

Natural

Software AG Mainframe Product Licensing

Version 8.2.7

October 2017

This document applies to Natural Version 8.2.7 and all subsequent releases.

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

Copyright © 1979-2017 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

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

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

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

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

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

Document ID: NATMF-OETSMFLIC-827-20180201

Table of Contents

| | |
|--|----|
| 1 Software AG Mainframe Product Licensing | 1 |
| Distributed License Check Software | 2 |
| License File | 3 |
| Copying a License File to Disk | 4 |
| Transferring a License File from PC to a z/OS Host Using FTP | 4 |
| Transferring a License File from PC to a z/VSE Host Using FTP | 5 |
| Transferring a License File from PC to a BS2000 Host Using FTP | 7 |
| Product License Check FAQs | 8 |
| Using the License Utility: LICUTIL | 13 |

1 Software AG Mainframe Product Licensing

| | |
|--|----|
| ▪ Distributed License Check Software | 2 |
| ▪ License File | 3 |
| ▪ Copying a License File to Disk | 4 |
| ▪ Transferring a License File from PC to a z/OS Host Using FTP | 4 |
| ▪ Transferring a License File from PC to a z/VSE Host Using FTP | 5 |
| ▪ Transferring a License File from PC to a BS2000 Host Using FTP | 7 |
| ▪ Product License Check FAQs | 8 |
| ▪ Using the License Utility: LICUTIL | 13 |

This document describes Software AG's mainframe product licensing procedures, license check software and license file.

Notations Used in this Document:

- *vrs, vr* or *v.r.s*:
When used in this document, the notation *vrs*, *vr* or *v.r.s* represents the three-digit or two-digit version number of the Software AG product.
- *ppp*:
When used in this document, the notation *ppp* represents the three-character product code of the Software AG product: ADA for Adabas, NAT for Natural, EXX for EntireX, COM for Complete, WCP for Entire Net-Work and NPR for Entire System Server.

Distributed License Check Software

This section describes the data sets (files) containing this software. All licensing data sets have names starting with the characters MLC. The *vrs* in the names represents the version of the license check software, which is not necessarily the same as the version of your Software AG mainframe product.

For instructions on copying the data sets from the product installation medium to disk and loading the libraries contained on the data set, refer to the appropriate installation procedure for your mainframe product.

- [z/OS Licensing Data Sets](#)
- [z/VSE Licensing Data Sets](#)
- [BS2000 Licensing Data Sets](#)

z/OS Licensing Data Sets

The following licensing data sets are provided in z/OS environments:

| Data Set Name | Description |
|----------------------|--------------------------------------|
| MLC <i>vrs</i> .LOAD | Mainframe license check load modules |
| MLC <i>vrs</i> .JOBS | Mainframe license check sample jobs |

z/VSE Licensing Data Sets

The following licensing data sets are provided in z/VSE environments:

| Data Set Name | Description |
|----------------------|--------------------------------------|
| MLC <i>vrs</i> .LIBR | Mainframe license check load modules |
| MLC <i>vrs</i> .LIBJ | Mainframe license check sample jobs |

BS2000 Licensing Data Sets

The following licensing data sets (files) are provided in BS2000 environments:

| Data Set Name | Description |
|----------------------|--------------------------------------|
| MLC <i>vrs</i> .MOD | Mainframe license check load modules |
| MLC <i>vrs</i> .JOBS | Mainframe license check sample jobs |

License File

You must install a valid license file on all mainframe platforms in which your Software AG mainframe product is installed.

In a Natural z/OS environment, additional license files are required for Natural zIIP Enabler: NAZ*vrs*.LICS for batch/TSO, NCI*vrs*.LICS for CICS and NCF*vrs*.LICS for Com-plete.

The license file is provided as an XML document (encoding is US-ASCII). If you receive the license file as an email attachment save it directly to disk. To avoid unwarranted modification, do not save it from a browsing tool or text editor. For example, Internet Explorer might change the XML encoding tag value to its local encoding.

The XML document can be viewed using a browsing tool or text editor on a PC. It can also be viewed on the mainframe using the **DISPLAY** function of the license utility, LICUTIL, described later in this document. The license file contains text, which represents the licensing information and a digital signature, the license key. Among other things, it displays Software AG legal notices and environmental information.

 **Caution:** The license file must remain in ASCII format -- even on the mainframe. It must not be modified. Any modification of the license file will invalidate the digital signature and the license check will fail. If the check fails, please contact your Software AG technical support representative.

Delivery and Installation

Your individual product license is shipped on the product installation medium as *pppvrs.LICS* data set (file) or as an e-mail attachment in the file format *pppvr.xml*.

Once the license file is received, either copy it from the product installation medium to disk or use native FTP commands to transfer it to your mainframe host before using it during the mainframe product installation.

