domain	Detected from `cf domains`.
Image version	The latest Figaf image, detected from Docker Hub.
Memory	`3700M`, with 50 % as maximum heap. Plan the quota for it.
Location ID	Only for SAP PI/PO through SAP Cloud Connector, see the next page.
CF services	`figaf-db` and `figaf-xsuaa` are required. Tick `figaf-connectivity` and `figaf-destination` for PI/PO. Existing instances are reused, missing ones are created.

Keep the defaults unless Figaf advises otherwise, then choose **Start deployment**.

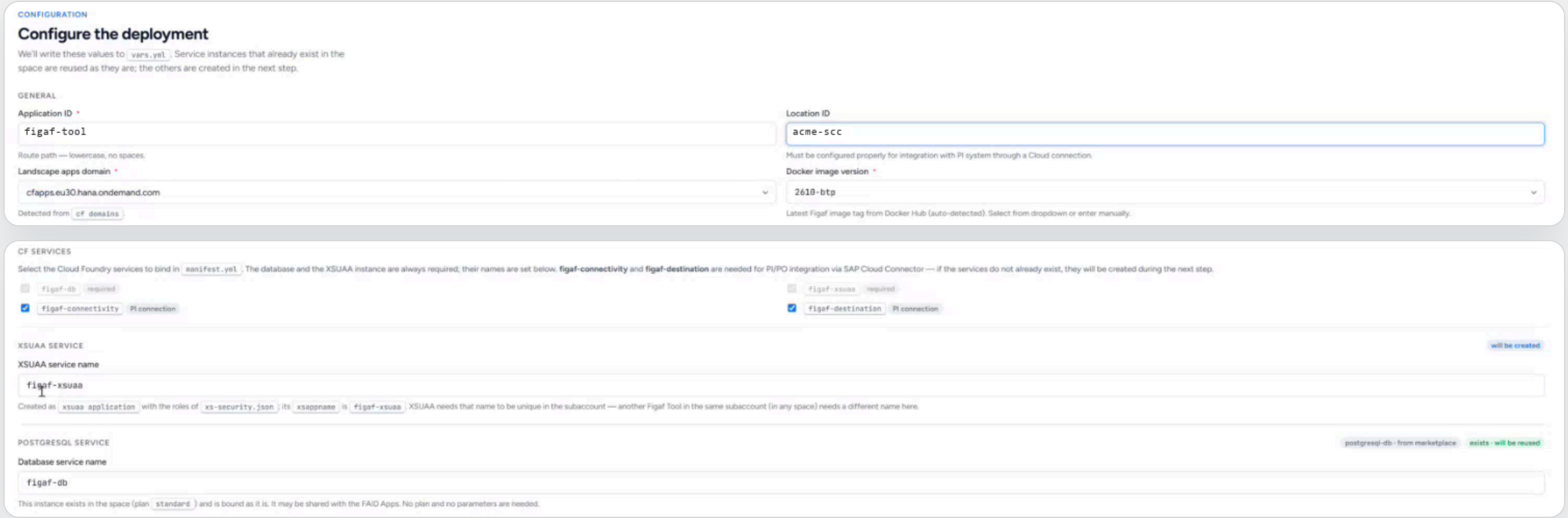
> **No secrets in variables**
>
> Anyone with Space Developer can read the environment variables.



The values go to `vars.yml`. Start deployment runs Provision, Deploy and Finish; the manager shows every command in its terminal drawer while it provisions and deploys.

Add the Cloud Connector Location ID

1. Find your Cloud Connector's **Location ID** in the BTP cockpit under **Connectivity → Cloud Connectors**.
2. In **Configure the deployment**, enter it under **Location ID**, spelled exactly as in the Cloud Connector.
3. Under **CF services**, tick `figaf-connectivity` and `figaf-destination`. Missing instances are created in the next step.
4. Running a second Figaf Tool in the same subaccount? Give it its own **XSUAA service name**.

