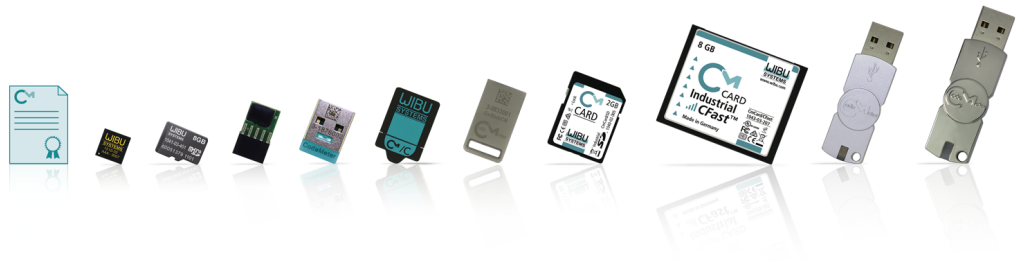
CodeMeter Embedded
API Help



Version 2.30 - October 2018

© Copyright 2018, WIBU-SYSTEMS AG,
Rueppurrer Strasse 52-54, 76137 Karlsruhe, Germany

Printed in Germany
All rights reserved. No part of this documentation, the accompanying software, or other components of the described product may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose other than the personal use of the purchaser without the express written permission of Wibu-Systems.
While the data contained in this document has been written with all due care, Wibu-Systems does not warrant or assume responsibility or represent that the data is free from errors or omissions.

Wibu-Systems expressly reserves the right to change programs or this documentation without prior notice.

WIBU®, CodeMeter®, SmartShelter®, SmartBind® and BLURRY BOX® are registered trademarks of WIBU-SYSTEMS AG.
Microsoft®, Windows®, Win32® , WinCE® are registered trademarks of Microsoft Corporation.
macOS® is a registered trademark of Apple Inc.
VxWorks® is a registered trademark of Wind River Corporation.
Intel® is a registered trademark of Intel-Corporation.

All other brand names and product names used in this documentation are trade names, service marks, trademarks, or registered trademarks of their respective owners.

Wibu-Systems is member of:

PCMCIA since 1993

USB Implementers Forum since 1997

SD Card Association since 2007

Bitkom, German Association of Information Technology, Telecommunications, and New Media since 2003

VDMA, German Engineering Federation since 2008

OPC Foundation since 2012

and also a member of the developers programs of Autodesk, Apple, HP, IBM, Intel and Microsoft.

Microsoft Gold Certified Partner

Microsoft Embedded Partner

Strategic Software Partner Industrial and Medical

Table of Contents

I Safety Instructions 5
II Overview 6
1 Introduction 6
III Functions / Methods 7
1 Accessing operations 7
1.1 CmAccess2 7
1.2 CmRelease 14
2 Authentication operations 14
2.1 CmCalculateSignature 14
2.2 CmCalculateHashMac 16
2.3 CmGetPublicKey 18
2.4 CmValidateSignature 19
3 Encryption operations 21
3.1 CmCrypt2 21
3.2 CmCryptEcies 26
3.3 CmGetSecureData 27
3.4 CmDecryptPioData 30
3.5 CmGetPioDataKey 30
4 Error Management operations 31
4.1 CmConvertString 31
4.2 CmGetLastErrorCode 32
4.3 CmGetLastErrorText 33
4.4 CmGetLastErrorText2 33
4.5 CmSetLastErrorCode 34
5 Managing operations 34
5.1 CmGetBoxContents2 34
5.2 CmGetBoxes 38
5.3 CmGetInfo 40
5.4 CmGetVersion 52
6 Programming operations 53
6.1 CmActLicenseControl 53
6.2 CmProgram 54
7 Remote Programming operations 64
7.1 CmExecuteRemoteUpdate 64
7.2 CmGetRemoteContextBuffer 66
8 Secure Disk operations 66
8.1 CmSecureDiscRead 66
8.2 CmSecureDiscWrite 67
8.3 CmExtendedDiscControl 69
9 Time Update operations 70
9.1 CmSetSystemTime 70
10 License Transfer operations 71
10.1 CmLtCreateContext 71
11 CmExecuteFieldUpdate() 72
IV Error Codes 75
V Knowledge Base 82
1 Access Modes 82
2 Encryption Modes 83

3 Little/Big Endian Conversion 83
4 License Quantity 83
5 Usage Period 84
VI Types, macros and helper functions 85
VII Version / Contact / Open Source Components 88
Index 0

1 Safety Instructions

The hardware of WIBU-SYSTEMS AG serves to protect and license digital products and has been developed, manufactured and inspected in accordance with state-of-the-art technology and recognized technical safety rules and regulations.

For further information on hardware certificates see the respective documents to be downloaded at the website of Wibu-Systems (http://www.wibu.com/en/certificates.html).

Before you use the hardware please observe the following safety instructions:

- If you follow the instructions regarding safety as described in this manual, the hardware will, in the normal case, neither cause personal injury nor damage to machinery and equipment. Connect the hardware only to matching intended interfaces. The use for other purposes, opening or own repair of the hardware may lead to damages of the product and its surroundings. Modifying the hardware affects the product safety. Caution: risk of injury!
- The hardware may warm up during operation - which is a normal operational parameter.
- Keep the hardware away from humidity and avoid strong vibration, dust, heat, and direct sunlight, in order to prevent operational interference.
- Depending on the used operating system the detection of the hardware device may take some seconds. Before disconnecting the hardware the user should wait several seconds to avoid loss of data during data saving.
- This product is not a toy, keep away from children!

Non-compliance with the safety instructions results in a loss of warranty.

© Copyright 2018 WIBU-SYSTEMS AG

2 Overview

This online help lists, describes and explains the functions and structures of the *CmEmbedded* API for embedded systems. For all supported embedded platforms available the functions and structures listed are a subset of the standard *CodeMeter Core API*.

Please note that depending on the modules activated for *CodeMeter Embedded* not all parameters are available.

Introduction

Functions

Error Codes (sortable)

Knowledge Base

Types and Macros

Version / Contact

Toggle Feature

For better usability this online help also uses a toggle feature.

Elements	Description
links in combination with ⊞ symbol	adds in-text information to the Topic Page or Parameter section without having to leave the actual displayed page
orange-colored links in the text	
"Show / Hide" symbol in the upper ribbon	hides or shows all toggle elements.

2.1 Introduction

For embedded systems Wibu-Systems provides an ANSI-C based *CodeMeter* Embedded to directly access *CodeMeter* licenses. For all supported platforms this API provides the same API calls. This documentation lists available functions and structures as a subset of the standard *API*.

Please note the following when applying the API:

- each function call should be followed by an error check. The return value of the called function should be read and checked, if it indicates an error or not.
- each new instance of a structure should be initialized to avoid unexpected results.

© Copyright 2018 WIBU-SYSTEMS AG

3 Functions / Methods

Accessing

CmAccess2
CmRelease

Authentication

CmCalculateSignature
CmCalculateHashMac
CmGetPublicKey
CmValidateSignature

Encryption

CmCrypt2
CmGetSecureData
CmDecryptPioData
CmGetPioDataKey

Error Management

CmConvertString
CmGetLastErrorCode
CmGetLastErrorText
CmSetLastErrorCode

Management

CmGetBoxContents2
CmGetBoxes
CmGetInfo
CmGetVersion

Programming

CmProgram
CmActLicenseControl

Remote Programming

CmExecuteRemoteUpdate
CmGetRemoteContextBuffer

Time Update

CmSetSystemTime

3.1 Accessing operations

3.1.1 CmAccess2

Description

CmAccess() opens the access to a:

© Copyright 2018 WIBU-SYSTEMS AG

- computer (SubSystem Access)
- CmContainer (Box Access)
- single Firm Item (FC, Firm Item Access) or a
- CmContainer entry (FC:PC, Entry Access) in a given subsystem.

Please note the detailed description of the Access Modes.

Please note that *CmAccess2()* provides enhanced functionality to access a CmContainer. This function is implemented since CodeMeter Version 3.30.

A SubSystem Access represents a superset of Box Access. A Box Access is a superset of Firm Item Access. A Firm Item Access is a superset of an Entry Access. Anything that can be done with a SubSystem Access can also be done with Box Access, but not vice versa!

Please note that on a Box Access the mask `m` and serial number s may change anytime depending on CodeMeter chip versions.
Please consider these possible changes for future CmContainer generations on programming accesses for CmContainer.

The subsystem is the local computer (CmLocal) or a local or wide area network system (CmLAN, CmWAN). For CmWAN please consult the CodeMeter Developer Guide. This implies that a CodeMeter License Server process must be started with a CmContainer connected on an other system in the network.
The communication between protected applications and CodeMeter License Server bases on the Transmission Control Protocol/Internet Protocol (TCP/IP). This is valid not only for locally existing licenses, but also for licenses which are provided via a network.
By default, CodeMeter uses the port 22350 registered by Wibu-Systems at IANA (Internet Assigned Numbers Authority) and uniquely assigned for the CodeMeter communication. The list of assigned ports can be viewed at www.iana.org/assignments/port-numbers.
In order to make available a client access to a CodeMeter License Server on the network, a communication using the CodeMeter port must be supported. If the server should locate in another network area, eventually the port must be made known and accessible as part of the infrastructure (router, firewall, etc.).
For the direct access to CodeMeter License Server on the network, the communication bases on TCP. For an automatic search of servers on the network, additionally a broadcast via UDP (User Datagram Protocol) is performed.

The access using the CodeMeter port is performed only for the access to CodeMeter License Servers and this only within the organization which runs the network server.
In particular, using this port **no** communication into the internet is performed.

In CodeMeter settings of CodeMeter WebAdmin an option exist to configure the CodeMeter port to a value other than the default of 22350. However, such a change should have plausible reasons, e.g. in the case of parallel test environments on the same network. In addition, such a change requires the same configuration of all affected CodeMeter License Servers.
The desired subsystem and the search order can be set in the special flags.
If a network access to another computer is performed, it will be available until the access is closed with *CmRelease()*. If more than one connection between two instances should be required, *CmAccess()* will open only one connection which is the medium for all communication between the two clients.

C, C++ Programming Language

Note

CmAccess2() provides enhanced functionality to access a CmContainer by using the CMCREDENTIAL feature.

Structure CMCREDENTIAL

CMCREDENTIAL holds parameters that can be used for a *CmAccess2()* call to refine the license allocation and the assignment to the CodeMeter License Server, for example, via an arbitrary user name.

Syntax

```
typedef struct __CMCREDENTIAL {
   CMULONG mulPID;
   CMULONG mulSession;
   CMULONG mulCleanupTime;
   CMULONG mulMaxLifeTime;
```


© Copyright 2018 WIBU-SYSTEMS AG

```
    CMULONG mulCreationTime;
    CMULONG mulAccessCtrl;
    CMULONG mulUserDefinedID;
    char mszUserDefinedText[128];
    char mszUsername[32];
    CMULONG mulCurrentCmInstanceUid;
    CMULONG mulPreviousCmInstanceUid;
    char mszDomainName[16];
    char mabReserved2[4]
} CMCREDENTIAL;
```

Parameter

mulPID contains the process ID that is set by the CodeMeter library (read only).

mulSession is always zero on the user site.

mulCleanupTime this parameter overwrites the CleanupTime profiling value for license releasing in minutes.

mulMaxLifeTime defines the maximum lifetime (in minutes) of the license that is allocated.

mulCreationTime holds the creation time of the license that was accessed in seconds since `01.01.2000`.

mulAccessCtrl holds the type of access (since 4.20):

The first group of flags hold information about local or Lan access:

- *CM_CRED_LICENSE_LOCATION_LOCAL* handle to the local system.
- *CM_CRED_LICENSE_LOCATION_LAN* handle to a network system.

The second group defines the access type:

- *CM_CRED_HANDLE_ACCESS_ENTRY* an entry handle.
- *CM_CRED_HANDLE_ACCESS_FI* a Firm Item handle.
- *CM_CRED_HANDLE_ACCESS_BOX* a box handle.
- *CM_CRED_HANDLE_ACCESS_SUBSYSTEM* a subsystem handle.

mulUserDefinedID contains the user defined ID.

mszUserDefinedText contains a user defined text (UTF-8 encoded).

mszUsername holds an arbitrary user name (read only).

It is available since CodeMeter Version 3.30 and offers a user-defined timing to release licenses that were allocated locally or in the network.

This release interval overwrites the default timing value **CleanUpTimeOut** of the CodeMeter profiling. The benefit is that the user cannot influence this timing anymore.

Additionally, the validity period of the handle returned by *CmAccess2()* can be determined. This feature can be used to limit the usage period of an application that allocates a license using *CmAccess2*().

The mapping of the licenses on a CodeMeter server can be simplified by giving an user-defined string or number (session-ID) that is displayed on the server once the license was allocated.

Syntax

```
HCMSysEntry CMAPIENTRY CmAccess2(CMULONG flCtrl, CMACCESS *pcmAcc)
```

Parameter



Please note, that if your module configuration includes Module Runtime Bridge but without Module LAN, then the flag *CM_ACCESS_LAN* cannot be used and the error code CM_ERROR_FUNCTION_NOT_SUPPORTED is set.
The flag *CM_ACCESS_LOCAL_LAN* tries to access first the local subsystem and subsequently cannot find the LAN subsystem. Here the error code CM_ERROR_FUNCTION_NOT_SUPPORTED is set.

flCtrl specifies one of the following commands to set the desired subsystems. Only one of these constants can be set:

- *CM_ACCESS_LOCAL* only searches in CmLocal for a CmContainer with a matching entry.
- *CM_ACCESS_LAN* only searches in CmLAN for a CmContainer with a matching entry.

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_ACCESS_LOCAL_LAN* searches CmLocal first for a CmContainer with a matching entry and after that in CmLAN, if no CmContainer was found locally.
- *CM_ACCESS_LAN_LOCAL* searches CmLAN first for a CmContainer with a matching entry and after that in CmLocal, if no CmContainer was found in the network.
- *CM_ACCESS_LANWAN* searches in CmLAN and CmWAN for a CmContainer with a matching entry.
- *CM_ACCESS_LOCAL_LANWAN* searches CmLocal first after that in CmLAN and CmWAN for a CmContainer with a matching entry.
- *CM_ACCESS_LANWAN_LOCAL* searches first in CmLAN and CmWAN and after that in CmLocal for a CmContainer with a matching entry.

pcmAcc sets the needed data for the desired access (see structure CMACCESS2).

Structure CMACCESS2

CMACCESS2 specifies additional data for the Access Modes to a computer (Subsystem Access), to a CmContainer (Box Access), to a single Firm Item (FC, Firm Item Access) or a CmContainer entry (FC:PC, Entry Access) in a given subsystem.