During the mainframe product installation process, the license file is then loaded and processed as required by your Software AG mainframe product. Refer to the appropriate installation procedure for your Software AG mainframe product for further information.

Copying a License File to Disk

For instructions on copying a *pppvrs.LICS* data set (file) from the product installation medium to disk, refer to the appropriate installation procedure for your Software AG mainframe product.

Transferring a License File from PC to a z/OS Host Using FTP

If a license file is supplied as an e-mail attachment, you must transfer the attached license file from the PC to the mainframe using native FTP commands provided in this section.

 **Caution:** Using utilities instead of native FTP commands for the license file transfer may corrupt the license key.

 **Important:** Make sure to switch to binary transfer and verify that the resulting data set has RECFM=FB or RECFM=F and LRECL=80.

➤ **To transfer a license file from the PC to a z/OS host, perform the following steps:**

- 1 Save the product license file e-mail attachment as *pppvr.xml* on your PC's hard disk.
- 2 Start an FTP session for communication with the z/OS host using the following FTP command:

```
ftp host-name
```

where *host-name* is the name of the z/OS host.

Enter your z/OS host login ID and password.

- 3 Switch to binary data mode (the license file must retain its format as ASCII during the transfer):

```
binary
```

- 4 Specify that the data set for the license file must be written with RECFM=FB and LRECL=80:

```
quote site RECFM=FB LRECL=80 BLKSIZE=4000
```

RECFM=F is also supported for product license files.

- 5 Write the license file as a data set on the z/OS system:

```
put pppvr.xml 'hilev.pppvrs.LICS'
```

where *hilev* is the high-level qualifier (for example, SAG) to be used for the data set.

This command will create a data set called *hilev.pppvrs.LICS* with RECFM=FB, LRECL=80, and the license information stored in the data set will be in ASCII format.

Transferring a License File from PC to a z/VSE Host Using FTP

If a license file is supplied as an e-mail attachment, you must transfer the attached license file from the PC to the mainframe using native FTP commands described in this section.

The FTP sample session described in these steps loads the license file onto a member of a z/VSE library. It is possible to load the license file onto a sequential z/VSE data set instead. For more information, refer to your FTP documentation.

-  **Caution:** Using utilities instead of native FTP commands for the license file transfer may corrupt the license key.
-  **Important:** Make sure to switch to binary transfer and verify that the resulting data set has RECFM=FB or RECFM=F and LRECL=80.

➤ To transfer a license file from the PC to a z/VSE host, perform the following steps:

- 1 Save the product license file e-mail attachment as *pppvr.xml* on your PC's hard disk.
- 2 Start an FTP session for communication with the z/VSE host using the following FTP command:

```
ftp host-name
```

where *host-name* is the name of the z/VSE host.

Enter your z/VSE host login ID and password.

- 3 Once the FTP session has been established, you can optionally delete the default working prefix. To determine what the default working prefix is, enter the following command:

```
pwd
```

To delete the default working prefix, enter the following command:

```
cd ..
```

- 4 Point to the z/VSE sublibrary for the license file, for example, USERLIB.MYSUBLIB:

```
cd USERLIB.MYSUBLIB
```

- 5 Switch to binary data mode (the license file must retain its ASCII format during the transfer):

```
binary
```

- 6 Specify that the data set for the license file must be written with RECFM=FB and LRECL=80:

```
quote site recfm fb 80
```

RECFM=F is also supported for product license files.

- 7 Write the license as a member in the libraries on the z/VSE system you identified in Step 4:

```
put pppvr.xml pppvrs.LICS
```

This command will write the license file *pppvr.xml* to a member called *pppvrs.LICS*. If you requested a library and sublibrary of USERLIB and MYSUBLIB in Step 4, the *pppvrs.LICS* file would reside in the USERLIB.MYSUBLIB library. The data set will have RECFM=FB, LRECL=80, and the license information stored in the member will be in ASCII format.

Transferring a License File from PC to a BS2000 Host Using FTP

If a license file is supplied as an e-mail attachment, you must transfer the attached license file from the PC to the mainframe using native FTP commands described in this section.



Caution: Using utilities instead of native FTP commands for the license file transfer may corrupt the license key.



Important: Make sure to switch to binary transfer and verify that the resulting data set has file type SAM and `RECFORM=V`.

» To transfer a license file from the PC to a BS2000 host, perform the following steps:

- 1 Save the product license file e-mail attachment as `pppvr.xml` on your PC's hard disk.
- 2 Start an FTP session for communication with the BS2000 host using the following FTP command:

```
ftp host-name
```

where `host-name` is the name of the BS2000 host.

Enter your BS2000 host user ID, password and account number (if relevant).