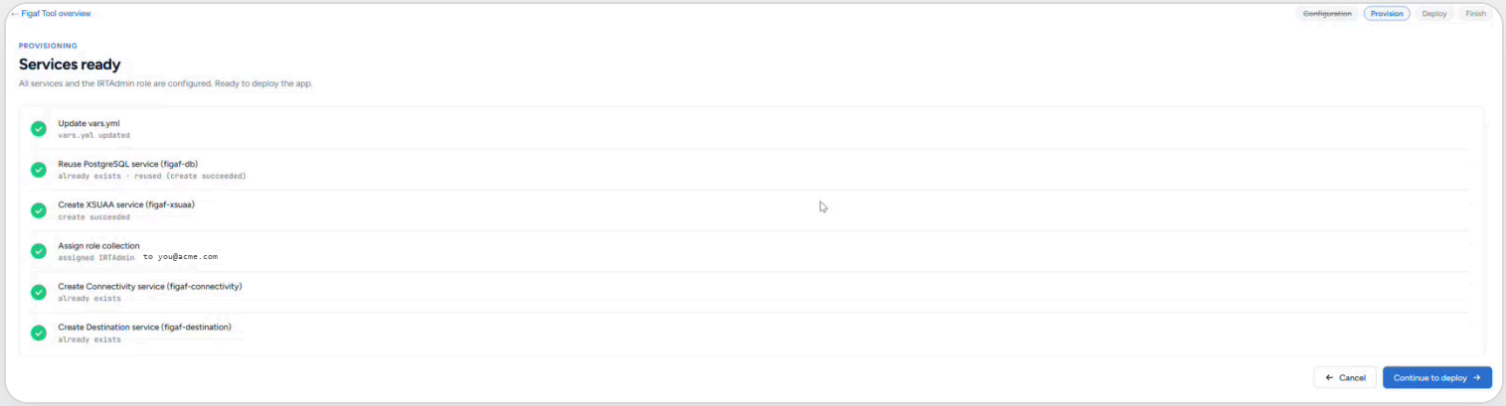


From the dry run. Names are placeholders. Without on-premise SAP PI/PO, leave the Location ID empty and the two services unticked.

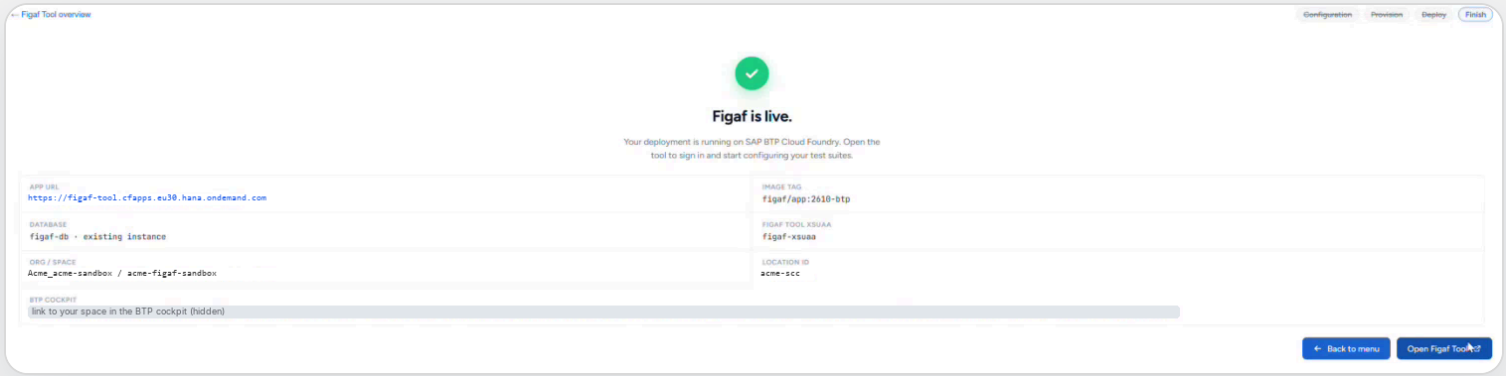
From Start deployment to "Figaf is live"

1. **Provision.** The manager updates `vars.yml`, reuses `figaf-db`, creates `figaf-xsuaa` and the PI/PO services if needed, and assigns the `IRTAdmin` role collection to you. Choose **Continue to deploy**.
2. **Deploy.** The manager pushes the Figaf Tool to Cloud Foundry. In the dry run this took about a minute.
3. **Finish.** *Figaf is live* shows the app address, the image and the services it uses. Choose **Open Figaf Tool**.

Every command runs in the terminal drawer at the bottom of the page, so you can follow what happens in your space.



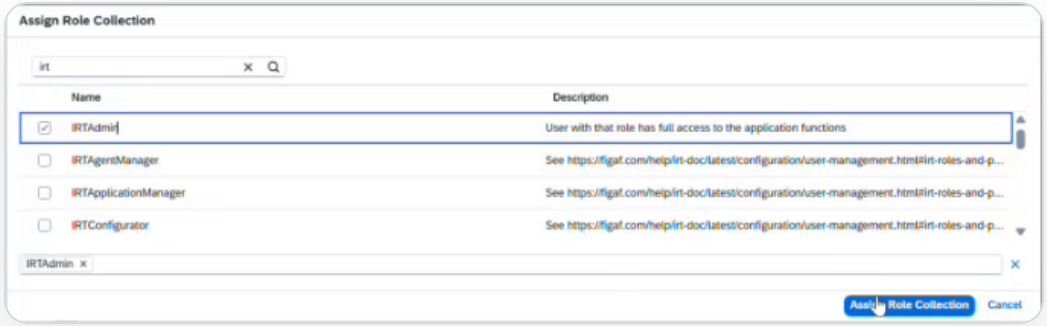
Provision done. The e-mail is a placeholder.



