

The Siemens logo is displayed in a white rectangular box in the upper left corner of the page. The logo consists of the word "SIEMENS" in a bold, teal, sans-serif font. The background of the entire page is a low-angle, upward-looking photograph of a modern glass skyscraper with a teal-tinted facade, set against a bright blue sky with scattered white clouds.

**SIEMENS**

# Installation Guide for Teamcenter based products

# Contents

**Preface** 5

**Introduction** 1-1

## Supported Environment

<b>Teamcenter Compatibility Matrix</b>	2-1
<b>Web Browser</b>	2-1
<b>Install Hosts and Locations</b>	2-1
<b>Sizing Considerations</b>	2-2
Minimum CPU and Memory Requirements	2-2
Job Pool memory and storage size	2-3
Calculate Log File Storage Size	2-3
<b>Operating Systems</b>	2-4

## Admin UI

<b>Administrative User Interface</b>	3-1
<b>Admin UI Troubleshooting</b>	3-3

**The Active Integration (PL4x) Architecture** 4-1

## Installation Instructions

<b>Overview of Installation Steps</b>	5-1
<b>Active Integration (PL4x) Installation</b>	5-1
Active Integration (PL4x) Installer Introduction	5-1
Install and Configure the PL4x BGS Using PL4x Installer	5-2
Install and Configure the PL4x GS Using PL4x Installer	5-5
<b>Active Integration (PL4x) Configuration Using Admin UI</b>	5-6
Configure Active Integration (PL4x) BGS Using BGS Admin UI	5-6
Configure Active Integration (PL4x) GS Using GS Admin UI	5-10
<b>Configure Teamcenter Environment for Active Integration (PL4x)</b>	5-13
Deploy Active Integration (PL4x) GS Template with TEM	5-13
Set PL4x GS Environment for a Teamcenter 2-Tier Environment	5-15
Set PL4x GS Environment for a Teamcenter 4-Tier Environment	5-16
Add PL4x Error Message Texts to Teamcenter	5-16
<b>Set Teamcenter Connection from PL4x</b>	5-16
<b>Configure Enterprise Application for PL4x</b>	5-17
<b>Configure Active Integration (PL4x) for TLS/SSL</b>	5-17
Certificates	5-18
Server Authentication	5-20
Client Authentication	5-22
Encrypted Logging	5-24

Troubleshooting	5-27
<b>Install More Than One BGS on the Same Host</b>	<b>5-29</b>
<b>Start Active Integration (PL4x) BGS and GS as Windows Service (Windows only)</b>	<b>5-29</b>
<b>Stop Active Integration (PL4x) BGS and GS Service with Script (Windows only)</b>	<b>5-30</b>
<b>Installation of additional components</b>	<b>5-31</b>
Install a JDBC Driver to connect to a database	5-31
<b>PL4x Job Server Installation</b>	
<b>PL4x Job Server Configuration</b>	<b>6-1</b>
<b>PL4x Job Agent Configuration</b>	<b>6-2</b>
<b>Set up Teamcenter Multi Connect for PL4x Jobs</b>	<b>6-5</b>
<b>Troubleshooting with Active Integration (PL4x) Process Start</b>	<b>7-1</b>
<b>Use Nagios to monitor the Active Integration (PL4x) Infrastructure</b>	
<b>Nagios Introduction</b>	<b>8-1</b>
<b>Base Server Module</b>	<b>8-1</b>
<b>Log Server Module</b>	<b>8-2</b>
<b>Job Server Module</b>	<b>8-3</b>
<b>Job Agent Module</b>	<b>8-3</b>
<b>Glossary</b>	<b>A-1</b>



# Preface

This documentation cannot be used as a substitute for consulting advice, because it can never consider the individual business processes and configuration. Despite our best efforts it is probable that some information about functionality and coherence may be incomplete.

**Issue: July 2018**

## **Legal notice:**

All rights reserved. No part of this documentation may be copied by any means or made available to entities or persons other than employees of the licensee of the Closed Loop Manufacturing for Teamcenter or those that have a legitimate right to use this documentation as part of their assignment on behalf of the licensee to enable or support usage of the software for use within the boundaries of the license agreement.

© 2018 Siemens Product Lifecycle Management Software Inc.

## **Trademark notice:**

Siemens, the Siemens logo and SIMATIC IT are registered trademarks of Siemens AG.

Camstar and Teamcenter are trademarks or registered trademarks of Siemens Product Lifecycle Management Software Inc. or its subsidiaries in the United States and in other countries.

Oracle is a registered trademark of Oracle Corporation.

SAP, R/3, SAP S/4HANA®, SAP Business Suite® and mySAP are trademarks or registered trademarks of SAP or its affiliates in Germany and other countries.

TESIS is a registered trademark of TESIS GmbH.

All other trademarks, registered trademarks or service marks belong to their respective holders.



# 1. Introduction

This manual explains the installation of the Active Integration (PL4x) software in version 18.1.

The term **PL4x** stands for the entire Active Integration product family including:

<b>T4S</b>	Teamcenter Gateway for SAP Business Suite
<b>T4O</b>	Teamcenter Gateway for Oracle EBS
<b>T4EA</b>	Teamcenter Gateway for Enterprise Applications
<b>T4S4</b>	Teamcenter Gateway for SAP S/4HANA
<b>T4CEP</b>	Teamcenter Gateway for Camstar Enterprise Platform
<b>TCRA4S</b>	Teamcenter Reporting and Analytics Gateway for SAP S/4HANA
<b>CLM4T</b>	Closed Loop Manufacturing for Teamcenter

**Caution:**

As this document describes the generic installation of the Active Integration (PL4x) software, these products are further referenced as PL4x.

The Active Integration (PL4x) software solution is a general-purpose integration software that provides data and process integration between Teamcenter® by Siemens PLM Software and SAP Business Suite® and SAP S/4HANA®, Oracle E-Business Suite by Oracle Corporation, Camstar Enterprise Platform, Teamcenter Reporting and Analytics, SIMATIC IT Unified Architecture Discrete Manufacturing and/or any other Enterprise Application, respectively.

For more details about general PL4x, please refer to other PL4x documentation.

For more information about new components and new versions of PL4x, please visit

[http://www.plm.automation.siemens.com/en\\_us/products/active-integration/index.shtml](http://www.plm.automation.siemens.com/en_us/products/active-integration/index.shtml)





## 2. Supported Environment

### 2.1 Teamcenter Compatibility Matrix

Generally Active Integration (PL4x) supports two major versions of Teamcenter, i.e. PL4x 18.1 supports Teamcenter 11.x and Teamcenter 12.x.

For detailed information on the compatibility of Active Integration products with operating systems, Teamcenter, Active Workspace, SAP Business Suite®, SAP S/4HANA®, Oracle EBS, Camstar and SIMATIC IT Unified Architecture Discrete Manufacturing, please visit [Active Integration Software Certifications](#).

### 2.2 Web Browser

For administrative PL4x tasks and configuring PL4x software, an Admin UI is provided for PL4x BGS and GS. In order to use it, you need an up-to-date web browser. We recommend using the following versions:

- Mozilla Firefox 31.5.0 ESR or 36.0.1 or higher up to 46.0
- Google Chrome 42.0.2311.90m
- Microsoft Internet Explorer 10 or 11

Caution:

- Using other web browsers is not recommended.
- There is no guarantee that older versions than the ones documented will work correctly with PL4x Admin UI.
- Newer versions of those browsers are supported based on the respective vendors' claims of compatibility.
- If any problems occur, please refer to [Troubleshooting with a PL4x Admin GUI](#) of this installation guide.

### 2.3 Install Hosts and Locations

PL4x only supports a 64 bit BGS (Basic Gateway Service), so you need to install PL4x BGS on a 64 bit server platform.

PL4x GS (Gateway Service) needs to run on the same host(s) as the Teamcenter server:

- in a Teamcenter **2-Tier** environment, PL4x GS should be installed on **every** Teamcenter client machine, because a Teamcenter client is also a Teamcenter server in the 2-Tier environment.
- in a Teamcenter **4-tier** environment: PL4x GS should be installed on every Teamcenter pool manager host.

For more information about PL4x BGS and GS, please refer to [The Active Integration \(PL4x\) Architecture](#).

In a production environment, there may be many transactions at the same time, which could cause problems because of excessive CPU and RAM demands. Thus, do not install the PL4x BGS and GS on the same server. For best performance, please install PL4x BGS on the host that is supposed to store log files. If not specified, the log files are stored in `<T4x_BGS_ROOT>/var/log`.

Caution:

- Do not use shared drives (NFS, SMB/CIFS...) for PL4x installations, log file storage or job file storage. Please use local disk and direct attached Storage, iSCSI, Fibre Channel or an equivalent technology.
- If you use a firewall, you need an open TCP and UDP port for the PL4x services.
- In case you are using a firewall with a content filter, please note that PL4x operates two different protocols on the same TCP/UDP port (HTTP and TPRPC). TPRPC is a PL4x native TCP protocol.

## 2.4 Sizing Considerations

- [Minimum CPU and Memory Requirements](#)
- [Job Pool memory and storage size](#)
- [Calculate Log File Storage Size](#)

### 2.4.1 Minimum CPU and Memory Requirements

Minimum CPU recommendation:

Operating System	CPU Type	Number of CPUs
Windows	Intel/AMD 32/64 bit	2
Linux	Intel/AMD 64 bit	2
Solaris	Ultra Sparc III	2
AIX	POWER 5	2

Minimum memory recommendation (free process memory):

Operating System	BGS	GS in 4-Tier Environment	GS in 2-Tier Environment
Windows	16 GB	8 GB	8 GB
Linux	16 GB	8 GB	8 GB
Solaris	16 GB	8 GB	
AIX	16 GB	8 GB	

## 2.4.2 Job Pool memory and storage size

PL4x jobs have a representation in the main memory as well as on the disk. The default job pool size is 100,000 jobs; maximum 4,000,000. A job pool needs a minimum of 32 GB and can grow up to 64 GB disk memory.

## 2.4.3 Calculate Log File Storage Size

Each GS and BGS (without job pool and log storage) typically requires a minimum of 2 GB on the file system. After installation, the GS (2-tier and 4-tier) does not write large files to the file system, while the BGS stores jobs and log files. The following table shows a recommendation of disk space for the BGS log storage depending on the number of Teamcenter users, assuming that log compression is on.

Number of Teamcenter users	Minimum disk space for the log storage
< 50	100 GB
50 - 500	500 GB
> 500	1 TB or more

PL4x compresses log files that have not been accessed for a certain time to save storage capacity. By default, log files that have not been accessed for two days will be compressed.

You can modify this threshold by adapting the **BGS Admin UI** → **Configuration** → **Log server** → **Advanced Settings** tab → **Compression** setting. For more information about the PL4x Admin UI, please refer to **Admin UI**. In rare cases, it could happen that the original log file is somehow blocked (e.g., because someone accessed it right in that moment), so that the compression cannot be successfully finished. Then you might see an error log line similar to this one, "*tpco\_udpCompressLogChannel :: cannot delete original log file ...*", but your log files will work as usual.

```
24/05/17 16:00:09.139337 tpbgs tpcp_apSrvCommandQueue :: INIT CMD QUEUE ID=241 DEPTH=256
24/05/17 16:01:51.340752 tpbgs tpcp_udpCompressLogChannel :: cannot delete original log file D:/Secure/work/ci_
24/05/17 16:02:51.483156 tpbgs tpcp_udpCompressLogChannel :: cannot delete original log file D:/Secure/work/ci_
24/05/17 16:02:51.499704 tpbgs tpcp_udpCompressLogChannel :: cannot delete original log file D:/Secure/work/ci_
24/05/17 16:02:51.515074 tpbgs tpcp_udpCompressLogChannel :: cannot delete original log file D:/Secure/work/ci_
24/05/17 16:02:51.529505 tpbgs tpcp_udpCompressLogChannel :: cannot delete original log file D:/Secure/work/ci_
```

## 2.5 Operating Systems

Active Integration (PL4x) is developed and distributed for all operating systems supported by Teamcenter. They are:

### with Teamcenter 11.x

- IBM AIX
  - 6.1, 7.1
- Red Hat Linux Enterprise Server x64
  - RHEL 6.3+ Server At Tc10.1.3 RHEL 6.4
- Solaris
  - Solaris 10 and 11
- SUSE Linux Enterprise Server x64
  - SUSE Linux Enterprise Server 11 SP2, SLES 11 SP3
- Windows Server 32 bit on x86-64
  - Win Server 2008 R2 SP1 x64 (Stand & Ent); Win Server 2012 x64 (Stand & Datacenter)
- Windows Server x64
  - Win Server 2008 R2 SP1 x64 (Stand & Ent); Win Server 2012 x64 (Stand & Datacenter) Win Server 2012 R2

### with Teamcenter 12.x

- IBM AIX
  - 6.1, 7.1
- Red Hat Linux Enterprise Server x64
  - RHEL 6.4+ and 7.x Server
- Solaris
  - Solaris 10 and 11

- SUSE Linux Enterprise Server x64
  - SUSE Linux Enterprise Server 11 SP2 & SP3, SUSE Linux Enterprise Server 12 SP1
- Windows Server x64
  - Win Server 2008 R2 SP1 x64 (Stand & Datacenter); Win Server 2012 R2 x64 (Stand & DC) Win Server 2012 R2

For more information about the certified version of operating systems, please refer to **Active Integration Software Certifications** and click on *Active Integration Compatibility Matrix*.

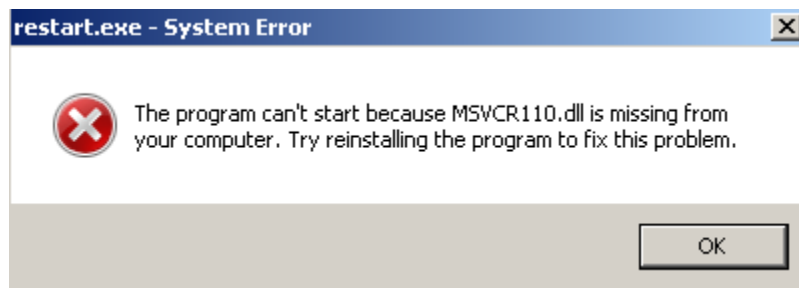
**Caution:**

On every machine running PL4x (both BGS and GS) on Linux, Solaris or AIX, make sure that the operating system has the “allowed number of open files” greater than 2048. We recommend the number 4096. Please consult the operating system documentation for how to check and configure that.

**Windows only:** Microsoft Visual C++ Redistributable Packages are required for Active Integration (both BGS and GS) on any Windows system. The requirement of the patches depends on the Operating System and the Teamcenter version in use. In principle, for PL4x 18.1 the following packages are required:

- Microsoft Visual C++ 2010 SP1 Redistributable Package (x86): <http://www.microsoft.com/en-us/download/details.aspx?id=8328>
- Microsoft Visual C++ 2010 SP1 Redistributable Package (x64): <http://www.microsoft.com/en-us/download/details.aspx?id=13523>
- Microsoft Visual C++ 2012 Redistributable Package: <http://www.microsoft.com/en-us/download/details.aspx?id=30679>
- Microsoft Visual C++ 2015 Redistributable Package: <https://www.microsoft.com/en-us/download/details.aspx?id=53840>

If Microsoft Visual C++ Redistributable Packages are not installed correctly, PL4x BGS or GS cannot be started properly. An error message will pop-up to mention it.



# 3. Admin UI

## 3.1 Administrative User Interface

The Active Integration Administrative User Interface (Admin UI) is an application that allows performing administrative tasks with regard to PL4x.

This documentation will give you basic information about the Admin UI and how to reach it. Detailed information on the single applications contained in it can be found in the according online help of the Admin UI.

### Menu Functionalities Overview

Both the BGS and the GS have their own interface with common and unique functionalities.

Common menu entries are:

- Monitoring: View current statistics of the system and monitor PL4x activity.
- Script: Execute PL4x Test Scripts. E.g., to check mappings (GS only) or encrypt passwords (BGS only).
- Diagnosis: Download of a stacktrace for our software support.
- System information: View the persistent data of the system.
- Configuration: Display and edit the configuration of PL4x. The configuration options and functionalities are different for BGS and GS.
- Restart: Restart of the application (recommended way is to use the executable *bin64/restart*).
- About: View service, credits and copyright.