- 3 Switch to binary data mode (the license file must retain its ASCII format during the transfer):

```
binary
```

- 4 Specify that the file for the license file must be written with `FCBTYPE=SAM` and `RECFORM=V`:

```
quote file pppvrs.LICS,FCBTYPE=SAM,RECFORM=V
```

- 5 Write the license file to the BS2000 host:

```
put pppvr.xml pppvrs.LICS
```

Product License Check FAQs

- Why has Software AG introduced a license check for products on the mainframe?
- Which information is contained in a product license?
- What exactly is meant by the CPU ID?
- Which Software AG mainframe products require a product license?
- How is the product license file installed?
- How can I read the product license file?
- How can I display machine-specific data?
- When is the product license checked?
- Which items of a product license are checked?
- Does the Software AG mainframe license check software support a disaster recovery scenario?
- How do I find out when the product license expires?
- What happens if the product license is incorrect, insufficient or not installed?
- How can I get a new product license file if the delivered license file is insufficient for my environment?
- Do I need separate product licenses for different machines?
- If I use the same Natural environment-independent nucleus on different machines, how can I handle the license files?
- Is it possible to have the license module separate from the Natural environment-independent nucleus?

Why has Software AG introduced a license check for products on the mainframe?

Software AG wants to ensure that customers run our software products only on mainframe machines for which they have valid and sufficient product licenses. This ensures that we have better control of the use of our software products. On other platforms (UNIX, Windows), the product license check has been established for many years.

Which information is contained in a product license?

A product license is a sequential file with US-ASCII text in XML format containing the following items:

- Software AG header
- customer information (Name, ID)
- encrypted license key
- license expiration date (or unlimited)
- product information (product code, version, name)
- environment information, including operating system type, CPU ID, LPAR ID, system name, and capacity measured in million service units (MSUs)

What exactly is meant by the CPU ID?

IBM defines the CPU ID as the “central processing complex node descriptor sequence number”. It is the unique hexadecimal machine serial number without the machine model number.

The CPU ID and other machine data can be viewed by using the **DISPLAY** function of the license utility, LICUTIL, described later in this document.

Natural:

When Natural is installed and runs, you can also display CPU information by issuing the following Natural system command:

```
NATQVS
```

Which Software AG mainframe products require a product license?

- Adabas
- Natural
- EntireX
- Com-plete
- Entire Net-Work
- Entire System Server

How is the product license file installed?

Transfer the license file from the product installation medium or PC to disk as described in the installation documentation for your Software AG mainframe product or the various *Transferring a License File from PC to a xxx Host* section sections respectively. Then proceed as described in the appropriate installation steps for your mainframe product.

How can I read the product license file?

- The product license file can be read on a PC by means of an XML editor (file type .xml) or any PC text editor.
- The license utility, LICUTIL, provided with the Software AG mainframe license check software includes a function that allows you to display the license (**DISPLAY** function). For more information about the license utility, read *Using the License Utility: LICUTIL*.
- **Natural:**

When Natural is installed and runs, the license information can be displayed by using function **L (Natural License Information)** of the Natural SYSTP utility (refer to the Natural *Utilities* documentation).

How can I display machine-specific data?

The license utility, LICUTIL, provided with the Software AG mainframe license check software includes a function that allows you to display machine-specific data (**DISPLAY** function). For more information about the license utility, read [Using the License Utility: LICUTIL](#).

Natural:

When Natural is installed and runs, you can also display CPU information by issuing the following Natural system command:

```
NATQVS
```

When is the product license checked?

The product license is checked every time the product is initialized. In addition, the product license is checked once a day.

Thirty days before the license expires, license check failure messages will warn you that it is time to obtain a new license.

If a license check fails, contact your Software AG technical support representative.

Which items of a product license are checked?

Product license items are divided into three logical license groups (License Key, Product-Specific Information and Machine-Specific Information) and checked in that order. If an inconsistency in any one of these groups is detected by the license checker, the succeeding items of that logical license group are also checked. Then the check terminates with corresponding error messages. For example, if the license key expires, the license key check terminates with an error before the license checker checks any product-specific or machine-specific information.

The following table identifies the product license items comprising each logical group:

| Checking Order | Logical License Group Name | Product License Item Included |
|----------------|------------------------------|---|
| 1 | License Key | The encrypted license key |
| 2 | Product-Specific Information | <ul style="list-style-type: none"> ■ The expiration date (if any) ■ The operating system (z/OS, z/VSE or BS2000) ■ The product code (for example, ADA, NAT or EXX) ■ The product version, in either <i>v.r.</i> or <i>v.r.s</i> format. |
| 3 | Machine-Specific Information | <ul style="list-style-type: none"> ■ The machine CPU ID ■ The machine capacity measured in MSU (million service units); only in z/OS and z/VSE environments |

| Checking Order | Logical License Group Name | Product License Item Included |
|----------------|----------------------------|---|
| | | <ul style="list-style-type: none"> ■ The logical partition (LPAR) ID (if applicable) |

Does the Software AG mainframe license check software support a disaster recovery scenario?

