Natural ISPF

Mainframe Product Licensing

Version 9.2.3

September 2025

ADABAS & NATURAL

This document applies to Natural ISPF Version 9.2.3 and all subsequent releases.

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

Document ID: ISP-OETSMFLIC-923-20250930

Table of Contents

1 About this Documentation

Document Conventions

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

Online Information and Support

Product Documentation

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

Product Training

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

Tech Community

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

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

Product Support

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

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

Data Protection

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

I Mainframe Product Licensing

2 Mainframe Product Licensing

This document describes the 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 product.
- *ppp*:
  When used in this document, the notation *ppp* represents the product code. The product code *ppp* can have three to five characters.

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 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

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

Data Set Name	Description
MLC*vrs*.LOAD	Mainframe license check load modules
MLC*vrs*.JOBS	Mainframe license check sample jobs

License File

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

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

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 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 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 mainframe product. Refer to the appropriate installation procedure for your 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 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.

Product License Check FAQs

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

We want 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:

- company 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 mainframe products require a product license?

- Adabas and the following add-ons:
  - Adabas for zIIP (AZPAD), Event Replicator for zIIP (AZPRP), Adabas Cluster Services (ALS), Adabas Parallel Services (ASM), Adabas Caching Facility (ACF), Adabas Online Services (AOS), Adabas Delta Save (ADE), Adabas Review (REV)
- Natural and the following add-ons:
  - Natural for zIIP (NAZBT, NAZCI, NAZCO, NAZNI), Natural for Db2 for zIIP (NDZ), Natural Optimizer Compiler (NOC)
- Com-plete
- Entire Net-Work
- Entire System Server
- Jopaz
- Predict
- Natural ISPF
- Entire Output Management
- Entire Operations

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 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 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 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.

License check failure messages will warn you that it is time to obtain a new license every day for a set warning period.

For MLC versions 1.3.8 and lower, the warning period starts thirty days before your license expires.

For MLC versions 1.4.1 and higher, the warning period starts forty days before your license expires.

If a license check fails, contact your 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	■ The expiration date (if any) ■ The operating system (z/OS) ■ The product code (for example, ADA, or NAT) ■ The product version, in either *v.r.* or *v.r.s* format.

Checking Order	Logical License Group Name	Product License Item Included
3	Machine-Specific Information	■ The machine CPU ID ■ The machine capacity measured in MSU (million service units) ■ The logical partition (LPAR) ID (if applicable)

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

Yes. Please contact your 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?

You will receive a license check failure message that prompts you to get a new license every day for a set warning period.

For MLC versions 1.3.8 and lower, the warning period starts thirty days before your license expires.

For MLC versions 1.4.1 and higher, the warning period starts forty days before your license expires.

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

Natural:

An error message is issued and the product for which the license is checked stops. The license check error message is output as a primary message.

In case of Natural (product code NAT), the session initialization terminates with the following message:

```
NAT9966 Session terminated due to license check failure.
```

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

Contact your 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 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.

How to install an Emergency Key?

An Emergency Key is a special type of license file which can be valid for all products. This kind of license is meant to have a short duration and is normally issued by the Global Support.

The customer should convert the license to the specific product license formats, in the same way as with normal license files.

Is there a better way to install Emergency Key licenses?

Instead of creating one LIC module for each licensed product, you could create a single LIC module with ALIAS to all required products.

You can modify the last step of the LICMAKE sample job to build a multiple ALIAS module. In this example, we named it EMGLIC:

```
//L.SYSIN    DD  *
 ALIAS ADALIC
 ALIAS ASMLIC
 ALIAS AOSLIC
 ALIAS WTCLIC
 NAME EMGLIC(R)
/*  ↩
```

The result is a single module with an ALIAS for each product.

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 the mainframe license check software contained on the MLC*vrs*.LOAD data set.

Error messages for the license utility are documented with the rest of the 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:

CHECK {ZIIP}
DISPLAY {ALL | LICENSE | MACHINE}
END | EXIT
HELP
LOADLIC *module-name*
MAKE
SET { DATE={*yyyy-mm-dd* | *yyyymmdd*} | FORMAT={FREE|LEFT|STRUCTURED} | PRODVER={*pppv.r.s*|*ppp,v.r.s*} | RMODE={ANY|24} }

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, read the documentation for your specific mainframe product.

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. LICUTIL terminates with one of the following return codes:

Return Code	Meaning
0	The license check has been successful and the product continues to run.
4	Warning message(s) are written to the job log but the product continues to run.
8	Warning message(s) are written to the user terminal (or job log) but the product continues to run.
12	Error message(s) are written to the user terminal (or job log) and the product terminates.

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 Batch for zIIP.

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) 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).