Syntax

```
typedef struct __CMACCESS2{
    CMULONG mflCtrl;
    CMULONG mulFirmCode;
    CMULONG mulProductCode;
    CMULONG mulFeatureCode;
    CMULONG mulUsedRuntimeVersion;
    CMULONG mulProductItemReference;
    CMBYTE mbMinBoxMajorVersion;
    CMBYTE mbMinBoxMinorVersion;
    CMUSHORT musBoxMask;
    CMULONG mulSerialNumber;
    CMULONG mulLicenseQuantity;
    char mszServername[128];
    CMTIME mcmReleaseDate;
    CMBYTE mabCmActId[4];
    CMCREDENTIAL mcmCredential;
} CMACCESS2;
```

Parameter

mflCtrl:specifies the kind of the subsystem access by setting one of the following flags.

- *CM_ACCESS_USERLIMIT* needs a license for each *CmAccess2 ()* call. The License Quantity will be decreased by one at each access.
- *CM_ACCESS_NOUSERLIMIT* needs no license for each *CmAccess2 ()* call. The License Quantity will not be decreased in this mode.
- *CM_ACCESS_EXCLUSIVE* needs a license for each *CmAccess2 ()* call and on a computer only one access to the specified entry is permitted. The License Quantity will be decreased by one at each access.
- *CM_ACCESS_FIRMITEM* forces a Firm Item access to an CmContainer entry.
- *CM_ACCESS_STATIONSHARE* needs a license for each *CmAccess2 ()* call and on a computer only one license to the specified entry is blocked. The License Quantity will be decreased by one once per computer.
- *CM_ACCESS_CONVENIENT* needs in the local subsystem (CmLocal) no license (works here as *CM_ACCESS_NOUSERLIMIT*) and in the network (CmLAN) one license per access (works here as *CM_ACCESS_USERLIMIT*).

These flags can be or'ed with the following flags:

- *CM_ACCESS_FORCE* forces the access to an entry, which has a expired Expiration Time, an invalid Activation Time or a Unit Counter, which is 0. Usually, *CmAccess2 ()* does not return a valid handle, if one of the three Product Item Options has not a true value.
- *CM_ACCESS_CHECK_FSB* specifies that the desired access must be done to a Product Code item of a Firm Security Box (FSB) (usually 99). If the specified entry is not an FSB item, an error is returned. Only checked for Entry Access.

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_ACCESS_CHECK_CTSB* specifies that the desired access must be done to an entry in a Certified Time Security Box (CTSB) item. If the specified entry is not inside a CTSB item, an error is returned. Only checked for Entry Access.
- *CM_ACCESS_SUBSYSTEM* specifies that the desired access can also be a Subsystem Access. In this case, **mulFirmCode** must be set to 0.
- *CM_ACCESS_IGNORE_LINGERTIME* specifies that a possibly set Linger Time (a Product Item Option) is ignored for this access. Only useful in combination with *CM_ACCESS_USERLIMIT* or *CM_ACCESS_STATIONSHARE*.

mulFirmCode contains the Firm Code of the access mode. Which Firm Code must be set for the different access modes you can see at the description of the *CmAccess2 ()* function

mulProductCode contains the Product Code of the access mode. Which Product Code must be set for the different access modes you can see at the description of the *CmAccess2 ()* function

mulFeatureCode contains the Feature Code, which must map a given Feature Map. If the Feature Code is 0, there is no search for a matching Feature Map.

mulUsedRuntimeVersion contains the minimum runtime version of CodeMeter. To calculate the version you may want to use the macro (see CM_SET_VERSION).

Macro CM_SET_VERSION

`CM_SET_VERSION(a, b, c)` sets the complete version in a CMULONG value.

- `a`: sets the major version.
- `b`: sets the minor version.
- `c`: sets the build version.

Syntax

```
#define CM_SET_VERSION(a,b,c) (((a) & 0xff) \
(((a) & 0xff) << 24) | (((b) & 0xff) << 16) | ((c) & 0xffff)
```

mulReserved1 is reserved for future purposes.

musProductItemReference Product Item Reference of the Product Item to access.

mbMinBoxMajorVersion contains the minimum major Firmware version of the CmContainer to access.

mbMinBoxMinorVersion contains the minimum minor Firmware version of the CmContainer to access.

musBoxMask contains the box mask of the CmContainer to access.

mulSerialNumber contains the serial number of the CmContainer to access.

mulLicenseQuantity contains the requested number of licenses to be allocated using a single handle.

Using this parameter requires that licenses must have a PIO License Quantity which is bigger than 1 and CodeMeter Version 6.20 must be installed.
For compatibility reasons Wibu-Systems recommends to set Version 6.20 as mimimum used version (**mulUsedRuntimeVersion**).
If the set runtime version is smaller than the minimum version, the access is canceled and the error message CMERROR_INVALID_PARAMETER displays.
If the requested license quantity is bigger than the number of available licenses, the error message CMERROR_NO_MORE_LICENSES displays.

mszServername contains the name or IP-address of the CodeMeter server to access. The type is detected automatically.

mcmReleaseDate contains a structure where the Release Date can be set that is checked against the Maintenance Period of the license (see CMTIME).

Structure CMTIME

CMTIME sets or gets the time to or from a CmContainer.

Syntax

```
typedef struct __CMTIME {
    CMUSHORT musYear;
    CMUSHORT musMonth;
```

© Copyright 2018 WIBU-SYSTEMS AG

```
    CMUSHORT musDay;
    CMUSHORT musHours;
    CMUSHORT musMinutes;
    CMUSHORT musSeconds;
    CMULONG mulSecondsSince01_01_2000;
} CMTIME;
```

Parameter

musYear specifies the value of the year (e.g. `2003`).
musMonth specifies the value of the month (`1..12`).
musDay specifies the value of the day (`1..31`).
musHours specifies the value of the hours (`0..23`).
musMinutes specifies the value of the minutes (`0..59`).
musSeconds specifies the value of the seconds (`0..59`).
mulSecondsSince01_01_2000 specifies the time in seconds since `01/01/2000`.

mabCmActId allows to specify a CmAct-ID using the following ranges: 1**** for CmDongle, 2****for CmActLicense, 3**** for CmCloud (not yet implemented).

Please note that a handle is returned only, if the entries match. Otherwise the error message CMERROR_CMACTID_DOES_NOT_MATCH displays.

mabReserved is reserved for future purposes.

mcmBorrowData contains a CMBORROWDATA structure, where the settings for License borrowing can be set. License borrowing is available since CodeMeter Version 4.0 and requires special programming of a license container. For easy access to borrowing functionality please refer to *CmBorrow()*.

Structure CMBORROWDATA

CMBORROWDATA stores all needed parameters needed for borrowing a license. It is used within *CmAccess2()* and can be filled with *CmGetInfo*.

Please note that borrowing in this context does not involve a license transfer. Rather it refers to borrowing using Firm Codes smaller than 6,000,000 and especially prepared CmContainer.

Syntax

```
typedef struct __CMBORROWDATA {
   CMULONG mulStatus;
   CMUSHORT midEnableBlock;
   CMUSHORT midCps;
   CMULONG mulFirmCode;
   CMULONG mulProductCode;
   CMULONG mulFeatureMap;
   CMUSHORT musReserved;
   CMUSHORT musBoxMask;
   CMULONG mulBoxSerial;
   CMBYTE mabSerialPublicKey[CM_PUBLIC_KEY_LEN];
   CMBYTE mabServerID[CM_BORROW_SERVERID_LEN];
   char mszClientName[128];
   CMBYTE mabTrailingValidationBlock[CM_TVB_LEN];
   CMULONG mulClientRequestTime;
   CMULONG mulMaintenancePeriodStart;
   CMULONG mulMaintenancePeriodEnd;
} CMBORROWDATA;
```

Parameter

mulStatus contains the current status of the enabling block:

- *CM_EBI_DISABLED* the entry is disabled, the license can not be used at the client.
- *CM_EBI_ENABLED* the entry is enabled, the license can be used at the client.

© Copyright 2018 WIBU-SYSTEMS AG

midEnableBlock contains the index of the enabling block used for the borrowing operation.
midCps contains the used copy protection system on client side:
- *CM_CPS_CODEMETER* The borrowing client is using CmDongle.
- *CM_CPS_CODEMETERACT* The borrowing client is using CmActLicense.

mulFirmCode contains Firm Code of the borrow license on the client side.
mulProductCode contains the Product Code of the borrow license on the client side.
mulFeatureMap contains the Feature Map of the borrow license on the client side.
musReserved is reserved for future purposes.
musBoxMask contains the box mask of the license container on the client side.
mulBoxSerial contains the serial number of the license container on the client side.
mabSerialPublicKey contains the Public Serial Key of the license container on the client side.
mabServerID contains the id of the borrow server.
mszClientName contains the DNS name of the borrowing client.
mabTrailingValidationBlock contains the trading validation block that allows a validation of the contents of this structure.
mulClientRequestTime contains the system time of the client.
mulMaintenancePeriodStart contains the start date of the Maintenance Period of the Product Item, if available.
mulMaintenancePeriodEnd contains the end date of the Maintenance Period of the Product Item, if available

mcmCredential contains a ⊟ CMCREDENTIAL structure, where some settings for the access to a license server can be set.

Structure CMCREDENTIAL

CMCREDENTIAL holds parameters that can be used for a *CmAccess2()* call to refine the license allocation and the assignment to the CodeMeter License Server, for example, via an arbitrary user name.

Syntax

```
typedef struct __CMCREDENTIAL {
   CMULONG mulPID;
   CMULONG mulSession;
   CMULONG mulCleanupTime;
   CMULONG mulMaxLifeTime;
   CMULONG mulCreationTime;
   CMULONG mulAccessCtrl;
   CMULONG mulUserDefinedID;
   char mszUserDefinedText[128];
   char mszUsername[32];
   CMULONG mulCurrentCmInstanceUid;
   CMULONG mulPreviousCmInstanceUid;
   char mszDomainName[16];
   char mabReserved2[4]
} CMCREDENTIAL;
```

Parameter

mulPID contains the process ID that is set by the CodeMeter library (read only).
mulSession is always zero on the user site.
mulCleanupTime this parameter overwrites the CleanupTime profiling value for license releasing in minutes.
mulMaxLifeTime defines the maximum lifetime (in minutes) of the license that is allocated.
mulCreationTime holds the creation time of the license that was accessed in seconds since 01.01.2000.
mulAccessCtrl holds the type of access (since 4.20):
The first group of flags hold information about local or Lan access:
- *CM_CRED_LICENSE_LOCATION_LOCAL* handle to the local system.

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_CRED_LICENSE_LOCATION_LAN* handle to a network system.

The second group defines the access type:

- *CM_CRED_HANDLE_ACCESS_ENTRY* an entry handle.
- *CM_CRED_HANDLE_ACCESS_FI* a Firm Item handle.
- *CM_CRED_HANDLE_ACCESS_BOX* a box handle.
- *CM_CRED_HANDLE_ACCESS_SUBSYSTEM* a subsystem handle.

mulUserDefinedID contains the user defined ID.
mszUserDefinedText contains a user defined text (UTF-8 encoded).
mszUsername holds an arbitrary user name (read only).

Return

The handle to the desired CmContainer entry or CmContainer in the desired subsystem is returned. If no matching CmContainer entry or CmContainer is found, NULL is returned.

Notes

Access to Firm Item 0 can only be performed locally. If tried using CmLAN, *CmAccess2* will trigger Error 200 (Entry not found).
CmAccess2() is supported since CodeMeter Version 3.30.
The following flags (and the corresponding functionality) are only available since CodeMeter Version 5.0:

- *CM_ACCESS_LANWAN*
- *CM_ACCESS_LOCAL_LANWAN*
- *CM_ACCESS_LANWAN_LOCAL*

3.1.2 CmRelease

Description

CmRelease closes a handle opened by *CmAccess2* and all related subsystem accesses.
The runtime will be cleaned up internally. Subsystem accesses will only be closed, if there is no other handle/entry using it.
This still open subsystem access will be closed from the last open handle to an entry using this subsystem when *CmRelease* is called for this handle.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmRelease(HCMSysEntry hcmse)
```

Parameter

hcmse contains a handle opened by *CmAccess2*.

Return

`TRUE` is returned, if no error has occurred - `FALSE`, if not.

3.2 Authentication operations

3.2.1 CmCalculateSignature

Description

CmCalculateSignature() calculates a signature using the ECDSA algorithm of the given digest directly via a CmContainer entry.

C, C++ Programming Language

© Copyright 2018 WIBU-SYSTEMS AG

Syntax

```
int CMAPIENTRY CmCalculateSignature(HCMSysEntry hcmse, CMAUTHENTICATE *pcmAuth,
CMBYTE *pbSignature, CMUINT cbSignature);
```

The digest is available as **mabDigest** of the structure CMAUTHENTICATE.

Structure CMAUTHENTICATE

CMAUTHENTICATE contains all information needed for an authentication process.

Syntax