Yes. Please contact your Software AG sales representative to obtain a license file which is able to support disaster recovery scenarios. The parameter DisasterRecoveryCPUIDs in this license file allows you to specify one or more CPU IDs of the machines to be used when your production machine fails. If you do not have these CPU IDs available, it is possible to specify the value `Unknown`. In this case, the license checker issues a warning message when a disaster recovery scenario is assumed.

In a disaster recover scenario, no machine data besides the CPU ID is checked.

How do I find out when the product license expires?

Thirty days before the license expires, license check failure messages will warn you that it is time to obtain a new license.

What happens if the product license is incorrect, insufficient or not installed?

An error message is issued and the product for which the license is checked either continues to run or terminates. The message is either output as a primary message or on the system console.

Natural:

When Natural is installed, there are three possible reactions: An initialization error message is issued and the Natural session terminates (1), an initialization error message is issued, but the Natural session continues execution (2), or a warning is issued on the system console (3).

1. An error message is issued on the user terminal and the session terminates in one of the following cases:
 - the product license module is missing or incorrect, or it has been modified
 - incorrect operating system, product code or product version
2. An error message is issued on the user terminal, but the Natural session continues execution:
 - the license expiration date has been reached.
3. The session starts but a warning message is issued on the system console in one of the following cases:
 - Your machine CPU ID is not defined in the license or the machine capacity is higher than the value specified in the product license.
 - The logical partition (LPAR) ID has been checked but is not defined in the license.

- The product license will expire within the next 30 days.

This behavior may change in a future version of Natural, that is, the Natural session will terminate after the warning message is issued.

The warning message is not repeated on the system console for every Natural session. It is issued only once per Natural buffer pool per day.

How can I get a new product license file if the delivered license file is insufficient for my environment?

Contact your Software AG sales representative to get a correct product license file according to your contract. It can be shipped by e-mail either in ASCII format or in the converted assembler input format.

Do I need separate product licenses for different machines?

No, it is possible to have multiple CPU IDs defined in one license file.

If I use the same Natural environment-independent nucleus on different machines, how can I handle the license files?

When Natural is installed, it is possible to have multiple CPU IDs defined in one license file. If all your CPU IDs are defined in your license file, you only need one license module that you link to all environment-independent nuclei. For detailed information, contact your Software AG sales representative.

Is it possible to have the license module separate from the Natural environment-independent nucleus?

Yes. When Natural is installed, the license module can be defined to be loaded dynamically during session start by means of the profile parameters `RCA` and `RCALIAS` (described in the *Natural Parameter Reference* documentation). Specify the following in the Natural parameter module or dynamically:

```
RCA=NATLIC,RCALIAS=(NATLIC, name)
```

where *name* is the name of your separately linked license module in the Natural load library.

Using this technique causes any statically linked license module to be replaced. This can be helpful for testing.

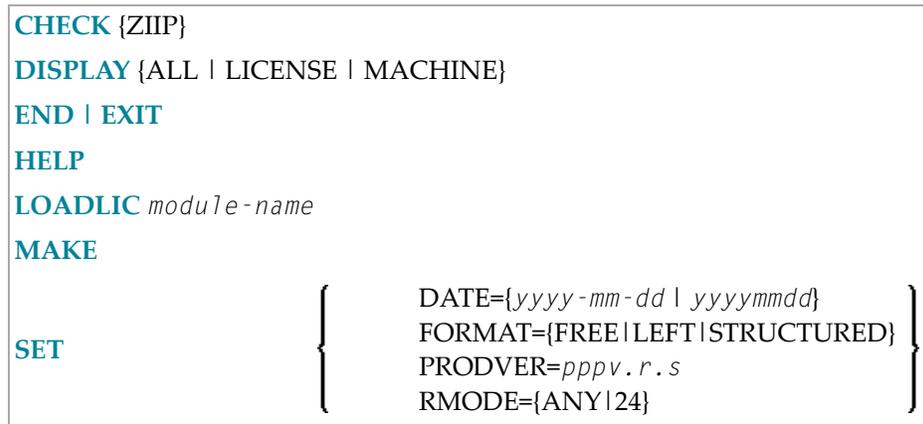
Using the License Utility: LICUTIL

The license utility, LICUTIL, is a batch utility that is used to check the license file, convert the license file into an assembler source module and display the license file and machine data.