Finish, from the dry run in a sandbox space. Address and names are placeholders.

Open the Figaf Tool and load the license

1. Choose **Open Figaf Tool** on the finish page. *Access is denied?* Your browser still holds an older session: log in with another user, or open the address in a private window.
2. Give your colleagues the **IRT role collections** in the BTP cockpit: **Security → Users → Assign Role Collection**, for example `IRTAdmin`. You already have it.
3. In the Figaf Tool, open **Configuration → License**, paste the **License ID** you received from Figaf and choose **Load license**.
4. The page lists what the license covers and until when. Choose **Sync** after Figaf has changed your license.



BTP cockpit → Security → Users → Assign Role Collection.

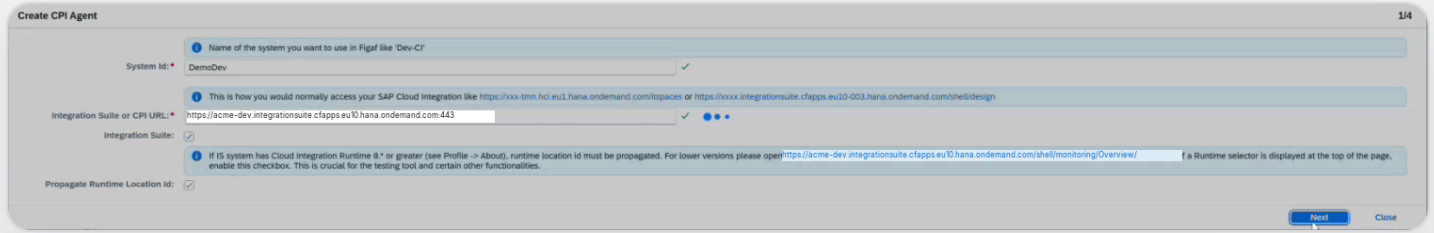


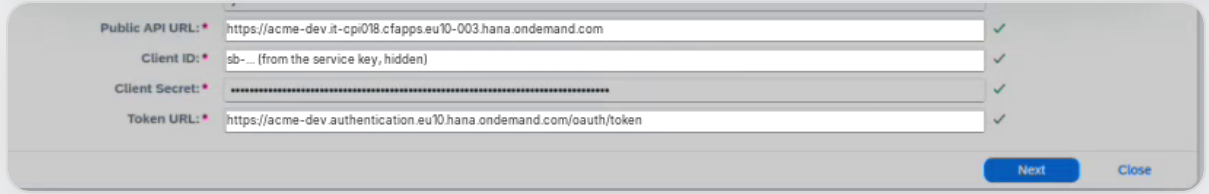
Configuration → License, after **Load license**. The ID and the company name are hidden.

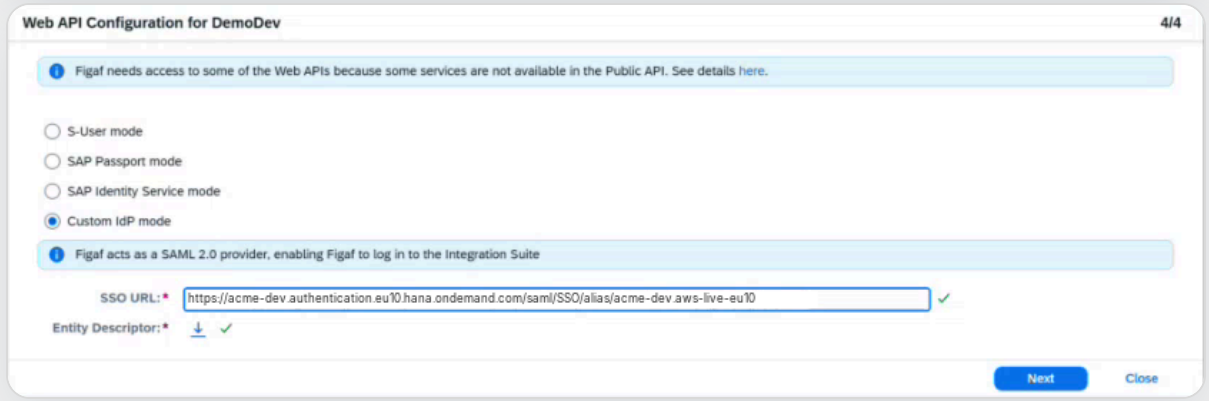
Create an agent for each Integration Suite tenant

An agent is a system the Figaf Tool connects to. Every agent you create here shows up in Figaf Manager under **Connections → SAP Integration Suite systems**.

