abYsis Installation Guide

This guide covers installing and running abYsis from a standalone Docker image. The standalone image bundles the entire application and database into a single container with a single data volume, making it the simplest way to run abYsis on-premises.

Prerequisites

You need Docker installed on your machine. No other software is required.

Linux

Install Docker Engine following the official instructions for your distribution: https://docs.docker.com/engine/install/

After installation, ensure your user can run Docker commands:

```
sudo usermod -aG docker $USER
newgrp docker
```

You can also install Docker Compose to take advantage of our docker-compose.yml configuration

```
mkdir -p ~/.docker/cli-plugins
curl -SL https://github.com/docker/compose/releases/latest/download/docker-compose-linux-x86_64 \
  -o ~/.docker/cli-plugins/docker-compose
chmod +x ~/.docker/cli-plugins/docker-compose
```

macOS

Install Docker Desktop for Mac: https://www.docker.com/products/docker-desktop/

Download the version matching your Mac:

- **Apple Silicon (M1/M2/M3/M4):** Apple chip installer (arm64)
- **Intel Mac:** Intel chip installer (x86_64)

Important: The abYsis image architecture must match your machine. Use the `aarch64` tarball for Apple Silicon Macs and the `x86_64` tarball for Intel Macs.

Windows

Install Docker Desktop for Windows: https://www.docker.com/products/docker-desktop/

Docker Desktop requires either **WSL 2** (recommended) or **Hyper-V**. The installer will guide you through enabling WSL 2 if needed.

Important: Use the `x86_64` tarball for Windows machines with Intel/AMD processors.

Important: Ensure that the **Docker disk image location** is set to a storage with at least 80GB. The disk image location can be set in **Docker Desktop** by going to **Settings > Resources > Advanced** and choosing an appropriate location. If you are running abYsis on a Windows or Mac you will have to monitor that the host disk does not become full.

Step 1: Load the Image

You will have received a zip file like `abysis-standalone-v5-x86_64.zip` containing the distribution files. Extract it to somewhere suitable on your machine.

Open a terminal (Terminal on macOS/Linux, PowerShell on Windows) and navigate to the extracted directory containing the distribution files. Then load the image:

```
cd ./path/to/extracted/abysis-standalone-v5-x86_64
docker load -i ./abysis-standalone.tar
```

Replace the filename with the actual file you received. You should see output like:

```
Loaded image: abysis-standalone:latest
```

Step 2: Run abYsis

From the same directory, start abYsis using Docker Compose:

```
docker compose up -d
```

This will:

- Start abYsis in the background
- Make it accessible at http://localhost:8001
- Store all data in a Docker volume called `abysis_data`

The first startup takes a minute or two while the database initialises.

If you wish to start abYsis without docker compose then use this

```
docker run -d \
  --name abysis \
  -p 8001:80 \
  -v abysis_data:/data \
  --shm-size=512m \
  abysis-standalone
```

Step 3: Access abYsis

Once the container is running, open your web browser and go to:

```
http://localhost:8001
```

On first startup, allow a minute or two for the database to initialize. You can check the progress by viewing the container logs:

Command line:

```
docker logs -f abysis
```

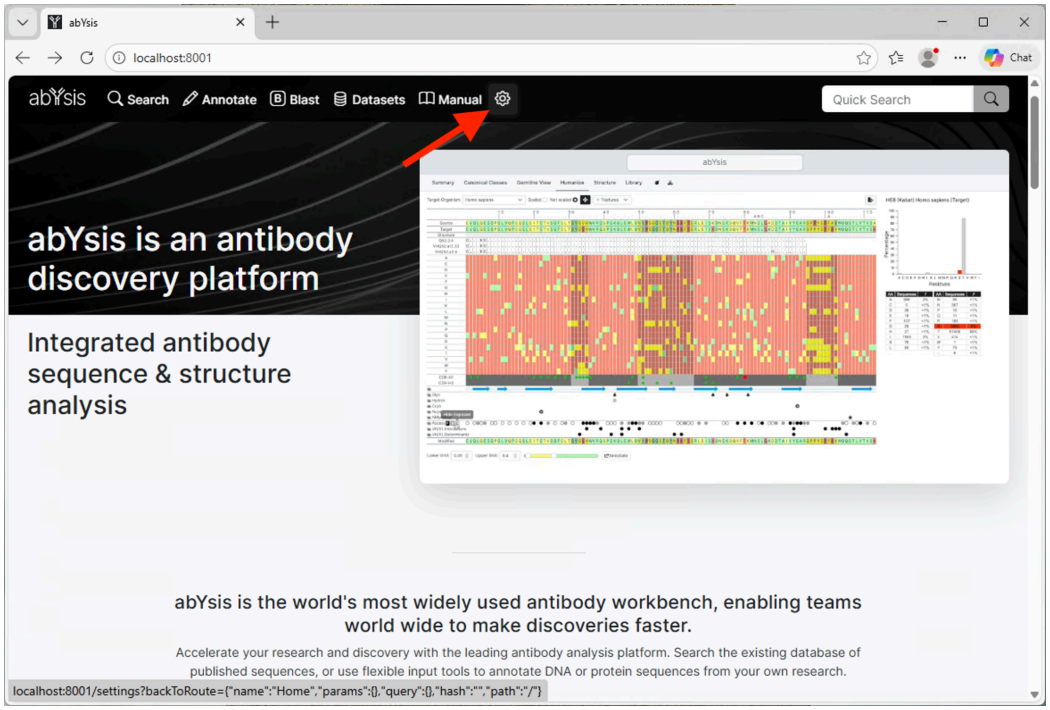
Look for the message `=== abysis is ready ===` which indicates the application is fully started.

