Tamino

Setting Up Tamino

Version 10.15

October 2022

WEBMETHODS

This document applies to Tamino Version 10.15 and all subsequent releases.

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

Document ID: INS-AFTERINSTALL-1015-20230910

Table of Contents

Preface

Tamino is installed using the Software AG Installer. Please refer to the Software AG Installer documentation, which you can find at **http://documentation.softwareag.com/**.

Before You Start Using Tamino

- **Linux Customization**
- **Configuring Tamino Access**
- **Other Issues**
- **Next Steps**

1 About this Documentation

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
`Monospace font`	Identifies service names and locations in the format *`folder.subfolder.service`*, APIs, Java classes, methods, properties.
`Italic`	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

Product Documentation

You can find the product documentation on our documentation website at **https://documentation.softwareag.com**.

In addition, you can also access the cloud product documentation via **https://www.softwareag.cloud**. Navigate to the desired product and then, depending on your solution, go to "Developer Center", "User Center" or "Documentation".

Product Training

You can find helpful product training material on our Learning Portal at **https://knowledge.softwareag.com**.

Tech Community

You can collaborate with Software AG experts on our Tech Community website at **https://tech-community.softwareag.com**. From here you can, for example:

- Browse through our vast knowledge base.
- Ask questions and find answers in our discussion forums.
- Get the latest Software AG news and announcements.
- Explore our communities.
- Go to our public GitHub and Docker repositories at **https://github.com/softwareag** and **https://hub.docker.com/publishers/softwareag** and discover additional Software AG resources.

Product Support

Support for Software AG products is provided to licensed customers via our Empower Portal at **https://empower.softwareag.com**. Many services on this portal require that you have an account. If you do not yet have one, you can request it at **https://empower.softwareag.com/register**. Once you have an account, you can, for example:

- Download products, updates and fixes.
- Search the Knowledge Center for technical information and tips.
- Subscribe to early warnings and critical alerts.
- Open and update support incidents.
- Add product feature requests.

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 Before You Start Using Tamino

Linux Customization

After installing Tamino, ensure that the shared memory limit is at least 512 megabytes. To check the current value, examine the value of the Linux kernel parameter `shmmax` as follows:

```
cat /proc/sys/kernel/shmmax
```

To set `shmmax` to a value of 512 megabytes (which should be sufficient for medium-size databases), enter the following command:

```
echo "512000000" > /proc/sys/kernel/shmmax
```

We recommended setting the variable `shmmax` to the value of the size of the physical memory in bytes. You can find out how much memory there is in your machine with

```
cat /proc/meminfo
```

For very large databases, it might be necessary to increase the value of the parameter `shmmax`.

Note that this change is not persistent, so after a reboot the parameter will have the old (default) value. In order to change this, add the following line to the */etc/sysctl.conf* script:

```
kernel.shmmax = 512000000
```

To ensure that */etc/sysctl.conf* is parsed after a reboot, enter the following command:

```
/sbin/chkconfig boot.sysctl on
```

Configuring Tamino Access

This document contains information about accessing Tamino. The simplest means of accessing a Tamino database is via HTTP in webserverless mode. For this purpose it suffices to set the database parameter `communication mode` to `HTTP` and the parameter `HTTP port` to an unused port of your choice.

Also, client applications can access a Tamino database using HTTP via a web server.

To specify that a particular database may only be accessed through a particular web server, define the web server as described below and assign it to a particular database via the Web Servers object in the tree-view frame.

The following topics are covered:

Overview of Web Servers

Besides accessing Tamino via HTTP in webserverless mode, client applications can access Tamino using HTTP via a web server. Each web server that is required to provide access to Tamino must be configured accordingly.

Note: Tamino version 4.1 or later can also be used without a web server. This is done by some Tamino APIs and tools, for example.

If you use a web server, it does not need to be located physically on the machine on which Tamino is installed; it can be located anywhere in the network where Tamino is installed. For example, it is possible to have Tamino running on a Windows machine and the web server running on a UNIX machine in the same network or vice versa.

A single web server can be configured to access Tamino servers on several machines.

Apache

General

The Tamino interface is a DSO (dynamic shared object: Apache terminology for DLL/shared library), so your Apache must have DSO support. DSO support is included in the default configuration of Apache. Apart from this, there are no special requirements for Tamino. For Apache configuration, tuning etc., it makes no difference whether the requested resources are in a file system or in a Tamino database.