```
typedef struct __CMAUTHENTICATE {
   CMULONG mflCtrl;
   CMULONG mulKeyExtType;
   CMULONG mulFirmCode;
   CMULONG mulProductCode;
   CMULONG mulEncryptionCodeOptions;
   CMULONG mulFeatureCode;
   CMBOXINFO mcmBoxInfo;
   CMBYTE mabDigest[CM_DIGEST_LEN];
} CMAUTHENTICATE;
```

Parameter

mflCtrl contains one of the following commands:

- *CM_AUTH_FIRMKEY* specifies that the signature calculation is done with the Firm Key.
- *CM_AUTH_HIDDENDATA* specifies that the signature calculation is done with a Hidden Data field.
- *CM_AUTH_SECRETDATA* specifies that the signature calculation is done with a Secret Data field.
- *CM_AUTH_SERIALKEY* specifies that the signature calculation is done with the SerialKey.
- *CM_AUTH_BOXKEY* specifies that the signature calculation is done with the Box Key.
- *CM_AUTH_PISK* specifies that the signature calculation is done with the PISK (Product Item Security Key).
- *CM_AUTH_KSMASK* can be used to mask bits used for specifying the key source.
- *CM_AUTH_AMMASK* can be used to mask bits used for specifying the authentication mode.

This commands can be or'ed with one of the following flags:

- *CM_AUTH_DIRECT* uses the specified digest directly as input of the ECDSA algorithm.
- *CM_AUTH_EXTENDED* defines that the specified Encryption Code Options and Feature Code influence the use of the Product Item and influences the signature calculation.
- *CM_AUTH_EXTENDED2* improves the definition that the specified Encryption Code Options and Feature Code influence the use of the Product Item and influences the signature calculation . The flag has been introduced with CodeMeter Version 5.10. Please note that using *CM_AUTH_EXTENDED2* requires at least hardware version 2.03.
- *CM_AUTH_SERIAL* defines that the specified Encryption Code Options influence the use of the Product Item and influences together with the Serial Number of the CmContainer the signature calculation.

mulKeyExtType contains the extended type of the Hidden Data or Secret Data PIO, if in **mflCtrl** *CM_AUTH_HIDDENDATA* or CM_AUTH_SECRETDATA is specified. Otherwise this member must be set to 0.

mulFirmCode contains the Firm Code used for the authentication process.

mulProductCode contains the Product Code used for the authentication process.

mulEncryptionCodeOptions contains the Encryption Code Options, this are the higher 32 bits of the Encryption Code. This value is only used, if **mflCtrl** has set the *CM_AUTH_EXTENDED*, *CM_AUTH_EXTENDED*, *CM_AUTH_EXTENDED2* or *CM_AUTH_SERIAL* option. Otherwise it must be set to 0.

mulFeatureCode contains the used Feature Code, which must match the used Feature Map. This value is only used, if **mflCtrl** has set the *CM_AUTH_EXTENDED*, *CM_AUTH_EXTENDED*, *CM_AUTH_EXTENDED2* option. Otherwise it must be set to 0.

© Copyright 2018 WIBU-SYSTEMS AG

mcmBoxInfo contains the CmContainer information used for the authentication process.
mabDigest contains the 32-byte res. 256-bit digest which is signed.

hcmse contains a handle opened with *CmAccess2* (Entry Access).
pcmAuth contains all information needed for the authentication process (see structure CMAUTHENTICATE.
pbSignature contains the buffer for the signature.
cbSignature contains the length of the buffer of **pbSignature**. The length has to be `CM_PUBLIC_KEY_LEN` which is 64.

Return

True (1), if the signature was calculated successful. If an error occurs, `0` is returned and an error code is set.

Notes

- The hash value used for the signature must always be set into **pcmAuth->mabDigest**.
- The value **pcmAuth->mflCtrl** has to be a combination of signature method and key source. The values are or'ed together.
- Signature method is one of *CM_AUTH_DIRECT*, *CM_AUTH_EXTENDED*, *CM_AUTH_EXTENDED2* or *CM_AUTH_SERIAL*.
- Key source is one of *CM_AUTH_HIDDENDATA*, *CM_AUTH_SECRETDATA*, *CM_AUTH_SERIALKEY*.
- In the case of Hidden Data or Secret Data, the extended type must be set as **pcmAuth->mulKeyExtType**. Otherwise this value should be 0.
- *CM_AUTH_DIRECT*: All other values are ignored and should be set to `0`.
- *CM_AUTH_EXTENDED(2)*: **pcmAuth->mulEncryptionCodeOptions** has to have the same value as for *CmValidateSignature*. **pcmAuth->mulFeatureCode** can be set to a specific value. The value will be integrated into the signature but the signature will only be calculated, if the requested Feature Code conforms with the current Feature Map. The signature incorporates the serial number of the signing CodeMeter and the Firm Code / Product Code values used for *CmAccess2*. These values have to be remembered to correctly verify the signature later using *CmValidateSignature*.
  The flag has been introduced with CodeMeter Version 5.10. Please note that using *CM_AUTH_EXTENDED2* requires at least hardware version 2.03.
- `CM_AUTH_SERIAL`: **pcmAuth->mulEncryptionCodeOptions** has to have the same value as for *CmValidateSignature*. All other values should be set to 0. The signature incorporates the serial number of the signing CodeMeter and the Firm Code / Product Code values used for *CmAccess2*. These values have to be remembered to correctly verify the signature later using *CmValidateSignature*.
- *CM_AUTH_PISK* (Product Item Secret Key) has been introduced with CodeMeter Version 6.0 (License Transfer).

3.2.2 CmCalculateHashMac

Description

CmCalculateHashMac calculates message authentication codes or hashes based on an algorithm which can be selected. The related secret keys are stored in a CmContainer entry.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmCalculateHashMac(HCMSysEntry hcmse, const CMHASHMAC
*pcmHASHMAC, const void *pvInput, CMUINT cbInput, void *pvResult, CMUINT
cbResult);
```

Parameter

hcmse contains an Entry Access handle opened by *CmAccess2*.

© Copyright 2018 WIBU-SYSTEMS AG

pcmHASHMAC contains the required information for calculate message authentication codes or hashes in the structure ⊟ CMHASHMAC.

Structure CMHASHMAC

CMHASHMAC contains all information needed for calculating message authentication codes or hashes.

Syntax

```
typedef struct __CMHASHMAC {
  CMULONG mflCtrl;
  CMULONG mulKeySrc;
  CMULONG mulAlgoType;
  CMULONG mulKeyExtType;
  CMBYTE mabReserved[48];
} CMHASHMAC;
```

Parameter

mflCtrl contains flags for future use.
mulKeySrc contains the selection of the key source *CM_HASHMAC_SECRETDATA*.
mulAlgoType contains the algorithm type: available are:

- *CM_HASHMAC_SIPHASH24* or
- *CM_HASHMAC_CMAC_AES128*.

mulKeyExtType contains the extended type of the Secret Data PIO.
mabReserved is reserved for future purposes.

pvInput contains the buffer for the input to be processed.
cbInput contains the length of the buffer of **pvInput** in bytes.
pvResult contains the result, i.e. target buffer.
cbResult contains the size of the target buffer **pvResult** in bytes.
Depending on the algorithm type:

- *CM_HASHMAC_CMAC_AES128* a minimum of 16 bytes.
- *CM_HASHMAC_SIPHASH24* a minimum of 8 bytes.

Return

Length in bytes of the specified algorithm result.
If an error occurs, `0` is returned.
With the exception that if *CMERROR_BUFFER_OVERFLOW* is set, the minimum result length of the respective algorithm is returned.

Notes

- Empty messages are supported only for CmDongle Firmware versions higher than 4.10. Otherwise, the error code *CMERROR_COMMANDINCOMPLETE* is set.
- The maximum input size is currently 3824 bytes.
- The *CM_HASHMAC_CMAC_AES128* algorithm can handle input which is not a multiple of the key length (16 bytes) only with available Firmware Version higher than 4.10. Otherwise the error code *CMERROR_INTERNAL* is set.
- The byte sequence of the *SipHash* tag in **pvResult** is big endian ordered.

Error Codes

CMERROR_INVALID_PARAMETER

- If **pcmHASHMAC** == NULL, ((**pvInput** == NULL && **cbInput** != 0) || (**pvInput** != NULL && **cbInput** == 0)).
- If **pvResult** == NULL and (**cbResult** >= 8 || **cbResult** >= 16).

© Copyright 2018 WIBU-SYSTEMS AG

- If key source is not known **mulKeySrc** `== 0` or **mulKeySrc** `> CM_HASHMAC_KSMAX`.
- If algorithm is not known **mulAlgoType** `== 0` or **mulAlgoType** `> CM_HASHMAC_ALGMAX`.
- If **cbInput** `> UINT16_MAX (65535)`.

CMERROR_WRONG_HANDLE_TYPE

- If a Subsystem, Box or Firm Item handle is used.

CMERROR_WRONG_BOX_VERSION

- If the used Firmware not yet supports the feature.

CMERROR_FUNCTION_NOT_IMPLEMENTED

- In the case of CmLAN of Runtime Bridge modules.

CMERROR_FUNCTION_NOT_SUPPORTED

- For CmActLicense Firm Codes and CmActLicense Universal Firm Code container.

CMERROR_BUFFER_OVERFLOW

- If **cbResult** is smaller, than the respose length of the respective algoritm.

3.2.3 CmGetPublicKey

Description

CmGetPublicKey returns a public key from the CmContainer. The CmContainer usually stores private keys only. With this request the CmContainer calculates the corresponding public key using Elliptic Curve Cryptography (ECC).

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmGetPublicKey(HCMSysEntry hcmse, CMAUTHENTICATE *pcmAuth,
CMBYTE *pbPubKey, CMUINT cbPubKey);
```

Parameter

hcmse contains a handle opened by *CmAccess2* (Entry Access).
pcmAuth contains the specification of the desired public key (see structure ⊟ CMAUTHENTICATE).

Structure CMAUTHENTICATE

CMAUTHENTICATE contains all information needed for an authentication process.

Syntax

```
typedef struct __CMAUTHENTICATE {
   CMULONG mflCtrl;
   CMULONG mulKeyExtType;
   CMULONG mulFirmCode;
   CMULONG mulProductCode;
   CMULONG mulEncryptionCodeOptions;
   CMULONG mulFeatureCode;
   CMBOXINFO mcmBoxInfo;
   CMBYTE mabDigest[CM_DIGEST_LEN];
} CMAUTHENTICATE;
```

Parameter

mflCtrl contains one of the following commands:

- *CM_AUTH_HIDDENDATA* specifies that the signature calculation is done with a Hidden Data field.

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_AUTH_SECRETDATA* specifies that the signature calculation is done with a Secret Data field.
- *CM_AUTH_SERIALKEY* specifies that the signature calculation is done with the Serial Key.
- *CM_AUTH_BOXKEY* specifies that the signature calculation is done with the Box Key.
- *CM_AUTH_PISK* specifies that the signature calculation is done with the PISK (Product Item Security Key).
- *CM_AUTH_KSMASK* can be used to mask bits used for specifying the key source.
- *CM_AUTH_AMMASK* can be used to mask bits used for specifying the authentication mode.

This commands can be or'ed with one of the following flags:

- *CM_AUTH_DIRECT* uses the specified digest directly as input of the ECDSA algorithm.
- *CM_AUTH_EXTENDED* defines that the specified Encryption Code Options and Feature Code influence the use of the Product Item and influences the signature calculation.
- *CM_AUTH_EXTENDED2* improves the definition that the specified Encryption Code Options and Feature Code influence the use of the Product Item and influences the signature calculation . The flag has been introduced with CodeMeter Version 5.10.
  Plese note that using *CM_AUTH_EXTENDED2* requires at least hardware version 2.03.
- *CM_AUTH_SERIAL* defines that the specified Encryption Code Options influence the use of the Product Item and influences together with the Serial Number of the CmContainer the signature calculation.

mulKeyExtType contains the extended type of the Hidden Data or Secret Data PIO, if in **mflCtrl** *CM_AUTH_HIDDENDATA* or *CM_AUTH_SECRETDATA* is specified. Otherwise this member must be set to `0`.

mulFirmCode contains the Firm Code used for the authentication process.

mulProductCode contains the Product Code used for the authentication process.

mulEncryptionCodeOptions contains the Encryption Code Options, this are the higher 32 bits of the Encryption Code. This value is only used, if **mflCtrl** has set the *CM_AUTH_EXTENDED(2)* or *CM_AUTH_SERIAL* option. Otherwise it must be set to 0.

mulFeatureCode contains the used Feature Code, which must match the used Feature Map. This value is only used, if **mflCtrl** has set the *CM_AUTH_EXTENDED(2)* option. Otherwise it must be set to `0`.

mcmBoxInfo contains the CmContainer information used for the authentication process.

mabDigest contains the 32-byte res. 256-bit digest which is signed.

pbPubKey contains the buffer for the public key.

cbPubKey contains the length of the **pbPubKey** buffer.

Return

The return value is TRUE, if the public key is returned, FALSE if not.

Notes

The **pcmAuth** structure has to be almost completely empty. The only value that has to be set is **mflCtrl** which has to be set to one of the following values: *CM_AUTH_HIDDENDATA*, *CM_AUTH_SECRETDATA*, *CM_AUTH_SERIALKEY*.

The value *CM_AUTH_FIRMKEY* is invalid here as this function cannot directly calculate the public key for the Firm Key. To calculate the public key for the Firm Key for Firm Item X calculate the public key for FSB entry 99:x Secret Data 128.

CM_AUTH_PISK (Product Item Secret Key) has been introduced with CodeMeter Version 6.0 (License Transfer).

3.2.4 CmValidateSignature

Description

CmValidateSignature validates a passed signature using the passed public key and the ECDSA algorithm.

C, C++ Programming Language

© Copyright 2018 WIBU-SYSTEMS AG

Syntax

```
int CMAPIENTRY CmValidateSignature(CMAUTHENTICATE *pcmAuth, CMBYTE
*pbSignature, CMUINT cbSignature, CMBYTE *pbPubKey, CMUINT cbPubKey);
```

Parameter

pcmAuth contains the specification of the validation process (see structure CMAUTHENTICATE).

Structure CMAUTHENTICATE

CMAUTHENTICATE contains all information needed for an authentication process.

Syntax