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 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 the 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

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 *yyyy-mm-dd* or *yyyymondd*, where *yyyy* is the four-character year, *mm* is the two-digit month, *mon* is the three-character month abbreviation and *dd* is the two-digit day.
FORMAT	Sets the display format for the **DISPLAY** function. There are three different formats: ■ 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 *pppv.r.s* **or** *ppp,v.r.s*. The product code can consist of up to five characters.
RMODE	Sets the RMODE for the assembler source produced by the **MAKE** function: ■ Specify ANY (the default) for a 31-bit addressing mode. ■ Specify 24 for a 24-bit addressing mode.

JCL Examples and Requirements

Sample JCL of the license utility functions can be found in the following members of the MLC*vrs*.JOBS licensing data set:

Member	Functions Included	Demonstrates how to...
LICCHECK	SET PRODVER=*pppv.r.s* DISPLAY CHECK	Check a license file with LICUTIL. The product code can consist of up to five characters.
LICMAKE	MAKE	Convert the license file to assembler source.
LICMDATA	DISPLAY MACHINE	Display machine data. This function can be used to send the required data for a license file.

This section covers the following topics:

- z/OS 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 (*pppvrs*.LICS 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		

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 (*cpuid*) is not defined in your product license. Either your license file is invalid or you have attempted to run the mainframe product on a machine for which it is not licensed.
Action	Install and run the product on the machine for which it is licensed. If the problem persists, contact your sales representative for assistance.

MLC1002	**LPAR name {lparname} is not defined in your product license**
Explanation	The LPAR name listed in the message (*lparname*) is not defined in your product license. Either your license file is invalid or you have attempted to run the mainframe product on a machine for which it is not licensed.
Action	Install and run the product on the machine for which it is licensed. If the problem persists, contact your sales representative for assistance.

MLC1003	**Machine capacity {value} higher than MSUs in your product license**
Explanation	The machine or the partition capacity (*value*) of the CEC or LPAR respectively on which you have attempted to install and run the mainframe product exceeds the capacity allowed in your product license. Either your license file is invalid or you have attempted to run the mainframe product on a machine for which it is not licensed.
Action	Contact your sales representative for assistance.

MLC1004	**The product license will expire on {date}**
Explanation	This warning message provides the date (YYYY/MM/DD) when the license will expire for this mainframe product. The product cannot be started with the current product license after this date. For example, an expiration date of 2018/09/05 indicates that you can still work on 2018/09/05 until 23:59 (local time) but need a new license from the next day (2018/09/06).
Action	No action is required for this warning message. However, you should contact your 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 (*cpuid*) 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 sales representative to obtain an updated license.

MLC2001	**The product license is invalid**
Explanation	Your product license is invalid.
Action	Contact your sales representative for assistance.

MLC2002	**The product license key is invalid**
Explanation	The license file for the 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 sales representative for assistance.

MLC2003	**The product license is not for {value}**
Explanation	The license file for the mainframe product is not for the operating system (OS), product code, or product version listed in the message (*value*).
Action	Use the correct license file. If the problem persists, contact your sales representative for assistance.

MLC2004	**The product license has expired on {date}**
Explanation	The product license has expired. The date (YYYY/MM/DD) it expired is given in the message (*date*).
Action	Contact your sales representative for assistance.

MLC2005	**Tag missing in product license: {tagname}**
Explanation	The tag with the name listed in the message (*tagname*) is missing in your product license.
Action	Contact your 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 (*tagname*) is invalid in your product license.
Action	Contact your 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 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 sales representative for assistance.

MLC2009 **ExtendedRights setting {value} not allowed under this system**

Explanation The ExtendedRights setting listed in the message (*value*) is not supported by your system.

Action Contact your 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 (*value*).

Action Contact your 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 (*value*).

Action Contact your 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 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 sales representative for assistance.

MLC3001 **Invalid LCPCB length**

Explanation The license check routine has been incorrectly called.

Action Contact your 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 support representative for assistance.

MLC3003	**Invalid function code**
Explanation	The license check routine has been incorrectly called.
Action	Contact your 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 support representative for assistance.

MLC3005	**Invalid product code**
Explanation	The product code passed through the license check routine is invalid.
Action	Contact your support representative for assistance.

MLC3006	**Invalid product version**
Explanation	The product version passed through the license check routine is not numeric.
Action	Contact your 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 *yyyy-mm-dd*. The dash (-) delimiters are not checked but the numbers must be a valid year, month and day, respectively.
Action	Contact your support representative for assistance.

MLC3008	**Tag not found**
Explanation	The license check routine cannot find the requested XML tag.
Action	Contact your support representative for assistance.

MLC3009	**Component Id tag not found**
Explanation	The license check routine cannot find the XML tag `Component Id`.
Action	Contact your support representative for assistance.

MLC3010 **Invalid search tag**

Explanation The XML search tag requested by the license check routine is invalid.

Action Contact your 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 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 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 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 support representative for assistance.

MLC9001 **License exceeds maximum size of 8000**

Explanation Your license file is too large.

Action Contact your 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 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 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 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 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 (*errorcode*) from the operating system.
Action	Correct the LICUTIL input, removing duplicate LICUTIL function calls. Check that the license module (*modulename*) 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 *modulename*).

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 *after* 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.