Source Distribution

The Tamino interface is provided in binary form; these binary files work for current standard Apache distributions. However, under certain circumstances, it may be desirable or necessary to compile and link the Tamino interface. Possible reasons are:

- Your Apache is built with compile/link options that are incompatible with the provided binary files.
- You prefer to use open source software.

The Tamino installation provides the Apache interface in the directory *X_Port\Apache*. This directory contains the source code, the necessary header files and link libraries, and an example *makefile* that can be adapted to the current platform. If the platform supports both 32-bit and 64-bit versions, the appropriate link libraries can be found in corresponding subdirectories.

Configure Apache

Apache must be configured as described below.

Note: We strongly recommend that you make a backup copy of the Apache configuration file *httpd.conf* before proceeding.

Copy the file *Apache22ModuleIno.dll* (Windows) or *Apache22ModuleIno.so* (UNIX, Linux) to your Apache *modules* directory. For Windows additionally copy the libraries icudtsag538.dll, icuucsag538.dll, sagovo5.dll, sagrgs5.dll, sagsmp2.dll and sagxts5.dll from <INSTALL_ROOT>\Tamino\v107\bin to your Apache modules directory.

Add the following to the Apache *httpd.conf* file:

```
LoadModule  ino_module modules/Apache22ModuleIno.dll

<Location /tamino>
SetHandler ino
InoRegfile "<INSTALL_ROOT>/Tamino/cfg/regfile"
</Location>
```

This results in an attempt to establish a connection between Tamino and the web server using XTS. When an XTS directory server is not used and instead a flat-file XTS directory is activated using the XTSDIR setting, the XTSDIR environment variable must be defined for the Apache web server process (default setting: XTSDIR = <INSTALL_ROOT>\Tamino\cfg). If this fails, e.g. because the server property `communication method` has been set to "TCP/IP", the local configuration information is searched for the XML port of the database. In this case a native TCP/IP connection is established, if the database is local.

Note: If you use Apache 2.4.*x*, the name of this file is *Apache24ModuleIno*.

If the Apache web server and the Tamino server are not on the same machine, then in order to enable the Apache web server to access Tamino databases via native TCP/IP, add the following section to the *httpd.conf* file for each database:

```
<Location /tamino/mydb>
SetHandler ino
InoHost xyz.abc.de
InoTcpipPort 4711
</Location>
```

Replace *mydb* by the name of the database, replace *xyz.abc.de* by the name of the host where the database server is running, and replace *4711* by the number of the database's XML port, which you can find via Tamino Manager (in the **Properties** dialog of the database).

Configuring Apache for Authorization in Conjunction with Tamino

Note that it is not possible to use Apache *.*htaccess* files (“distributed configuration files”) in conjunction with Tamino to specify authorization information. These files are typically located in the same directory as the files to be read, but there is no such directory in the case of Tamino. This problem can easily be circumvented by specifying exactly what would otherwise be the content of your *.*htaccess* file in a <Location> directive in your *httpd.conf* file, for example:

```
<Location /tamino>
# Put here exactly the contents of your *.htaccess file
</Location>
```

Apache Portability Runtime Library Name

On UNIX (including Linux) systems, Tamino's Apache interface is linked against the Apache portability runtime (APR) library (*libapr.so*). The name of this library can be customized in Apache. If *libapr.so* has a different name in your Apache (e.g. a versioned name like *libapr-0.so*), please create a file system link between the real name and *libapr.so*, or use the source distribution (see above).

Microsoft IIS

This section describes installation and configuration aspects of Tamino and Microsoft Internet Information Services (IIS). The following topics are covered here:

- Installation of modiis.dll as a Filter in IIS
- Correcting the Configuration of Microsoft IIS if HTTP Error 403 (read access forbidden) Occurs when Trying to Access Tamino in Conjunction with IIS

Installation of modiis.dll as a Filter in IIS

To install *modiis.dll* as a filter in IIS

1 Ensure that IIS is installed and ISAPI Filters and ISAPI Extensions are activated. In order to make the IIS Manager available, follow the steps below depending on the specific platform.

On Windows:

Go to "Start > Control Panel > Programs and Features" and click "Turn Windows features on or off".

Go to "Internet Information Services > World Wide Web Services > Application Development Features" and mark "ISAPI Extensions" and "ISAPI Filters".

Go to "Internet Information Services > Web Management Tools" and mark "IIS Management Console". Click "OK".

On Windows Server

Go to "Start > Control Panel > Programs and Features" and click "Turn Windows features on or off". The Server Manager is started.

Mark "Roles" and click "Add Roles" in the action pane.