BGS exclusive menu entries are:

- Job management: Control jobs and job agents.
- Log files: View and analyze transaction, system, session and user log files.

### Connection and Access to the Admin UI

To access the BGS or GS Admin UI follow these steps:

- Be sure the BGS or GS is installed and configured correctly. Please see [Installation Instructions](#).

- Be sure the BGS or GS is running. If not, start it with `<T4x_BGS_ROOT>/bin64/restart` or `<T4x_GS_ROOT>/bin64/restart` or start the according service.
- The Admin UI is available by entering and loading the following URL in your web browser:  
The BGS Admin UI can be reached by default by `https://<URL of BGS>:11320`  
The GS Admin UI can be reached by default by `https://<URL of GS>:11321`
- Login with the default **Username** "t4adm". The default **Password** is "geheim" (the German word for "secret"). Please consider to change or to replace the default username and password.  
For further information on user management, roles and rights please see the Admin UI Online Help (see below).

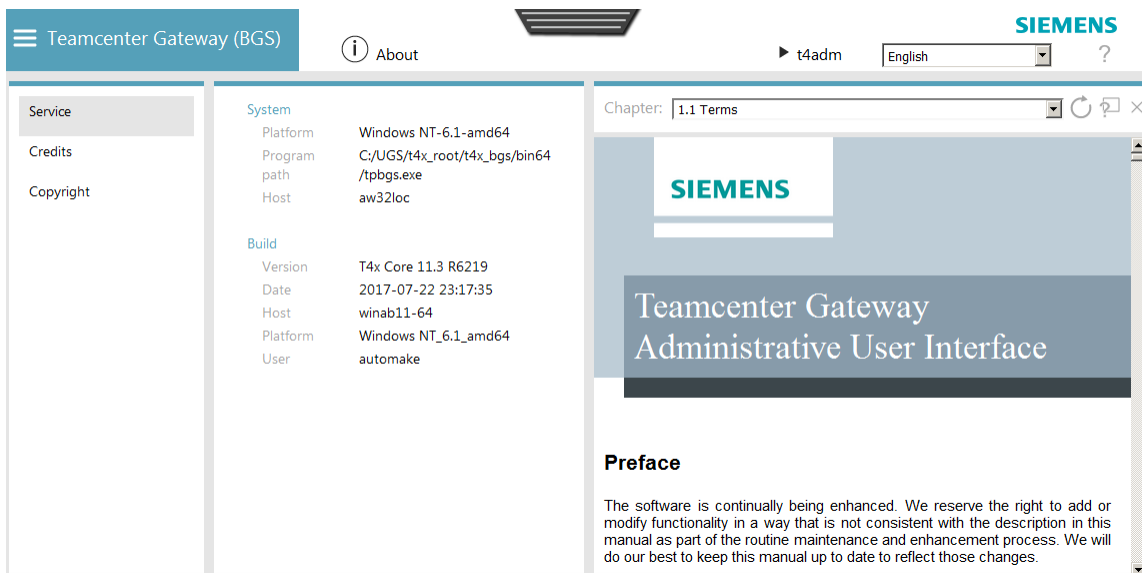
**Caution:**


The port number to reach the Admin UI and whether to use HTTP or HTTPS can be changed. This is described in the chapter **Active Integration (PL4x) Configuration Using Admin UI**.

For troubleshooting and web browser compatibility please refer to **Admin UI Troubleshooting** and **Web Browser** of this installation guide.

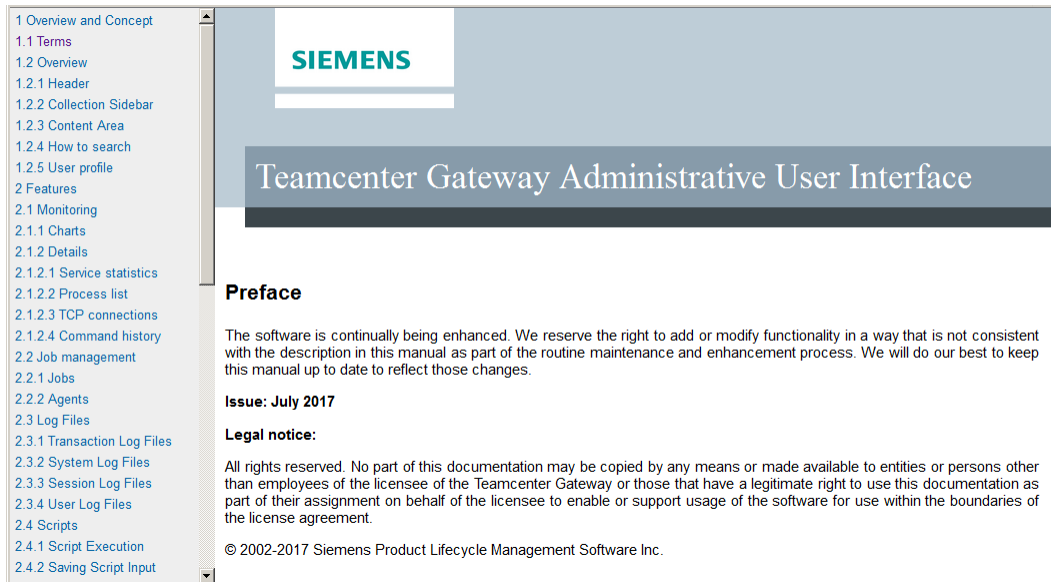
## Admin UI Online Help

The Admin UI has built-in online help. It can be accessed by clicking on the question mark (?) in the upper right corner:



By default it will show the help content that applies to the opened menu point. The help can be opened in a separate window, by click on the  symbol.





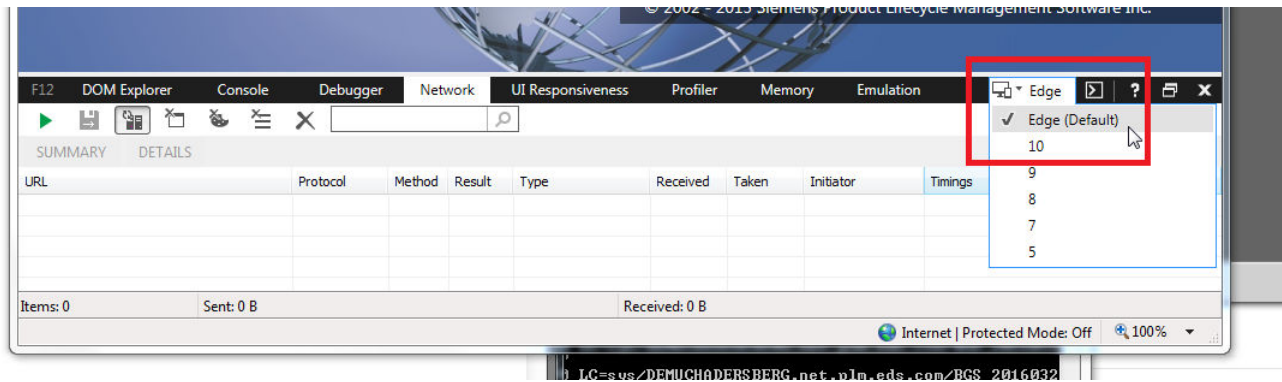
## 3.2 Admin UI Troubleshooting

If anything seems strange with the PL4x Admin UI (or behaving differently from the description here), try the following:

- Be sure to use a web browser that is supported by your PL4x version. For more information about supported web browser, please refer to [Web Browser](#).
- Activate Java Script for full PL4x functionality
- Delete the web browser cache and cookies
- Restart the web browser after the cleaning

If you use Internet Explorer and the Admin UI web page stays blank, please check the document mode settings of your browser:

- Open the Admin UI web page in your Internet Explorer
- Open **Settings** → **F12 Developer Tools** or press F12
- Set the document mode to **Edge** in the upper right corner of the tools
- The UI login screen should now appear and you can close the developer tools

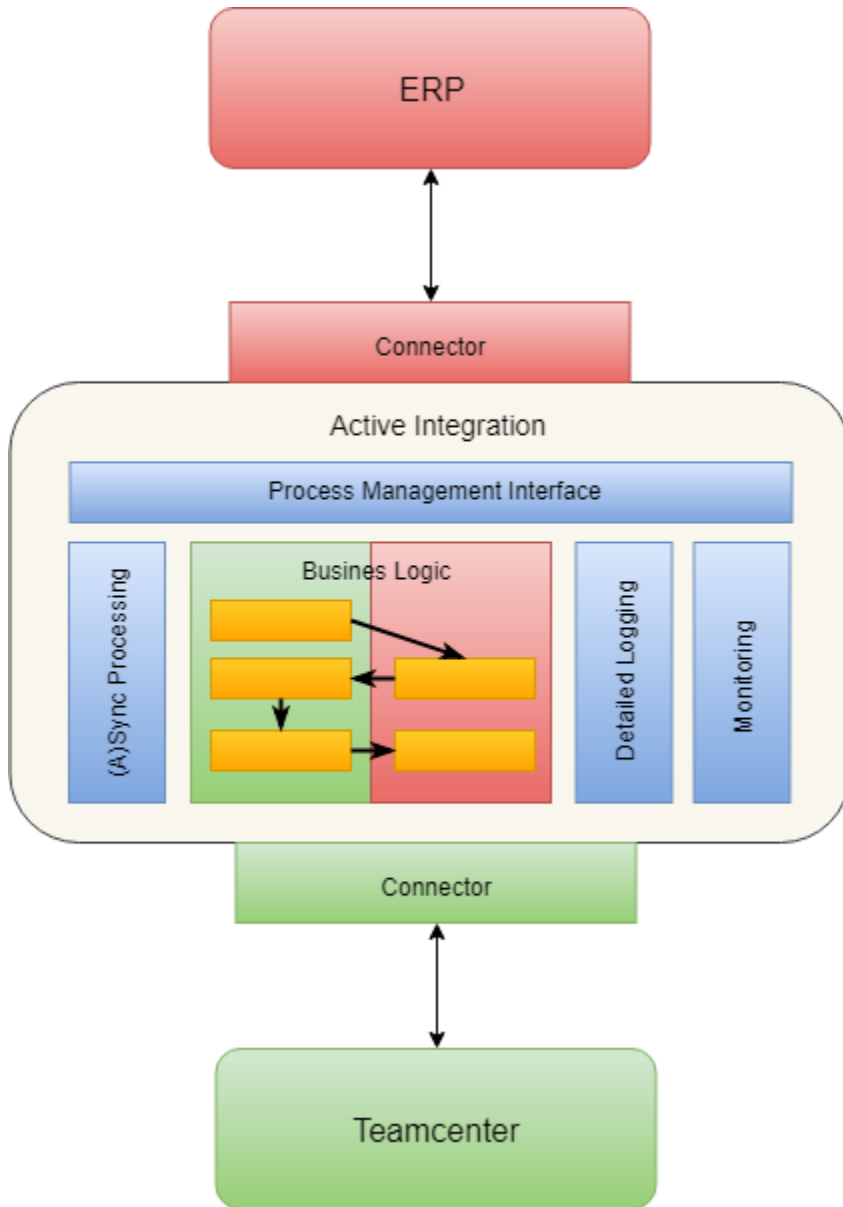


Usually this behavior is caused by the so called *Compatibility View* of the Internet Explorer. It can also be disabled for all pages following these steps:

- Open the **Tools** → **Compatibility View Settings** of your Internet Explorer
- Remove the check mark next to **Display intranet sites in Compatibility View**
- Close the dialog

# 4. The Active Integration (PL4x) Architecture

PL4x is the integration software to enable bidirectional data integration and process coupling between Teamcenter and SAP Business Suite® and SAP S/4HANA®, Oracle EBS or other enterprise applications.



PL4x consists of two services:

- **BGS:** PL4x Basic Gateway Service is responsible for the licensing and logging. This central service has to be installed at least once per site and does not need any target system (e.g., SAP, Oracle EBS, ...) or Teamcenter environment (except possibly for job execution, depending on the configuration). The PL4x job server is a part of the PL4x BGS that manages transactions - that may be large and numerous - in the background in order to not keep the Teamcenter user blocked while processing the data. To install and configure PL4x job server, please refer to Job Server Installation. Each PL4x process writes logs and debug messages to this central BGS using the UDP/IP protocol. The PL4x log server is a part of the BGS which writes these messages into log files and stores them in the log server's file system. Depending on the configuration, the "log cleaner" clears the log files and directories (roll files over, delete files...). The log files can be viewed with the PL4x Admin UI from anywhere in the network.

**Caution:**

Any log information is sent via the UDP protocol. If a network connection is down, no PL4x process will be blocked but the sender will not be informed if a log data package is lost. Definitely logging information will be lost if clients cannot connect to the BGS.

- **GS:** PL4x Gateway Service drives the process mapping. It contains the complete PL4x software (includes all PL4x servers, but not the BGS). Several PL4x instances can be installed using this package in the network and they all can use the same PL4x BGS. It manages the connection to target enterprise applications, operates the mapping, etc. Therefore it needs a configured target system (e.g., SAP, Oracle EBS, ...) and Teamcenter environment. This package contains the client software as well as the programmable TCL code (mapping) that manages the transfers/imports. Large and numerous transactions can be executed asynchronously in the background using the job server (BGS) and job agents (GS).

# 5. Installation Instructions

## 5.1 Overview of Installation Steps

The PL4x installation is divided into following five steps:

1. Acquire the PL4x installation file from **GTAC**
2. Install and configure the PL4x BGS
3. Install and configure the PL4x GS
4. Configure Teamcenter environment for PL4x
5. Configure target enterprise application (such as SAP, Oracle EBS, CEP...) for PL4x

## 5.2 Active Integration (PL4x) Installation

The released Active Integration (PL4x) installation files are uploaded in **GTAC** and available for all customers to download. Before installing PL4x, please acquire the right PL4x installation files according to your operating system and Teamcenter version.

PL4x installation files are located in folder **Teamcenter and Teamcenter Rapid Start → Full Products → Integrations and Solutions** in **GTAC**. The PL4x installation files are packed as a zip file.

Complete these steps to install PL4x:

- **Active Integration (PL4x) Installer Introduction**
- **Install and Configure the PL4x BGS Using PL4x Installer**
- **Install and Configure the PL4x GS Using PL4x Installer**

### 5.2.1 Active Integration (PL4x) Installer Introduction

PL4x provides an additional tool, i.e., PL4x Installer, to facilitate the procedures for installing, upgrading or extending PL4x:

- **Install:** newly install a PL4x product.
- **Upgrade:** update or patch an already installed PL4x product.
- **Extend:** extend an already installed PL4x product by one or more other PL4x products in the Active Integration product family.

**Caution:**

When installing PL4x GS, the installer will modify the Teamcenter installation by copying the PL4x jar files to the Teamcenter `<TC_ROOT>/portal/plugins` folder.

For upgrade scenarios from an old PL4x version to PL4x 18.1, please refer to for migrating PL4x mapping, preferences and workflows.

The PL4x Installer is provided for all OS platforms supported by PL4x, i.e., Windows, Linux, Solaris and AIX.

The PL4x Installer requires a Java runtime environment (minimum version 1.7). Usually, that can be the same Java runtime installation as the one Teamcenter uses.

If a JRE is installed, the environment variable `PATH` should include the whole directory to the Java executable file. The java path will be checked at the first step to start PL4x Installer.

