software AG

Tamino

High Availability

Version 10.7

May 2021

Table of Contents

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

Software AG Documentation Website

You can find documentation on the Software AG Documentation website at **https://documenta-tion.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 TECHcommunity

You can find documentation and other technical information on the Software AG TECHcommunity website at **http://techcommunity.softwareag.com**. You can:

- Access product documentation, if you have TECHcommunity 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   **High Availability**

Tamino is often part of a mission-critical infrastructure. Downtime of the infrastructure means downtime of the underlying business. One example: A major European airport uses Tamino in a high availability scenario - one hour of downtime would mean a severe interruption of the entire European air traffic network. This, of course, would also have a heavy business impact.

An increasing number of applications require such a high degree of availability. Clients cannot afford to wait to get mission-critical information. Applications have to provide a continuous service. Tamino addresses these requirements by providing support for high availability configurations.

In the following sections, you will find information about high availability in general and how it is implemented in Tamino. Also, you are pointed to other parts of the documentation that contain relevant information.

High-availability data processing configurations are necessarily associated with considerably higher costs and complexity than standard configurations. Issues such as licensing costs for standby CPUs and professional services consultancy should be taken into consideration. It is not possible to cover all the technical implications within the scope of this documentation. Software AG experts will be pleased to advise you in all aspects of your implementation. Please contact your nearest Software AG account manager or the professional services consultant assigned to you for further information.

This chapter is subdivided into the following topics:

What is High Availability?

This section will first give you some general background information about high availability. In the second part, the possibilities that Tamino offers with regard to high availability are introduced

- Definition
- High Availability in Tamino

Definition

Generally, high availability is defined as the ability of a system to perform its function continuously (without interruption) for a significantly longer period of time than the reliability of its components would suggest. As high availability cannot easily be quantified, it can be considered as a design goal rather than an actual design of a system. It is a trade-off between the cost of downtime and the cost of the protective measures that are available to avoid or reduce downtime. With increasing investment, availability increases also.

Several technological aspects need to be considered for providing high availability:

1. Good system administration practices

2. Reliable backups

3. Disk and volume management

4. Reliable network

5. Reliable local environment

6. Good client management

7. Error-free services and applications

8. Failover

9. Replication management

10. Procedures for disaster recovery

High availability is achieved by applying the technologies from the bottom up - the more aspects considered, the higher the availability of your system. The trend is clear: Reaching higher levels of availability will cost more money, and the amount of money it takes to get from one level to the next level increases.

High Availability in Tamino

What does high availability mean with regard to Tamino? High availability depends on many factors, such as the system environment, the application requirements and the balance between cost versus risk. The strategy for achieving high availability aims at eliminating single points of failure in a system.

Improving high availability is more than just implementing one failover solution for Tamino. Tamino provides several solutions for the different building blocks of availability mentioned above that need to be considered for improving availability:

- *Good system administration*

  Good system administration practices are the foundation of other high availability technologies. For Tamino, system administration is done with the inoadmin command line utility. Detailed information about the Tamino Manager can be found in the *Tamino Manager documentation*. For high availability aspects, take a look at the following sections:

  - *inoadmin Features*

  - *Tamino Security*

- *Internal and external backups*

  Backups prevent loss of data. Tamino provides several possibilities for backing up your data. For detailed information about backups in Tamino, refer to the *Backup Guide*.

- *Failovers*

  This guide deals with system failures and failover situations. It describes the basic hardware and software configuration as well as the actions that are taken if a failure occurs.

- *Replication*

In Tamino, replication is always asymmetric (a one-sided master/slave relationship) and asynchronous (to avoid conflicting updates, the replication database reflects changes in the data with a certain time delay only). This means that the replication database must be made available as the master database manually (in a high availability environment, this is done automatically). Loss of data is possible because the replication is done asynchronously, and hence the last transactions may not be recovered. For detailed information, refer to the *Replication Guide*.