LICUTIL is provided with Software AG's mainframe license check software contained on the data sets MLC *vrs*.LOAD (z/OS), MLC *vrs*.MOD (BS2000) and MLC *vrs*.LIBR (z/VSE).

Error messages for the license utility are documented with the rest of your Software AG product messages.

This section provides detailed information of each function provided by the LICUTIL utility. The functions can be executed by using the sample JCL described later in this section. The syntax of the functions that can be supplied with the LICUTIL command is shown in the following diagram:



The parameters and variable values that can be supplied with each function, are described in the following section. For explanations of the syntax symbols used in the diagram, please read the documentation for your specific Software AG mainframe product.

- CHECK Function: Check the Product License
- DISPLAY Function: Display License Items and Machine Data
- END/EXIT Functions: Terminate the License Utility
- HELP Function: View LICUTIL Help
- MAKE Function: Convert the License File to Assembler Source
- LOADLIC Function
- SET Function: Set LICUTIL Options
- JCL Examples and Requirements

- LICUTIL Licensing Messages

CHECK Function: Check the Product License

The CHECK function can be used to check the license file against the product's execution environment. The original license file must be provided as input along with the product code and version. If the license file is not valid, appropriate error messages are returned.

 **Important:** In order to run the CHECK function successfully, you must have previously specified the **SET PRODVER** function to provide the product code and version number. The parameter ZIIP must be specified if the license is for a zIIP enabling product, for example, Natural zIIP Enabler.

The following items are checked, in the following order:

1. License key.
2. Product-specific data, including the product code, version, operating system and expiration date.
3. Machine-specific data, including the CPU IDs, the machine capacity (measured in MSU, million service units) on only z/OS and z/VSE systems, and the logical partition (LPAR) ID.

DISPLAY Function: Display License Items and Machine Data

Use the DISPLAY function to display the contents of the product license items (license file, product-specific information and machine-specific information) and to display machine data (CPU ID, machine type, model ID, LPAR ID, MSU) about the machine on which the utility is being run.

The parameters of this function are described in the following table. At least one parameter is required.

| Parameter | Description |
|-----------|---|
| ALL | Display the product license items and machine data. ALL is the default. |
| LICENSE | Display only the product license items. |
| MACHINE | Display only the machine data. |

END/EXIT Functions: Terminate the License Utility

Use the END or EXIT function to terminate the license utility. These functions should be used if LICUTIL is running interactively in environments other than batch (for example, TSO under z/OS).

HELP Function: View LICUTIL Help

Use the HELP function to review a list of the available LICUTIL functions and their syntax.

MAKE Function: Convert the License File to Assembler Source

Use the MAKE function to convert the license file into an assembler source. After assembling this source (and linking it), the resulting license module can be used by defined Software AG products (for example, Adabas or Natural) instead of the license files. For more information about using license modules, refer to the installation documentation provided with your Software AG product.

Sample Output

Here is some sample output from this function when run on a z/OS system:

```
MLC9006 License converted to assembler output. 042 records written  
to SYSUT2.
```

LOADLIC Function

This function applies only on z/OS.

Use the LOADLIC function to load the license module after you converted (**MAKE** function), assembled and linked the assembler source. *module-name* denotes the name of the license module in the load library.

LOADLIC can be specified only once per LICUTIL session.

After the license module has been successfully loaded by LOADLIC, you can further process the module using other LICUTIL functions such as **CHECK** and **DISPLAY**.

SET Function: Set LICUTIL Options

Use the SET function to set options for the license utility.

The parameters of this function are described in the following table. At least one parameter is required.

| Parameter | Description |
|-----------|---|
| DATE | Sets the date for the license check. The default is today. The format of the date can be either <i>yyyy-mm-dd</i> or <i>yyyymondd</i> , where <i>yyyy</i> is the four-character year, <i>mm</i> is the two-digit month, <i>mon</i> is the three-character month abbreviation and <i>dd</i> is the two-digit day. |
| FORMAT | Sets the display format for the DISPLAY function. There are three different formats: <ul style="list-style-type: none"> ■ Specify FREE (the default) to produce output without XML tags. ■ Specify LEFT to produce output with each XML tag in a new line. ■ Specify STRUCTURED to produce output with each XML tag indented. |
| PRODVER | Sets the product code and version for the CHECK function. The format in which the product code and version should be specified is <i>pppv.r.s</i> . For a zIIP enabling product such as Natural zIIP Enabler, the product code can consist of up to five characters: <i>pppppv.r.s</i> . |
| RMODE | Sets the RMODE for the assembler source produced by the MAKE function: <ul style="list-style-type: none"> ■ Specify 24 (the default) for 24-bit addressing mode. ■ Specify ANY for either 24- or 31-bit addressing mode. |