1. **Configuration → Agents → Create CPI Agent.** Step 1/4: a **System ID** such as `DemoDev`, the tenant URL, **Integration Suite** and **Propagate Runtime Location Id** ticked.
2. Step 2/4, **Public API**: paste the service-key JSON of the Process Integration Runtime (plan `api`) and choose **Parse**. URL, client and token URL are filled in.
3. Step 3/4, **IFlow Messages API**: a key for sending test messages. Left empty in the dry run.
4. Step 4/4, **Web API**: choose **Custom IdP mode**, enter the tenant's **SSO URL** and **download the Entity Descriptor**. You need the file on the next page.







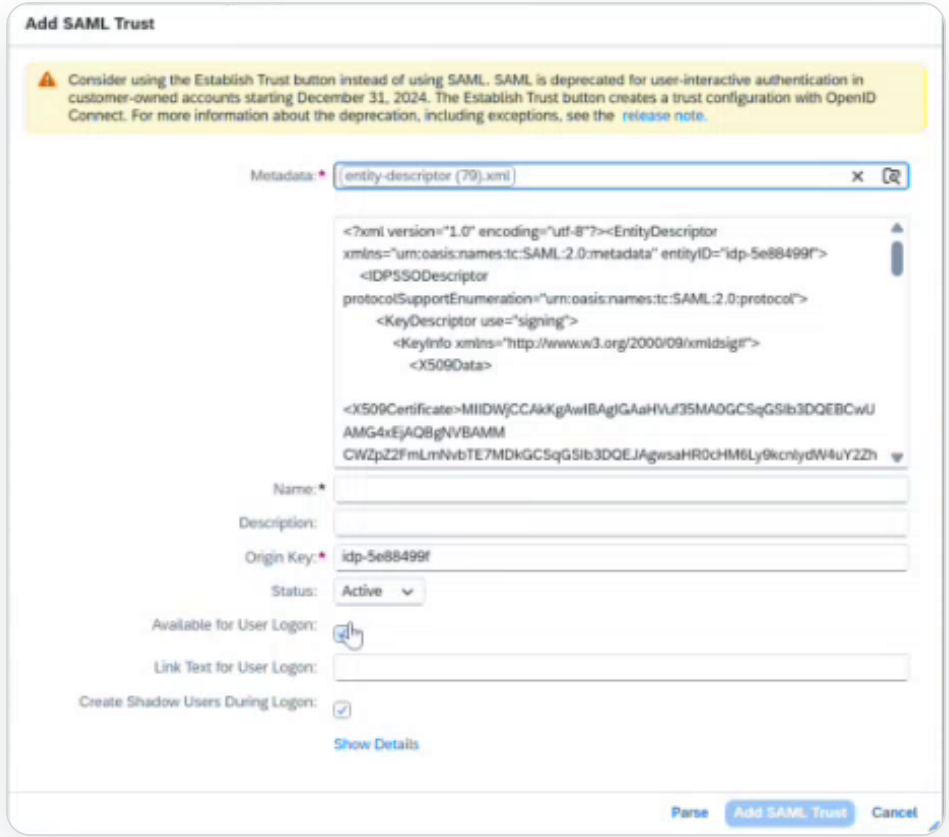
Steps 1, 2 and 4 of the wizard. Tenant addresses are placeholders; the client ID is hidden.

Trust the Figaf Tool as identity provider

1. In the BTP cockpit, open the **subaccount of your Integration Suite tenant → Security → Trust Configuration**.
2. Choose **Add SAML Trust** (the link above the list, not *Establish Trust*).
3. Under **Metadata**, upload the `entity-descriptor.xml` you downloaded in step 4/4 and choose **Parse**. The origin key is filled in.
4. Enter a **Name**, for example `figaf-sandbox`. Untick **Available for User Logon**: people never sign in with it. Keep **Create Shadow Users** ticked.
5. Choose **Add SAML Trust**.

> **Too many trusts?**
> A subaccount allows a limited number of trust configurations. In the dry run, old test trusts had to be deleted first.

SAP now recommends OpenID Connect for people's sign-in. This SAML trust is only for the Figaf Tool's technical access to the Integration Suite Web API.