```
typedef struct __CMAUTHENTICATE {
   CMULONG mflCtrl;
   CMULONG mulKeyExtType;
   CMULONG mulFirmCode;
   CMULONG mulProductCode;
   CMULONG mulEncryptionCodeOptions;
   CMULONG mulFeatureCode;
   CMBOXINFO mcmBoxInfo;
   CMBYTE mabDigest[CM_DIGEST_LEN];
} CMAUTHENTICATE;
```

Parameter

mflCtrl contains one of the following commands:

- *CM_AUTH_HIDDENDATA* specifies that the signature calculation is done with a Hidden Data field.
- *CM_AUTH_SECRETDATA* specifies that the signature calculation is done with a Secret Data field.
- *CM_AUTH_SERIALKEY* specifies that the signature calculation is done with the Serial Key.
- *CM_AUTH_BOXKEY* specifies that the signature calculation is done with the Box Key.
- *CM_AUTH_KSMASK* can be used to mask bits used for specifying the key source.
- *CM_AUTH_AMMASK* can be used to mask bits used for specifying the authentication mode.

This commands can be or'ed with one of the following flags:

- *CM_AUTH_DIRECT* uses the specified digest directly as input of the ECDSA algorithm.
- *CM_AUTH_EXTENDED* defines that the specified Encryption Code Options and Feature Code influence the use of the Product Item and influences the signature calculation.
- *CM_AUTH_EXTENDED2* improves the definition that the specified Encryption Code Options and Feature Code influence the use of the Product Item and influences the signature calculation . The flag has been introduced with CodeMeter Version 5.10.
  Plese note that using *CM_AUTH_EXTENDED2* requires at least hardware version 2.03.
- *CM_AUTH_SERIAL* defines that the specified Encryption Code Options influence the use of the Product Item and influences together with the Serial Number of the CmContainer the signature calculation.

mulKeyExtType contains the extended type of the Hidden Data or Secret Data PIO, if in **mflCtrl** *CM_AUTH_HIDDENDATA* or *CM_AUTH_SECRETDATA* is specified. Otherwise this member must be set to `0`.

mulFirmCode contains the Firm Code used for the authentication process.

mulProductCode contains the Product Code used for the authentication process.

mulEncryptionCodeOptions contains the Encryption Code Options, this are the higher 32 bits of the Encryption Code. This value is only used, if **mflCtrl** has set the *CM_AUTH_EXTENDED*, *CM_AUTH_EXTENDED2* or *CM_AUTH_SERIAL* option. Otherwise it must be set to `0`.

mulFeatureCode contains the used Feature Code, which must match the used Feature Map. This value is only used, if **mflCtrl** has set the *CM_AUTH_EXTENDED*, *CM_AUTH_EXTENDED2* option. Otherwise it must be set to `0`.

mcmBoxInfo contains the CmContainer information used for the authentication process.

© Copyright 2018 WIBU-SYSTEMS AG

mabDigest contains the 32-byte res. 256-bit digest which is signed.

pbSignature contains the buffer for the signature to validate.
cbSignature contains the length of the **pbSignature** buffer.
pbPubKey contains the buffer for the public key needed for validation.
cbPubKey contains the length of the **pbPubKey** buffer.

Return

The hash value used for the signature must always be set into **pcmAuth->mabDigest**. The value **pcmAuth->mflCtrl** is set to one of the following values: *CM_AUTH_DIRECT*, *CM_AUTH_EXTENDED*,*CM_AUTH_EXTENDED2*, *CM_AUTH_SERIAL*. Depending on this value the following values must be set accordingly:

- *CM_AUTH_DIRECT*: All other values are ignored and can be set to 0.
- *CM_AUTH_EXTENDED*, *CM_AUTH_EXTENDED2*: **pcmAuth->mulEncryptionCodeOptions** and **pcmAuth->mulFeatureCode** have to have the same value as for *CmValidateSignature*. **pcmAuth->mulFirmCode** and **pcmAuth->mulProductCode** have to match the values used for *CmAccess2* performed for *CmCalculateSignature*.
- *CM_AUTH_SERIAL*: **pcmAuth->mulEncryptionCodeOptions** has to have the same value as for *CmCalculateSignature*. **pcmAuth->mcmBoxInfo.musBoxMask** and **pcmAuth->mcmBoxInfo.mulSerialNumber** have to be set to identify the signing box. **pcmAuth->mulFirmCode** and **pcmAuth->mulProductCode** have to match the values used for *CmAccess2* performed for *CmValidateSignature*.

  performed for *CmValidateSignature*.
- *Serial*: **Auth->EncryptionCodeOptions** has to have the same value as for *CmCalculateSignature:* **Auth->BoxInfo.BoxMask** and **Auth->BoxInfo.SerialNumber** have to be set to identify the signing box. **Auth->FirmCode** and **Auth->ProductCode** have to match the values used for *CmAccess2* performed for *CmValidateSignature*.

3.3 Encryption operations

3.3.1 CmCrypt2

Description

CmCrypt2 encrypts or decrypts data using direct or indirect encryption (see Encryption Modes) via a CmContainer entry.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmCrypt2(HCMSysEntry hcmse, CMULONG flCtrl, CMCRYPT2 *pcmCrypt,
void* pvDest, CMUINT cbDest)
```

Parameter

hcmse contains a handle opened by *CmAccess2*. Access Mode must be Entry Access to the Firm Code:Product Code entry used for en-/decryption. For software encryption (**CM_CRYPT_AES_ENC_CBC_DIRECT** or **CM_CRYPT_AES_DEC_CBC_DIRECT**) this has to be NULL.

flCtrl contains the encryption or decryption command. Following commands are supported:

- *CM_CRYPT_DIRECT_ENC* executes the **direct** symmetric encryption via a CmContainer entry. The length of bytes to encrypt must be exactly 16 bytes (for *CM_CRYPT_AES*) or between 0 and 32 bytes (for *CM_CRYPT_ECIES* and *CM_CRYPT_ECIES_STD*) - see structure CMBASECRYPT2.

Structure CMBASECRYPT2

CMBASECRYPT2 stores all needed parameters needed for every possible en- or decryption.

© Copyright 2018 WIBU-SYSTEMS AG

Syntax

```
typedef struct __CMBASECRYPT2 {
  CMULONG mflCtrl;
  CMULONG mulKeyExtType;
  CMULONG mulEncryptionCode;
  CMULONG mulEncryptionCodeOptions;
  CMULONG mulFeatureCode;
  CMTIME mcmReleaseDate;
  CMULONG mulCrc;
} CMBASECRYPT2;
```

Parameter

mflCtrl contains commands and flags influencing the en- or decryption. The parameter can use the following flags:

Used Key:

- *CM_CRYPT_FIRMKEY* specifies that the encryption key is stored in the Firm Key of a Firm Item.
- *CM_CRYPT_HIDDENDATA* specifies that the encryption key is stored in a Hidden Data of a Product Item. The Key Source must also be specified here.
- *CM_CRYPT_SECRETDATA* specifies that the encryption key is stored in a Secret Data of a Product Item. The Key Source must also be specified here.

Used Encryption Algorithm:

- *CM_CRYPT_AES* specifies that the Advanced Encryption Standard (AES) algorithm is used for encryption. The key may be *CM_CRYPT_FIRMKEY*, *CM_CRYPT_HIDDENDATA* or *CM_CRYPT_SECRETDATA*. The encryption will be affected by other parameters such as Firm Code, Product Code, Encryption Code etc.
- *CM_CRYPT_ECIES* specifies that the Elliptic Curve Integrated Encryption Scheme (ECIES) algorithm is used for encryption. This flag is only allowed for direct encryption (see Encryption Modes).
- *CM_CRYPT_ECIES_STD* specifies that the Elliptic Curve Integrated Encryption Scheme (ECIES) algorithm is used for encryption. This option executes the standard ECIES. This flag only works with direct encryption. Encryption only works with simulated encryption (*CmCryptSim*). The length of the buffer must be at least 64 bytes (*CM_PUBLIC_KEY_LEN*) larger than the input sequence to add the public key after the encrypted sequence. Decryption only works with real decryption (*CmCrypt2*). Though the returned length is 64 bytes smaller than the input sequence. This flag is only valid for direct encryption (see Encryption Modes)
- *CM_CRYPT_AES_DIRECT* specifies that the direct Advanced Encryption Standard (AES) algorithm is used for encryption without hash calculation. The length of the buffer must be at least 16 bytes. The key to use may be *CM_CRYPT_HIDDENDATA* or *CM_CRYPT_SECRETDATA*. *CM_CRYPT_FIRMKEY* is not implemented. This algorithm is compatible to standard AES. Required CmContainer Firmware version: 1.10.
- *CM_CRYPT_RSA* specifies that the standard RSA algorithm is used. The length of the buffer must be at least 128 bytes (*CM_RSA1024_KEY_LEN*). The keys must locate in a Secret Data entry of a Firm Code or Product Code:
  - d (private key) in SecretData ExtType 130
  - p (secret prime) in SecretData ExtType 131
  - q (secret prime) in SecretData ExtType 132

  Requirements:
  - For d, p, and q exactly these ExtTypes must be programmed.
  - For security reasons these SecretData types must be programmed in this exact sequence. First ExtType 130, then 131, and finally 132. If programmed in another order, the respective preceeding SecretData fields are overwritten.
  - All three numbers must be specified using Little Endian.
    Please note that some tools, for example, openssl create keys using Big Endian.

  CM_CRYPT_RSA requires at least a CmContainer Firmware version 1.11. Then key lengths up to 1024 bytes are supported. Firmware version 2.02 and higher supports key lengths up to 2048 bytes.

CRC calculation:

The algorithm CRC-32 is used to calculate the CRC checksum of the data.

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_CRYPT_CHKCRC* specifies that the CRC value stored in **mulCrc** should be checked using the CRC of the current operation.
- *CM_CRYPT_CALCCRC* specifies that the CRC of the current operation should be calculated and stored in **mulCrc**.

mulKeyExtType specifies the key source, if the encryption is done with a Hidden Data or a Secret Data.

mulEncryptionCode contains the Encryption Code set by the user.

mulEncryptionCodeOptions contains one or more of the following flags:

- *CM_CRYPT_RES1MASK* reserved for future use.
- *CM_CRYPT_UCMASK* specifies all flags for the Unit Counter options.
- *CM_CRYPT_UCCHECK* specifies that the Unit Counter is checked and reduced, if available as Product Item Option.
- *CM_CRYPT_UCIGNORE* specifies that the Unit Counter is neither checked nor reduced, independent of its availability as Product Item Option.
- *CM_CRYPT_UCREQUIRED* specifies that a Unit Counter is checked and reduced and must be available as Product Item Option.
- *CM_CRYPT_ATMASK* specifies all flags for the Activation Time.
- *CM_CRYPT_ATCHECK* specifies that the Activation Time is checked, if available as Product Item Option.
- *CM_CRYPT_ATIGNORE* specifies that the Activation Time is not checked, independent of its availability as Product Item Option.
- *CM_CRYPT_ATREQUIRED* specifies that a Activation Time is checked and must be available as Product Item Option.
- *CM_CRYPT_ETMASK* specifies all flags for the Expiration Time options.
- *CM_CRYPT_ETCHECK* specifies that Expiration Time or Usage Period are checked, if available as Product Item Option.
- *CM_CRYPT_ETIGNORE* specifies that Expiration Time and Usage Period are not checked, independent of its availability as Product Item Option.
- *CM_CRYPT_ETREQUIRED* specifies that Expiration Time or Usage Period are checked and one of them must be available as Product Item Option.
- *CM_CRYPT_CERTTIME* specifies that the Certified Box Clock (CBC) has to be updated by a Certified Time Stamp (CTS) since the CmDongle has received power. This option is only evaluated, if an Expiration Time Product Item Option is available.
- *CM_CRYPT_FACDECREMENT* specifies that the Access Counter is reduced by 1.
- *CM_CRYPT_MPREQUIRED* specifies that a Maintenance Period is checked and must be available as Product Item Option.
- *CM_CRYPT_TOPLAINONLY* specifies that the CmContainer can execute only the decryption but not the encryption variant. This flag can only be used in combination with the *CmCrypt2* flag *CM_CRYPT_DIRECT_DEC*. The corresponding encryption has to be done by using *CmCryptSim2*.
- *CM_CRYPT_RES2MASK* reserved for future use.
- *CM_CRYPT_SAMASK* specifies all flags for network access. These bits affect the encryption.



Please make sure that these flags correspond to the access modes flags you have set the in *CmAccess2*.

- *CM_CRYPT_SAEXCLUSIVE* specifies that the current access is done using Exclusive Mode.
- *CM_CRYPT_SAUNLIMITED* specifies that the current access is done using No User Limit.
- *CM_CRYPT_SAUSERLIMIT* specifies that the current access is done using User Limit.
- *CM_CRYPT_SASTATIONSHARE* specifies that the current access is done using Stationshare Mode.
- *CM_CRYPT_RES3MASK* reserved for future use.
- *CM_CRYPT_UCDELTAMASK* contains the mask to specify the delta value to reduce the Unit Counter value during encryption. The value is limited to the range of 0 to 9999, values between 10000 and 16383 are reserved for future use.

mulFeatureCode specifies the Feature Code mapping for the existing Feature Map. The operation fails, if the Feature Code contains bits which do not exist in the Feature Map.

© Copyright 2018 WIBU-SYSTEMS AG

mcmReleaseDate specifies the Release Date in a ⊟ CMTIME structure that has to map to the existing Maintenance Period. The operation fails, if the Release Date is not in the Maintenance Period.

Structure CMTIME

CMTIME sets or gets the time to or from a CmContainer.

Syntax

```
typedef struct __CMTIME {
  CMUSHORT musYear;
  CMUSHORT musMonth;
  CMUSHORT musDay;
  CMUSHORT musHours;
  CMUSHORT musMinutes;
  CMUSHORT musSeconds;
  CMULONG mulSecondsSince01_01_2000;
} CMTIME;
```

Parameter

musYear specifies the value of the year (e.g. `2003`).
musMonth specifies the value of the month (`1..12`).
musDay specifies the value of the day (`1..31`).
musHours specifies the value of the hours (`0..23`).
musMinutes specifies the value of the minutes (`0..59`).
musSeconds specifies the value of the seconds (`0..59`).
mulSecondsSince01_01_2000 specifies the time in seconds since `01/01/2000`.

mulCrc contains the CRC value of the current operation. It depends on the `CM_CRYPT_CALCCRC` or `CM_CRYPT_CHKCRC` flags, if this value is set or checked.

- *`CM_CRYPT_DIRECT_DEC`* executes the **direct** symmetric decryption via a CmContainer entry. The length of bytes to decrypt must be exactly 16 bytes (for *`CM_CRYPT_AES`*) or between 0 and 32 bytes (for *`CM_CRYPT_ECIES`* and *`CM_CRYPT_ECIES_STD`*) - see CMBASECRYPT2.

  Wibu-Systems recommends this mode since CBC is the most commonly used mode of operation.
  Although *CodeMeter* supports also other older cryptographic modes. However, their use is not recommended.
  For more information see the technical guideline TR-02102-1 "Cryptographic Mechanisms: Recommendations and Key Lengths" (courtesy translation) edited by the German Federal Office for Information Security, BSI (Bundesamt für Sicherheit in der Informationstechnik) (BSI).

- *`CM_CRYPT_AES_ENC_CBC`* executes the **indirect** encryption using the AES block cipher algorithm with CBC.
  By default, the method **CTS** is used. CTS stands for ciphertext stealing, i.e. swapping the final two ciphertext blocks, also in the case that `cbDest` is a multiple of the blocksize.
  Moreover, CTS allows unpadded data, that has a length not a multiple of the block site (16 bytes), to be encrypted to the same length of the cipher text.

  `cbDest` must always be equal to or larger than 16 bytes. The initialization vector (IV) is never stolen.
  If `cbDest` equals 16 bytes, the encryption results corresponds to the encryption using standard AES CBC without CTS.
  If `cbDest` equals 16 bytes and the initialization vector (IV) equals a value of `0`, the encryption result corresponds to ECB.

- *`CM_CRYPT_AES_DEC_CBC`* executes the **indirect** decryption using the AES block cipher algorithm with CBC.
  By default, the method **CTS** is used. CTS stands for ciphertext stealing, i.e. swapping the final two ciphertext blocks, also in the case that `cbDest` is a multiple of the blocksize.

© Copyright 2018 WIBU-SYSTEMS AG

Moreover, CTS allows unpadded data, that has a length not a multiple of the block site (16 bytes), to be encrypted to the same length of the cipher text.

> `cbDest` must always be equal to or larger than 16 bytes. The initialization vector (IV) is never stolen.
> If `cbDest` equals 16 bytes, the encryption results corresponds to the encryption using standard AES CBC without CTS.
> If `cbDest` equals 16 bytes and the initialization vector (IV) equals a value of `0`, the encryption result corresponds to ECB.