JCL Examples and Requirements

Sample JCL of the license utility functions can be found in the following members of the licensing data sets (files): *MLCvrs.JOBS* for z/OS and BS2000, and *MLCvrs.LIBJ* for z/VSE.

| Member | Functions Included | Demonstrates how to... |
|----------|---|--|
| LICCHECK | SET PRODVER= <i>pppv.r.s</i> DISPLAY CHECK | Check a license file with LICUTIL. For a zIIP enabling product such as Natural zIIP Enabler, the product code can consist of up to five characters: <i>pppppv.r.s</i> . |
| LICMAKE | MAKE | Convert the license file to assembler source. |
| LICMDATA | DISPLAY MACHINE | Display machine data. This function can be used to send Software AG the required data for a license file. |

This section covers the following topics:

- [z/OS JCL Requirements](#)
- [z/VSE JCL Requirements](#)

- BS2000 JCL Requirements

z/OS JCL Requirements

The following file assignments are used by the license utility in z/OS environments:

| DD Name | Description |
|----------|--|
| SYSIN | Input of LICUTIL control parameters. |
| SYSPRINT | Output of LICUTIL run. |
| SYSUT1 | Input of original license file (<i>pppvrs.LICS</i> data set). |
| SYSUT2 | Output of license converted to assembler source. |

The following table shows which file assignments are required in the JCL by each LICUTIL function (an X in a table cell indicates that the file assignment is required; a blank cell indicates that file assignment is not required):

| License Utility Function | DD Name | | | |
|--------------------------|---------|----------|--------|--------|
| | SYSIN | SYSPRINT | SYSUT1 | SYSUT2 |
| CHECK | X | X | X | |
| DISPLAY ALL | X | X | X | |
| DISPLAY LICENSE | X | X | X | |
| DISPLAY MACHINE | X | X | | |
| MAKE | X | X | X | X |
| HELP | X | X | | |

z/VSE JCL Requirements

The following file assignments are used by the license utility in z/VSE environments:

| File Assignment | Description |
|-----------------|---|
| SYSIPT | Input of LICUTIL control parameters and the input of original license file (<i>pppvrs.LICS</i> data set). Note: The license file has to be stored in a sublibrary. |
| SYSLIST | Output of LICUTIL run. |
| SYSPCH | Output of license converted to assembler source. |

The following table shows which file assignments are required in the JCL by each LICUTIL function (an X in a table cell indicates that the file assignment is required; a blank cell indicates that file assignment is not required):

| License Utility Function | File Assignment | | |
|--------------------------|-----------------|---------|--------|
| | SYSIPT | SYSLIST | SYSPCH |
| CHECK | X | X | |
| DISPLAY ALL | X | X | |
| DISPLAY LICENSE | X | X | |
| DISPLAY MACHINE | X | X | |
| MAKE | X | X | X |
| HELP | X | X | |

BS2000 JCL Requirements

The following file assignments are used by the license utility in BS2000 environments:

| System File | Description |
|-------------|--|
| SYSDTA | Input of LICUTIL control parameters. |
| SYSOUT | Output of LICUTIL run. |
| Link Name | |
| SYSUT1 | Input of original license file (<i>pppvrs.LICS</i> data set). |
| SYSUT2 | Output of license converted to assembler source. |

The following table shows which file assignments are required in the JCL by each LICUTIL function (an X in a table cell indicates that the file assignment is required; a blank cell indicates that file assignment is not required):

| License Utility Function | File Assignment | | | |
|--------------------------|-----------------|--------|--------|--------|
| | SYSDTA | SYSOUT | SYSUT1 | SYSUT2 |
| CHECK | X | X | X | |
| DISPLAY ALL | X | X | X | |
| DISPLAY LICENSE | X | X | X | |
| DISPLAY MACHINE | X | X | | |
| MAKE | X | X | X | X |
| HELP | X | X | | |

LICUTIL Licensing Messages

The messages in this document may be produced when using the LICUTIL utility.