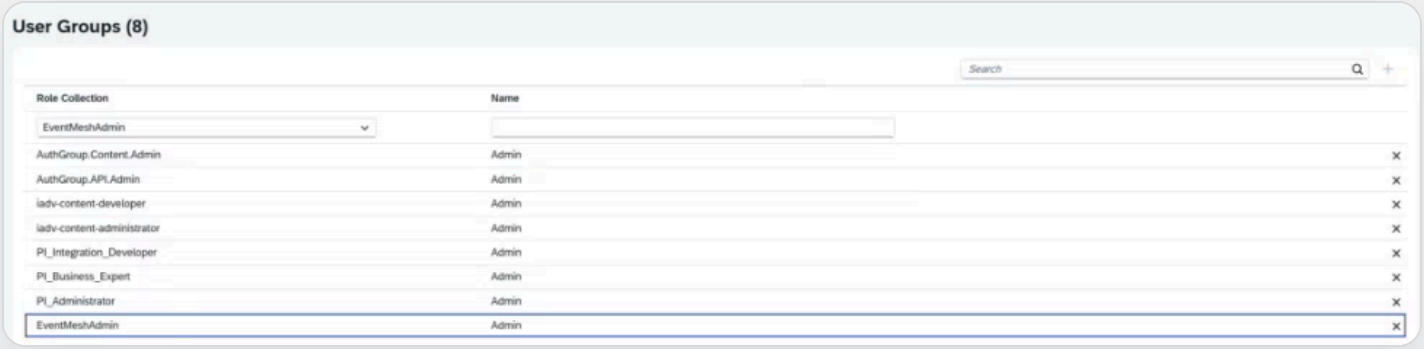


Add SAML Trust after **Parse**: the metadata from the Figaf Tool and the generated origin key.

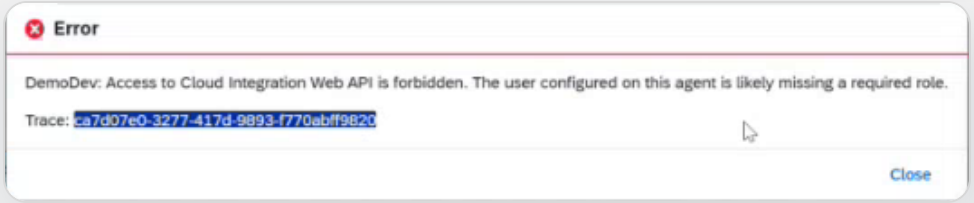
Map the role collections to the user group

1. Open the new trust → **Identity Provider** → **Edit** → **User Groups**.
2. Add one row per role collection, all with the group name `Admin`:

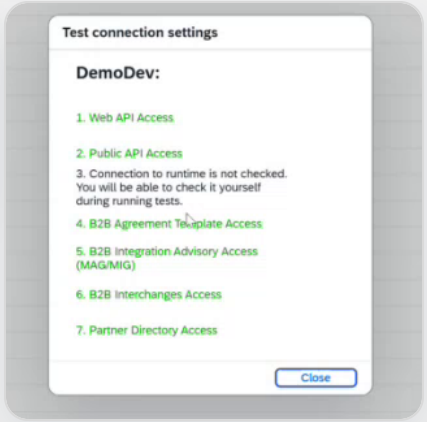
   `AuthGroup.Content.Admin` `AuthGroup.API.Admin` `iadv-content-developer` `iadv-content-administrator` `PI_Integration_Developer` `PI_Business_Expert` `PI_Administrator` `EventMeshAdmin`
3. Choose **Save**, go back to the agent in the Figaf Tool and finish the wizard.
4. Choose **Test configuration**. Every line must be green.



The eight role collections from the dry run. Role collections your landscape does not have are left out.



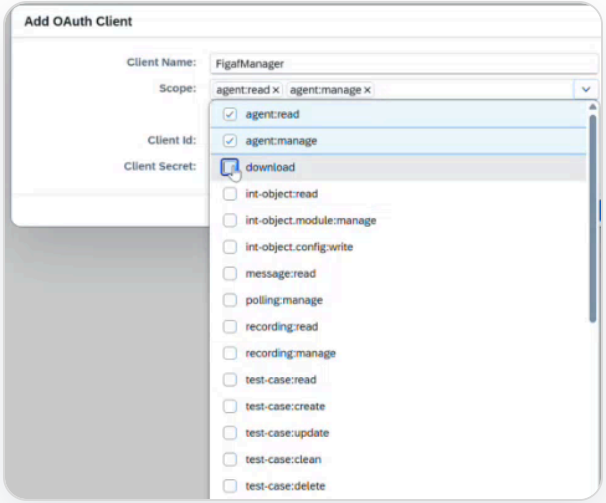
Without the user groups the Web API is forbidden.