■ *Disaster Recovery*

In Tamino, disaster recovery stands for the restoration of a Tamino database when everything is lost except for a backup of the database and recovery files. This may be the case after a total loss of the computing environment, in which backup and recovery files are still available on an archive. Detailed information of how to proceed in this situation can be found in the section *Disaster Recovery* of the *Backup Guide*.

Failures and Failover

Failures

Tamino usually runs in a very complex environment, which consists of many components. All of them can fail individually, leading to a failure of larger components, and in the worst case, of the whole environment. Points of failure could be:

■ network wiring, network routers, network switches, network bridges

■ power supply, air conditioning

■ CPUs, memory, disk drives, disk controllers, network controllers, fans

■ application software, web servers, operating systems.

Failover

Failover means that there is a second system configured as a standby system for the primary one. In case of a failure, the second system can take over: After a short interruption in service, all the functions of the primary system are accomplished by the secondary system. The take-over process is done automatically. Downtime is limited to the takeover time needed for the migration. Users should not be aware of the failover process.

Basic Architecture of a Failover System

To avoid failure and loss of data, Tamino should be built into a cluster system which reduces the amount of time that an application is unavailable. The basic architecture of a high-available failover system is shown in the following graphic. It consists of (a minimum of) two nodes, a disk shared between the two nodes, and a heartbeat connection connecting the two nodes. This setup is controlled by a cluster software. Each of the two nodes is identified by its own IP address. The high availability cluster is accessed by the clients, using a third IP address. This IP address is attached to only one node at a time under the control of the cluster software:
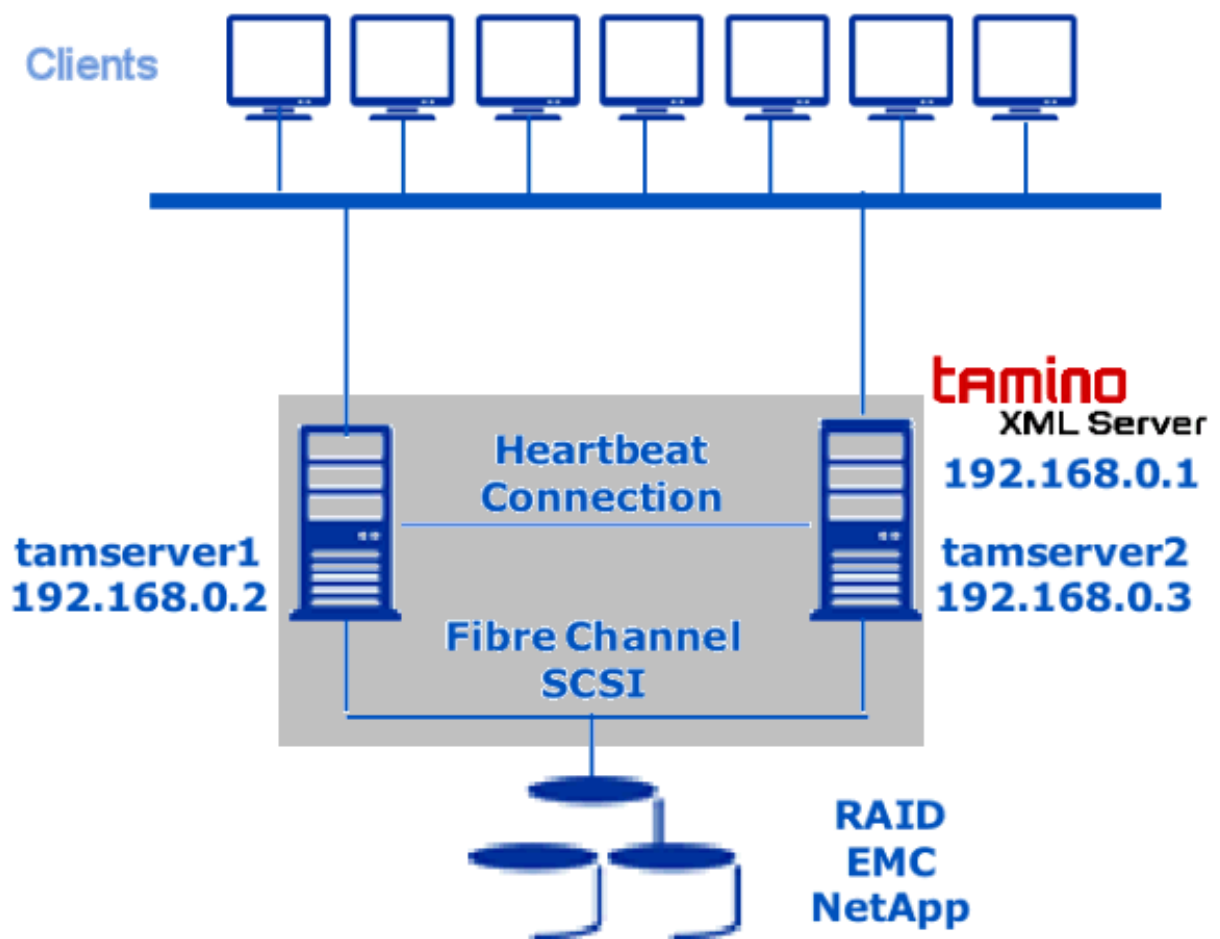
Basic Failover Configuration