| | |
|--------------------|---|
| MLC1001 | CPU-ID {cpuid} is not defined in your product license |
| Explanation | The CPU ID listed in the message (<i>cpuid</i>) is not defined in your product license. Either your license file is invalid or you have attempted to run the Software AG mainframe product on a machine for which it is not licensed. |
| Action | Install and run the Software AG product on the machine for which it is licensed. If the problem persists, contact your Software AG sales representative for assistance. |
| MLC1002 | LPAR name {lparname} is not defined in your product license |
| Explanation | The LPAR name listed in the message (<i>lparname</i>) is not defined in your product license. Either your license file is invalid or you have attempted to run the Software AG mainframe product on a machine for which it is not licensed. |
| Action | Install and run the Software AG product on the machine for which it is licensed. If the problem persists, contact your Software AG sales representative for assistance. |
| MLC1003 | Machine capacity {value} higher than MSUs in your product license |
| Explanation | The machine or the partition capacity (<i>value</i>) of the CEC or LPAR respectively on which you have attempted to install and run the Software AG mainframe product exceeds the capacity allowed in your product license. Either your license file is invalid or you have attempted to run the Software AG mainframe product on a machine for which it is not licensed. |
| Action | Contact your Software AG sales representative for assistance. |
| MLC1004 | The product license will expire on {timestamp} |
| Explanation | This warning message provides the date and time (<i>timestamp</i>) when the license will expire for this Software AG mainframe product. |
| Action | No action is required for this warning message. However, you should contact your Software AG sales representative to obtain an updated license soon. |
| MLC1005 | We assume CPU-ID {cpuid} as your DR or DR testing environment |
| Explanation | The CPU ID listed in the message (<i>cpuid</i>) is not defined in your product license. Therefore we assume that you are running in a disaster recovery environment. |
| Action | If you are running in a disaster recovery environment, no action is required for this warning message. If you are not running in a disaster recovery environment, contact your Software AG sales representative to obtain an updated license. |

| | |
|--------------------|---|
| MLC2001 | The product license is invalid |
| Explanation | Your product license is invalid. |
| Action | Contact your Software AG sales representative for assistance. |
| MLC2002 | The product license key is invalid |
| Explanation | The license file for the Software AG mainframe product is invalid. |
| Action | Verify that the license file was transferred correctly to the mainframe host and that it is still in ASCII format. If the problem persists, contact your Software AG sales representative for assistance. |
| MLC2003 | The product license is not for {value} |
| Explanation | The license file for the Software AG mainframe product is not for the operating system (OS), product code, or product version listed in the message (<i>value</i>). |
| Action | Use the correct license file. If the problem persists, contact your Software AG sales representative for assistance. |
| MLC2004 | The product license has expired on {timestamp} |
| Explanation | The product license has expired. The date and time it expired is given in the message (<i>timestamp</i>). |
| Action | Contact your Software AG sales representative for assistance. |
| MLC2005 | Tag missing in product license: {tagname} |
| Explanation | The tag with the name listed in the message (<i>tagname</i>) is missing in your product license. |
| Action | Contact your Software AG sales representative for assistance. |
| MLC2006 | Invalid tag value in product license: {tagname} |
| Explanation | The value of the tag with the name listed in the message (<i>tagname</i>) is invalid in your product license. |
| Action | Contact your Software AG sales representative for assistance. |
| MLC2007 | Current CPU-ID not found for product license check |
| Explanation | The CPU ID of the CPU on which the product license check was run is not found in the license file. |
| Action | Contact your Software AG sales representative for assistance. |

| | |
|--------------------|---|
| MLC2008 | The number of MSU values in the product license is incorrect |
| Explanation | The machine capacity covered by the product license is incorrect. |
| Action | Contact your Software AG sales representative for assistance. |
| MLC2009 | ExtendedRights setting {value} not allowed under this system |
| Explanation | The ExtendedRights setting listed in the message (<i>value</i>) is not supported by your system. |
| Action | Contact your Software AG sales representative for assistance. |
| MLC2010 | Soft capping is not allowed with ExtendedRights {value} |
| Explanation | Your product license does not allow soft capping with the ExtendedRights setting listed in the message (<i>value</i>). |
| Action | Contact your Software AG sales representative for assistance. |
| MLC2011 | Defined hardware class in the license exceeded {value} |
| Explanation | The hardware class (PriceQuantity) in your product license is too small for the current machine (<i>value</i>). |
| Action | Contact your Software AG sales representative for assistance. |
| MLC2012 | The number of zIIPs ({value}) exceeds the license definition |
| Explanation | The number of zIIP processors in the current machine exceeds the number of zIIPs (PriceQuantity) in your product license. |
| Action | Contact your Software AG sales representative for assistance. |
| MLC2013 | The number of zIIP values in the product license is incorrect |
| Explanation | The number of zIIP values (zIIPs) in your product license does not match with the number of LPAR values (LPARs). |
| Action | Contact your Software AG sales representative for assistance. |
| MLC3001 | Invalid LCPCB length |
| Explanation | The license check routine has been incorrectly called. |
| Action | Contact your Software AG support representative for assistance. |