- *CM_CRYPT_AES_ENC_CBC_DIRECT* executes the encryption using the original AES block cipher algorithm with CBC.
  By default, the method **CTS** is used. CTS stands for ciphertext stealing, i.e. swapping the final two ciphertext blocks, also in the case that `cbDest` is a multiple of the blocksize.
  Moreover, CTS allows unpadded data, that has a length not a multiple of the block site (16 bytes), to be encrypted to the same length of the cipher text.

  > `cbDest` must always be equal to or larger than 16 bytes. The initialization vector (IV) is never stolen.
  > If `cbDest` equals 16 bytes, the encryption results corresponds to the encryption using standard AES CBC without CTS.
  > If `cbDest` equals 16 bytes and the initialization vector (IV) equals a value of `0`, the encryption result corresponds to ECB.

  It does not use any CodeMeter license, therefore use NULL for **hcmse**. The AES key has to be placed in **mabDirectAesKey**, the initialization vector in **mabInitKey** - both see structure CMCRYPT2).

Structure CMCRYPT2

CMCRYPT2 contains all needed parameters for an encryption with a CmContainer using *CmCrypt2*

Syntax

```
typedef struct __CMCRYPT2 {
  CMBASECRYPT2 mcmBaseCrypt;
  CMBYTE mabDirectAesKey[CM_BLOCK_SIZE];
  CMBYTE mabInitKey[CM_BLOCK_SIZE];
  CMULONG maulReserved[8]
} CMCRYPT2;
```

Parameter

mcmBaseCrypt contains all needed parameters for an encryption (see CMBASECRYPT2).

mabDirectAesKey contains the AES key for an encryption/decryption without CmContainer using original AES. This is only used for *CmCrypt2* with the flags **`CM_CRYPT_AES_ENC_CBC_DIRECT`** or **`CM_CRYPT_AES_DEC_CBC_DIRECT`** (see also Encryption Modes).

mabInitKey contains the key for the indirect encryption (see Encryption Modes). For an encryption/decryption without hardware using *CmCrypt2* with the flags **`CM_CRYPT_AES_ENC_CBC_DIRECT`** or **`CM_CRYPT_AES_DEC_CBC_DIRECT`** use this member to set the initialization vector (IV).

- *CM_CRYPT_AES_DEC_CBC_DIRECT* executes the decryption using the original AES block cipher algorithm with CBC.
  By default, the method **CTS** is used. CTS stands for ciphertext stealing, i.e. swapping the final two ciphertext blocks, also in the case that `cbDest` is a multiple of the blocksize.
  Moreover, CTS allows unpadded data, that has a length not a multiple of the block site (16 bytes), to be encrypted to the same length of the cipher text.

  > `cbDest` must always be equal to or larger than 16 bytes. The initialization vector (IV) is never stolen.
  > If `cbDest` equals 16 bytes, the encryption results corresponds to the encryption using standard AES CBC without CTS.
  > If `cbDest` equals 16 bytes and the initialization vector (IV) equals a value of `0`, the encryption

© Copyright 2018 WIBU-SYSTEMS AG

result corresponds to ECB.

It does not use any CodeMeter license, therefore use NULL for **hcmse**. The AES key has to be placed in **mabDirectAesKey**, the initialization vector in **mabInitKey** - both see structure CMCRYPT2).

Structure CMCRYPT2

CMCRYPT2 contains all needed parameters for an encryption with a CmContainer using *CmCrypt2*

Syntax

```
typedef struct __CMCRYPT2 {
  CMBASECRYPT2 mcmBaseCrypt;
  CMBYTE mabDirectAesKey[CM_BLOCK_SIZE];
  CMBYTE mabInitKey[CM_BLOCK_SIZE];
  CMULONG maulReserved[8]
} CMCRYPT2;
```

Parameter

mcmBaseCrypt contains all needed parameters for an encryption (see CMBASECRYPT2).

mabDirectAesKey contains the AES key for an encryption/decryption without CmContainer using original AES. This is only used for *CmCrypt2* with the flags **CM_CRYPT_AES_ENC_CBC_DIRECT** or **CM_CRYPT_AES_DEC_CBC_DIRECT** (see also Encryption Modes).

mabInitKey contains the key for the indirect encryption (see Encryption Modes). For an encryption/decryption without hardware using *CmCrypt2* with the flags **CM_CRYPT_AES_ENC_CBC_DIRECT** or **CM_CRYPT_AES_DEC_CBC_DIRECT** use this member to set the initialization vector (IV).

pcmCrypt sets some needed data for the en-/decryption process (see CMCRYPT2).

pvDest contains the data to en-/decrypt.

cbDest contains the number of bytes to en-/decrypt stored in **pvDest**.

Return

This function returns the number of en-/decrypted bytes. If an error occurred, `0` will be returned.

Notes

CmCrypt2() is supported since CodeMeter Version 4.30.

Crypt sets some needed data for the en-/decryption process (see CmCrypt2).

Dest contains the data to encrypt or decrypt.

Return

This function returns true. If an error occurred, `false` will be returned.

Notes

CmCrypt2() is supported since CodeMeter Version 4.30.

3.3.2 CmCryptEcies

Description

CmCryptEcies encrypts a given sequence (up to 32 bytes) with the standard ECIES algorithm. This function is a helper function to use the complete functionality within CodeMeter API.

C, C++ Programming Language

Syntax

© Copyright 2018 WIBU-SYSTEMS AG

```
int CMAPIENTRY CmCryptEcies(CMBYTE* pbPublicKey, CMUINT cbPublicKey, CMBYTE*
pbData, CMUINT cbData)
```

Parameter

pbPublicKey contains the public key of the other side.

cbPublicKey contains the length of the public key **pbPublicKey**. Has to be exactly 64 bytes.

pbData is an encryption buffer which contains the data to be encrypted. Its length must be equal to the length of this data plus the length of the public key.

cbData contains the length of **pbData**.

Return

This function returns the number of en-/decrypted bytes. If an error occurred, `0` will be returned.

3.3.3 CmGetSecureData

Description

CmGetSecureData reads encrypted Hidden Data from a CmContainer entry. The retrieved encrypted data has to be decrypted individually.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmGetSecureData(HCMSysEntry hcmse, CMSECUREDATA *pcmSecureData,
CMENTRYDATA *pcmEntryData);
```

Parameter

hcmse contains a handle opened by *CmAccess2* (must be Entry Access).

pcmSecureData contains a ⊟ CMSECUREDATA structure.

Structure CMSECUREDATA

CMSECUREDATA stores all needed parameters for an encrypted reading of a PIO.

Syntax

```
typedef struct __CMSECUREDATA {
    CMBASECRYPT mcmBaseCrypt;
    CMUSHORT musPioType;
    CMUSHORT musExtType;
    CMBYTE mabPioEncryptionKey[CM_BLOCK_SIZE];
    CMULONG mulReserve;
} CMSECUREDATA;
```

Parameter

mcmBaseCrypt contains all needed parameters for an encryption (see structure ⊟ CMBASECRYPT for parameter settings for data to be encrypted.

Structure CMBASECRYPT

CMBASECRYPT stores all needed parameters needed for every possible en- or decryption.

Syntax

```
    typedef struct __CMBASECRYPT {
    CMULONG mflCtrl;
    CMULONG mulKeyExtType;
    CMULONG mulEncryptionCode;
    CMULONG mulEncryptionCodeOptions;
```

© Copyright 2018 WIBU-SYSTEMS AG

```
    CMULONG mulFeatureCode;
    CMULONG mulCrc;
    } CMBASECRYPT;
```

Parameter

mflCtrl contains commands and flags influencing the en- or decryption. The parameter can use the following flags:

Used Key:

- *CM_CRYPT_HIDDENDATA* specifies that the encryption key is stored in a Hidden Data of a Product Item. The Key Source must also be specified here.

Used Encryption Algorithm:

- *CM_CRYPT_AES* specifies that the Advanced Encryption Standard (AES) algorithm is used for encryption. The key *CM_CRYPT_HIDDENDATA*. The encryption will be affected by other parameters such as Firm Code, Product Code, Encryption Code etc.
- *CM_CRYPT_ECIES* specifies that the Elliptic Curve Integrated Encryption Scheme (ECIES) algorithm is used for encryption. This flag is only allowed for direct encryption (see Encryption Modes).
- *CM_CRYPT_ECIES_STD* specifies that the Elliptic Curve Integrated Encryption Scheme (ECIES) algorithm is used for encryption. This option executes the standard ECIES. This flag only works with direct encryption. Encryption only works with simulated encryption (*CmCryptSim*). The length of the buffer must be at least 64 bytes (*CM_PUBLIC_KEY_LEN*) larger than the input sequence to add the public key after the encrypted sequence. Decryption only works with real decryption (*CmCrypt*). Though the returned length is 64 bytes smaller than the input sequence. This flag is only valid for direct encryption (see Encryption Modes)
- *CM_CRYPT_AES_DIRECT* specifies that the direct Advanced Encryption Standard (AES) algorithm is used for encryption without hash calculation. The length of the buffer must be at least 16 bytes. The key to use is *CM_CRYPT_HIDDENDATA*. This algorithm is compatible to standard AES. Required CmContainer Firmware version: 1.10.

CRC calculation:

The algorithm CRC-32 is used to calculate the CRC checksum of the data.

- *CM_CRYPT_CHKCRC* specifies that the CRC value stored in mulCrc should be checked using the CRC of the current operation.
- *CM_CRYPT_CALCCRC* specifies that the CRC of the current operation should be calculated and stored in **mulCrc**.

mulKeyExtType specifies the key source, if the encryption is done with a Hidden Data.

mulEncryptionCode contains the Encryption Code set by the user.

mulEncryptionCodeOptions contains one or more of the following flags:

- *CM_CRYPT_RES1MASK* reserved for future use.
- *CM_CRYPT_UCMASK* specifies all flags for the Unit Counter options.
- *CM_CRYPT_UCCHECK* specifies that the Unit Counter is checked and reduced, if available as Product Item Option.
- *CM_CRYPT_UCIGNORE* specifies that the Unit Counter is neither checked nor reduced, independent of its availability as Product Item Option.
- *CM_CRYPT_UCREQUIRED* specifies that a Unit Counter is checked and reduced and must be available as Product Item Option.
- *CM_CRYPT_ATMASK* specifies all flags for the Activation Time.
- *CM_CRYPT_ATCHECK* specifies that the Activation Time is checked, if available as Product Item Option.
- *CM_CRYPT_ATIGNORE* specifies that the Activation Time is not checked, independent of its availability as Product Item Option.
- *CM_CRYPT_ATREQUIRED* specifies that a Activation Time is checked and must be available as Product Item Option.
- *CM_CRYPT_ETMASK* specifies all flags for the Expiration Time options.
- *CM_CRYPT_ETCHECK* specifies that Expiration Time or Usage Period are checked, if available as Product Item Option.

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_CRYPT_ETIGNORE* specifies that Expiration Time and Usage Period are not checked, independent of its availability as Product Item Option.
- *CM_CRYPT_ETREQUIRED* specifies that Expiration Time or Usage Period are checked and one of them must be available as Product Item Option.
- *CM_CRYPT_CERTTIME* specifies that the Certified Box Clock (CBC) has to be updated by a Certified Time Stamp (CTS) since the CmDongle has received power. This option is only evaluated, if an Expiration Time Product Item Option is available.
- *CM_CRYPT_FACDECREMENT* specifies that the Access Counter is reduced by 1.
- *CM_CRYPT_TOPLAINONLY* specifies that the CmContainer can execute only the decryption but not the encryption variant. This flag can only be used in combination with the *CmCrypt* flag *CM_CRYPT_DIRECT_DEC*. The corresponding encryption has to be done by using *CmCryptSim*.
- *CM_CRYPT_RES2MASK* reserved for future use.
- *CM_CRYPT_SAMASK* specifies all flags for network access. These bits affect the encryption.

> Please make sure that these flags correspond to the access modes flags you have set the in *CmAccess2*.

- *CM_CRYPT_SAEXCLUSIVE* specifies that the current access is done using Exclusive Mode.
- *CM_CRYPT_SAUNLIMITED* specifies that the current access is done using No User Limit.
- *CM_CRYPT_SAUSERLIMIT* specifies that the current access is done using User Limit.
- *CM_CRYPT_SASTATIONSHARE* specifies that the current access is done using Stationshare Mode.
- *CM_CRYPT_RES3MASK* reserved for future use.
- *CM_CRYPT_UCDELTAMASK* contains the mask to specify the delta value to reduce the Unit Counter value during encryption. The value is limited to the range of 0 to 9999, values between 10000 and 16383 are reserved for future use.

mulFeatureCode specifies the Feature Code mapping for the existing Feature Map. The operation fails, if the Feature Code contains bits which do not exist in the Feature Map.

mulCrc contains the CRC value of the current operation. It depends on the *CM_CRYPT_CALCCRC* or *CM_CRYPT_CHKCRC* flags, if this value is set or checked.

musExtType contains the extended type (ID) of the PIO to read.

musPioType contains the type of the PIO to read out in an encrypted way.

- *CM_GF_HIDDENDATA* specifies that the PIO to read is a Hidden Data.

mabPioEncryptionKey [*CM_BLOCK_SIZE*] contains the PIOEK (Product Item Option Encryption Key).

mulReserve is reserved for future purposes.

pcmEntryData contains a CMENTRYDATA structure receiving the encrypted data.

Structure CMENTRYDATA

CMENTRYDATA contains a dynamic added data of a Product Item with all its properties.

Syntax

```
typedef struct __CMENTRYDATA {
    CMULONG mflCtrl;
    CMULONG mflDependency;
    CMUINT mcbData;
    CMBYTE mabData[2 * CM_MAX_STRING_LEN];
} CMENTRYDATA;
```

Parameter

mflCtrl specifies the type of the dynamic added entry. It can contain the following flag:

- *CM_GF_HIDDENDATA* specifies that the Product Item contains one or more HiddenData Data.

For the entry type Hidden Data the index of this data type is or'ed in the upper two bytes. See remarks for a sample.

© Copyright 2018 WIBU-SYSTEMS AG

mflDependency contains the dependency of the dynamic added entry. It can be have one or more of the following flags:
- *CM_GF_DATA* specifies that the parameter depends on the contents of the data of the Firm Item.
- *CM_GF_FUC* specifies that the parameter depends on the Firm Update Counter of the Firm Item.
- *CM_GF_SERIAL* specifies that the parameter depends on the Serial Number of the CmContainer.

mcbData contains the length of **mabData**, for text entries (*CM_GF_TEXT*) the number of characters, for all other types the number of stored bytes.

mabData[*2 * CM_MAX_STRING_LEN*] contains the data of the dynamic added entry with the length stored in **mcbData**. For text entries (CM_GF_TEXT) the byte array contains an UCS2-LE encoded null-terminated string.

Notes

To check the **mflCtrl** you can use the **CM_GF_PIO_MASK** to retrieve only the data type flag, e.g. *mflCtrl & CM_GF_PIO_MASK*. To retrieve the index of Hidden Data you need to extract the upper two bytes of **mflCtrl**, e.g. by **mflCtrl >> 16**.

For an easier conversion of **mabData** it may be helpful to use *CmConvertString()*.

Return

The return value contains the number of bytes containing the plain data returned encrypted in CMENTRYDATA::mabData. If an error occurred, 0 is returned.

Notes

For Hidden Data, the Hidden Data Access Code (HDAC) must specified and match the stored value.

3.3.4 CmDecryptPioData

Description

CmDecryptPioData decrypts a given encrypted Hidden Data sequence using the Product Item Option Data Key (PIODK).

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmDecryptPioData(CMBYTE* pabDest, CMUINT cbDest, CMBYTE*
pabPiodkDest, CMUINT cbPiodkDest)
```

Parameter

pabDest points to the data received by *CmGetSecureData* to be decrypted in this buffer.
cbDest contains the length of data at **pabDest**. Has to be a multiple of 16.
pabPiodkDest points to the input Product Item Option Decryption Key (PIODK) controlling the decryption.
cbPiodkDest contains the length of data at **pabDest**. Usually the length is 16.