The basic components are:

- the *clients*, which are connected to at least two servers (in this example tamserver1 and tamserver 2). The clients are not directly connected to the servers, but to the IP address of the high availability cluster. Thus in a failover situation, the clients are not influenced by the switch from one server to another.

- at least two *nodes* (tamserver1 and tamserver2), used for failover of the Tamino XML Server.

- a *heartbeat connection* between the two nodes, which is used by the cluster software to monitor the status of the servers by sending network packets through the connection.

- *shared disks*, which must be RAID for securing the availability of data, connected to the servers by fiber channel or SCSI.

Tamino XML Server is installed on both nodes (two licenses are required). The disks of the RAID array are mounted either on node 1 or on node 2, but never on both nodes at the same time. The Tamino database containers are located on the RAID array. For the clients, the two-node architecture is a "black box". They are connected to the virtual IP address of the high availability cluster, and not to a specific node in the cluster. The cluster software is installed on the servers and uses plug-ins for monitoring, starting and stopping the Tamino XML Server.


The Failover Process

Once you have established a basic architecture for failover situations in Tamino and a failure occurs, the software handles the situation automatically.

Basically, the failover process is simple: The cluster software detects a problem on tamserver1. It terminates all parts of Tamino on node 1. It also releases all resources needed by Tamino (disk, mount, IP address). Then, on tamserver2, the cluster software re-establishes the needed resources and starts Tamino. After a failover, the database recovers to a consistent state of the database.

However, clients need to be aware of the fact that during a failover, the Tamino database session is lost and there might be a timeout response during the failover period. Client programming should be adapted in a way that the corresponding response codes are not handled as an unrecoverable failure.

The following graphics give a more detailed description of the process.