| | |
|--------------------|---|
| MLC3002 | Work area too small |
| Explanation | The size of the work area for the license check routine is too small. |
| Action | Contact your Software AG support representative for assistance. |
| MLC3003 | Invalid function code |
| Explanation | The license check routine has been incorrectly called. |
| Action | Contact your Software AG support representative for assistance. |
| MLC3004 | Invalid operating system |
| Explanation | The ID of the operating system passed through the license check routine is invalid. |
| Action | Contact your Software AG support representative for assistance. |
| MLC3005 | Invalid product code |
| Explanation | The product code passed through the license check routine is invalid. |
| Action | Contact your Software AG support representative for assistance. |
| MLC3006 | Invalid product version |
| Explanation | The product version passed through the license check routine is not numeric. |
| Action | Contact your Software AG support representative for assistance. |
| MLC3007 | Invalid current date |
| Explanation | The current date passed through the license check routine is invalid. The date must be in the format <i>yyyy-mm-dd</i> . The dash (-) delimiters are not checked but the numbers must be a valid year, month and day, respectively. |
| Action | Contact your Software AG support representative for assistance. |
| MLC3008 | Tag not found |
| Explanation | The license check routine cannot find the requested XML tag. |
| Action | Contact your Software AG support representative for assistance. |
| MLC3009 | Component Id tag not found |
| Explanation | The license check routine cannot find the XML tag <code>Component Id</code> . |
| Action | Contact your Software AG support representative for assistance. |

| | |
|--------------------|---|
| MLC3010 | Invalid search tag |
| Explanation | The XML search tag requested by the license check routine is invalid. |
| Action | Contact your Software AG support representative for assistance. |
| | |
| MLC3011 | Insufficient I/O area size |
| Explanation | The size of the I/O area for the license check routine is too small. |
| Action | Contact your Software AG support representative for assistance. |
| | |
| MLC3012 | Output record length too small |
| Explanation | The length of the output record for the license check routine is too small. |
| Action | Contact your Software AG support representative for assistance. |
| | |
| MLC3013 | Error message text not found |
| Explanation | The license check routine cannot find the text that belongs to the requested error message. |
| Action | Contact your Software AG support representative for assistance. |
| | |
| MLC3014 | Too many errors occurred |
| Explanation | The license check routine detected too many errors. Some messages are lost. |
| Action | Contact your Software AG support representative for assistance. |
| | |
| MLC9001 | License exceeds maximum size of 8000 |
| Explanation | Your license file is too large. |
| Action | Contact your Software AG sales representative for assistance. |
| | |
| MLC9002 | Error opening license file input |
| Explanation | An error occurred during an attempt to open the license file input. |
| Action | Investigate the cause of the error, paying special attention to the corresponding error messages produced by the operating system. Contact your Software AG sales or support representative for assistance. |

| | |
|--------------------|---|
| MLC9003 | Error reading license file input |
| Explanation | An error occurred during an attempt to read the license file input. |
| Action | Investigate the cause of the error, paying special attention to the corresponding error messages produced by the operating system. Contact your Software AG sales or support representative for assistance. |
| MLC9004 | Error opening license file assembler output |
| Explanation | An error occurred during an attempt to open the license file assembler output. |
| Action | Investigate the cause of the error, paying special attention to the corresponding error messages produced by the operating system. Contact your Software AG sales or support representative for assistance. |
| MLC9005 | Error writing license file assembler output |
| Explanation | An error occurred during an attempt to write the license file assembler output. |
| Action | Investigate the cause of the error, paying special attention to the corresponding error messages produced by the operating system. Contact your Software AG sales or support representative for assistance. |
| MLC9006 | License converted to assembler output records written to output file |
| Explanation | The license file was successfully converted to an assembler source module. Output records were written to the output file. |
| Action | No action is required for this informational message. |
| MLC9007 | Error loading license module {modulename} {errorcode} |
| Explanation | The LOADLIC function (z/OS only) tried to load a license module from the load library and received an error (<i>errorcode</i>) from the operating system. |
| Action | Correct the LICUTIL input, removing duplicate LICUTIL function calls. Check that the license module (<i>modulename</i>) is in the load library, that the correct library is referenced in the JCL, and that the module name is specified correctly in the LOADLIC statement (case sensitive, a blank between LOADLIC and <i>modulename</i>). |
| MLC9009 | DMS Error |
| Explanation | A DMS error occurred (only in BS2000 environments). |
| Action | Investigate the cause of the BS2000 DMS error. If you need assistance, contact your Software AG support representative. |

| | |
|--------------------|--|
| MLC9010 | LOADLIC can only be issued once, before any DISPLAY, CHECK, MAKE or READ |
| Explanation | The LOADLIC function (z/OS only) can only be performed once per LICUTIL session. LOADLIC must not be preceded by the DISPLAY, CHECK or READ function. These functions can only locate the license module <i>after</i> the LOADLIC. |
| Action | Correct the LICUTIL input, removing duplicate LICUTIL function calls. Change the call order so that the LOADLIC precedes any DISPLAY, CHECK or READ. |