Mark "Web Server (IIS)", click "Next" and, on the introduction page, click "Next" again. That opens the list of role services for IIS.

Mark "ISAPI Extensions" and "ISAPI Filters" under "Application Development" and "IIS Management Console" under "Management Tools".

Click "Next" and then click "Install" to install the selected role services.

The "World Wide Web Publishing Service" should now be started. Expect the IIS Manager to be available under "Start > Control Panel > Administrative Tools > Internet Information Services (IIS) Manager".

2 Go to "Start > Control Panel > Administrative Tools" and start "Internet Information Services (IIS) Manager".

3 Ensure that a virtual directory inoScripts exists for the desired site. If it does not exist create it by right-clicking on the desired site and activate the "Add Virtual Directory" menu item.

4 Copy the file modiis.dll from the directory \X_Port\IIS folder of your installed Tamino environment to the directory to which the IIS virtual directory inoScripts points to.

For Tamino versions 9.12 and higher modiis.dll is no longer self-contained (statically linked with all required libraries). Therefore it is required to copy six additional libraries from "<INSTALL_ROOT>/Tamino/v107/bin" to the inoScripts directory, namely icudtsag538.dll, icuucsag538.dll, sagovo5.dll, sagrgs5.dll sagsmp2.dll, and sagxts5.dll.

5 Add modiis.dll as an ISAPI filter: Click on the desired site in the IIS manager. Double-click "ISAPI Filters" and activate the "Add" item in the action pane.

Enter "Tamino" as Filter Name and the modiis.dll of the scripts directory (C:\Inetpub\inoScripts) as executable.

6 Set the "execute" permission for the ISAPI filter:

Click on the desired site in the IIS manager.

Double-click "Handler Mappings".

Select "ISAPI-dll" and activate the "Edit Feature Permissions" item in the action pane.

Check the Execute box and click "OK".

"ISAPI-dll" is now enabled.

7 Allow the modiis.dll ISAPI Extension to run on the web server:

Click on the top-level local machine name in the IIS manager.

Double-click "ISAPI and CGI Restrictions" and activate the "Add" item in the action pane.

Enter the modiis.dll of the scripts directory (C:\Inetpub\inoScripts) as ISAPI or CGI path, "Tamino" as Description.

Check the Allow extension path to execute box and click "OK".

Restarting the IIS web server enables it to access Tamino databasess using an XTS directory server.

If XTS communication is to be used in absence of an XTS directory server, but instead using a flat-file XTS directory activated using the XTSDIR setting, then in order to enable an IIS web server to access Tamino databases via XTS, locate the following registry key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Software AG\Tamino\IIS Mapping\Prefix
```

Under this key you may have a list of keys for the individual path prefixes you want to map (for each Tamino database you want to access, enter a key `tamino/`*`dbname`*, where *`dbname`* is to be replaced by the name of your database). This sub key has the following subordinate string values:

- "Location" and its value is the actual path prefix;
- "XTSDIR" and its value is the path to the flat-file XTS directory (see TaminoInstallation Guide).

Example of database *`mydb`* (in regedit format):

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Software AG\Tamino\IIS Mapping]
[HKEY_LOCAL_MACHINE\SOFTWARE\Software AG\Tamino\IIS Mapping\Prefix]
[HKEY_LOCAL_MACHINE\SOFTWARE\Software AG\Tamino\IIS Mapping\Prefix\tamino/mydb]
"location"="tamino/mydb"
"XTSDIR"="C:\SoftwareAG\Tamino\cfg"
```

Note that the forward slash between the strings “tamino” and “mydb” is correct and necessary, because *tamino/mydb* is part of a URL.

After changing the registry, the World Wide WEB Publishing Service must be restarted (item **Services** in the Windows control panel).

Correcting the Configuration of Microsoft IIS if HTTP Error 403 (read access forbidden) Occurs when Trying to Access Tamino in Conjunction with IIS

HTTP Error 403 (read access forbidden) may occur due to incorrect configuration of IIS. You can verify this situation and correct the configuration as follows:

To verify and correct the configuration for using Tamino with IIS

Verification: To test this, enter a Tamino-URL in the browser, e.g. *http://MYHOST/tamino/myDB?_diagnose=ping*. If HTTP error 403 occurs and you are using the Microsoft Internet Information Server (IIS) web server, you should proceed as follows:

1 Open the Microsoft Management Console (MMC) in order to configure the IIS;

2 With your host name, open “Default Web Site”;

3 Click the right mouse button on “inoScripts”;

4 Change the properties of this folder;

5 Make sure that under “Virtual Directory” the “Permissions” are set to “Execute”.

Note: This value may be set to “Scripts” e.g. by the “Microsoft Lockdown” utility.