Docker Desktop:

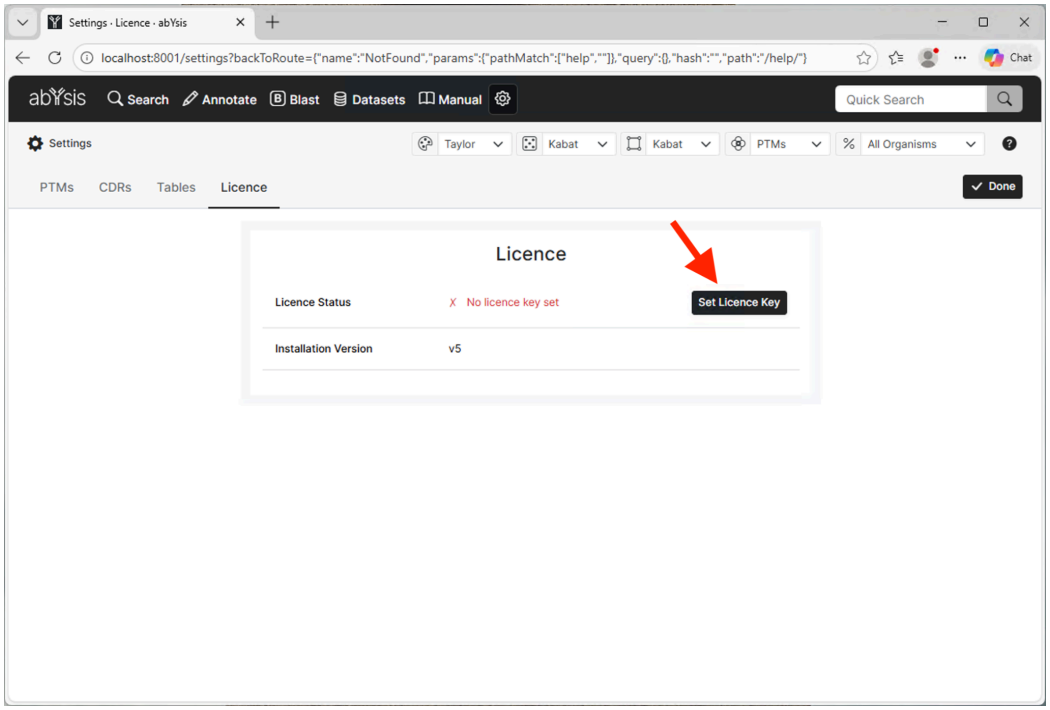
1. Go to the **Containers** tab
2. Click on the `abysis` container
3. The **Logs** tab shows the startup progress