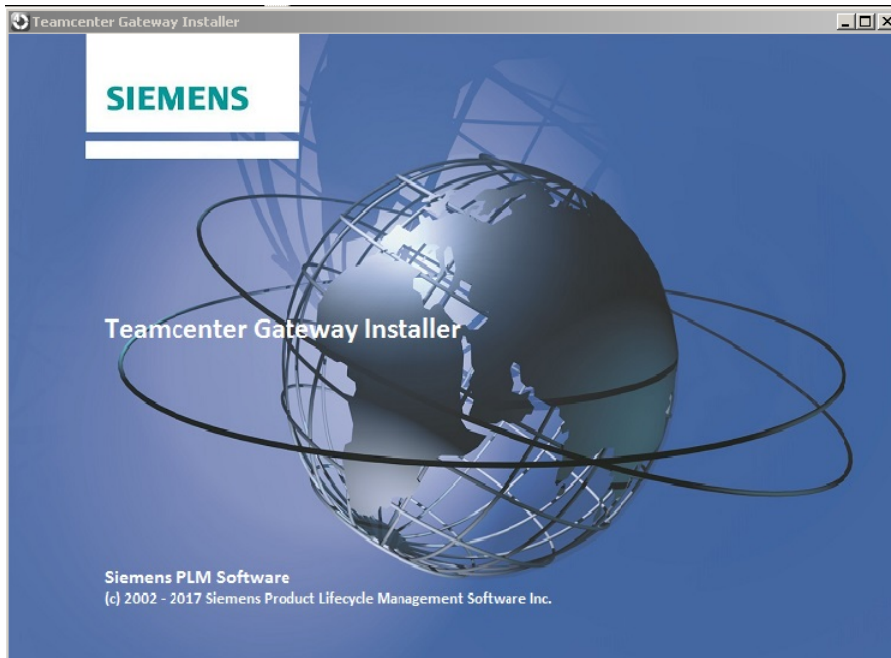
## 5.2.2 Install and Configure the PL4x BGS Using PL4x Installer

Download PL4x Installer, a zip file with the name *install and patch tool*, from **GTAC**.

Unpack this zip file and execute one of the following start scripts depending on your operating system to start the PL4x Installer:

- on Windows platforms: `__installer_18.1.bat`
- on other platforms: `__installer_18.1.sh`

After the tool starts, it shows a window with Siemens PLM Software company information and PL4x product information. This will disappear after a few seconds and the "License Agreement" will be shown.



After accepting the license agreement, the PL4x installation archive, downloaded from **GTAC**, has to be selected for unpacking and installing. Then the **Readme** shows the version information of the PL4x product to be installed and a short introduction of this PL4x Installer.

In order to install PL4x BGS, please select or specify the correct values in the following dialogs:

1. select **install** for Install action
2. select **BGS** for PL4x package type
3. specify the root directory **T4x\_Root** for PL4x BGS
4. specify the port numbers for BGS
5. specify the host and port of the SPLM license server

Once you finish all selections, a summary of the chosen settings will be displayed. Click **Previous** to go back to the previous selection and give new values, if needed. Click **Save** to save the shown data into a file, click **Start** to start the installation or **Exit** to cancel the installation.

**Caution:**

Do not install the BGS together with any PL4x GS installation in the same directory. You must specify one directory for BGS and another directory for GS.

Do not install PL4x BGS or GS on a shared (mounted) drive, including drives that are physically located on the same machine but connected by a network connection! UNC paths (\\server\share) are not allowed as well.

Avoid long path names and blanks in the path names.

Be sure to have write access for PL4x BGS or GS, and make sure that the files are not write-protected after you copied them. In UNIX/Linux, the permission 755 is required for the entire directory tree.

As it might cause file system problems, be sure to exclude the PL4x directory from an automatic backup. If required, only the directory `<T4x_BGS_ROOT>/var` should be included.

The folder name of BGS root can be changed only before the first time BGS is started.

Start PL4x BGS by executing `<T4x_BGS_ROOT>/bin64/restart`. The **restart** executable does the same start as **start**, but it additionally calls **stop** to stop all running PL4x BGS processes of the same PL4x BGS installation cleanly before starting it. The **restart** executable is recommended to start or restart PL4x after the first time starting this PL4x installation.

To see whether PL4x BGS is running, you should check with operating system tools if the process *tpbgs* is running.

- on Windows platforms: Task Manager
- on UNIX/Linux platforms: use the command `ps -aef | grep tpbgs`

Alternatively, you can check the BGS status by executing `<T4x_BGS_ROOT>/bin64/status.exe` (on Windows platform) or `status.sh` (on UNIX/Linux platforms).

```

D:\t4x_install\t4x_root\t4x_bgs\bin64\status.exe
DESCRIPTION          PID  PPID  USER      MEMORY      CPU  COMMAND
T4x watch process    9660 10120  mtv2k5     53899264    2   D:/t4x_i
ninstall/t4x_root/t4x_bgs/bin64/tpbgs.exe ./var/init/start.bgs INTERN
T4x server process   11148 9660   mtv2k5     688058368   8   D:/t4x_i
ninstall/t4x_root/t4x_bgs/bin64/tpbgs.exe ./var/init/start.bgs INTERN
T4x helper           11132 9660   mtv2k5     44638208    1   D:/t4x_i
ninstall/t4x_root/t4x_bgs/bin64/tps.exe var/init/start.tps D:/t4x_install/t4x_root
/t4x_bgs/var/init/garbage_collector.tcl

```

By default, the BGS is configured with three server instances, each has its own port number:

- **SERVER** provides the default service functions (e.g., via TPRPC or HTTP); default port: **11300**
- **LOG\_SERVER** provides the log functions; default port: **11300**



- **ADMIN\_UI20** provides all services needed for the Admin UI (default configuration using HTTPS); default port: **11320**

You can log into BGS Admin UI to check and configure PL4x BGS. For more information, please refer to [Configure Active Integration \(PL4x\) BGS Using BGS Admin UI](#).

### 5.2.3 Install and Configure the PL4x GS Using PL4x Installer

For installing PL4x GS, start PL4x Installer, check the version information of your PL4x product at first and then select or specify the correct values in the upcoming dialogs:

1. select **install** for Install action
2. select **PL4x GS** for PL4x package type
3. specify your Teamcenter **TC\_ROOT**
4. specify the root directory **T4x\_Root** for PL4x GS
5. specify the **host** and **port number** of your BGS Service

Once you finish all the selections, a summary of the chosen settings will be displayed. Click **Previous** to go back to the previous selection and give new values, if needed. Click **Save** to save the shown data into a file, click **Start** to start the installation or **Exit** to cancel the installation.

#### Caution:

Because of the data communication between PL4x GS and Teamcenter server, there is an essential difference to install PL4x GS in Teamcenter 2-tier and 4-tier environment:

- In the 2-tier environment, PL4x GS should be installed in all Teamcenter 2-tier clients.
- In the 4-tier environment, PL4x GS should be installed only on all Teamcenter pool manager hosts.

For additional cautions, please refer to [Install and Configure the PL4x BGS Using PL4x Installer](#).

After installing the PL4x GS, you should configure the Teamcenter environment in the additional script *t4xcust* of PL4x GS (*t4xcust.bat* or *t4xcust.unix*, respectively, in <T4x\_GS\_ROOT>/etc). **TC\_ROOT**, **TC\_DATA** and call of *tc\_profilevars.bat* should be specified. An Example for the Teamcenter environment configuration on Windows:

```
rem Set the Teamcenter environment:

set TC_ROOT=C:\work\teamcenter
```

```
set TC_DATA=C:\work\tcdata

call %TC_DATA%\tc_profilevars.bat
```

**Caution:**

- Avoid long path name and blanks in the path name.
- The folder name of GS Root can be changed only before the first time GS is started.

Start PL4x GS by executing `<T4x_GS_ROOT>/bin64/restart`. To see whether PL4x GS is running, you should check with operating system tools if the process `tpapps` is running.

Besides the operating system tools, you can check the GS status by executing `<T4x_GS_ROOT>/bin64/status.exe` (on Windows platform) or `status.sh` (on UNIX/Linux platforms).

DESCRIPTION	PID	PPID	USER	MEMORY	CPU	COMMAND
T4x watch process	11024	8064	mtv2k5	58691584	2	D:/t4x_i
install\t4x_root\t4x_gs\bin64\tpapps.exe			./var/init/start.apps INTERN			
T4x server process	10096	11024	mtv2k5	200552440	2	D:/t4x_i
install\t4x_root\t4x_gs\bin64\tpapps.exe			./var/init/start.apps INTERN			
T4x helper	11060	11024	mtv2k5	44601344	2	D:/t4x_i
install\t4x_root\t4x_gs\bin64\tps.exe			var/init/start.tps			D:/t4x_install/t4x_root/
t4x_gs/var/init/garbage_collector.tcl						

By default, the GS is configured with two server instances, each has its own port number:

- **SERVER** provides the default service functions (e.g., via TPRPC or HTTP); default port: **11301**
- **ADMIN\_UI20** provides all services needed for the Admin UI (default configuration using HTTPS); default port: **11321**

You can log into GS Admin UI to check and configure PL4x GS. For more information, please refer to [Configure Active Integration \(PL4x\) GS Using GS Admin UI](#).

## 5.3 Active Integration (PL4x) Configuration Using Admin UI

[Configure Active Integration \(PL4x\) BGS Using BGS Admin UI](#)

[Configure Active Integration \(PL4x\) GS Using GS Admin UI](#)

### 5.3.1 Configure Active Integration (PL4x) BGS Using BGS Admin UI

Login to the BGS Admin UI. Please see chapter [Administrative User Interface](#).

Click **About** → **Service** to get basic information about the installed BGS.

Teamcenter Gateway (BGS) About t4adm English ?

System	
Platform	Windows NT-6.1-amd64
Program path	C:/UGS/t4x_root/t4x_bgs/bin64/tpbgs.exe
Host	aw32loc
Build	
Version	T4x Core 11.3 R6219
Date	2017-07-22 23:17:35
Host	winab11-64
Platform	Windows NT_6.1_amd64
User	automake

### 5.3.1.1 Set Teamcenter License Service in PL4x BGS

The Active Integration products are licensed software. They are protected by a license key.

As a member of the Teamcenter product family, the license for PL4x products is included in the Siemens PLM Software license file.

PL4x BGS directly gets the license information from the Siemens PLM Software license server. Thus, you need to configure PL4x BGS to connect to the license server. Click **Configuration** → **General** → **License server** to specify your Siemens PLM Software license server.

Teamcenter Gateway (BGS) Configuration t4adm English ?

Basic Gateway Service

Installation instance name: BGS\_20180205-131228

Number of threads: 3

**License server**

IP address of the license server: localhost

Port of the license server: 28000

Privacy

Participation in the Product Excellence Program:  On  Off

When there is an active Teamcenter ITK connection, Teamcenter Gateway will retrieve a license from the ITK connection if PL4x BGS fails to retrieve the license directly from the Siemens PLM License Server.

**Caution:**

If any setting of your PL4x BGS installation is changed in Admin UI, you have to save the modified settings and restart the BGS in Admin UI.



The configuration has been saved. The server needs to be restarted for the new configuration to take effect. Do you want to restart the server now?

[Restart later manually](#)

[Restart now](#)

### 5.3.1.2 Change the Port of BGS Instances

If needed, the port number and other communication settings of BGS server instances can be modified in the Admin UI. Open **Configuration** → **Server instances** and then click the edit button in the **Actions** column of the table.

The screenshot shows the Admin UI for Teamcenter Gateway (BGS) in the Configuration section. The left sidebar contains navigation options: General, Server instances (selected), Communication channels, Java, Log server, Job server, Task management, and User management. The main content area displays a table of server instances with the following data:

Name	Port	Interf...	IP stack	Tran...	Prot...	HTTP r...	Encr...	Server...	CA ce...	TLS/...	TLS/S...	Basi...	Shar...	Actions
SERVER	11300	Any	IPv6 + IPv4	TCP	All	D:/work/...	Off					On		
LOG_SERVER	11300	Any	IPv6 + IPv4	UDP	LOG		Off					Off		
ADMIN_UI20	11320	Any	IPv6 + IPv4	TCP	HTTP	D:/work/...	On	serverC...		TLSv1	AES128...	Off		

Specify your port in the pop-up dialog.

**Edit server instance**

Name: ADMIN\_UI20

Port: 11320

Interface binding:  Any  
 Define IP address

IP stack:  IPv4 + IPv6  
 IPv4  
 IPv6

Transport mode: TCP

Protocol: HTTP

HTTP root: D:/work/bgs/var/httpd/au20

Basic authentication: Off

Encryption:  On  Off

TLS/SSL type: TLSv1

TLS/SSL cipher:  AES512-SHA  
 AES256-SHA

Server certificate: demo\_server\_certificate.pem  
/C=DE/ST=Bavaria/L=Munich/O=INVALID/OU=INVALID/CN=PL4x  
Default Delivery Server  
Certificate/emailAddress=NEVER@USE.THIS.CERTIFICATE

CA certificate:

Function binding: [Edit function binding](#)

[Cancel](#) [Apply](#)

Save the modification and restart BGS in the Admin UI.

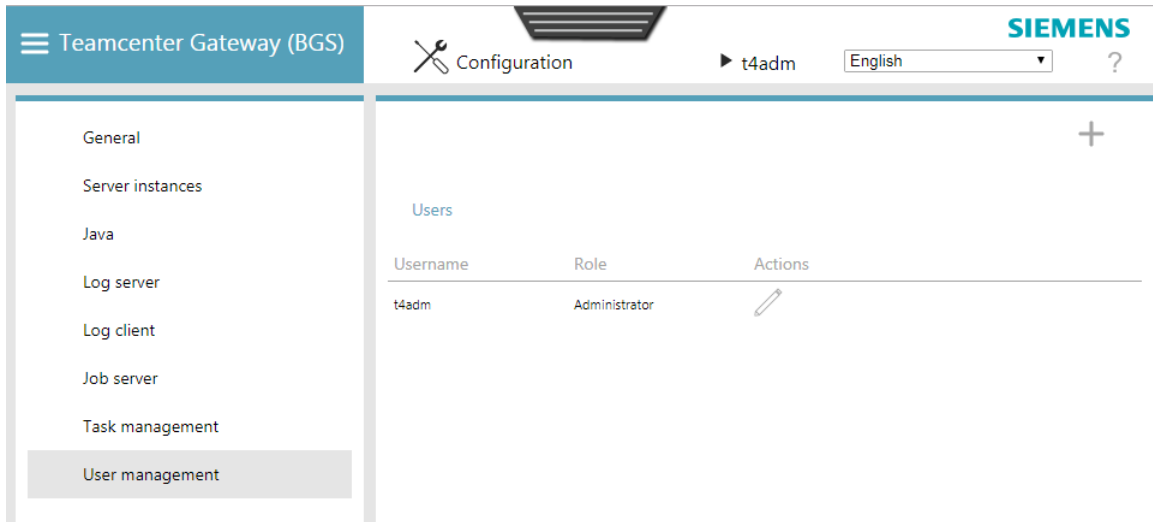
### 5.3.1.3 User Management in PL4x BGS Admin UI

PL4x offers a user management where you can add users who are allowed to access the Admin UI. For each user, you can choose from three predefined roles to define which areas of the Admin UI should be accessible. For more information about roles please refer to Admin UI online help.

In the PL4x BGS Admin UI, the user management is only accessible to users with the role of Administrator. By default there is one predefined user *t4adm* (password *geheim*) with this role on a newly installed system.

Click **Configuration** → **User management** to open the user management page. The following actions are available

- Add a new user ID, enter a password and select a role for it.
- Edit an existing user ID, i.e., change the password or assign a different role. Please note that in order to avoid a lock out, you cannot change the role of the current user ID.
- Delete a user ID. The current user ID cannot be deleted.