Return

TRUE is returned, if the decryption was successful - FALSE, if not.

3.3.5 CmGetPioDataKey

Description

CmGetPioDataKey calculates the Product Item Option Data Key (PIODK) from the Product Item Option Encryption Key (PIOEK) and the Product Item Option Core Key (PIOCK).

© Copyright 2018 WIBU-SYSTEMS AG

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmGetPioDataKey( CMBYTE* pabPiodkDest, CMUINT cbPiodkDest,
CMBYTE* pabPioCk, CMUINT cbPioCk, CMBYTE* pabPioEk, CMUINT cbPioEk)
```

Parameter

pabPiodkDest points to a memory block of 16 bytes which receives the calculated PIODK.
cbPiodkDest contains the length of **pabPiodkDest**.Usually the length is 16.
pabPioCk points to the input PIOCK.
cbPioCk contains the length of **pabPioCk**. Usually the length is 16.
pabPioEk points to the input PIOEK.
cbPioEk contains the length of **pabPioEk**. Usually the length is 16.

Return

TRUE is returned, if the calculation executed successfully - FALSE if not.

3.4 Error Management operations

3.4.1 CmConvertString

Description

This function converts an input string into the specified character set.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmConvertString(CMUINT flCtrl, const void *pszSrc, void *pszDst,
CMUINT cbDst);
```

Parameter

flCtrl

- *CM_CONVERT_UTF8_TO_WCHAR*
- *CM_CONVERT_UTF8_TO_UCS2LE*
- *CM_CONVERT_UTF8_TO_ASCII*
- *CM_CONVERT_UTF8_TO_UTF16*
- *CM_CONVERT_UTF8_TO_LOCAL*
- *CM_CONVERT_WCHAR_TO_UCS2LE*
- *CM_CONVERT_WCHAR_TO_UTF8*
- *CM_CONVERT_WCHAR_TO_ASCII*
- *CM_CONVERT_WCHAR_TO_UTF16*
- *CM_CONVERT_WCHAR_TO_LOCAL*
- *CM_CONVERT_UCS2LE_TO_WCHAR*
- *CM_CONVERT_UCS2LE_TO_UTF8*
- *CM_CONVERT_UCS2LE_TO_ASCII*
- *CM_CONVERT_UCS2LE_TO_UTF16*
- *CM_CONVERT_UCS2LE_TO_LOCAL*
- *CM_CONVERT_ASCII_TO_WCHAR*
- *CM_CONVERT_ASCII_TO_UCS2LE*
- *CM_CONVERT_ASCII_TO_UTF8*
- *CM_CONVERT_ASCII_TO_UTF16*
- *CM_CONVERT_ASCII_TO_LOCAL*
- *CM_CONVERT_UTF16_TO_UTF8*
- *CM_CONVERT_UTF16_TO_UCS2LE*
- *CM_CONVERT_UTF16_TO_WCHAR*

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_CONVERT_UTF16_TO_ASCII*
- *CM_CONVERT_UTF16_TO_LOCAL*
- *CM_CONVERT_LOCAL_TO_UTF8*
- *CM_CONVERT_LOCAL_TO_UCS2LE*
- *CM_CONVERT_LOCAL_TO_WCHAR*
- *CM_CONVERT_LOCAL_TO_ASCII*
- *CM_CONVERT_LOCAL_TO_UTF16*

pszSrc pointer to memory that holds the string to convert. It must be null-terminated and not longer than 64 Kbytes.

pszDst destination buffer.

cbDest holds the length of **pszDst** in bytes.

Return

Returns on success the number of converted characters (including terninating null character). The number of bytes varies depending on the encoding. For example, if using tag *CM_CONVERT_UTF8_TO_UTF16*, the function returns 15, the resulting string has also 15 characters but a size of 30 bytes. If an error occurs, zero is returned.

Notes

ASCII: 7bit encoding, ASCII-Norm.

WCHAR: wchar_t encoded, on Windows/CE UCS-2 and on Unix UTF-32.

UCS2LE: Little Endian encoded, without white blank mark.

LOCAL: 8bit local encoding.

UTF-8: UTF-8 encoded strings.

UTF-16: UTF-16 encoded strings.

3.4.2 CmGetLastErrorCode

Description

CmGetLastErrorCode returns the last set error code. Every API function/method call sends a request from the calling driver (on Windows: `WibuCm32.dll`) directly to CodeMeter License Server (on Windows: `CodeMeter.exe`). With the answer CodeMeter License Server always sends the last occurred error code back to the calling driver.

At this location the error code is stored in a global variable declared as Thread Local Storage (TLS). *CmGetLastErrorCode* returns this variable. If a problem in the calling driver occurs, an error code is also set at this global variable. Because of using Thread Local Storage the calling driver is thread safe.



Please note that with CodeMeter Version 6.10 for programming in C# the function *CmGetLastErrorCode* has become obsolete. Please use the new function *CmGetLastErrorCode2* instead.
If using a CodeMeter Version smaller than 6.10 please use function *CmGetLastErrorCode*.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmGetLastErrorCode()
```

Return

The return value contains the error code set from the last called API function.

© Copyright 2018 WIBU-SYSTEMS AG

3.4.3 CmGetLastErrorText

Description

This function / method gets the error text of the last set error. The language of the text is the current set system language or English, if there's no language support for the system language. If the *CM_GLET_DIALOG* flag is set, the error text and some useful information are displayed in a message box (Windows / Windows CE only).

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmGetLastErrorText(CMULONG flCtrl, char *pszErrorText, CMUINT
cbErrorText)
```

Parameter

flCtrl

- *CM_GLET_ERRORTEXT* returns the related error text of the last set error code.
- *CM_GLET_DIALOG* displays a message box with the related error text of the last set error code. This flag is only considered on Windows/Windows CE.
- *CM_GLET_DIALOG* and *CM_GLET_ERRORTEXT* can be or'ed with the following flags:
- *CM_GLET_LOCAL_ENCODING* uses local 8bit encoding.
- *CM_GLET_UTF8_ENCODING* represents any character in the Unicode standard.
- *CM_GLET_WCHAR_ENCODING* uses wide character encoding in ANSI/ISO C (UCS-2 on Windows/Windows CE and UTF-32 on Unix).

pszErrorText contains a pointer to a string to receive the error text.
cbErrorText contains the length of **pszErrorText** in bytes.

Return

This function returns the length of the error text or zero, if an error occurred. If *pszErrorText* does not contain enough memory, the number of needed bytes is returned.

3.4.4 CmGetLastErrorText2

Description

This function gets the error text of the last set error. The language of the text is the current set system language or English if there is no language support for the system language. If the *CM_GLET_DIALOG* flag is set, the error text and some useful information are displayed in a message box (Windows / Windows CE only).

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmGetLastErrorText2(CMULONG flCtrl, void *pvErrorText, CMUINT
cbErrorText);
```

Parameter

flCtrl

- *CM_GLET_ERRORTEXT* returns the related error text of the last set error code.

CM_GLET_ERRORTEXT can be or'ed with the following flags

- *CM_GLET_UTF8_ENCODING* represents any character in the Unicode standard.
- *CM_GLET_WCHAR_ENCODING* uses wide character encoding in ANSI/ISO C (UCS-2 on Windows/Windows CE and UTF-32 on Unix).

pvErrorText pointer to memory to hold the data.
cbErrorText holds the length of **pvErrorText** in bytes.

© Copyright 2018 WIBU-SYSTEMS AG

Return

Returns on success the number of returned bytes or zero, if an error occurred. If *pvErrorText* does not contain enough memory, the number of required bytes is returned.

3.4.5 CmSetLastErrorCode

Description

CmSetLastErrorCode sets an error code into the internal global error code variable.

> Please note that with CodeMeter Version 6.10 for programming in C# the function *CmSetLastErrorCode* has become obsolete. Please use the new function *CmSetLastErrorCode2* instead.
> If using a CodeMeter Version smaller than 6.10 please use function *CmGetLastErrorCode*.

C, C++ Programming Language

Syntax

```
void CMAPIENTRY CmSetLastErrorCode(int idError)
```

Parameter

idError specifies the error code to be set.

3.5 Managing operations

3.5.1 CmGetBoxContents2

Description

This function returns the entries of a CmContainer.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmGetBoxContents2(HCMSysEntry hcmse, CMULONG flCtrl, CMULONG
ulFirmCode, CMBOXINFO *pcmBoxInfo, CMBOXENTRY2 *pcmBoxEntry, CMUINT cbBoxEntry)
```

Parameter

hcmse contains a handle opened by *CmAccess2* (can be Box Access, Entry Access or Subsystem Access).

flCtrl contains one of the following commands:

- *CM_GBC_FI*: All entries returned must match the Firm Item provided in **ulFirmCode**.
- *CM_GBC_BOX*: All entries returned must match the CmContainer properties provided in **pcmBoxInfo**.
- *CM_GBC_ALLENTRIES*: All entries of all connected CmContainers available at *hcmse* will be returned.

The above flags can be or'ed with

- *CM_GBC_USELOCALTIME*: Returns the time values using the local time zone instead of UTC.

The flags *CM_GBC_FI* and *CM_GBC_BOX* can be combined. If the value of **ulFirmCode** or **pcmBoxInfo** is not relevant for the desired flag operation, the value is ignored.

ulFirmCode contains the Firm Code of the Firm Item the returned entries should have (used with *CM_GBC_FI* flag).

pcmBoxInfo contains the CmContainer information of the CmContainer the entries should be returned from (used with *CM_GBC_BOX* flag).

pcmBoxEntry contains an array of CMBOXENTRY2 structure.

© Copyright 2018 WIBU-SYSTEMS AG

Structure CMBOXENTRY2

CMBOXENTRY2 contains data of one entry. An entry is containing one Firm Item and one Product Item. In this structure all static data are stored. The dynamic data must be read out of the CmContainer explicitly.

Syntax

```
typedef struct __CMBOXENTRY2 {
    CMULONG mflFiCtrl;
    CMULONG mulFirmCode;
    CMUSHORT musFirmAccessCounter;
    CMULONG mulFirmUpdateCounter;
    CMULONG mulFirmPreciseTime;
    CMUSHORT mausFirmItemText[CM_MAX_STRING_LEN];
    CMULONG mulLicenseTransferUpdateCounter;
    CMULONG mflSetNativePios;
    CMULONG mflSetEffectivePios;
    CMULONG mflSetPios;
    /* Product Item data */
    CMULONG mulProductCode;
    CMULONG mulFeatureMap;
    CMTIME mcmMaintenancePeriodStart;
    CMTIME mcmMaintenancePeriodEnd;
    CMULONG mulUnitCounter;
    CMTIME mcmExpirationTime;
    CMTIME mcmActivationTime;
    CMULONG mulUsagePeriodLifeTime;
    CMTIME mcmUsagePeriodStartTime;
    CMULONG mulLicenseQuantity;
    CMULONG mflLicenseQuantityFlags;
    CMULONG mulLicenseTag;
    CMULONG mflTransferStatus;
    CMBYTE mabProductItemId[10];
    CMUSHORT musMaxEncryptionRate;
    CMBYTE mfbDependencyPC;
    CMBYTE mfbDependencyFM;
    CMBYTE mfbDependencyMP;
    CMBYTE mfbDependencyUC;
    CMBYTE mfbDependencyET;
    CMBYTE mfbDependencyAT;
    CMBYTE mfbDependencyUP;
    CMBYTE mfbDependencyLQ;
    CMBYTE mfbDependencyLT;
    CMBYTE mfbDependencyMV;
    CMUSHORT musModuleRef;
    CMUSHORT musProductItemRef;
    CMUSHORT musTransferBoxMask;
    CMULONG mulTransferSerialNumber;
    CMTIME mcmTransferBorrowExpirationTime;
    CMBYTE mabTransferId[10];
} CMBOXENTRY2;
```

Parameter

mflFiCtrl specifies the type of the Firm Item. The following flags can be set:

- *CM_GF_BOXBASED* specifies that the Firm Item has been programmed in the Box-Based-Mode.
- *CM_GF_FILEBASED* specifies that the Firm Item has been programmed in the File-Based-Mode.
- *CM_GF_TRANSFERBASED* specifies that the Firm Item has been created based on a license transfer.
- *CM_GF_FI_ALLOWED_TO_PULL* specifies the license transfer can be actively performed.
- *CM_GF_FI_RESERVED* specifies that the Firm Item is a Reserved Firm Item.

mulFirmCode contains the Firm Code of the Firm Item.