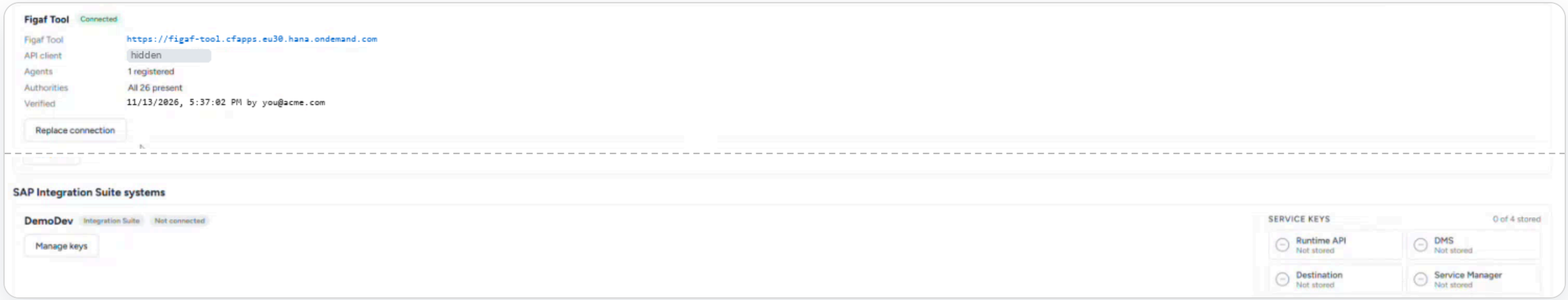
With them, all green.

Create the API client and connect the manager

1. In the Figaf Tool, open **Configuration → OAuth Clients → Add OAuth Client**. Name it `FigafManager`.
2. Under **Scope**, tick every authority that Figaf Manager lists in its Connect dialog, for example `agent:read`, `agent:manage`, `download`. Choose **Register** and copy the client ID and secret.
3. In Figaf Manager, open **Connections → Figaf Tool → Connect**, pick the deployment, paste the client and choose **Verify & save**.
4. Your agents now appear under **SAP Integration Suite systems**. Choose **Manage keys** and store at least the **Runtime API** key (step 5).



Add OAuth Client in the Figaf Tool.

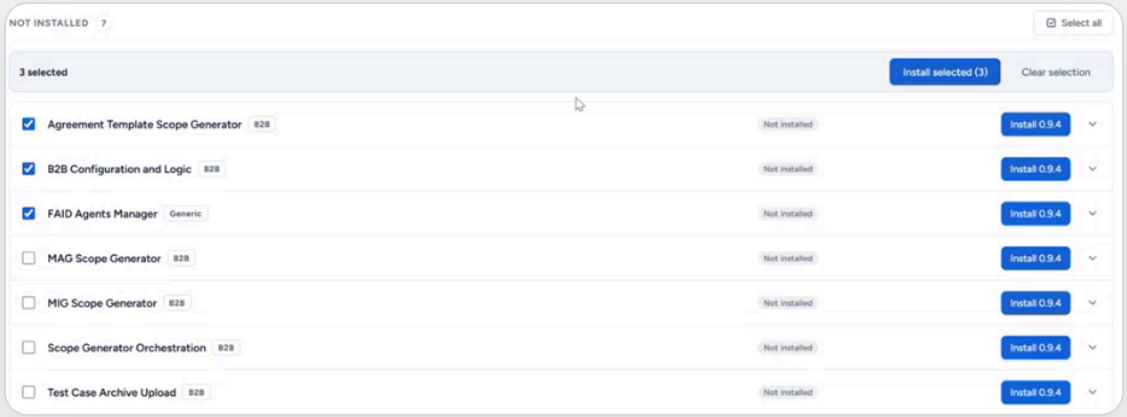


Figaf Manager → Connections: the Figaf Tool is connected with all authorities, and the agent `DemoDev` is listed as a system.

Install the first FAID Apps

1. Open **FAID Apps**. It lists every app in the release, grouped into installed and not installed.
2. Tick the apps you want and choose **Install selected**, or install one app with its own button.
3. The **shared backend** is deployed automatically with the first app.
4. From here you also start, stop, update and remove apps, and open them.

Filter by **Group** or **Area** to find the apps for your use case, for example B2B or Generic.