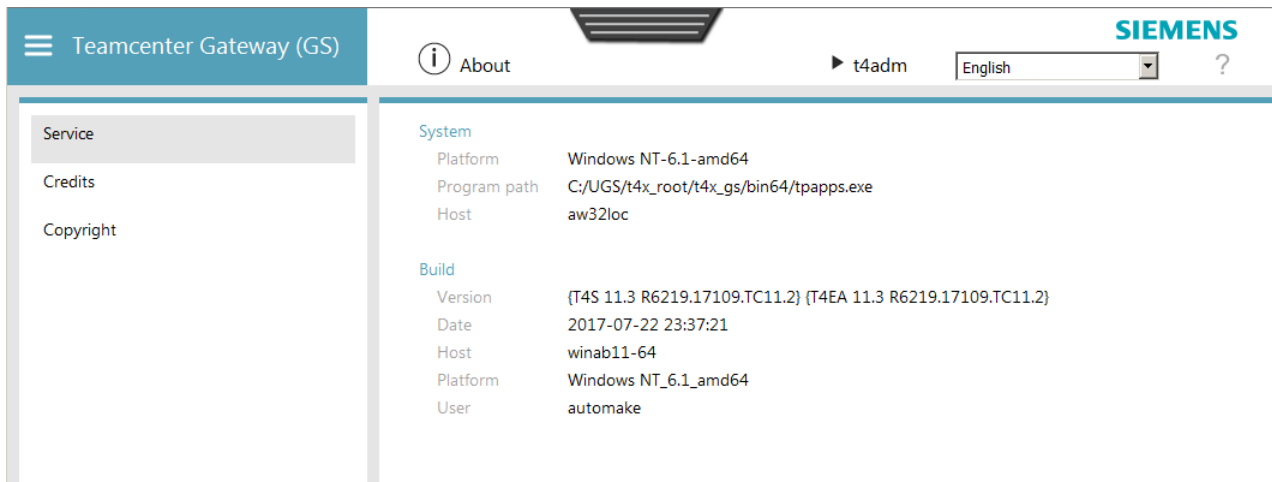


In order to avoid a lock out, e.g., if the password of the last administrator is lost, a tool `<T4x_BGS_ROOT>\bin64\resetadmin.exe` (on Linux/Unix fileNam<T4x\_BGS\_ROOT>/bin64/resetadmin) is provided on machines running a BGS. If you run it, the default administrator `t4adm` with password `geheim` will be restored.

### 5.3.2 Configure Active Integration (PL4x) GS Using GS Admin UI

Login to the GS Admin UI. Please see chapter [Administrative User Interface](#).

Click **About** → **Service** to get basic information about the installed GS.



#### 5.3.2.1 Change the Port of GS Instances

If needed, the port number of GS server instances can be modified in the Admin UI. Click **Configuration** → **Server instances**, then click **Actions** of server instances and specify your port in the pop-up dialog.

The screenshot shows the 'Configuration' page for 'Teamcenter Gateway (GS)'. The left sidebar contains navigation options: General, Server instances, Communication channels, Java, Job agent, and Task management. The main content area is titled 'Server instances' and contains a table with the following data:

Name	Port	Interf...	IP stack	Tran...	Prot...	HTTP ...	Encr...	Server...	CA ce...	TLS/...	TLS/S...	Basi...	Shar...	Actions
SERVER	11301	Any	IPv6 + IPv4	TCP	All	D/work...	Off					On		
ADMIN_UI20	11321	Any	IPv6 + IPv4	TCP	HTTP	D/work...	On	serverC...		TLSv1	AES128...	Off		

**Caution:**

If any setting of your PL4x GS installation is changed in Admin UI, you have to save the modified settings and restart the GS in Admin UI.

The dialog box contains the following text: "The configuration has been saved. The server needs to be restarted for the new configuration to take effect. Do you want to restart the server now?" Below the text are two buttons: "Restart later manually" and "Restart now".

**5.3.2.2 Set BGS Server in PL4x GS**

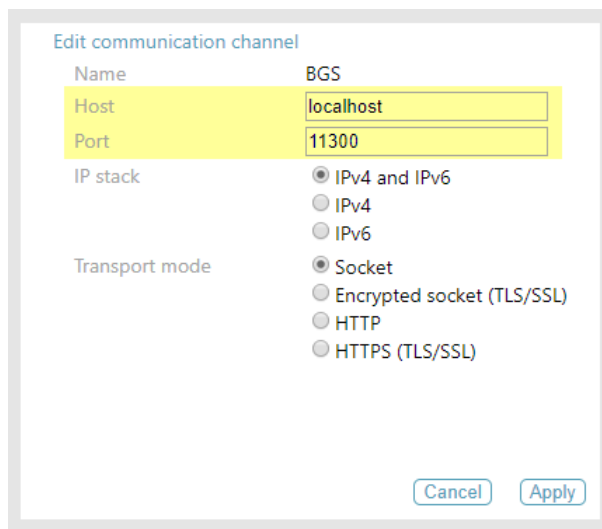
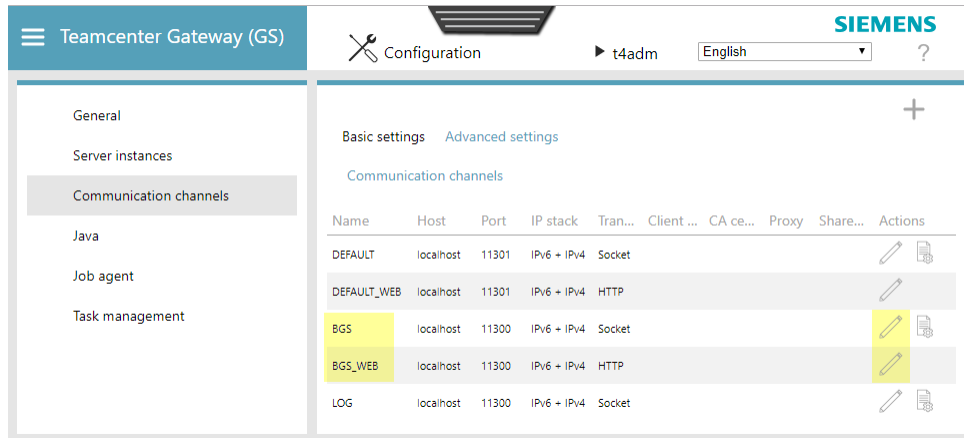
To check or modify the set BGS server in PL4x GS, open **Configuration** → **Communication channels**.

When using the default configuration mode for communication channels the BGS can be set in the **Basic settings** tab by entering IP address and port.

The screenshot shows the 'Basic settings' page for 'Teamcenter Gateway (GS)'. The left sidebar is the same as in the previous screenshot. The main content area is titled 'Basic settings' and contains the following settings:

- Settings for all communication channels**
- Configuration mode**:
  - Use default settings (no encryption)
  - Use advanced settings
- IP stack**:
  - IPv4 and IPv6
  - IPv6
  - IPv4
- Basic Gateway Service** (highlighted in yellow):
  - IP address of the BGS: localhost
  - Plain port of the BGS: 11300

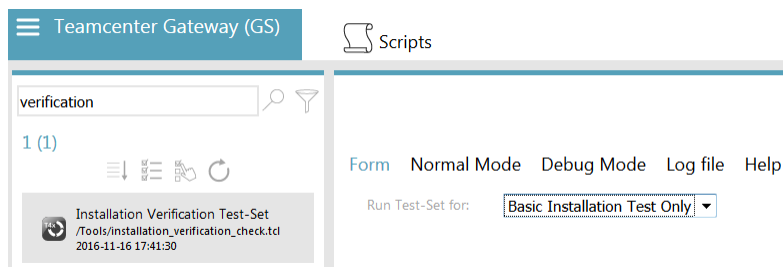
When the advanced settings configuration mode is used, the BGS and BGS\_WEB communication channels in the table have to be edited by clicking the edit button in the **Actions** column. Enter the host and port of the BGS and apply the settings to close the popup.



Save the modification and restart GS in the Admin UI.

### 5.3.2.3 Verify the PL4x Installation

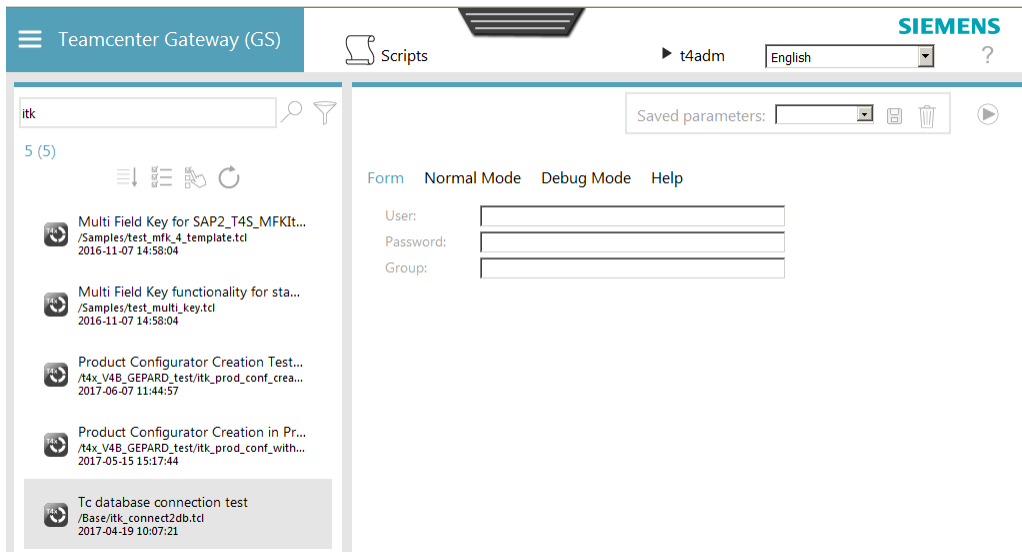
You can check the installation of PL4x GS and PL4x license information by executing test script **Installation Verification Test-Set**. Click **Script** → **Scripts** and search for it and run this script.



In the script output, you will find the message about PL4x installation, Teamcenter parameters and PL4x license information.



A script **Tc database connection test** is offered to test the connection from PL4x to Teamcenter. Click **Script** → **Scripts** and search for it, give the Teamcenter user information and run this script.



## 5.4 Configure Teamcenter Environment for Active Integration (PL4x)

- **Deploy Active Integration (PL4x) GS Template with TEM**
- **Set PL4x GS Environment for a Teamcenter 2-Tier Environment**
- **Set PL4x GS Environment for a Teamcenter 4-Tier Environment**
- **Add PL4x Error Message Texts to Teamcenter**

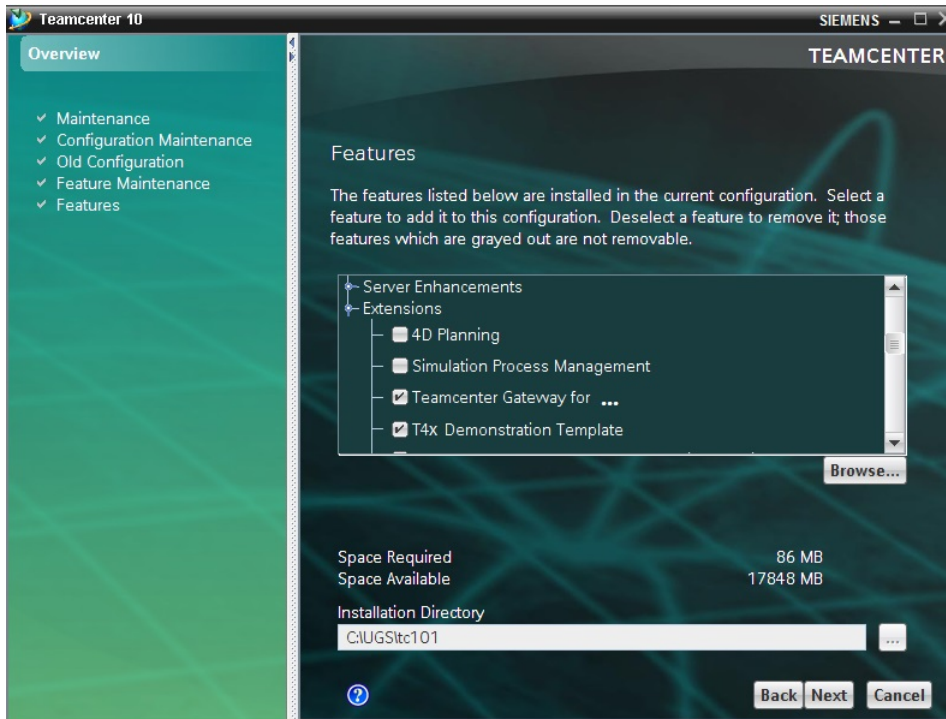
### 5.4.1 Deploy Active Integration (PL4x) GS Template with TEM

For the PL4x installation, the Teamcenter database and configuration should be adapted by deploying the PL4x template with the Teamcenter Environment Manager (TEM).

For deploying the PL4x template with TEM, please perform the following steps:

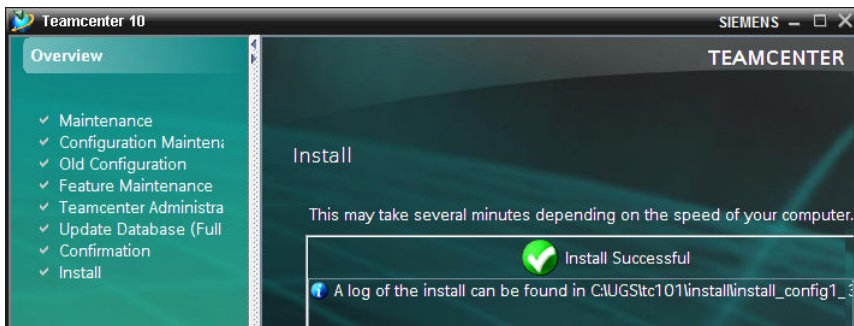
1. Execute *tem.bat* in Windows or the corresponding file *tem.sh* in UNIX/Linux in directory *%TC\_ROOT%/install* to start TEM
2. In the first screen **Maintenance**, select **Configuration Manager** and click **Next**
3. Select **Perform maintenance on an existing configuration**
4. Select your desired database

5. Feature Maintenance: select **Teamcenter – Add/Remove Features**
6. In the next window **Features**, click **Browse** and select the PL4x template file: `<T4x_GS_ROOT>/var/template/<t4x>/BMIDE/full_update/feature_<t4x>.xml`
7. Then, in the same window, the new feature appears and has to be checked. To deploy template, the new feature under **Extensions** should be selected.



Usually this will cause TEM to also install the RAC extensions if the PL4x template has client extensions. However in some environments this will not always happen. To make sure client extensions are installed, please also select the corresponding template "<PL4x> for Rich Client".

8. Click **Next** and type the dba password for this database and click **Next**
9. In the next window **Confirmation**, you should see the selected features. Click **Start** to start the template installation
10. The next window **Install** shows the progress with a moving bar. This may take some minutes and should end with the message **Install Successful**:



#### Caution:

- Before starting TEM, please make sure to have all the configuration XML files of any adaptation that was done up to now (i.e., before the beginning of this PL4x installation) in the directory `%TC_DATA%/model`. Especially after adaptations using database dumps, there may be missing files. As TEM will try to delete or modify those files, it will run into an error if any of them is missing (find the messages in `%TC_DATA%/model/delta.xml`)
- Be sure to use the correct TEM for the desired Teamcenter installation: check that it shows the correct **Installation Directory** (`%TC_ROOT%`) in the **Select Features** window (see above). If not, exit TEM and start the one from the correct Teamcenter directory: `%TC_ROOT%/install/tem.bat` or `tem.sh`.
- As a result the following library `lib` must be included under the **TC\_customization\_libraries** preference in Teamcenter
- The TEM instance under `%TC_ROOT%/install` will only install the client extensions to the portal installation under `%TC_ROOT%/install`. If your installation has a separate 4-tier RAC installation or your host has only a client installation, the TEM instance from that specific installation can be used to install the client extensions of the template. In this TEM run, you only need to select template "<PL4x> for Rich Client".

## 5.4.2 Set PL4x GS Environment for a Teamcenter 2-Tier Environment

In order to integrate PL4x functionalities (i.e., PL4x workflow handlers) into Teamcenter, Teamcenter needs to know the PL4x environment and PL4x libraries which should be loaded during Teamcenter start. As Teamcenter clears the environment settings while processing its start-up scripts (`portal.bat`, `start_imr.bat`...), the call of PL4x environment file has to be done immediately before the start of the Teamcenter Server process.

Open the file `%TC_ROOT%\iopservers\start_TcServer1.bat` (the number and file extension may be different depending on the platform and installation) in a text editor and add the line to call the PL4x environment before the line to start Teamcenter Server:

- For Windows: `call <T4x_GS_ROOT>\etc\t4x_env.bat`
- For Linux/Unix: `. <T4x_GS_ROOT>/etc/t4x_env.sh`

In case you have more than one Teamcenter database in your Rich Client installation, there is more than one file like that, e.g., *start\_TcServer1.bat* and *start\_TcServer2.bat*... Be sure to do this modification in the file(s) corresponding to the database(s) where PL4x should be used.

**Caution:**

As a Teamcenter 2-Tier environment, the adjustment of the Teamcenter start script should be done to all Teamcenter clients.