Example: Web Server fails
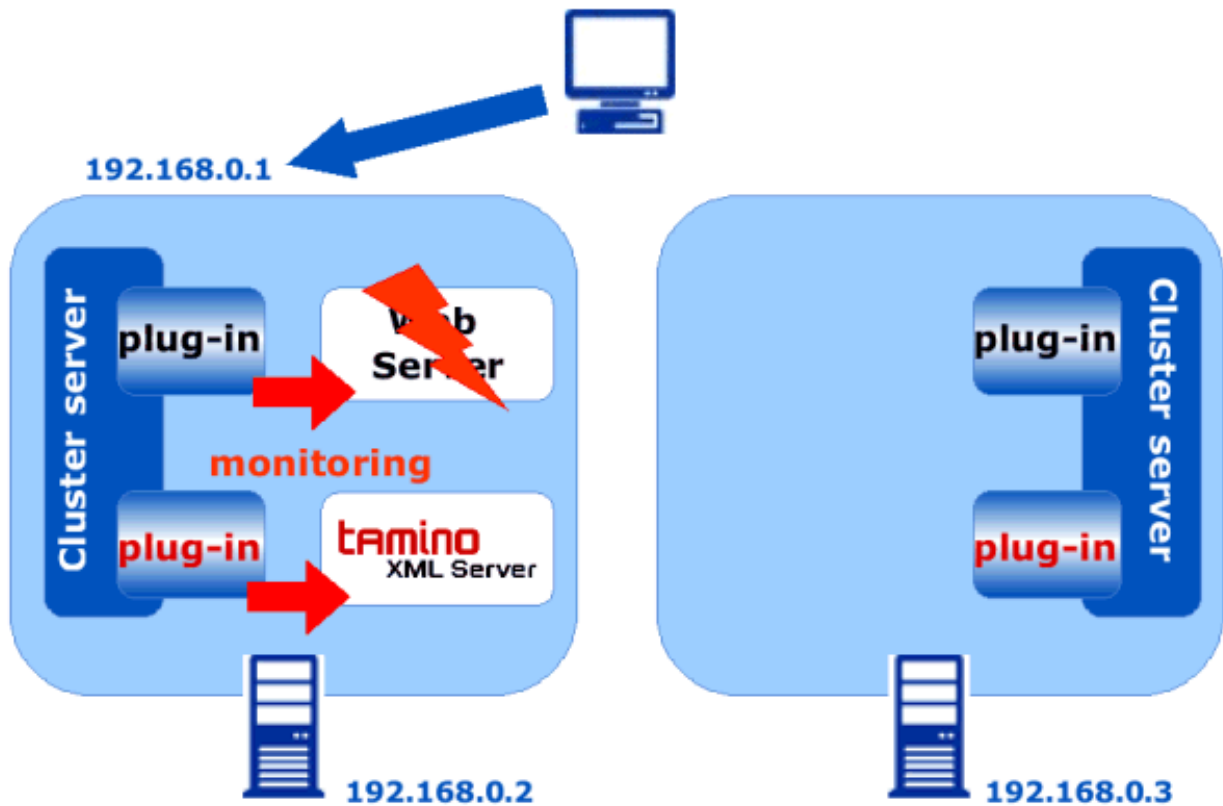
A cluster server uses plug-ins to control the software, in this case the web server and Tamino XML Server, which are installed on node 1. Controlling means that the cluster plug-ins can start, stop and monitor the software. The client is connected to the virtual IP address of the high availability cluster. A failure occurs if one of the components is no longer available, for example due to an hardware error. In this example, the web server fails.