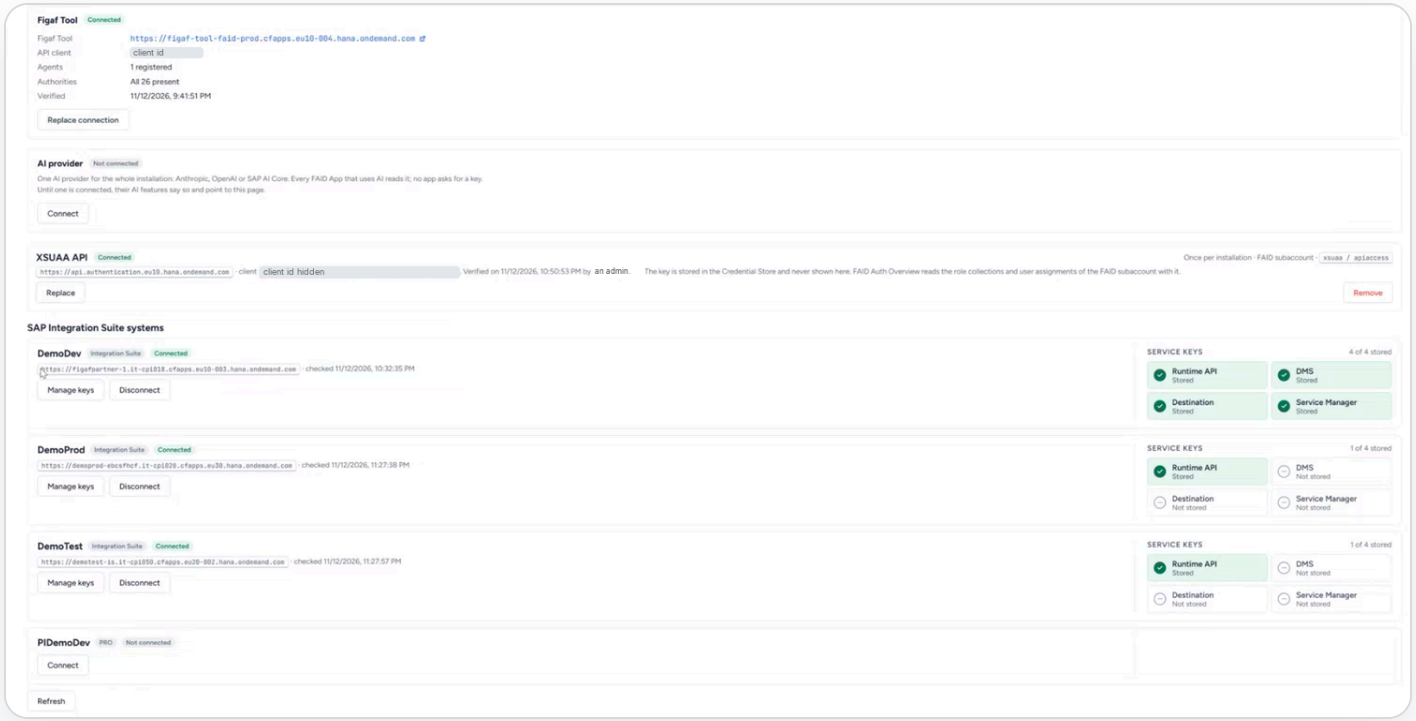


Three apps selected for installation in one go.

Connect the Figaf Tool and your systems

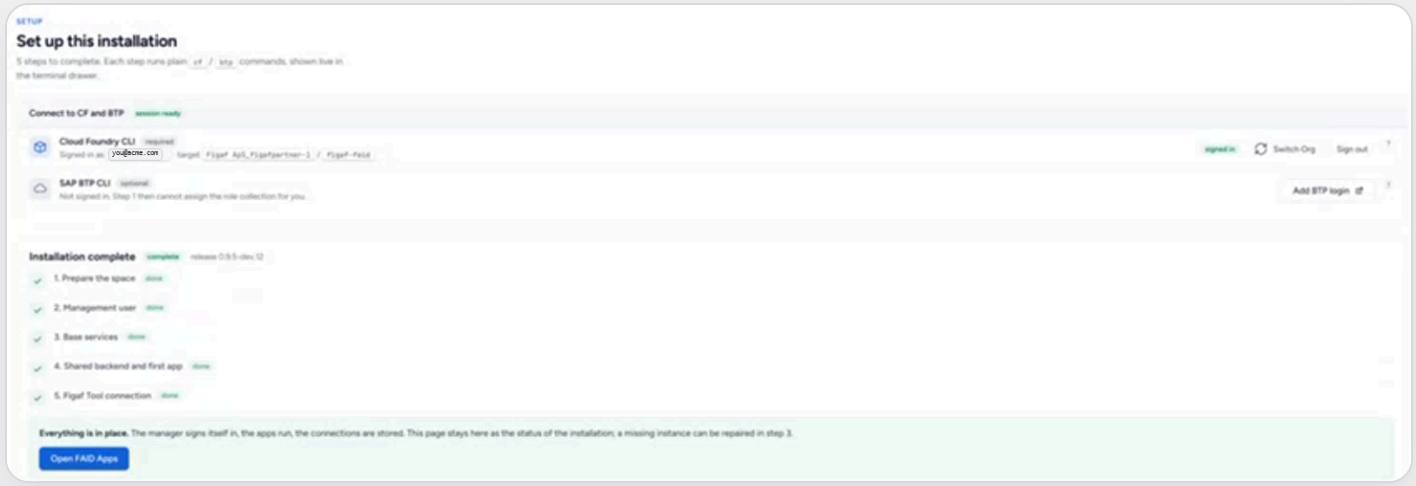
Figaf Tool	**required** Pick the deployment and enter its API client, once. Apps read the system list through the shared backend.
Integration Suite	Per tenant, the **Runtime API** service key **required**. DMS, Destination and Service Manager keys are optional.
XSUAA API	Once per installation, for the FAID Auth Overview app.
AI provider	**optional** Anthropic, OpenAI or SAP AI Core, for apps with AI features.
SAP PI/PO	**optional** Connected through the PI/PO services from step 1.