Tomcat

A standalone Tomcat server cannot be used to communicate to X-Machine via HTTP. Programs running under Tomcat can use the webserverless interface instead.

Activating Web Server Authentication Mechanisms

This section discusses how to activate the authentication mechanisms of the various web servers that Tamino supports.

- For Apache
- For Microsoft IIS

For Apache

The following example is an extract of a suitable Apache configuration file *httpd.conf* to set the authentication mechanism for all databases:

```
# ------------------------------------------------------------------
# Enable Apache Web Server Authentication for Tamino Security
# Note: The AuthUserFile is created with pgm ../apache/bin/htpasswd.exe
#       Enter command 'htpasswd -h' for pgm usage description
#       Restart Apache Web Server after modification
# If authorization fails:
# - Check Apache error.log file
# - Enter browser command http://localhost/tamino/secdb?_diagnose=echo

<Location /tamino>
AllowOverride AuthConfig
AuthName "Tamino"
AuthType Basic
AuthUserFile "bin/userids"   <== file created with htpasswd.exe
require valid-user
</Location>
```

It is also possible to set the authentication mechanism for individual databases. To do so, add the database name to the location:

```
<Location /tamino/mydb>
```

For Microsoft IIS

Activate the checkbox **Integrated Windows Authentication** in the "Authentication Methods" dialog box.

Caution: Do not activate the checkbox "Basic Authentication (password is sent in clear text)", because this option can cause severe problems.

To get to the "Authentication Methods" dialog box, choose the **File Security** tab in the "modiis.dll Properties" window of the IIS, then choose **Edit** in the "Anonymous access and authentication control" section.

Other Issues

To enable clients to locate Tamino databases by name, even in a distributed environment, Tamino uses the XTS (eXtended Transport Service) directory service.

XTS provides a uniform mechanism for Software AG products to communicate with each other across diverse platforms using multiple protocols and different types of hardware.

XTS offers two methods of maintaining the directory:

1. **Software AG directory server**

   A standalone service, included in the distribution as an installable package

2. **Flat file implementation**

   Can be used if all participants are on the local machine. The flat file implementation is used under the condition that an environment variable XTSDIR is defined and contains a valid path pointing to a directory where the flat file is created and maintained by XTS.

Starting with Tamino v8.0 the default mechanism for XTS was changed from the Software AG directory server to the flat file implementation.

That means that for each client accessing a Tamino database the XTSDIR environment variable must be defined and contain the same value as the database server uses (the XTSDIR setting of the Tamino 10.7 databases can be found in the registry under *HKEY_LOCAL_MACHINE\SOFTWARE\Software AG\Tamino\10.7\environment*).

The script useXTSDIR.[cmd|sh] on <INSTALL_ROOT>/Tamino/v107/bin can be used to set the XTSDIR default value (parameter: enable) or to remove the XTSDIR entry in the registry (parameter: disable).

"Client" in this case means the instance addressing the database server via XTS. For example, if Tamino is accessed via an Apache web server then the XTSDIR environment variable must be defined for the Apache web server process.

However, for replication, due to the fact that the databases reside on different nodes, it is necessary to use the Software AG directory server.

To use XTS with the Software AG directory server, Tamino must be able to resolve SAGXTSDShost and SAGXTSDSport to IP addresses (if SAGXTSDSport cannot be resolved, a default value is used). This can either be done by a DNS server (or by entering the two well-known names in your local hosts file).

The entries in the hosts file are as follows:

```
<IP address of the master node>         SAGXTSDShost
x.y.0.0                                 SAGXTSDSport
```

where *master node* is the node where the directory of the Extended Transport Service is running.

The port number is calculated as 256 * x + y. Thus, for example, if the port number should be set to 12731, then SAGXTSDSport must be defined as 49.187.0.0 (12731/256 = 49, remainder 187).

This information is provided in case you want to custom configure the `SAGXTSDSport`.

Note: Tamino registers with XTS using host names, not IP addresses. For a remote client to resolve the host name into an IP address, the host name either has to be present in DNS or an entry for that host name must be entered in the client's local *etc/hosts* file.

Next Steps

You have now completed the installation of Tamino and set up the environment, and are ready to start using Tamino. The next steps are described in the section *Getting Started*.

Index