

Tamino

Tamino with Docker

Version 10.11

November 2021

This document applies to Tamino Version 10.11 and all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 1999-2021 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA, Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <http://softwareag.com/licenses>.

Use of this software is subject to adherence to Software AG's licensing conditions and terms. These terms are part of the product documentation, located at <http://softwareag.com/licenses/> and/or in the root installation directory of the licensed product(s).

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at <http://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with Software AG.

Document ID: INS-DOCKER-1011-20211101

Table of Contents

Preface	v
1 About this Documentation	1
Document Conventions	2
Online Information and Support	2
Data Protection	3
2 Prerequisites for Building a Docker Image	5
3 Building Tamino Docker Images	7
4 Bring your own License	9

Preface

Docker is an open-source technology that allows users to deploy applications to software containers. A Docker container is an instance of a Docker image, where the Docker image is the application, including the file system and runtime parameters.

You can create a Docker image containing one runnable Tamino database and then run the Docker image as a Docker container. To facilitate running Tamino in a Docker container, Tamino provides a script to use to build a Docker image and then load or push the resulting Docker image to a Docker registry.

Support for Tamino with Docker 18 and later is available on Linux and UNIX systems for which Docker provides native support.

For details on Docker and container technology, see [Docker documentation](#).

This Guide covers the following topics:

Prerequisites for Building a Docker Image
Building Tamino Docker Images
Bring your own License

1

About this Documentation

■ Document Conventions	2
■ Online Information and Support	2
■ Data Protection	3

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
Monospace font	Identifies service names and locations in the format <i>folder.subfolder.service</i> , APIs, Java classes, methods, properties.
<i>Italic</i>	Identifies: Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
Monospace font	Identifies: Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the symbol.
[]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information and Support

Software AG Documentation Website

You can find documentation on the Software AG Documentation website at <https://documentation.softwareag.com>.

Software AG Empower Product Support Website

If you do not yet have an account for Empower, send an email to empower@softwareag.com with your name, company, and company email address and request an account.

Once you have an account, you can open Support Incidents online via the eService section of Empower at <https://empower.softwareag.com/>.

You can find product information on the Software AG Empower Product Support website at <https://empower.softwareag.com>.

To submit feature/enhancement requests, get information about product availability, and download products, go to [Products](#).

To get information about fixes and to read early warnings, technical papers, and knowledge base articles, go to the [Knowledge Center](#).

If you have any questions, you can find a local or toll-free number for your country in our Global Support Contact Directory at https://empower.softwareag.com/public_directory.aspx and give us a call.

Software AG Tech Community

You can find documentation and other technical information on the Software AG Tech Community website at <https://techcommunity.softwareag.com>. You can:

- Access product documentation, if you have Tech Community credentials. If you do not, you will need to register and specify "Documentation" as an area of interest.
- Access articles, code samples, demos, and tutorials.
- Use the online discussion forums, moderated by Software AG professionals, to ask questions, discuss best practices, and learn how other customers are using Software AG technology.
- Link to external websites that discuss open standards and web technology.

Data Protection

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

2 Prerequisites for Building a Docker Image

Prior to building a Docker image containing a Tamino database, you must complete the following:

- Install Docker client on the machine on which you are going to install Tamino and start Docker as a daemon. The Docker client should have connectivity to Docker server to create images.
- Install Tamino on a Linux system using the instructions in Installing Software AG products.

3 Building Tamino Docker Images

To run a Tamino database in a Docker container Tamino provides the script `create_dockerimage.sh` located at `<INSTALL_ROOT>/Tamino/v107/files` to build the Docker images. Start the script by running the command `./create_dockerimage.sh <INSTALL_ROOT>`.

Per default, a RedHat UBI18 docker base image is pulled from the Docker server upon which the Tamino docker image is built. Adapt the script if a different base image is required. For more information of the referenced image refer to [RedHat Developer Site](#). Software AG is not responsible for the contents of this image.

To start a database within the thus started Docker container either create a new database when starting the container or provide a backup.

To create a new Tamino database pass the name of the database to-be to the container by using the environment variable `CREATE_DB`: `docker run -d -p 3250:3250 --name inocontainer -e CREATE_DB=<dbname> taminoimage`.

To create a Tamino database from an existing backup make sure the backup file has the same owner id and group id as the user that runs the database in the docker container (by default: 1724:1724) and mount the backup file into the directory `/backup` at the new container: `docker run -d -p 3250:3250 -v <backupfile.1B0>:/backup/<backupfile.1B0> --name inocontainer taminoimage`. Specify the absolute path for the existing backup. You can provide a new name for the database using the environment variable `CREATE_DB`.

To shut down the Docker container execute the shutdown script inside the Tamino container, i.e. `docker exec -it inocontainer /bin/bash -c "/shutdown.sh"`.

To persist Tamino database container it is possible to mount your Tamino database containers into the mountpoint `/data` with `docker run -d -p 3250:3250 --name inocontainer -v TaminoDB:/data taminoimage`. Tamino administration is available via `docker exec -it inocontainer /bin/bash`. `docker exec -it inocontainer /bin/bash` lands at the system folder level. To execute `inoadmin` it is required to go to `/Tamino/v107/bin`.

4 Bring your own License

You can copy your own license into the volume `/license` of the Tamino container.

- Make sure the license file has the same owner id and group id as the user that runs the database in the docker container. By default this is 1724:1724.
- Mount the license file into the directory `/license` at the new container via `docker run -d -p 3250:3250 -v <licensefile.xml>:/licenses/<licensefile.xml> taminoimage`.
- Start a shell inside the Tamino container with `docker exec -it inocontainer /bin/bash`.
- Use `regutil` to apply the new license.