### 5.4.3 Set PL4x GS Environment for a Teamcenter 4-Tier Environment

In order to integrate PL4x functionalities (i.e., PL4x workflows handlers) into Teamcenter, Teamcenter needs to know the PL4x environment and PL4x libraries which should be loaded during Teamcenter start. In a Teamcenter 4-Tier environment, the PL4x environment file should be called during pool or net server manager start.

In a Teamcenter 4-Tier environment, we recommend editing the start script file: *%TC\_ROOT%\pool\_manager\confs\config1\tcenv.bat*. The following PL4x environment file should be called after executing the file *tc\_profilevars.bat* and before starting Teamcenter Server:

- For Windows: `call <T4x_GS_ROOT>\etc\t4x_env.bat`
- For Linux/Unix: `. <T4x_GS_ROOT>/etc/t4x_env.sh`

### 5.4.4 Add PL4x Error Message Texts to Teamcenter

The PL4x error messages are stored in an additional file separate from other Teamcenter error messages (which are stored in the files *ue\_errors.xml*). It is *sap\_errors.xml*. This file is copied to the target Teamcenter folder during PL4x template deployment.

The PL4x error messages use their own number range (starting from 212000) within the Teamcenter error handling, so there will be no conflicts with other error messages.

PL4x supports several languages that are selected automatically according to the Teamcenter GUI language: if it is started in a language that is not supported by PL4x, the PL4x GUI and its error messages will be presented in English.

Use the Teamcenter environment variable *TC\_MSG\_ROOT* or *TC\_USER\_MSG\_DIR* to control where Teamcenter will search for error message files (e.g., *%TC\_USER\_MSG\_DIR%\en\_US\sap\_errors.xml*). If these are not found there, Teamcenter will use the default directory *%TC\_ROOT%\lang\textserver\en\_US*.

## 5.5 Set Teamcenter Connection from PL4x

The standard default connection to Teamcenter is configured by the following procedure:

```
set State [::ITK::setConnectionParameters <user> <password> <group> \
        <?EncryptFlag?> <?LocalOnlyFlag?>]
```

## 5.6 Configure Enterprise Application for PL4x

For the information about the configuration of for PL4x, please refer to the .

## 5.7 Configure Active Integration (PL4x) for TLS/SSL

PL4x supports TLS (Transport Layer Security) only. The term SSL (Secure Sockets Layer) may be used for simplification as those two terms are often used interchangeably. The usage of TLS/SSL encryption for PL4x is optional and depends on your requirements. However, a properly installed and tested BGS and GS are required before you begin. Furthermore, a basic knowledge of TLS/SSL and how to obtain valid certificates is assumed, as a complete description of TLS/SSL, certificates and certificate authorities is beyond the scope of this manual.

Configuring PL4x for TLS is currently supported for PL4x 11.4 and higher only. The TLS implementation of PL4x is based on the OpenSSL libraries and using TLS version 1.2 solely. Please be aware, that TLS/SSL based communication between PL4x and Teamcenter server is currently limited to the following Teamcenter versions (64-bit only) on Linux and Windows only:

- Teamcenter 12
- Teamcenter 11.5
- Teamcenter 11.4
- Teamcenter 11.3
- Teamcenter 11.2.3

### Caution:

It may happen that you lose the connection to the PL4x server due to misconfiguration, so that you will not be able to fix the configuration using the Admin UI. Therefore, it is highly recommended to backup your configuration before changing any encryption settings by copying the file <T4x\_BGS\_ROOT>/var/conf/tpds.overlay or <T4x\_GS\_ROOT>/var/conf/tpds.overlay respectively.

## 5.7.1 Certificates

### Caution:

PL4x provides some self-signed demo certificates out of the box, which are *not* secure and have to be replaced with your own for production use. These demo certificates are bound to the *localhost* domain name and will not work for installations on separate hosts. Active Integration can not and does not provide any certificates for your installation or any consulting on how to obtain these certificates, as this has to match the detailed IT and security requirements of your organization. Your organization may use an independent certificate authority or use certificates generated by a third-party vendor. Please contact your IT support to obtain valid certificates, accordingly.

PL4x requires X.509 pem encoded certificates using the \*.pem file extension. Other files with no or a different extension will not be shown in the UI and cannot be used during the configuration. The server and client certificates used need to contain the public certificate and its associated private key (usually the key is inserted before the certificate). The private key of the certificate file *must not* be encrypted, as PL4x does not support specifying a pass phrase at the moment.

The CA certificate has to contain the whole chain of CA certificates to verify the validity of the server or client certificate. Usually the certificates defined in the CA certificate begin with the most specific one (the one nearest to the server or client certificate) and end with the most generic one, i.e., the one closest to the certificate root.

If you are using client authentication for the **ADMIN\_UI20** server instance, you have to import your client certificate to Firefox (PKCS#12 format) or the OS certificate storage (PEM format), depending on the browser you use. For detailed information on the needed formats and how to import and use those certificates, please consult the documentation of your operating system and/or web browser.

To check the properties of your certificates before configuration follow these steps:

1. Check that each certificate has a \*.pem file extension. PEM encoded certificates can also have the file extension \*.cer or \*.crt, therefore it is necessary to check the content of the file as mentioned in the next step. Since PL4x will only view \*.pem files in the Admin UI you have to replace or add this file extension if necessary.

2. Open the certificate file using a text editor and check that each of the following sections can be found once in the server and client certificate:

```
-----BEGIN RSA PRIVATE KEY----- ... -----END RSA PRIVATE KEY-----
-----BEGIN CERTIFICATE----- ... -----END CERTIFICATE-----
```

If you cannot read the content of the file then this is probably not a PEM encoded file. The certificate will not work in PL4x if one of the sections is missing in the file.

The CA certificate (chain) file has to contain one or more certificate sections, but no private key sections.

3. If necessary you can use the following OpenSSL commands - assuming that OpenSSL is installed in your test system - to check the properties of your certificates in details. For more information on OpenSSL please consult the official website at <https://www.openssl.org/>.

- a. `openssl x509 -inform PEM -in yourCertificate.pem`  
`openssl rsa -inform PEM -in yourCertificate.pem`  
 These commands can be used to test if your certificate and your private key contained in the certificate file are actually PEM encoded. If working correctly the content of the certificate or private key is printed between the tags mentioned in section 2. above. If the certificate or private key is missing or in the wrong format an *unable to load* error message is shown.
- b. `openssl verify -CAfile yourChain.pem yourCertificate.pem`  
 Verifies that your server or client certificate has been issued by the CA defined in your CA certificate (chain). The output has to end with *OK*.
- c. `openssl rsa -check -noout -in yourCertificate.pem`  
 Check the consistency of the private key contained in the certificate file. *RSA key ok* indicates that the private key is correct, otherwise *RSA key error* is shown.
- d. `openssl x509 -noout -dates -in yourCertificate.pem`  
 Prints the validation dates of the certificate. Make sure that the current date is in between those dates. Otherwise the certificate is already expired or not valid yet.
- e. `openssl x509 -in yourCertificate.pem -noout -pubkey | openssl md5`  
`openssl rsa -in yourCertificate.pem -pubout | openssl md5`  
 The output of both commands has to match exactly to make sure that the public key contained in the certificate section matches the public key portion contained in the private key section. Otherwise the wrong private key has been copied to the wrong certificate file.
- f. `openssl x509 -noout -modulus -in yourCertificate.pem | openssl md5`  
`openssl rsa -noout -modulus -in yourCertificate.pem | openssl md5`  
 The output of both commands has to match exactly to make sure that the public and private key of your file form a matching key pair.
- g. `openssl crl2pkcs7 -nocrl -certfile yourChain.pem | openssl pkcs7 -print_certs -noout`  
 Prints the subject and issuer chain in the order contained in the CA certificate (chain) file. The recommended order is from the most specific certificate to the most generic root (or most proximate to the root) certificate. Though PL4x does not consider the order of the certificates in the CA certificate (chain) file you have to make sure that the chain is complete without any gaps.
- h. To view the content of the different certificates as human-readable text you can use the following commands, depending on the file format:  
 PEM encoded file: `openssl x509 -noout -text -in yourCertificate.pem`  
 PKCS#12 (i.e. \*.p12) file: `openssl pkcs12 -info -in yourCertificate.p12`

Place your certificate files in the `<T4x_BGS_ROOT>/etc/cert` and `<T4x_GS_ROOT>/etc/cert` respectively to make them available for PL4x and the configuration dialog of the Admin UI.

**Caution:**

Since the private key portion of the certificate files is not encrypted you should make sure that the `cert` folders are only accessible for PL4x, i.e., for the OS user operating PL4x.

## 5.7.2 Server Authentication

Using server authentication (also known as *standard authentication* or *one-way TLS*) the client (e.g., your GS or web browser) verifies the certificate sent by the server (e.g., your BGS or GS) before continuing any communication. You can choose to use server authentication for BGS and/or GS as well as which server instances are affected.

### 5.7.2.1 Configuring Server Authentication for the BGS

For enabling server authentication in the BGS the following certificates have to be available:

- The BGS server certificate has to be placed in the `<T4x_BGS_ROOT>/etc/cert` directory.
- The corresponding CA certificate (chain) file has to be available in the BGS (`<T4x_BGS_ROOT>/etc/cert`) and each connected GS (`<T4x_GS_ROOT>/etc/cert`).

Follow these steps to configure server authentication in the BGS Admin UI:

1. Open **Configuration** → **Server instances** and edit all server instances that should send a certificate for verification. For each server instance enable the **Encryption** in the dialog for editing and select the BGS server certificate in the **Server certificate** editing dialog. **Apply** the changes to close the pop-up and proceed with the next server instance if needed.  
To enable server authentication for the default BGS (web)services edit the properties of the **SERVER** server instance. To operate the Admin UI using TLS, edit the **ADMIN\_UI20** server instance.
2. Since the BGS needs to be able to communicate with itself properly, e.g. when running a script, you have to modify the properties for the communication additionally. The settings of the **SERVER** server instance configured in step 1. have to match the properties of the communication channels **DEFAULT** and **DEFAULT\_WEB**.  
Open the **Configuration** → **Communication channels** settings, select **Use advanced settings** for the **Configuration mode** if needed and modify these communication channels in the shown table:
  - a. Edit the **DEFAULT** channel. Enter the correct **Host**, i.e., the one that is used in the certificates. Switch the **Transport mode** to **Encrypted socket (TLS/SSL)** and select the corresponding CA certificate (chain file) in the **CA certificate** editing dialog.
  - b. **Apply** the changes to close the pop-up.
  - c. Repeat the configuration for the **DEFAULT\_WEB** communication channel, that is used for the URL composition of certain web services. Enter the **Host**, select **HTTPS (TLS/SSL)** for **Transport mode** and select the same CA certificate (chain file) for **CA certificate**.



- d. **Apply** the changes to close the pop-up.
3. **Apply** all changes and restart the BGS.
4. Not only the BGS, but also each connected GS has to know the properties for a secure communication with the BGS, i.e., the settings of the **SERVER** server instance of the configured BGS have to match the properties of the communication channels **BGS** and **BGS\_WEB** in each connected GS.  
Therefore, open the **Configuration** → **Communication channels** settings in the Admin UI of each connected GS, switch to the advanced **Configuration mode** if necessary and modify these communication channels:
  - a. Edit the **BGS** channel. Enter the correct **Host** of the BGS, i.e., the one that is used in the certificates. Switch the **Transport mode** to **Encrypted socket (TLS/SSL)** and select the corresponding CA certificate (chain file) in the **CA certificate** editing dialog.
  - b. **Apply** the changes to close the pop-up.
  - c. Repeat the configuration for the **BGS\_WEB** communication channel, that is used for the URL composition of certain web services. Enter the **Host** of the BGS, select **HTTPS (TLS/SSL)** for **Transport mode** and select the same CA certificate (chain file) for **CA certificate**.
  - d. **Apply** the changes to close the pop-up.
5. **Apply** all changes and restart the GS.

### 5.7.2.2 Configuring Server Authentication for the GS

For enabling server authentication in the GS the following certificates have to be available:

- The GS server certificate has to be placed in the `<T4x_GS_ROOT>/etc/cert` directory.
- The corresponding CA certificate (chain) file has to be placed in the `<T4x_GS_ROOT>/etc/cert` directory, too.

To enable TLS in the GS follow the steps beneath - depending on your platform - to rename some needed libraries properly. Users of Teamcenter 12 have to replace any files and commands containing "tc11" with "tc12" accordingly, e.g., `tpncitc12.dll` instead of `tpncitc11.dll`.

- For Windows backup and remove `tpncitc11.dll`, `tpncitc11.pdb`, `tpcoretc11.dll` and `tpcoretc11.pdb` in the `<T4x_GS_ROOT>\bin64` directory. Rename all four `tp*tc11tls.*` files in the same directory to remove "tls" from the file name. For example, rename `tpncitc11tls.dll` to `tpncitc11.dll`.
- For Linux backup and remove `libtpncitc11.so` and `libtpcoretc11.so` in the `<T4x_GS_ROOT>/lib64` directory. Rename `libtpcoretc11tls.so` in the same directory to `libtpcoretc11.so`. Open a command shell, navigate to this directory and create a soft link for the other library executing `ln -s libtpncitc11tls.so libtpncitc11.so`.

Follow the steps 1. to 3. as described in [Configuring Server Authentication for the BGS](#), thereby replacing BGS with GS, i.e., using the GS Admin UI, GS server certificate and restarting the GS in the end to configure server authentication for the GS.

### 5.7.2.3 Testing and Troubleshooting Server Authentication

To test the configuration start your BGS and GS using the `bin64/debug` executable. The debug start keeps the command shell open so that you can see all log messages in the shell directly. This way you can see error log messages, even if you cannot access the Admin UI to read log files (e.g., due to misconfiguration).

To test the server authentication configuration of the Admin UI open a web browser and try to access it via `https://<bgs-host-address>:<bgs-ui-port>` or `https://<gs-host-address>:<gs-ui-port>` respectively. Make sure that you use the correct host address of the BGS/GS in the URL, which has to be the same as used in the server certificates. For example, a certificate issued for the domain `my.test.domain.com` will show a certificate error in the browser if you try to access the Admin UI using `https://localhost:11320`.

Open **Script** → **Scripts** in the BGS and GS Admin UI and run the **Test Communication Channels** script to confirm the correct configuration. Check that all test cases have been completed successfully.

There is one test case for each main communication channel, i.e., **DEFAULT** and **DEFAULT\_WEB** for BGS and GS and additionally **BGS** and **BGS\_WEB** in the GS. If one or more test cases are failing, check the configuration of the according communication channel again. Additionally, have a look at the **tpbgs64\_netd.log** and **tpapps64\_netd.log** log files in the BGS Admin UI **Log files** → **System** or the debug command shell of your BGS/GS for any error log messages.

For some detailed error messages and hints to solutions please refer to the chapter [Troubleshooting](#).

## 5.7.3 Client Authentication

Using client authentication (also known as *mutual authentication* or *two-way TLS*) not only the client verifies the certificate of the server, but also the server (e.g., your BGS or GS) requests and verifies the certificate of the client (e.g., your GS or web browser) before continuing any communication. You can choose to use the server authentication for BGS and/or GS as well as which server instances are affected.

### 5.7.3.1 Configuring Client Authentication for the BGS

For enabling client authentication in the BGS make sure you have configured [server authentication in the BGS](#) successfully.

Additionally to the certificates needed for server authentication the BGS client certificate has to be available in the BGS (`<T4x_BGS_ROOT>/etc/cert`) and each connected GS (`<T4x_GS_ROOT>/etc/cert`).