Setting the License Key

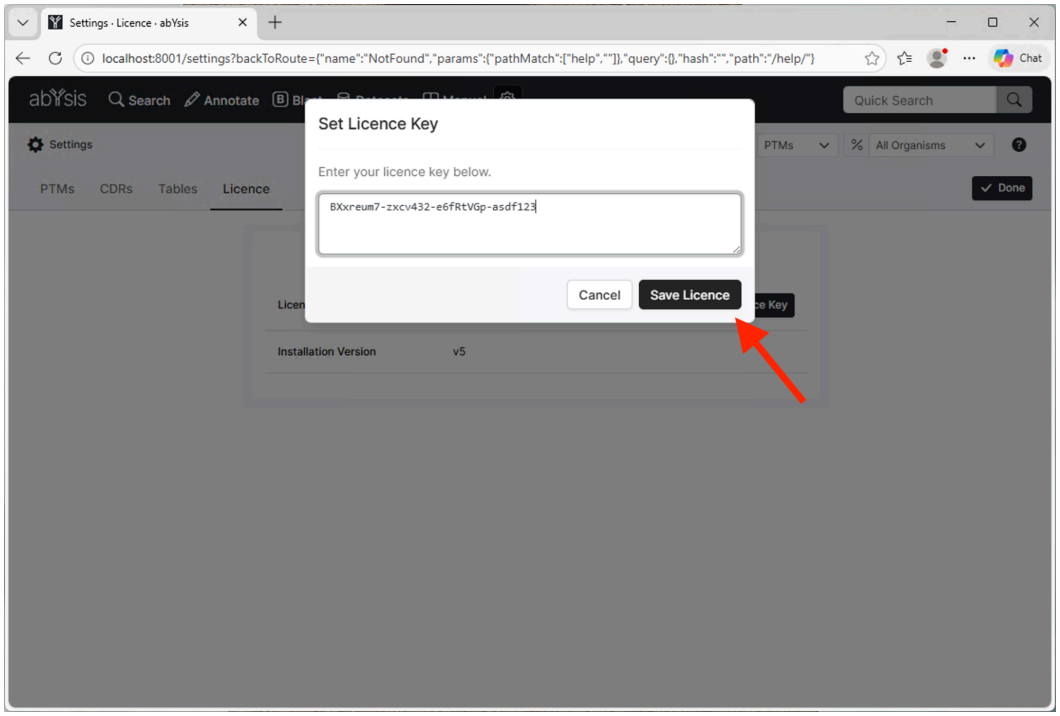
When you first start anYsis you will need to set the license key, this is done via the settings menu which is accessed by the betting button at the top of the seen



From the license status screen click “Set License Key”



Then enter it and press “Save License”



Once accepted the screen will refresh and you will be able to start using abYsis.

Configuration

The `docker-compose.yml` file can be edited to customise the deployment.

Changing the port

Edit `docker-compose.yml` and change the port mapping, or set the `ABYSIS_PORT` environment variable:

```
ABYSIS_PORT=9000 docker compose up -d
```

Environment variables

Environment variables can be added to the `environment` section of `docker-compose.yml`:

```
services:
  abysis:
    image: abysis-standalone
    shm_size: 512m
    ports:
```

```
  - "8001:80"
volumes:
  - abysis_data:/data
environment:
  - WEB_DIRECTORY=/abysis/
restart: unless-stopped
```

Variable	Default	Description
`WEB_DIRECTORY`	`/`	URL path prefix for the application
`SKIP_MIGRATIONS`	(unset)	Set to `true` to skip database migrations on startup

Managing abYsis

Stopping

To stop abYsis and remove the container:

```
docker compose down
```

Your data is preserved in the `abysis_data` volume and will be available when you start again.

Starting again

```
docker compose up -d
```

This creates a fresh container from the image and reattaches the existing data volume.

Restarting

To restart abYsis (e.g. after changing configuration):

```
docker compose down
docker compose up -d
```

Viewing logs

```
docker compose logs -f
```

Press `Ctrl+C` to stop following the logs (this does not stop abYsis).

Data Management

By default, all persistent data is stored in a Docker-managed volume named `abysis_data`. This includes both the PostgreSQL database and application files. The volume persists across stops, starts, and upgrades.

Using a custom data location

To store data in a specific directory on your host instead, edit `docker-compose.yml` and change the volume line to a bind mount path, and remove the `volumes:` section at the bottom of the file:

```
volumes:
  - /path/to/your/data:/data
```

Backup

```
docker run --rm -v abysis_data:/data -v "$(pwd)":/backup alpine \
tar czf /backup/abysis-backup-$(date +%Y%m%d).tar.gz -C /data .
```

On Windows (PowerShell):

```
docker run --rm -v abysis_data:/data -v "${PWD}:/backup" alpine `
tar czf /backup/abysis-backup.tar.gz -C /data .
```

Restore

```
docker compose down
docker run --rm -v abysis_data:/data -v "$(pwd)":/backup alpine \
sh -c "rm -rf /data/* && tar xzf /backup/abysis-backup-YYYYMMDD.tar.gz -C /data"
docker compose up -d
```

Upgrading

When you receive a new version:

1. Load the new image:

   ```
   docker load -i abysis-standalone-v5-x86_64.tar
   ```

2. Restart the container:

```
docker compose down
docker compose up -d
```

Your data volume is preserved automatically. The new container will run any pending database migrations on startup.

Removing everything

```
docker compose down -v
docker rmi abysis-standalone
```

Troubleshooting

Container won't start

Check the logs for error messages:

```
docker compose logs
```

Port already in use

Change the port by setting `ABYSIS_PORT`:

```
ABYSIS_PORT=9000 docker compose up -d
```

Or edit the port mapping in `docker-compose.yml`.

Architecture mismatch

If you see a message about platform mismatch, you have the wrong image for your machine:

- **Intel/AMD machines** (most Windows PCs, older Macs): use the `x86_64` image
- **Apple Silicon Macs** (M1/M2/M3/M4): use the `aarch64` image

Checking disk space

```
docker system df -v
```