Building Teamwork Cloud containers

$body

These procedures are intended for users with Docker knowledge. We do not support Docker, just provide guidelines on how to use it with our product. For more information about Docker, see https://docs.docker.com/

Example package

- TWC-Container-Example-Pkg.zip

This page provides an example package for building Cassandra and Teamwork Cloud components from a Teamwork Cloud No-Install package. The example package uses Docker Compose to build all the components required to deploy a fully-functional Teamwork Cloud containerized environment. Such an environment can be deployed in an orchestration framework to enhance scaling and management.

Teamwork Cloud Container Example Package Components

Docker Hub Pulled Images:

- Cassandra 4.0.3
- Zookeeper 3.7.0
- ZooNavigator 1.1.1 (elkozmon)
- Elasticsearch 7.16.2

External Packages:

- Teamwork Cloud 2022x No-Install Linux (downloaded from the 3DS software download page)
- ActiveMQ Artemis 2.26.0 (downloaded automatically after executing the containerization environment staging script)

Example with Docker Compose

The example package requires at least 32GB of available RAM.

1. Download and extract the example package into an empty working directory.
2. Download a 2022x or newer version of Teamwork Cloud No-Install Linux package.
3. Rename or symbolically link the no-install package to *twcoudsuite.zip*.
4. Execute the *config.tw-compose.sh* script to stage the containerization environment.
5. Run the following command to initialize Docker Compose container configurations:

```
docker compose build
```

6. Run the following command to launch containers:

```
docker compose up -d
```

   If you execute this command for the first time, it will build the containers as well.
7. Use the following URLs:
   - To access web UI:

```
https://host.docker.internal:8443/webapp
```

   - To connect from a modeling tool client (with default port 3579):

```
host.docker.internal
```

   - To access REST API/Swagger Page:

```
https://host.docker.internal:8111
```

You can apply the Teamwork Cloud license via the web UI or REST API.

Helpful commands

To follow the output (tail) the output of a specific container:

```
docker compose logs -f <Container Name or ID>
```

To stop all containers but retain data in non-persistent storage:

```
docker compose stop
```

To stop all and remove all running containers (non-persistent storage data will be lost):

```
docker compose down -v
```