Follow these steps to configure client authentication in the BGS Admin UI on top of the server authentication:

1. Open **Configuration** → **Server instances** and edit all server instances that should request and verify a client certificate. Edit each relevant server instance and select the according CA certificate (chain file) in the **CA certificate** editing dialog. **Apply** the changes to close the pop-up and proceed with the next server instance if needed.  
To enable client authentication for the default BGS (web)services edit the properties of the **SERVER** server instance. To operate the Admin UI using client authentication in the browser, edit the **ADMIN\_UI20** server instance.
2. As for the server authentication, the BGS needs to be able to communicate with itself properly, so you have to modify the properties for the communication additionally. The settings of the **SERVER** server instance configured in step 1. have to match the properties of the communication channels **DEFAULT** and **DEFAULT\_WEB**.  
Open the **Configuration** → **Communication channels** settings and edit each of these communication channels. Select the previously copied client certificate in the **Client certificate** editing dialog and press **Apply** to close the pop-up and continue with the next channel.
3. **Apply** all changes and restart the BGS.
4. Again, each connected GS has to know the properties for a secure communication with the BGS, i.e., the settings of the **SERVER** server instance of the configured BGS have to match the properties of the communication channels **BGS** and **BGS\_WEB** in each connected GS. Therefore, open the **Configuration** → **Communication channels** settings in the Admin UI of each connected GS and edit those communication channels to select the client certificate in the **Client certificate** editing dialog.
5. **Apply** all changes in each connected and edited GS and restart it.

### 5.7.3.2 Configuring Client Authentication for the GS

For enabling client authentication in the GS make sure you have configured **server authentication in the GS** successfully.

Additionally to the certificates needed for server authentication the GS client certificate has to be placed in the `<T4x_GS_ROOT>/etc/cert` directory.

Follow the steps 1. to 3. as described in **Configuring Client Authentication for the BGS**, thereby replacing BGS with GS, i.e., using the GS Admin UI, GS client certificate and restarting the GS in the end to configure client authentication for the GS.

### 5.7.3.3 Testing and Troubleshooting Client Authentication

Similar to the server authentication tests start your BGS and GS using the `bin64/debug` executable.

To test the client authentication configuration of the Admin UI you have to import the client certificate to the browser or OS certificate storage first. For more information on how to do this, please refer to the documentation of your web browser or operating system. Afterwards open your web browser and try to access the Admin UI via `https://<bgs-host-address>:<bgs-ui-port>` or `https://<gs-host-address>:<gs-ui-`

port> respectively. The browser will ask you which client certificate you want to use for this page. If everything works correctly, the login page will be shown.

Open **Script** → **Scripts** in the BGS and GS Admin UI and run the **Test Communication Channels** script again to confirm the correct configuration. Check that all test cases have been completed successfully.

If one or more test cases are failing, check the configuration of the according communication channel again. Additionally, have a look at the **tpbgs64\_netd.log** and **tpapps64\_netd.log** log files in the BGS Admin UI **Log files** → **System** or the debug command shell of your BGS/GS for any error log messages.

For some detailed error messages and hints to solutions please refer to the chapter **Troubleshooting**.

## 5.7.4 Encrypted Logging

Any log message sent from a log client to the log server is *not* encrypted by default. PL4x provides two different basic log configurations, which can be configured to send messages encrypted. For performance reasons it is highly recommended to use the logging via UDP, if there is not any need to force PL4x running on TCP only (e.g., due to firewall configuration).

Though the log messages sent between log client and server are encrypted no matter which technique you use, the log files themselves stored in the log root use an unencrypted binary format.

If it is required to run PL4x completely encrypted, i.e., all server instances are using TLS/SSL and log messages are sent encrypted, it is recommended to configure TLS/SSL first, before modifying the log configuration. Configure server or client authentication for all server instances except **LOG\_SERVER** first and make sure it is working properly. Afterwards enable and test the encrypted logging. This way it is easier to receive and view the error messages in the log files occurring during the configuration of TLS/SSL. Any misconfiguration of the log encryption might result in log lines and files being skipped and hence you will not be able to debug other issues.

### 5.7.4.1 Configuring Symmetric Encryption for Logging

By default PL4x is configured to use logging via UDP (User Datagram Protocol), a protocol which does not guarantee that every sent message is actually received, but provides a high performance. Therefore, it is highly recommended to prefer this method. To encrypt log messages sent by PL4x via UDP symmetric encryption is used, i.e., the sender and receiver use the exact same password (shared secret) to encrypt and decrypt messages. To enable the encryption in your log server and clients follow these steps:

1. Configure the log server to run in encrypted mode, i.e., the BGS expects all log messages that are received to be encrypted. Open **Configuration** → **Server instances** in the BGS Admin UI and edit the **LOG\_SERVER** server instance. Turn the **Encryption** on and enter a common password used for log server and clients in the **Shared secret** box (e.g., `my-P4ssw0rd`). **Apply** the settings to close the pop-up.

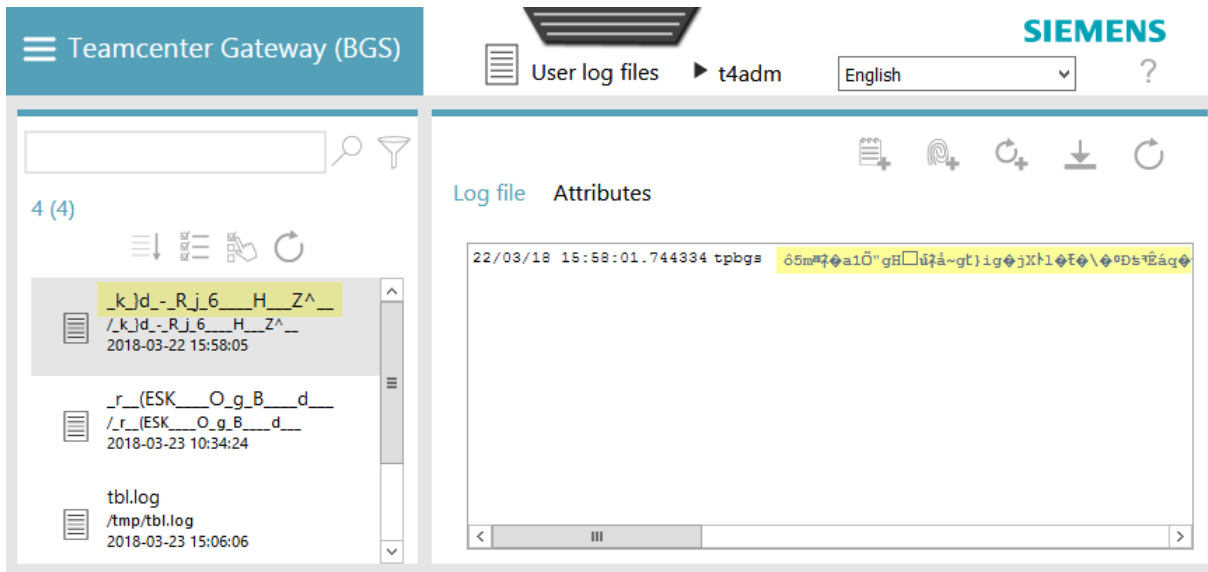
**Caution:**

A log server with log encryption enabled ignores any unencrypted log messages received and vice versa, encrypted log messages cannot be decrypted by a log server without log encryption and are discarded.

2. Since the BGS is not only a log server but also a client, additional settings have to be modified to make sure that messages can be logged.
  - a. Open **Configuration** → **Communication channels** in the BGS Admin UI. If you have not modified any advanced communication channel settings yet, you have to switch the **Configuration mode to Use advanced settings** first, before editing the **LOG** communication channel shown in the table.
  - b. Set the **Transport mode to Encrypted socket (shared secret)** and enter the exact same password as used for the log server in the **Shared secret** text box (e.g., `my-P4ssw0rd`). **Apply** the new settings to close the pop-up.
  - c. Apply all your changes and restart the BGS.
3. If the log configuration of the BGS has been tested successfully (see below), repeat the step 2. above for all GS installations connected and logging to this BGS.

To test the log communication after restart, login to the Admin UI of the BGS, open **Log** → **System** and check the most recent content (consider the timestamps in front of each log line) of several log files. For example, check the `tpbgs64*.log` log channels as the BGS is usually logging to these during the start. Similar you can check the `tpapps64*.log` log channels for each relevant GS in the same menu. Make sure that log lines have been written recently and can be read.

If you do not see any new log lines or channels, the configuration of the log server (server instance) and client (communication channel) do not match. Check the configuration again, make sure that the encryption of the server instance and communication channel is turned on and that both are using the exact same shared secret. Additionally run some tests to produce log lines (e.g., execute any test scripts) and do not only check if proper log files have been created where expected, but also check the content of the **Log** → **User** menu in the BGS Admin UI. It should not contain any log channels with cryptic names and content as shown in the image beneath. If you do see any of these log files one of your clients is using a different shared secret than the log server and hence producing unusable log content. Check the configuration to find the log client which is either logging cryptic content or not at all to fix its log configuration.



### 5.7.4.2 Configuring Logging via HTTP using TLS

If necessary (e.g., due to firewall settings) PL4x can be configured to log via TCP (Transmission Control Protocol) or more precisely HTTP instead of UDP. This is not the recommended way of logging since the performance decreases in contrast to UDP.

It is possible to enable encryption when using this option, by using HTTPS instead of HTTP for the log communication channel. In contrast to the UDP logging each log client does actually not send the log messages to the **LOG\_SERVER** but to the **SERVER** server instance of the BGS instead. Hence, when using this approach the complete BGS communication settings are affected. Follow these steps to enable encrypted logging using HTTPS:

1. Open **Configuration** → **Server instances** in the BGS Admin UI and modify the **SERVER** server instance to enable **server** or **client authentication** as described in previous chapters.
2. Since the BGS is not only a log server but also a client, additional settings have to be modified to make sure that messages can be logged.
  - a. Open **Configuration** → **Communication channels** in the same BGS Admin UI, switch the **Configuration mode** to **Use advanced settings** if needed and modify the **LOG** communication channel in the table shown.
  - b. Select **HTTPS (TLS/SSL)** for the **Transport mode** and the correct CA certificate (chain) file of the BGS in the **CA certificate** editing dialog to enable server authentication for the log communication. In case of client authentication you have to select the proper BGS client certificate in the **Client certificate** editing dialog. **Apply** your changes to close the pop-up.
  - c. Apply all your changes and restart the BGS.

3. If the log configuration of the BGS has been tested successfully (see beneath), repeat step 2. for all GS installations connected and logging to this BGS.

To test the log communication start the BGS using the `<T4x_BGS_ROOT>/bin64/debug` executable, so that you can check error messages in the command shell directly. Open **Log** → **System** in the BGS Admin UI and check the most recent content (consider the timestamps in front of each log line) of several log files. For example, check the `tpbgs64*.log` log channels as the BGS is usually logging to these during the start. Similar you can check the `tpapps64*.log` log channels for each relevant GS in the same menu.

If you do not see any new log lines or channels, there might be something wrong with the configuration. Check the output of the command shell for any TLS/SSL error messages. For some detailed TLS/SSL error messages and hints to solutions please refer to the chapter [Troubleshooting](#).

## 5.7.5 Troubleshooting

In this chapter some indications for typical errors occurring during the configuration and usage of TLS/SSL and encrypted logging are provided. If you are not able to access the BGS Admin UI or read any log files stop your BGS/GS and start it again using the `<T4x_BGS_ROOT>/bin64/debug` or `<T4x_GS_ROOT>/bin64/debug` executable. The debug executable will start your BGS/GS as usual but keep a command shell open showing the most recent log messages.

### SSL/TLS handshake failures in the log output

Any error shown in the `tpbgs64_netd.log`, `tpapps64_netd.log` or in the command shell can provide some more detailed information and hint regarding the error. Here are some messages you can encounter and what could potentially be a reason for their appearance. However, this list is not exhaustive. Other failure conditions may result in similar error messages.

- `SSL error:140760FC:SSL routines:SSL23_GET_CLIENT_HELLO:unknown protocol`  
Unencrypted communication with an encrypted server. Check if the communication channel is configured correctly.
- `SSL error:14094418:SSL routines:ssl3_read_bytes:tlsv1 alert unknown ca`  
Server certificate is damaged or a wrong CA certificate (chain) file is configured in the communication channel. Check the validity of your server certificate and if it matches the CA certificate you are using.
- `SSL error:14089086:SSL routines:ssl3_get_client_certificate:certificate verify failed`  
Client certificate could not be verified against CA certificate configured in the server.
- `SSL error:14094415:SSL routines:ssl3_read_bytes:ssl3 alert certificate expired`  
Server certificate is expired. Renew your certificates.
- `SSL error:14094412:SSL routines:ssl3_read_bytes:ssl3 alert bad certificate`

Probably the Subject Alternative Name of the server certificate is not matching the host configured in the according communication channel. Make sure that the correct host name is used in the communication channel and in the server certificate.

- `SSL error:0906D06C:PEM routines:PEM_read_bio:no start line`  
`SSL error:140B0009:SSL routines:SSL_CTX_use_PrivateKey_file:PEM lib`  
 Server certificate does not contain the private key. Make sure the server and client certificates contain a certificate and private key section.
- `SSL error:0906D06C:PEM routines:PEM_read_bio:no start line`  
`SSL error:140DC009:SSL routines:SSL_CTX_use_certificate_chain_file:PEM lib`  
 Server certificate file exists but it does not contain a certificate section. Make sure the server and client certificates contain a certificate and private key section.
- `SSL error:02001002:system library:fopen:No such file or directory`  
`SSL error:20074002:BIO routines:FILE_CTRL:system lib`  
`SSL error:140DC002:SSL routines:SSL_CTX_use_certificate_chain_file:system lib`  
 The selected certificate cannot be found. Check if the file exists in the `<T4x_BGS_ROOT>/etc/cert` or `<T4x_GS_ROOT>/etc/cert` respectively. Check the spelling of the file, maybe it has been renamed.

## Crash of the Teamcenter server

If the Teamcenter server crashes during the login with the following error

```
Teamcenter server failed to respond: .
```

This is most likely the result of a crash of the Teamcenter server, or the Teamcenter server encountering an unrecoverable error. Please check the Teamcenter server syslog for more details.

and you are using HTTP(S) instead of (encrypted) socket for the **DEFAULT** communication channel of the GS, you have to rename some libraries in the `<T4x_GS_ROOT>/bin64` or `<T4x_GS_ROOT>/lib64` directory. Follow the steps to rename the libraries described in [Configuring Server Authentication for the GS](#).

## Hanging PL4x asking for pass phrase

A hanging PL4x process showing `Enter PEM pass phrase:` in the command shell indicates that a certificate with an encrypted private key is used. Hence, a pass phrase would be needed during the start of each worker. Currently, PL4x does not support an encrypted private key in the certificate files.

## Certificate is not visible in the Admin UI

Make sure that the certificate file is available in `<T4x_BGS_ROOT>/etc/cert` or `<T4x_GS_ROOT>/etc/cert` respectively. Check that the file has a `*.pem` file extension. If the files have just been copied after the configuration pop-up in the Admin UI has already been opened, close and reopen the pop-up to refresh the list of available certificate files.



## 5.8 Install More Than One BGS on the Same Host

It is possible to install more than one BGS on the same host. To do so, perform the following steps:

- Modify your first BGS to use for all configured server instances (**SERVER**, **LOG\_SERVER** and **ADMIN\_UI20** by default) other ports than the default ports (otherwise, the second one will also try and start on the same ports, which will lead to errors). Do not forget to modify the corresponding settings for **Communication channels** if needed.
- Restart it by executing its **bin64/restart** or the Admin UI function to take this modification into account.
- If it has been installed as a Windows service, there is no need to modify that service.
- Install and configure the second BGS in a separate directory using the **PL4x Installer**.
- Start the second BGS by executing its **bin64/restart**.
- You can now run both BGSes in parallel and access their Admin UIs by using their different port numbers in the URL.

For information on how to change the ports of a BGS installation, please see chapter **Change the Port of BGS Instances**.

## 5.9 Start Active Integration (PL4x) BGS and GS as Windows Service (Windows only)

In Windows platforms, the start of PL4x BGS and GS may be included into an auto start queue, i.e., Windows "Startup" folder, or executed as "Windows service". The Windows service has the advantage that it will run whenever the system is running; no matter which user is logged on.

- To install a Windows service for the PL4x BGS

```
sc create PL4x_BGS binPath= "<T4x_BGS_ROOT>\bin64\t4xservice.exe 123"
start= auto
```

- To install a Windows service for the PL4x GS

```
sc create PL4x_GS binPath= "<T4x_GS_ROOT>\bin64\t4xservice.exe 123"
start= auto
```

Instead of *restart.exe*, the executable file **t4xservice.exe** should be used to start BGS and GS as Windows Services.