Keys are stored in the Credential Store and never shown again. No secret is ever typed into an app.



Connections, from a completed installation. IDs are hidden.

Installation complete



All five steps done. The Setup page stays as the status of the installation.

What is in place now

- Figaf Manager behind SAP sign-in, signing itself in with the management user.
- Database, roles and Credential Store in your space.
- The shared backend and your first FAID Apps, running.
- Connections to the Figaf Tool and SAP Integration Suite.

Next

- Assign app roles to your users in the subaccount.
- Repeat the installation in production when the sandbox is accepted.
- Check **About & updates** for new manager versions.

TROUBLESHOOTING

If a step stops

A red row always comes with a message. Open the terminal drawer at the bottom of the page for the exact command and its output.

YOU SEE	WHAT TO DO
Step 1: *does not offer the plan 'standard' … in this landscape*	The plan is not entitled. Assign it to the subaccount, or pick a plan the landscape offers, then choose **Try again**. Finished actions are skipped.
No `[SETUP] Token:` line, or the token is refused	The token is valid for 30 minutes and works once. Restart `figaf-manager` and use the newest line.
Passcode rejected	Codes work once and expire quickly. Choose **Get passcode** again and paste the new one immediately.
Unauthorized or forbidden after step 1	Sign out and in again, or open the new address in a private window. Check that `FAID-Manager-Admin` is assigned to you.
Step 2: *login check failed — wrong password, 2FA enabled, or the account is not registered*	Check the password, switch off two-factor authentication for the technical user, and make sure the account is registered with SAP.
Database is still being created	This can take up to about 15 minutes. Wait, then choose **Refresh** in step 3.
Step 3, or a FAID App that does not start: *Hostname/IP does not match certificate's altnames*	Known issue on Google Cloud landscapes. Step 3 is fixed in Figaf Manager 0.3.2: update under **About & updates**, then **Try again**. The fix for the shared backend is in progress; send the report to Figaf.
Figaf Tool: *BTP login required*	Sign in to the SAP BTP CLI under **Session & access** and choose the subaccount. If it persists, sign out and in again.
Figaf Tool: *Access is denied* on first open	Log in with another user or use a private window. Check that an IRT role collection is assigned to you.

CHECKLIST

Ready for the installation session?

Before the day

- [x] Subaccount with Cloud Foundry, sandbox first
- [x] PostgreSQL standard, Credential Store standard and XSUAA application entitled
- [x] Enough Cloud Foundry memory quota
- [x] Technical user registered, with password and without two-factor authentication
- [x] An installation name agreed, 3–20 characters
- [x] Integration Suite service keys at hand, at least the Runtime API
- [x] Figaf license at hand, if you deploy a new Figaf Tool

On the day

- [x] Space created, with you as Space Developer and Space Manager
- [x] Figaf Manager deployed and the setup token claimed
- [x] Steps 1 to 3 completed in Setup
- [x] New Figaf Tool deployed, if needed, with an API client
- [x] First FAID Apps installed
- [x] Figaf Tool and Integration Suite connected
- [x] New manager address bookmarked

FAID
FIGAF AI-DRIVEN

WE INSTALL IT WITH YOU

Questions before or during the installation?

Your Figaf contact joins the installation session and can help with entitlements, the technical user and any step that stops.

figaf.com

manager-download.figaf.com