© Copyright 2018 WIBU-SYSTEMS AG

musFirmAccessCounter contains the Firm Access Counter of the Firm Item.
mulFirmUpdateCounter contains the Firm Update Counter of the Firm Item.
mulFirmPreciseTime contains the FirmPreciseTime of the Firm Item.
mausFirmItemText [*CM_MAX_STRING_LEN*] contains the Firm Item Text of the Firm Item.
mulLicenseTransferUpdateCounter contains the Update Counter of the License Transfer.
mflSetNativePios specifies in the context of used Module Items the native representation of the PIOs as programmed.
mflSetEffectivePios specifies in the context of used Module Items the effective representation, i.e including eventually inherited PIOs.
mflSetPios specifies all available PIO's at this Product Item. The following flags can be set:

- *CM_BE_PRODUCTCODE* specifies that the Product Item contains a Product Code.
- *CM_BE_FEATUREMAP* specifies that the Product Item contains a Feature Map.
- *CM_BE_EXPTIME* specifies that the Product Item contains an Expiration Time.
- *CM_BE_ACTTIME* specifies that the Product Item contains an Activation Time.
- *CM_BE_UNITCOUNTER* specifies that the Product Item contains an Unit Counter.
- *CM_BE_PROTDATA* specifies that the Product Item contains a Protected Data.
- *CM_BE_EXTPROTDATA* specifies that the Product Item contains one or more Extended Protected Data.
- *CM_BE_HIDDENDATA* specifies that the Product Item contains one or more Hidden Data.
- *CM_BE_SECRETDATA* specifies that the Product Item contains one or more Secret Data.
- *CM_BE_USERDATA* specifies that the Product Item contains an User Data.
- *CM_BE_TEXT* specifies that the Product Item contains a Text.
- *CM_BE_USAGEPERIOD* specifies that the Product Item contains an Usage Period.
- *CM_BE_LICENSEQUANTITY* specifies that the Product Item contains a License Quantity.
- *CM_BE_BORROWCLIENT* specifies that the Product Item contains a Borrow Client entry.

  Please note that borrowing in this context does not involve a license transfer. Rather it refers to borrowing using Firm Codes smaller than 6,000,000 and especially prepared CmContainer.
- *CM_BE_BORROWSERVER* specifies that the Product Item contains a Borrow Server entry.

  Please note that borrowing in this context does not involve a license transfer. Rather it refers to borrowing using Firm Codes smaller than 6,000,000 and especially prepared CmContainer.
- *CM_BE_COLI* specifies that the Product Item contains a Customer Owned License Information (COLI).
- *CM_BE_MAINTENANCEPERIOD* specifies that the Product Item contains a Maintenance Period.
- *CM_BE_LINGERTIME* specifies that the Product Item contains a Linger Time.
- *CM_BE_MINIMUMVERSION* specifies that the Product Item contains a Minimal Runtime Version.
- *CM_BE_MODULE_ITEM* specifies that the Product Item contains a Module Item.
- *CM_BE_BORROWEXPIRATIONTIME* specifies that the Product Item contains a Borrow Expiration Time based on a license transfer.
- *CM_BE_HISTORY* specifies that the Product Item is part of a license transfer process and holds History Data.
- *CM_BE_TRANSFEROPTIONS* specifies that the Product Item contains license transfer options.
- *CM_BE_RETURNINFO* specifies that the Product Item contains a license return information.
- *CM_BE_ACCEPTSPULL* specifies that the Product Item can be pulled.
- *CM_BE_NAMEDUSER* specifies that the Product Item contains a Named User information.
- *CM_BE_MAX_ENCRYPTION_RATE* specifies that the Product Item contains a Maximum Encryption Rate information.
- *CM_BE_PIO_MASK* specifies all flags for the Product Item Options.

mulProductCode contains the Product Code of the Product Item.
mulFeatureMap contains the Feature Map of the Product Item.
mcmMaintenancePeriodStart contains the start date of the Maintenance Period of the Product Item, if available.

© Copyright 2018 WIBU-SYSTEMS AG

mcmMaintenancePeriodEnd contains the end date of the Maintenance Period of the Product Item, if available.
mulUnitCounter contains the Unit Counter of the Product Item, if available.
mcmExpirationTime contains the Expiration Time of the Product Item, if available (see CMTIME structure).

Structure CMTIME

CMTIME sets or gets the time to or from a CmContainer.

Syntax

```
typedef struct __CMTIME {
    CMUSHORT musYear;
    CMUSHORT musMonth;
    CMUSHORT musDay;
    CMUSHORT musHours;
    CMUSHORT musMinutes;
    CMUSHORT musSeconds;
    CMULONG mulSecondsSince01_01_2000;
} CMTIME;
```

Parameter

musYear specifies the value of the year (e.g. `2003`).
musMonth specifies the value of the month (`1..12`).
musDay specifies the value of the day (`1..31`).
musHours specifies the value of the hours (`0..23`).
musMinutes specifies the value of the minutes (`0..59`).
musSeconds specifies the value of the seconds (`0..59`).
mulSecondsSince01_01_2000 specifies the time in seconds since `01/01/2000`.

mcmActivationTime contains the Activation Time of the Product Item, if available (see CMTIME).
mulUsagePeriodLifeTime contains the period length of the Usage Period in seconds, if available.
mcmUsagePeriodStartTime contains the start time of the Usage Period, if available (see CMTIME).
mulLicenseQuantity contains the License Quantity of the Product Item, if available.
mflLicenseQuantityFlags contains the flags for license access options, i.e. local accesses.

- *CM_LQ_DENIEDUSERLIMIT*
- *CM_LQ_DENIEDNOUSERLIMIT*
- *CM_LQ_DENIEDEXCLUSIVE*
- *CM_LQ_DENIEDSTATIONSHARE*
- *CM_LQ_ALLOWCMWAN*
- *CM_LQ_TRIPLEMODEREDUNDANCY*
- *CM_LQ_LOCALACCESSONLY*
- *CM_LQ_DENIEDALL*
- *CM_LQ_MAXVALUE*

mulLicenseTag contains information in the context of Triple Mode Redundancy feature if available.
mflTransferStatus contains the flag of the transfer status. The following flags exist:

- *CM_TRANSFER_STATUS_ENABLED*
- *CM_TRANSFER_STATUS_DISABLED*
- *CM_TRANSFER_STATUS_RETURNED*
- *CM_TRANSFER_STATUS_EXPIRED*

mabProductItemId contains an unique identifier for the Product Item.
musMaxEncryptionRate contains the Maximum Encryption Rate.
mfbDependencyPC contains the dependency flags of the Product Code.
mfbDependencyFM contains the dependency flags of the Feature Map.

© Copyright 2018 WIBU-SYSTEMS AG

mfbDependencyUC contains the dependency flags of the Unit Counter.
mfbDependencyET contains the dependency flags of the Expiration Time.
mfbDependencyAT contains the dependency flags of the Activation Time.
mfbDependencyUP contains the dependency flags of the Usage Period.
mfbDependencyLQ contains the dependency flags of the License Quantity.
mfbDependencyLT contains the dependency flags of the Linger Time.
mfbDependencyMV contains the dependency flags of the Minimum Runtime Version.
musModuleRef contains the Product Item Reference of the super-ordinated Product Item of a Module Item. All items of a module then have the same **ModulRef** if available.
musProductItemRef holds the Product Item Reference.
musTransferBoxMask holds the box mask of the source CmContainer in the case of a transferred license if available.
mulTransferSerialNumber holds the serial number of the source CmContainer in the case of a transferred license if available.
mcmTransferBorrowExpirationTime holds the time at which a borrowed license (license transfer) expires if available.
mabTransferId[10] holds the Id of the transfer (corresponds to the Id in the history item of the sender) if available.

Notes

The dynamic data can be read out with *GetInfo()* / `CM_GEI_ENTRYDATA`.
The dependency parameters (**mfbDependencyXX**) can have one or more of the following flags:

- `CM_GF_DATA` specifies that the parameter depends on the contents of the data of the Firm Item.
- `CM_GF_FUC` specifies that the parameter depends on the Firm Update Counter of the Firm Item.
- `CM_GF_SERIAL` specifies that the parameter depends on the Serial Number of the CmContainer.

On Windows the **mausFirmItemText** array can be simply converted via *CmConvertString()* into the specified character set.
CMBOXENTRY2 is supported since CodeMeter Version 4.30.

cbBoxEntry contains the number of CMBOXENTRY2 elements stored in **pcmBoxEntry**.

Return

The number of found entries is returned. If the number is larger than **cbBoxEntry**, the array in **pcmBoxEntry** is too small. In this case only **cbBoxEntry** entries are stored in **pcmBoxEntry**. To get the rest call the function with the correct array length again. If the array is to small the error code CMERROR_BUFFER_OVERFLOW is set. If `0` is returned, then no entry was found and depending on **flCtrl** error code CMERROR_ENTRY_NOT_FOUND or CMERROR_BOX_NOT_FOUND is set.

Notes

CmGetBoxContents2() is supported since CodeMeter Version 4.30.

3.5.2 CmGetBoxes

Description

CmGetBoxes returns all connected CmContainers of the given licensing system.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmGetBoxes(HCMSysEntry hcmse, CMULONG idPort, CMBOXINFO
*pcmBoxInfo, CMUINT cbBoxInfo)
```

Parameter

hcmse contains the handle opened by *CmAccess2* (must be Subsystem Access).

© Copyright 2018 WIBU-SYSTEMS AG

idPort contains one of the following commands:

- *CM_GB_ALLPORTS*: All CmContainers will be scanned.
- *CM_GB_DONGLE*: All CmDongles (CodeMeter hardware) will be scanned. Old naming: *CM_GB_USB*.
- *CM_GB_ACT*: All CmActLicenses will be scanned.

pcmBoxInfo contains an array of CMBOXINFO structure.

Structure CMBOXINFO

CMBOXINFO contains information on a CmContainer.

Syntax

```
typedef struct __CMBOXINFO {
    CMBYTE mbMajorVersion;
    CMBYTE mbMinorVersion;
    CMUSHORT musBoxMask;
    CMULONG mulSerialNumber;
    CMUSHORT musBoxKeyId;
    CMUSHORT musUserKeyId;
    CMBYTE mabBoxPublicKey[CM_PUBLIC_KEY_LEN];
    CMBYTE mabSerialPublicKey[CM_PUBLIC_KEY_LEN];
    CMULONG mulReserve;
} CMBOXINFO;
```

Parameter

mbMajorVersion contains the major version of the CmContainer.
mbMinorVersion contains the minor version of the CmContainer.
musBoxMask contains the box mask of the CmContainer.
mulSerialNumber contains the serial number of the CmContainer.
musBoxKeyId contains the ID of the used BoxKey of the CmContainer.
musUserKeyId contains the ID of the used UserKey of the CmContainer.
mabBoxPublicKey`[CM_PUBLIC_KEY_LEN]` contains the public Box Key of the CmContainer.
mabSerialPublicKey`[CM_PUBLIC_KEY_LEN]` contains the public Serial Key of the CmContainer.
mulReserve is reserved for future purposes.

Notes

The length of the **mabBoxPublicKey** and **mabSerialPublicKey** is 64 byte long. The used public key are currently 224 bits = 28 bytes long. In Elliptic Curve Cryptography the public key is a 2 dimensional point, so the x-value is stored in the first 28 bytes (`0..27`) and the y-value is store in byte `32..59`.

cbBoxInfos contains the number of CMBOXINFO elements stored in **pcmBoxInfo**.

Return

The number of found CmContainers is returned. If the number is larger than **cbBoxInfo**, the array in **pcmBoxInfo** was too small. In this case, only **cbBoxInfo** elements were stored in **pcmBoxInfo**. To get the remaining call the function with the correct array length again. If the array is too small, error code CMERROR_BUFFER_OVERFLOW is set. If 0 is returned,then no entry was found and error code CMERROR_BOX_NOT_FOUND is set.

Notes

This function does not really scan all available or desired ports for connected CmDongles because of CodeMeter License Servers Plug&Play mechanism and the implementation of storing all CmContainers and their content in internal lists. The function call checks the internal lists only and returns the matching data.

© Copyright 2018 WIBU-SYSTEMS AG

3.5.3 CmGetInfo

Description

CmGetInfo returns data related to the opened entry.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmGetInfo(HCMSysEntry hcmse, CMULONG flCtrl, void *pvDest,
CMUINT cbDest)
```

Parameter

hcmse contains the handle opened by *CmAccess2*.

flCtrl The following flags can be set:

- *CM_GEI_BOXCONTROL* fills a CMBOXCONTROL structure specified in **pvDest**.

Structure CMBOXCONTROL

The CmDongle controller chip may support some digital output and some digital input lines, generally named as Control Lines. The CmDongle architecture supports following lines store in the CMBOXCONTROL structure:

- Up to 16 Indicator Control Lines are controlled (powered) by the chip. They may be set to 0 or 1, for example to supply a LED (Light Emitting Diode).
- Up to 16 Switcher Control Lines are read by the chip. Their read value may be 0 or 1, for example controlled by a switcher or button.

Syntax

```
typedef struct __CMBOXCONTROL {
     CMUSHORT musIndicatorFlags;
     CMUSHORT musSwitchFlags;
     CMULONG mulReserve;
} CMBOXCONTROL;
```

Parameter

musIndicatorFlags contains the current setting of the indicator control lines.

musSwitchFlags contains the current setting of the switcher control lines.

mulReserve is reserved for future purposes.

- *CM_GEI_BOXINFO* fills a CMBOXINFO structure specified in **pvDest**.

Structure CMBOXINFO

This structure contains information on a CmContainer.

Syntax

```
typedef struct __CMBOXINFO {
     CMBYTE mbMajorVersion;
     CMBYTE mbMinorVersion;
     CMUSHORT musBoxMask;
     CMULONG mulSerialNumber;
     CMUSHORT musBoxKeyId;
     CMUSHORT musUserKeyId;
     CMBYTE mabBoxPublicKey[CM_PUBLIC_KEY_LEN];
     CMBYTE mabSerialPublicKey[CM_PUBLIC_KEY_LEN];
     CMULONG mulReserve;
} CMBOXINFO;
```

Parameter

mbMajorVersion contains the major version of the CmContainer.

© Copyright 2018 WIBU-SYSTEMS AG

mbMinorVersion contains the minor version of the CmContainer.
musBoxMask contains the box mask of the CmContainer.
mulSerialNumber contains the serial number of the CmContainer.
musBoxKeyId contains the ID of the used BoxKey of the CmContainer.
musUserKeyId contains the ID of the used UserKey of the CmContainer.
mabBoxPublicKey[**CM_PUBLIC_KEY_LEN**] contains the public Box Key of the CmContainer.
mabSerialPublicKey[**CM_PUBLIC_KEY_LEN**] contains the public Serial Key of the CmContainer.
mulReserve is reserved for future purposes.

Notes

The length of the **mabBoxPublicKey** and **mabSerialPublicKey** is 64 byte long. The used public key are currently 224 bits = 28 bytes long. In Elliptic Curve Cryptography the public key is a 2 dimensional point, so the x-value is stored in the first 28 bytes (`0..27`) and the y-value is store in byte `32..59`.

CM_GEI_BOXTIME fills a CMBOXTIME structure specified in **pvDest**.

Structure CMBOXTIME

CMBOXTIME returns the three values of time clocks: Certified, Box and System Time.