The parameter 123 in `binPath` is only a dummy. To define a Windows Service, you have to provide a parameter for the executable. For PL4x BGS and GS, this parameter does not matter and is not used. You can give any text to this parameter.

More information about how to create, update and delete Windows Service, please refer to Windows Service Controller help page: <https://technet.microsoft.com/de-de/library/bb490995.aspx>

Caution:

- Depending on your Windows security guide lines, it may happen that Windows will prevent executing any file, if the service is running as the Windows **Local Service** user (default Windows service setting).

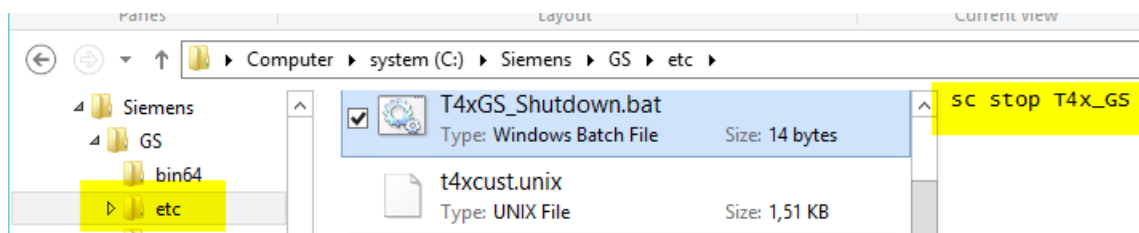
If PL4x BGS and GS start does not work as Windows Services, please try starting the same Windows Services within a command shell and check the error message. If some error message like *ERROR: invalid command name "rcwd"* is shown, please modify the service **Log On** to an administrator instead of **Local Service** user.

- In a Teamcenter 2-Tier environment, we do not recommend starting PL4x GS as a Windows Service. PL4x GS should be started and stopped together with Teamcenter.

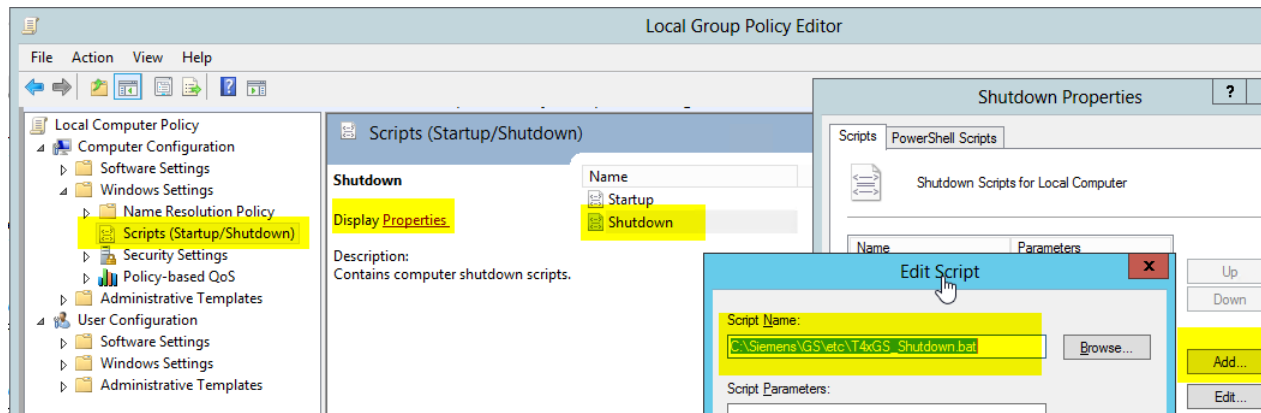
## 5.10 Stop Active Integration (PL4x) BGS and GS Service with Script (Windows only)

In order to stop PL4x BGS and GS services cleanly in a Windows environment, create a script to stop BGS and GS and add it in the Windows **Local Group Policy Editor** following these steps:

- Define a batch script to shutdown the service for BGS and GS. This script contains the command to stop the service only:



- Start `gpedit.msc`
- Open **Windows Settings** → **Scripts**.
- Select **Shutdown**.
- Open the **Properties** and **Add** the shutdown script.



## 5.11 Installation of additional components

### 5.11.1 Install a JDBC Driver to connect to a database

If you want to use JDBC connections to a database, you must first install a JDBC driver. The PL4x delivery does not contain any JDBC drivers. You must download each required driver on your own and accept the licensing agreements. To install a JDBC driver for PL4x, follow these instructions:

Download the JDBC driver or find it in the database installation. Here are some download URLs for common JDBC drivers:

- Oracle: <http://www.oracle.com/technetwork/indexes/downloads/index.html>. Click the link "JDBC drivers". Then select your database version, accept the license agreement, log in with your Oracle account and download the thin driver for JDK 6.
- Microsoft SQL Server: <http://msdn.microsoft.com/en-US/sqlserver/>. Click the link "JDBC Driver for SQL Server", then click "Download the Microsoft JDBC Driver for SQL Server" and then follow the download instructions. If you need an older version of the driver, review the links near the bottom of the page.
- MySQL: <http://dev.mysql.com/downloads/connector/j/>. See the "Archives" link for older versions.

Copy the jar file of the JDBC driver to the `<T4x_GS_ROOT>/lib/modules` directory.

Your JDBC driver can now be used from the PL4x mapping and test scripts.



# 6. PL4x Job Server Installation

## 6.1 PL4x Job Server Configuration

The PL4x job server is part of the PL4x BGS installation and therefore does not require a separate installation. It manages the jobs using a pool that caches all jobs. Furthermore, the BGS contains the management interface for the PL4x job server and the jobs. Therefore, it is also called job master. Whenever this documentation mentions the PL4x job server, it might imply client functionality as well. Therefore, this chapter describes the complete configuration of the PL4x job functionality – including the steps on the server and on the client side.

In the BGS Admin UI, select the menu entry **Configuration** and the topic **Job server** in the sidebar. Three settings, as displayed in the screen shot beneath are shown.

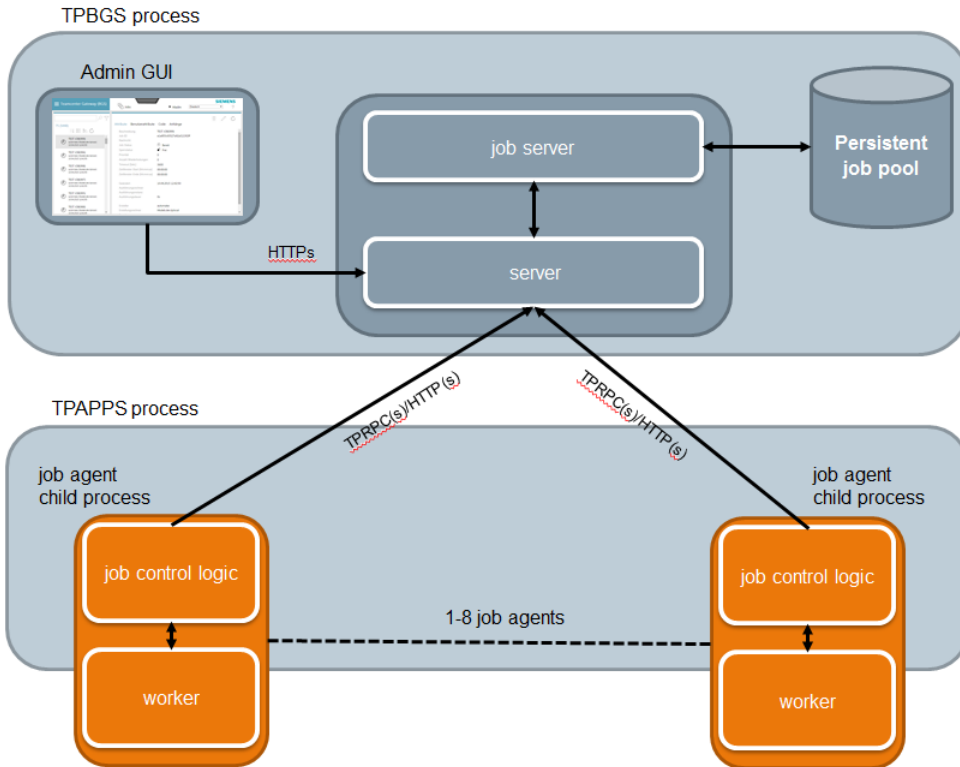
### Job server

Storage path	<input type="text" value="C:/UGS/BGS_cr3638/var/pool"/>
Storage time (in days)	<input type="text" value="7"/>
Maximum number of jobs	<input type="text" value="100000"/>

- **Storage path:** path to the folder where the PL4x job server stores the jobs. Default is: `<T4x_BGS_ROOT>/var/pool`.
- **Storage time (in days):** defines how long executed jobs are stored in the pool. The job pool is cleaned up regularly (every three minutes). All jobs which are in the state Finished, Application Error or Runtime Error and which are older than the storage time defined here are removed.
- **Maximum number of jobs:** maximum number of jobs in the job pool.

PL4x jobs are executed neither by the "tpbgs" nor by a "tpapps" process but rather by individual job agent processes that will be started as child processes by "tpapps". Each GS may handle up to eight of those job agents. In principle, a BGS can handle a very high number of job agents, but we recommend not using more than 128 job agents with one BGS. If you need more job agents, please use additional PL4x BGS installations.

The following figure shows the main interactions between job agents and GS ("tpapps") / BGS ("tpbgs"):

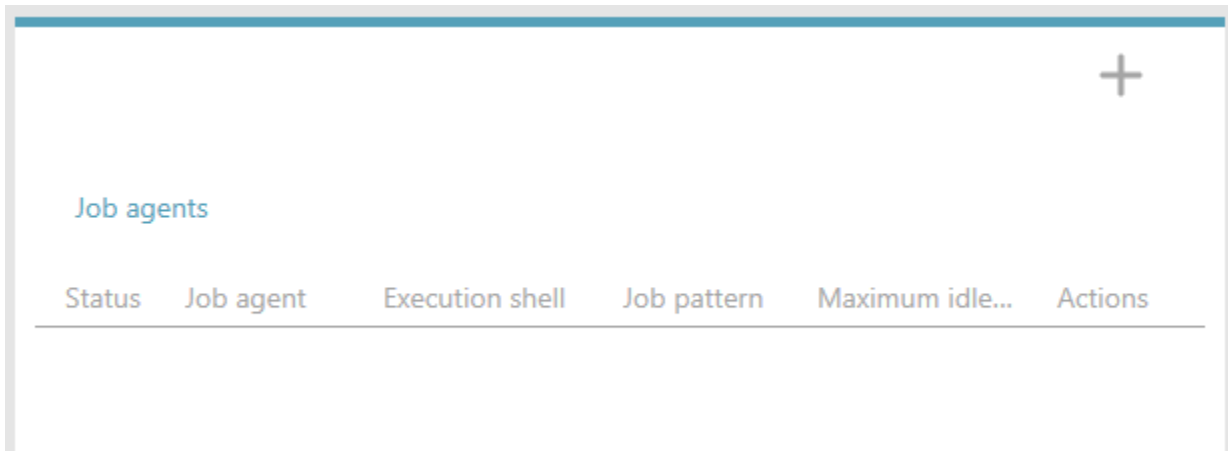
**Caution:**

Changing the **Maximum number of jobs** to a smaller number can only be done if the number of the currently stored jobs is much smaller than the new pool size.

## 6.2 PL4x Job Agent Configuration

Before doing anything in order to use a PL4x GS as a job agent, be sure to have its standard configuration completed, i.e., it has to be able to connect to the BGS, etc.

In the GS Admin UI, click **Configuration** → **Job agent**. By default, an empty table is displayed as shown in the screen shot beneath, which means that there is no job agent yet. So this GS does not *execute* any job.



To create a new job agent instance, click on the plus icon in the upper right corner of the screen, which will open a new pop-up window to configure a new agent (see screen shot beneath).

The 'Add a new job agent' dialog box contains the following settings:

- Status:** Radio buttons for 'Active' (selected) and 'Inactive'.
- Job pattern:** Radio buttons for 'Execute GS-jobs only', 'Execute all jobs' (selected), and 'Expert mode'.
- Maximum idle memory size (in MB):** A text input field containing the value '128'.

At the bottom right, there are two buttons: 'Cancel' and 'Apply'.

The following settings can be specified for each job agent:

- **Status:** defines whether the agent is activated or not.
  - **Active:** By default, an agent is activated.
  - **Inactive:** This setting can be used to deactivate an agent without losing its settings. An inactive job agent does not actually exist and does not process any job; the BGS will not even try to assign one to it.
- **Job pattern** defines the type of jobs this agent may execute, e.g., if not all of them have a correct Teamcenter environment.
  - Use **Execute all jobs** to allow this job agent to execute any job.
  - Use **Execute GS-jobs** to allow this job agent to execute only jobs having the appropriate ERP flag set.

- Use **Expert mode** to allow this job agent to execute only jobs that match an **Expert pattern** which can be specified. This expert pattern is matched against a job property value like the job description or the job filter. If there is a match, the job agent is allowed to execute that job.  
For proper functionality, different jobs have to be designed with different key words in their descriptions or filter attributes in order to be distinguished by the Job Master.  
Use `*` for any and `?` for one or more occurrences of unknown (wild-card) characters. For example, the pattern `*ar?` would match the key words `start`, `star1`, `care`, `car5`, `park` and `art`, but not `arch` or `warehouse`.  
Enter a comma separated list of patterns to enable this job agent to process multiple patterns. For example, if you enter `car*,wheel*` the job agent will first process all jobs that match the first pattern `car*`. If no more jobs matching this pattern have to be processed, the next pattern `wheel*` will be matched.

In fact this Job pattern setting does not allow this job agent to execute a specific job, but it tells the job master to assign a pending job to this job agent or not.

- **Maximum idle memory size (in MB):** in some cases a job leaves some memory allocated. In order to prevent the amount of blocked memory from growing continuously, define the maximum size allowed before the job agent will be restarted so that its memory will be released. The recommended setting is 128 MB in Windows or 256 MB in UNIX, respectively.

For the first tests of job server, we recommend setting as easy and few restrictions as possible.

- **1 job agent only**
- **Execute all jobs**
- **128 MB**

Be sure to do all the basic testing with those easy settings before modifying them as desired, because complex settings may result in complex error tracking.



**Caution:**

- The number of active job agents is the number of rows with status **activated**. Each GS can host up to eight job agent instances working independently, but in most cases, using only one is recommended. Consider the quantity of jobs to expect and decide on the required number of instances.
- To completely remove a job agent instance (not only deactivate it), click the "delete" icon in the table row of the respective agent.
- In order to take the modifications into account, you have to click the "apply" button in the upper right corner to save the changes and restart the GS.
- Afterwards, you should find three instead of two tpapps processes running, because the third one is the job agent process (one additional for each job agent)
- If you have activated the "external workers", you may find some more processes called "tpapps".
- The created job agent will now be visible in the **Job management** → **Agents** screen of the BGS Admin UI of the corresponding job server. Depending on network load, etc., this may take up to two minutes.

## 6.3 Set up Teamcenter Multi Connect for PL4x Jobs

It is possible to work with different connection data for different types of jobs or for the jobs executed by a specified job agent.

For configuring the default connection to Teamcenter, see chapter [Set Teamcenter Connection from PL4x](#).

In addition, it is possible to set the Teamcenter connection data in the context of the job processing, so that you can use different Teamcenter users for different job processing. The detailed connection data configuration is specified by the following two specific procedures:

- for import and transfer jobs : :ITK::MULTI::CONNECT::setConnectParameters4Batch
- for workflow jobs : :ITK::MULTI::CONNECT::setConnectParameters4WF

For a detailed description of the input parameters, see .