Failover: Software handles the process
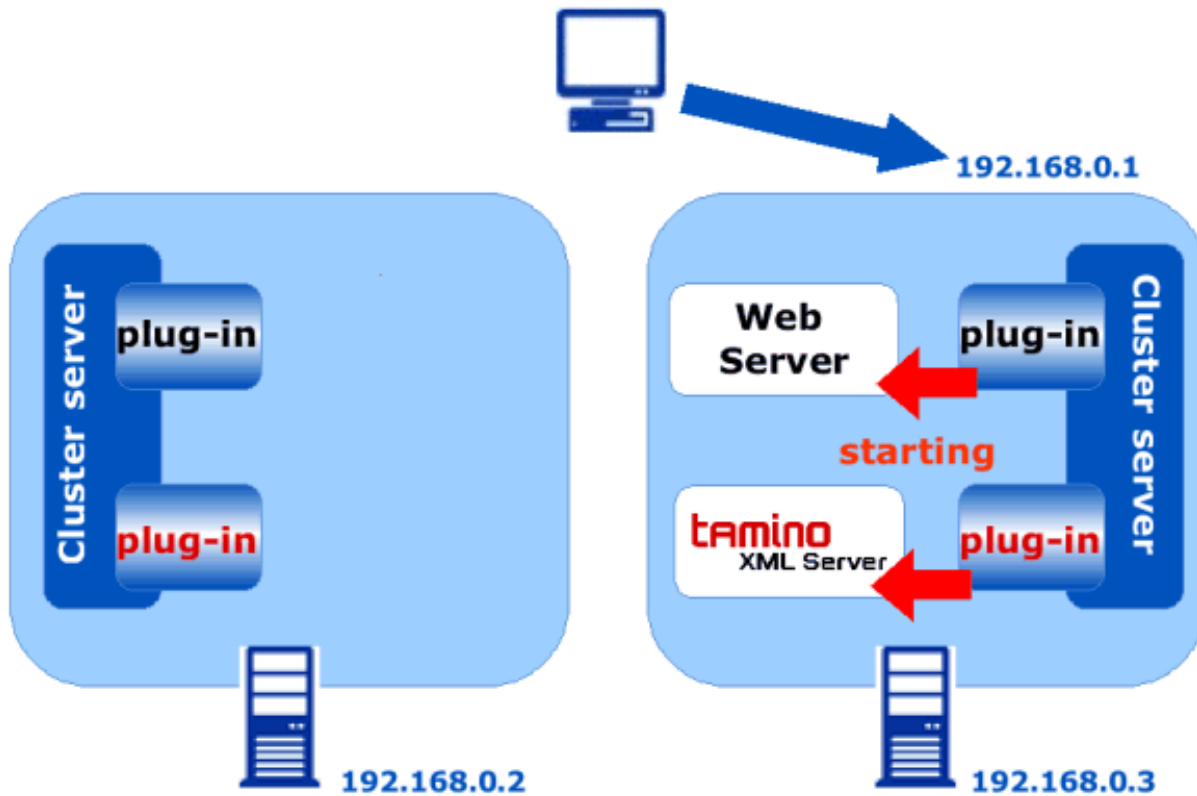
As the cluster server continuously monitors the software, failures are immediately recognized. In the case of a failure (if, for example, the web server fails), the cluster server stops both the web server and the Tamino XML Server. The IP address is switched from node 1 to node 2. Thus the applications are taken over by node 2, without the clients realizing that a switch has occurred (in the best case). The cluster server starts the web server, as well as the Tamino XML Server. Work can be resumed. The failover is done automatically and includes the following actions:

- switch of IP address
- unmounting/mounting of disks
- stopping and starting Tamino-related daemons and processes
- stopping and starting all other processes related to the application using Tamino.

High Availability Software

Cluster software monitors the health of the hardware and the software in a cluster. Tamino supports the major third-party cluster software products with respective plug-ins. These plug-ins allow an integrated control of Tamino servers in the context of the cluster software. The Tamino plug-ins are part of the Tamino Enterprise License.

Third Party Software

Tamino High Availability supports Veritas Cluster Server, providing a high availability kit for Solaris and Windows. For the details on supported Veritas Cluster Server versions and operating system versions contact your software supplier. Other high availability software (such as HP Service Guard or IBM HACMP) might be supported on request and under special conditions. Please contact your software supplier if you require support for other high availability software.

Tamino High Availability Software

The Tamino Enterprise kit contains a Tamino High Availability kit. The major components of this kit are:

■ **Tamino High Availability Monitor (inoham)**

The Tamino High Availability Monitor (an executable named *inoham*) includes all the functionality required to build a failover solution for Tamino. It provides features like starting, stopping, monitoring and cleaning a Tamino XML Server in a high availability environment. The Tamino High Availability Monitor can also be used to build failover solutions for non-supported high availability cluster software.

■ **Configuration Files**

These are the files defining the Software AG components for a specific high availability cluster software.

■ **Agents**

These are Perl scripts containing the logic for starting, stopping, monitoring and cleaning Tamino XML Server and the required Software AG components (like administration). These scripts can be used as a basis for integrating Tamino XML Server in non-supported environments.

Summary

High availability is defined by the demands placed on the system by its users, and it is met through appropriate protection methods by its administrators to deliver application availability. Tamino offers several possibilities to achieve high availability, among them good system practices, backups, failover and replication. Failover, as discussed in this guide, comprises a special hardware architecture and the corresponding third party software.

Index

C

cluster software
    high availability, 13

D

downtime, 5

F

failover
    high availability, 8
failover architecture, 9
failover process, 10
failure
    high availability, 8
failure of web server, 11

H

high availability, 5

I

inoham, 13

R

reliability, 5