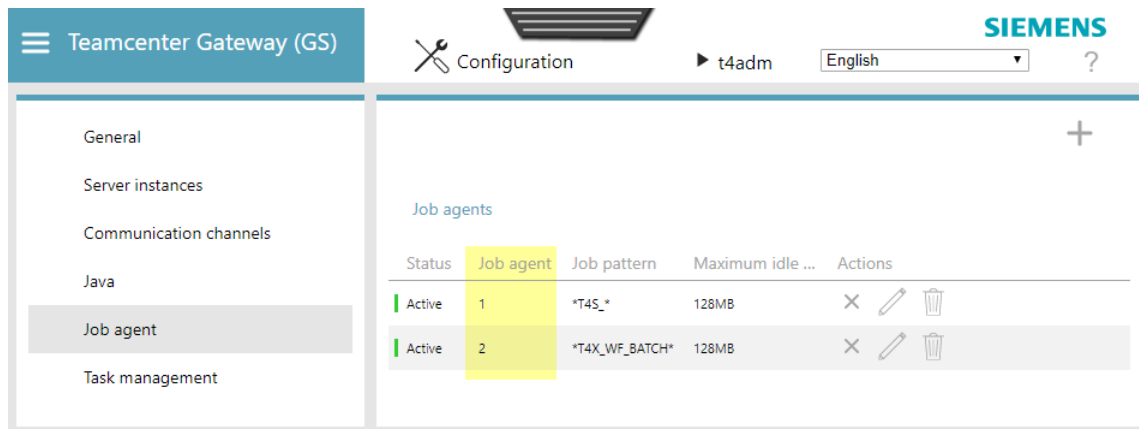
The mandatory parameters define the connection data itself, while the optional parameters define for which jobs this data shall be applied. This means that a call with only three parameters is valid for all kinds of jobs and the data is used as default if no more specific data is found. If no default data is set this way, the default is taken from the settings done by calling : :ITK::setConnectionParameters. If

no connection data was set using `setConnectParameters4Batch`  
 or `::ITK::setConnectionParameters`, an attempt to connect via auto connect will be done.

The following example shows how to use different Teamcenter users for different job agents:

```
set rc [::ITK::MULTI::CONNECT::setConnectParameters4Batch testuser1
testpw1 dba 0]
```

```
set rc [::ITK::MULTI::CONNECT::setConnectParameters4Batch testuser2
testpw2 dba 1]
```



#### Caution:

The number to set in this function call parameter `BatchClient` is the internal Job Agent number. As counters begin with zero in TCL, the internal job agent number is the external job number (shown in the Admin UI) minus one. Job agents are numbered and listed in the UI in creation order.

Job agent 0 (i.e. external job agent number 1 in the Admin UI) executes all "<T4x>\_" jobs with "testuser1" as Teamcenter user, whereas job agent 1 (i.e. external job agent number 2 in the Admin UI) processes all "T4X\_WF\_BATCH" jobs as Teamcenter user "testuser2".

Additionally, standard and user attributes of jobs can be used to set different connection data. For example:

- Execute jobs with a priority of 30 by any job agent:
 

```
::ITK::MULTI::CONNECT::setConnectParameters4Batch testuser testuser dba *
PRIO 30
```
- Execute jobs having a description starting with "<T4x>\_TRANSFER\_" to be executed by any job agent:
 

```
::ITK::MULTI::CONNECT::setConnectParameters4Batch infodba infodba dba
* DESC \
"^T4S_TRANSFER_.*"
```

- Execute jobs having a description containing "\_IMPORT\_" to be executed by any job agent:

```
::ITK::MULTI::CONNECT::setConnectParameters4Batch infodba infodba dba
* DESC \
".*_IMPORT_.*"
```

- Execute jobs with a description beginning with "<T4x>\_TRANSFER\_" by the second job agent (internal job agent number 1) using the "user1" connection data:

```
::ITK::MULTI::CONNECT::setConnectParameters4Batch user1 user1 dba 1
DESC \
"^T4S_TRANSFER_.*"
```

The process starts the check for the connection data from the most detailed job agent and attribute settings and iterates through the settings for job agents only down to default. The first fitting connection setting found is used.

#### Setting the connection parameters for workflow jobs

using `::ITK::MULTI::CONNECT::setConnectParameters4WF` works a bit different, since jobs created by a workflow have an attribute "ITK\_TC\_USER\_ID" filled with a user name of the reviewer or assigner (depending on workflow). At the beginning of the job execution, the matrix with the connection data will be checked for any data defined for the user name. If none is found, the default data will be used. In case of default data not being set, an attempt for auto login will be made.



# 7. Troubleshooting with Active Integration (PL4x) Process Start

If PL4x BGS or GS could not be started, please check the following points:

- First check for the `<T4x_BGS_ROOT>/tmp/scs.lock` or `<T4x_GS_ROOT>/tmp/scs.lock` file. Some PL4x commands (especially start and stop) create a small file `scs.lock` in the `tmp` directory to prevent other commands accessing this process during that time. After the successful execution, this file is deleted. In some cases (mostly because of an improper process interruption, e.g., its command window has been closed before it finished), this file remains and then the PL4x processes will fail to start.  
Be sure there is no hanging process to start or stop a PL4x process, then just delete the file `scs.lock` and try again.
- The integrity of the shared memory file (`<T4x_BGS_ROOT>/var/pef/share.ca` or `<T4x_GS_ROOT>/var/pef/share.ca`) is checked whenever it is opened. As a consequence, if the shared memory is broken (e.g., if it was damaged by a previous crash), PL4x will not start anymore. If you execute the start via command shell you will see the following error message:  

```
ERROR: .../var/pef/share.ca integrity check failed. If this is a shared memory (share.ca) from a T4x version smaller than 11.3, it must be migrated. How to migrate: rename <t4x_root>/var/pef/share.ca to <t4x_root>/var/pef/share.ca_pre then execute "bin64/shmmig -in var/pef/share.ca_pre"
```

  
If you have updated from a previous installation, please migrate your shared memory (see ).  
If your shared memory has already been migrated or if this is a new installation, the error message signals that your shared memory is broken.  
If PL4x of version 11.3 or higher does not start due to a failed shared memory integrity check, you can run the following steps trying to repair the shared memory manually:
  1. Stop all processes.
  2. Execute `bin64/shmdump` in BGS or GS root directory.
  3. Open the file `tmp/shm.dump` using a text editor.
  4. Repair all entries beginning with `# CHECK => damaged` by editing them manually and save the file.
  5. Delete (or move) the damaged shared memory file `var/pef/share.ca`.
  6. Start `bin64/tpshell` and execute `source tmp/shm.dump`.

After the execution of all these steps a new `share.ca` file will have been created and PL4x will be able to run again.



# 8. Use Nagios to monitor the Active Integration (PL4x) Infrastructure

## 8.1 Nagios Introduction

Nagios (now known as Nagios Core) is free and open source software for monitoring systems, networks and infrastructure. For more information, please visit <https://www.nagios.org/>.

Nagios can be used for monitoring the PL4x infrastructure (e.g., the core server, log server, job server, job agents). Therefore, Nagios needs access to the PL4x installations, to servers as well as to clients. If Nagios is already used in that environment for monitoring IT services, it can be used for checking PL4x as well. PL4x provides these Nagios modules for monitoring:

- Base Server Module
- Log Server Module
- Job Server Module
- Job Agent Module

The PL4x modules are tested with Nagios core 3.4.1.

Usually, Nagios is used to monitor the PL4x BGS. Therefore, the Nagios modules are included in the BGS installation. However, if you want to monitor your GS, simply copy the file `<T4x_BGS_ROOT>/var/init/start.ngs_server` to the `<T4x_GS_ROOT>/var/init` directory to enable the base server module for your GS.

The following examples show how to test each module, which data is returned by PL4x and how to define the command in the Nagios configuration file `commands.cfg`. When using Windows, do `set TP_NCONHIDE=1` in the command shell before executing the Nagios module for test to keep the command shell open.

## 8.2 Base Server Module

The PL4x base server module works with the BGS and GS server. It monitors the memory and CPU usage of the server as well as the server call statistics. The following optional parameters can be passed:

```
-memuse    warninglevel;errorlevel memory usage in MB (default=0,0)
-calls     warninglevel;errorlevel application calls per minute
           (default=0,0)
-ecalls    warninglevel;errorlevel application error calls per minute
           (default=0,0)
```

```
-wcmd      warninglevel;errorlevel application calls in waiting queue
(default=0,0)
```

To test this module navigate to your BGS or GS directory and execute the following command in an OS command shell:

```
bin64/tps var/init/start.ngs_server
```

```
T4x Base OK - MEM=278.6 MB CPU=4% WCMD=0 1/m CALLS=0 1/m ERRCALLS=0 1/m
|
MEM=278.6 CPU=4 WCMD=0 C=0 EC=0
```

Nagios command (example):

```
Nagios command (example):
define command {
    command_name    check_t4xbase
    command_line    cd /etc/nagios/plugins/bgs &&
                   bin64/tps var/init/start.ngs_server
}
```

## 8.3 Log Server Module

The PL4x log server module works only with the BGS. The following optional parameters can be passed:

```
-pkg      warninglevel;errorlevel packets per minute (default=0,0)
-epkg     warninglevel;errorlevel errors per minute (default=0,0)
```

To test this module navigate to your BGS directory and execute the following command in a command shell:

```
bin64/tps var/init/start.ngs_log
```

```
T4x Log OK - TRAFFIC=0.01 MB/m PKG=27 1/m EPKG=0 1/m |
TRAFFIC=0.01 PKG=27 EPKG=0
```

Nagios command (example):

```
define command{
    command_name    check_t4xlog
    command_line    cd /home/joerg/work/t4x_bgs &&
```



```

        bin64/tps var/init/start.ngs_log
    }

```

## 8.4 Job Server Module

The PL4x job server module works only with the BGS. The following optional parameters can be passed:

```

-poolsz      warninglevel;errorlevel job-pool-size in % (default=0,0)
-ready      warninglevel;errorlevel number of ready jobs (default=0,0)
-running    warninglevel;errorlevel number of running jobs (default=0,0)
-waiting    warninglevel;errorlevel number of waiting jobs (default=0,0)
-apperror   warninglevel;errorlevel number of jobs in application error
(default=0,0)
-rterror    warninglevel;errorlevel number of jobs in runtime error
(default=0,0)
-finish     warninglevel;errorlevel number of finish jobs (default=0,0)

```

To test this module navigate to your BGS directory and execute the following command in a command shell:

```

bin64/tps var/init/start.ngs_batch

T4x job OK - POOLSZ=0% READY=1 RUN=0 WAIT=2 AERR=2 RERR=4 FIN=4 |
POOLSZ=0 READY=1 RUN=0 WAIT=2 AERR=2 RERR=4 FIN=4

```

Nagios command (example):

```

define command{
    command_name    check_T4xJobs
    command_line    cd /home/joerg/work/t4x_bgs &&
                   bin64/tps var/init/start.ngs_batch
}

```

## 8.5 Job Agent Module

The PL4x job agent module works only with BGS. This module does not provide any additional parameters.

To test this module navigate to your BGS directory and execute the following command in a command shell:

```

bin64/tps var/init/start.ngs_batchclient

```

```
T4x JobAgent CRITICAL - oregon (SunOS)/16612 winab100-64 (Windows NT)/  
3516  
winab100 (Windows NT)/520 win28ab91r2 (Windows NT)/3672 winab100  
(Windows NT)/2516  
winab100 (Windows NT)/2516 win2008-t43-83 (Windows NT)/2628 - BCLCNT=19  
| BCLCNT=19
```

### Nagios command (example):

```
define command{  
    command_name    check_t4xjobagent  
    command_line    cd /home/joerg/work/t4x_bgs &&  
                   bin64/tps var/init/start.ngs_batchclient  
}
```

# A. Glossary

## A

### Admin

is the term used in this document for people who install and configure Teamcenter and its components. This is in contrast to the “user” role.

### Apps

See "GS".

## B

### BGS

Basic Gateway Service.

### BMIDE

Teamcenter Business Modeler IDE (Integrated Development Environment).

### BOM

A Bill Of Materials is a list of the parts or components and their quantities that are required to build a product.

### BOP

The Bill Of Process describes a manufacturing process and lists the operations and steps with all their instructions, consumed materials, resources, work places and machines.

## D

### Dataview mark-up

is the language understood by the Dataview. The Dataview receives messages written in this language from the T4x server. Such messages can be formatted as XML or JSON. Normally users do not see such messages. They may however appear in log files or error messages. The so called prop mapping (e.g. *t4s\_prop\_mapping\_template.sd*) contains TCL commands that compose messages in the Data View mark-up.

## E

### EA

stands for Enterprise Application, any software or set of computer programs used by business users to perform various business functions in context of current integration's portfolio with Teamcenter.

**ECN**

The Engineering Change Notice can also be called an Engineering Change Note, Engineering Change Order (ECO), or just an Engineering Change (EC).

**EPM**

Enterprise Process Modeling.

**G****GRM**

The Generic Relationship Management provides a general way in which two objects can be associated via a relationship.

**GS**

Gateway Service, manages the communication between Teamcenter and the Enterprise Application.

**GUI**

Graphical user interface.

**I****IDGEN**

The IDGEN is a mechanism to get an external ID from the ERP system when assigning a Teamcenter ID.

**ITK**

The Integration Toolkit (ITK) is a set of software tools provided by Siemens PLM Software that you can use to integrate third-party or user-developed applications with Teamcenter.

**J****JDBC**

Java Database Connectivity is an application programming interface (API) for the programming language Java, which defines how a client may access a database.

**L****LOV**

List of Values.

## **M**

### **MFK**

Multi-key functionality in Teamcenter.

## **O**

### **OOTB**

Out of the box.

## **R**

### **RAC**

stands for Rich Application Client also referred to as rich client or portal.

## **S**

### **SSL**

Secure Sockets Layer.

## **T**

### **T4x**

The entire Teamcenter Gateway product family.

### **TC**

Teamcenter

### **TCL**

is a high-level, general-purpose, interpreted, dynamic programming language.

### **TEM**

Teamcenter Environment Manager.

## **U**

### **UOM**

UOM stands for Unit of Measure.

### **URI**

Unified Resource Identifier: a generalized form of a resource locator (URL) and resource name (URN), which just identifies a resource, but is not necessarily sufficient to locate (find) the resource. URIs are

often used to identify configurations in Java and other languages. See [https://en.wikipedia.org/wiki/Uniform\\_Resource\\_Identifier](https://en.wikipedia.org/wiki/Uniform_Resource_Identifier) for more details.

## URL

Unified Resource Locator: a string with a certain format, allowing to load a resource from a network. URLs are a specific form or URNs.

## X

### XML

Extensible Markup Language is designed to store and transport data in a format that is both human- and machine-readable.

### XRT

stands for XML Rendering Template, also known as XML Rendering Stylesheet. These are XML documents stored in datasets that define how parts of the Teamcenter user interface are rendered. They are used for the Rich Client as well as the Active Workspace.

## Z

### Z-Table

"Z" is the prefix name for custom tables well-known in SAP world.



# Siemens Industry Software

## Headquarters

Granite Park One  
5800 Granite Parkway  
Suite 600  
Plano, TX 75024  
USA  
+1 972 987 3000

## Americas

Granite Park One  
5800 Granite Parkway  
Suite 600  
Plano, TX 75024  
USA  
+1 314 264 8499

## Europe

Stephenson House  
Sir William Siemens Square  
Frimley, Camberley  
Surrey, GU16 8QD  
+44 (0) 1276 413200

## Asia-Pacific

Suites 4301-4302, 43/F  
AIA Kowloon Tower, Landmark East  
100 How Ming Street  
Kwun Tong, Kowloon  
Hong Kong  
+852 2230 3308

## About Siemens PLM Software

Siemens PLM Software, a business unit of the Siemens Industry Automation Division, is a leading global provider of product lifecycle management (PLM) software and services with 7 million licensed seats and 71,000 customers worldwide.

Headquartered in Plano, Texas, Siemens PLM Software works collaboratively with companies to deliver open solutions that help them turn more ideas into successful products. For more information on Siemens PLM Software products and services, visit [www.siemens.com/plm](http://www.siemens.com/plm).

© 2018 Siemens Product Lifecycle Management Software Inc. Siemens, the Siemens logo and SIMATIC IT are registered trademarks of Siemens AG. Camstar, D-Cubed, Femap, Fibersim, Geolus, I-deas, JT, NX, Omneo, Parasolid, Solid Edge, Syncrofit, Teamcenter and Tecnomatix are trademarks or registered trademarks of Siemens Product Lifecycle Management Software Inc. or its subsidiaries in the United States and in other countries. All other trademarks, registered trademarks or service marks belong to their respective holders.