Syntax

```
typedef struct __CMBOXTIME {
     CMTIME mcmCertifiedTime;
     CMTIME mcmBoxTime;
     CMTIME mcmSystemTime;
} CMBOXTIME;
```

Parameter

mcmCertifiedTime contains the Certified Time(see CMTIME structure).

Structure CMTIME

CMTIME sets or gets the time to or from a CmContainer.

Syntax

```
typedef struct __CMTIME {
     CMUSHORT musYear;
     CMUSHORT musMonth;
     CMUSHORT musDay;
     CMUSHORT musHours;
     CMUSHORT musMinutes;
     CMUSHORT musSeconds;
     CMULONG mulSecondsSince01_01_2000;
} CMTIME;
```

Parameter

musYear specifies the value of the year (e.g. `2003`).
musMonth specifies the value of the month (`1..12`).
musDay specifies the value of the day (`1..31`).
musHours specifies the value of the hours (`0..23`).
musMinutes specifies the value of the minutes (`0..59`).
musSeconds specifies the value of the seconds (`0..59`).
mulSecondsSince01_01_2000 specifies the time in seconds since `01/01/2000`.

mcmBoxTime contains the Box Time (see CMTIME).
mcmSystemTime contains the System Time (see CMTIME).

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_GEI_ENTRYINFO2* fills a CMBOXENTRY2 structure specified in **pvDest**.

Structure CMBOXENTRY2

CMBOXENTRY2 contains data of one entry. An entry is containing one Firm Item and one Product Item. In this structure all static data are stored. The dynamic data must read out of the CmContainer explicitly.

Syntax

```
typedef struct __CMBOXENTRY2 {
    CMULONG mflFiCtrl;
    CMULONG mulFirmCode;
    CMUSHORT musFirmAccessCounter;
    CMULONG mulFirmUpdateCounter;
    CMULONG mulFirmPreciseTime;
    CMUSHORT mausFirmItemText[CM_MAX_STRING_LEN];
    CMULONG mflSetNativePios;
    CMULONG mflSetEffectivePios;
    CMULONG mflSetPios;
    /* Product Item data */
    CMULONG mulProductCode;
    CMULONG mulFeatureMap;
    CMTIME mcmMaintenancePeriodStart;
    CMTIME mcmMaintenancePeriodEnd;
    CMULONG mulUnitCounter;
    CMTIME mcmExpirationTime;
    CMTIME mcmActivationTime;
    CMULONG mulUsagePeriodLifeTime;
    CMTIME mcmUsagePeriodStartTime;
    CMULONG mulLicenseQuantity;
    CMULONG mflLicenseQuantityFlags;
    CMULONG mulLicenseTag;
    CMULONG mflTransferStatus;
    CMBYTE mabProductItemId[10];
    CMBYTE mabReserve2[20];/* aligned to 796 */
    CMUSHORT musMaxEncryptionRate;
    CMBYTE mfbDependencyPC;
    CMBYTE mfbDependencyFM;
    CMBYTE mfbDependencyMP;
    CMBYTE mfbDependencyUC;
    CMBYTE mfbDependencyET;
    CMBYTE mfbDependencyAT;
    CMBYTE mfbDependencyUP;
    CMBYTE mfbDependencyLQ;
    CMBYTE mfbDependencyLT;
    CMBYTE mfbDependencyMV;
    CMUSHORT musModuleRef;
    CMUSHORT musProductItemRef;
    CMUSHORT musTransferBoxMask;
    CMULONG mulTransferSerialNumber;
    CMTIME mcmTransferBorrowExpirationTime;
    CMBYTE mabTransferId[10];
} CMBOXENTRY2;
```

Parameter

mflFiCtrl specifies the type of the Firm Item. The following flags can be set:

- *CM_GF_BOXBASED* specifies that the Firm Item was programmed in the Box-Based-Mode.
- *CM_GF_FILEBASED* specifies that the Firm Item was programmed in the File-Based-Mode.
- *CM_GF_TRANSFERBASED* specifies that the Firm Item has been created based on a license transfer.

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_GF_FI_ALLOWED_TO_PULL* specifies the license transfer can be actively performed.
- *CM_GF_FI_RESERVED* specifies that the Firm Item is a Reserved Firm Item.

mulFirmCode contains the Firm Code of the Firm Item.

musFirmAccessCounter contains the Firm Access Counter of the Firm Item.

mulFirmUpdateCounter contains the Firm Update Counter of the Firm Item.

mulFirmPreciseTime contains the FirmPreciseTime of the Firm Item.

mausFirmItemText [*CM_MAX_STRING_LEN*] contains the Firm Item Text of the Firm Item.

mflSetNativePios specifies in the context of used Module Items the native representation of the PIOs as programmed.

mflSetEffectivePios specifies in the context of used Module Items the effective representation, i.e including eventually inherited PIOs.

mflSetPios specifies all available PIO's at this Product Item. The following flags can be set:

- *CM_BE_PRODUCTCODE* specifies that the Product Item contains a Product Code.
- *CM_BE_FEATUREMAP* specifies that the Product Item contains a Feature Map.
- *CM_BE_EXPTIME* specifies that the Product Item contains a Expiration Time.
- *CM_BE_ACTTIME* specifies that the Product Item contains a Activation Time.
- *CM_BE_UNITCOUNTER* specifies that the Product Item contains a Unit Counter.
- *CM_BE_PROTDATA* specifies that the Product Item contains a Protected Data.
- *CM_BE_EXTPROTDATA* specifies that the Product Item contains one or more Extended Protected Data.
- *CM_BE_HIDDENDATA* specifies that the Product Item contains one or more Hidden Data.
- *CM_BE_SECRETDATA* specifies that the Product Item contains one or more Secret Data.
- *CM_BE_USERDATA* specifies that the Product Item contains a User Data.
- *CM_BE_TEXT* specifies that the Product Item contains a Text.
- *CM_BE_USAGEPERIOD* specifies that the Product Item contains a Usage Period.
- *CM_BE_LICENSEQUANTITY* specifies that the Product Item contains a License Quantity.

  Please note that borrowing in this context does not involve a license transfer. Rather it refers to borrowing using Firm Codes smaller than 6,000,000 and especially prepared CmContainer.

- *CM_BE_BORROWCLIENT* specifies that the Product Item contains a Borrow Client entry.
- *CM_BE_BORROWSERVER* specifies that the Product Item contains a Borrow Server entry.
- *CM_BE_COLI* specifies that the Product Item contains a Customer Owned License Information (COLI).
- *CM_BE_MAINTENANCEPERIOD* specifies that the Product Item contains a Maintenance Period entry.
- *CM_BE_LINGERTIME* specifies that the Product Item contains a Linger Time entry.
- *CM_BE_MINIMUMVERSION* specifies that the Product Item contains a Minimal Runtime Version entry.
- *CM_BE_MODULE_ITEM* specifies that the Product Item contains a Module Item, if available.
- *CM_BE_BORROWEXPIRATIONTIME* specifies that the Product Item contains a borrowing entry based on a license transfer, if available.
- *CM_BE_HISTORY* specifies that the Product Item is part of a license transfer process, if available.
- *CM_BE_ACCESS_PASSWORD* specifies that the Product Item contains a password, if available.
- *CM_BE_TRANSFEROPTIONS* specifies that the Product Item contains license transfer options, if available.
- *CM_BE_RETURNINFO* specifies that the Product Item contains license transfer return information, if available.
- *CM_BE_ACCEPTSPULL* specifies that the Product Item contains the information that a license can be pulled, if available.
- *CM_BE_NAMEDUSER* specifies that the Product Item contains a Named User entry, if available.
- *CM_BE_MAX_ENCRYPTION_RATE* specifies that the Product Item contains a Maximum Encryption Rate information.

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_BE_PIO_MASK* specifies all flags for the Product Item Options.

mulProductCode contains the Product Code of the Product Item.
mulFeatureMap contains the Feature Map of the Product Item.
mcmMaintenancePeriodStart contains the start date of the Maintenance Period of the Product Item, if available.
mcmMaintenancePeriodEnd contains the end date of the Maintenance Period of the Product Item, if available.
mulUnitCounter contains the Unit Counter of the Product Item, if available.
mcmExpirationTime contains the Expiration Time of the Product Item, if available (see CMTIME structure).

Structure CMTIME

CMTIME sets or gets the time to or from a CmContainer.

Syntax

```
typedef struct __CMTIME {
    CMUSHORT musYear;
    CMUSHORT musMonth;
    CMUSHORT musDay;
    CMUSHORT musHours;
    CMUSHORT musMinutes;
    CMUSHORT musSeconds;
    CMULONG mulSecondsSince01_01_2000;
} CMTIME;
```

Parameter

musYear specifies the value of the year (e.g. `2003`).
musMonth specifies the value of the month (`1..12`).
musDay specifies the value of the day (`1..31`).
musHours specifies the value of the hours (`0..23`).
musMinutes specifies the value of the minutes (`0..59`).
musSeconds specifies the value of the seconds (`0..59`).
mulSecondsSince01_01_2000 specifies the time in seconds since `01/01/2000`.

mcmActivationTime contains the Activation Time of the Product Item, if available (see CMTIME).
mulUsagePeriodLifeTime contains the period length of the Usage Period in seconds, if available.
mcmUsagePeriodStartTime contains the start time of the Usage Period, if available (see CMTIME).
mulLicenseQuantity contains the License Quantity of the Product Item, if available.
mflLicenseQuantityFlags contains the flags for license access options, i.e. local accesses.

- *CM_LQ_DENIEDUSERLIMIT*
- *CM_LQ_DENIEDNOUSERLIMIT*
- *CM_LQ_DENIEDEXCLUSIVE*
- *CM_LQ_DENIEDSTATIONSHARE*
- *CM_LQ_ALLOWCMWAN*
- *CM_LQ_TRIPLEMODEREDUNDANCY*
- *CM_LQ_LOCALACCESSONLY*
- *CM_LQ_DENIEDALL*
- *CM_LQ_MAXVALUE*

mulLicenseTag contains information in the context of Triple Mode Redundancy feature if available.
mflTransferStatus contains the flag of the transfer status. The following flags exist:

- *CM_TRANSFER_STATUS_ENABLED 1*
- *CM_TRANSFER_STATUS_DISABLED 2*
- *CM_TRANSFER_STATUS_RETURNED 3*

© Copyright 2018 WIBU-SYSTEMS AG

CM_TRANSFER_STATUS_EXPIRED 4

mabProductItemId contains an unique identifier for the Product Item.

musMaxEncryptionRate contains the Maximum Encryption Rate.

mfbDependencyPC contains the dependency flags of the Product Code.

mfbDependencyFM contains the dependency flags of the Feature Map.

mfbDependencyUC contains the dependency flags of the Unit Counter.

mfbDependencyET contains the dependency flags of the Expiration Time.

mfbDependencyAT contains the dependency flags of the Activation Time.

mfbDependencyUP contains the dependency flags of the Usage Period.

mfbDependencyLQ contains the dependency flags of the License Quantity.

mfbDependencyLT contains the dependency flags of the Linger Time.

mfbDependencyMV contains the dependency flags of the Minimum Runtime Version.

musModuleRef contains the Product Item Reference of the super-ordinated Product Item of a Module Item. All items of a module then have the same **ModulRef** if available.

musProductItemRef holds the Product Item Reference.

musTransferBoxMask holds the box mask of the source in the case of a transferred license if available.

mulTransferSerialNumber holds the serial number of the source in the case of a transferred license if available.

mcmTransferBorrowExpirationTime holds the time at which a borrowed license (license transfer) expires if available.

mabTransferId[10] holds the Id of the transfer (corresponds to the Id in the history item of the sender) if available.

Notes

The dynamic data can be read out with *GetInfo()* / *CM_GEI_ENTRYDATA*.

The dependency parameters (**mfbDependencyXX**) can be have one or more of the following flags:

- *CM_GF_DATA* specifies that the parameter depends on the contents of the data of the Firm Item.
- *CM_GF_FUC* specifies that the parameter depends on the Firm Update Counter of the Firm Item.
- *CM_GF_SERIAL* specifies that the parameter depends on the Serial Number of the CmContainer.

On Windows the **mausFirmItemText** array can be simply converted via *CmConvertString()* into the specified character set.

CMBOXENTRY2 is supported since CodeMeter Version 4.30.

The flags *CM_GF_TRANSFERBASED* and *CM_GF_FI_ALLOWED_TO_PULL* of **mflFiCtrl** are available since CodeMeter Version 6.00.

The members **musModuleRef**, **musTransferBoxMask**, **mulTransferSerialNumber**, **mcmTransferBorrowExpirationTime**, **mabTransferId**, **mflLicenseQuantityFlags** are available since CodeMeter Version 6.00.

The member **musMaxEncryptionRate** is available since CodeMeter Version 6.50.

For **mflTransferStatus** the flags *CM_TRANSFER_STATUS_ENABLED*, *CM_TRANSFER_STATUS_DISABLED*, *CM_TRANSFER_STATUS_RETURNED*, *CM_TRANSFER_STATUS_EXPIRED*: are available since CodeMeter Version 6.30.

For **mflSetPio** the flag *CM_BE_NAMEDUSER* is available since CodeMeter Version 6.30.

For **mflSetPio** the flags *CM_BE_ACCESS_PASSWORD* and *CM_BE_MAX_ENCRYPTION_RATE* is available since CodeMeter Version 6.50.

CM_GEI_ENTRYDATA fills a CMENTRYDATA structure specified in **pvDest**. If more than one item is available, an array with matching or larger size must be passed.

Structure CMENTRYDATA

CMENTRYDATA contains a dynamic added data of a Product Item with all its properties.

Syntax

```
typedef struct __CMENTRYDATA {
```

© Copyright 2018 WIBU-SYSTEMS AG

```
    CMULONG mflCtrl;
    CMULONG mflDependency;
    CMUINT mcbData;
    CMBYTE mabData[2 * CM_MAX_STRING_LEN];
} CMENTRYDATA;
```

Parameter

mflCtrl specifies the type of the dynamic added entry. It can contain one of the following flags:

- *CM_GF_USERDATA* specifies that the Product Item contains a User Data.
- *CM_GF_TEXT* specifies that the Product Item contains a Text.
- *CM_GF_PROTDATA* specifies that the Product Item contains a Protected Data.
- *CM_GF_EXTPROTDATA* specifies that the Product Item contains one or more Extended Protected Data.
- *CM_GF_HIDDENDATA* specifies that the Product Item contains one or more HiddenData Data.
- *CM_GF_SECRETDATA* specifies that the Product Item contains one or more Secret Data.
- *CM_GF_USAGEPERIOD* specifies that the Product Item contains a Usage Period.

For the three entry types Extended Protected Data, Hidden Data and Secret Data the index of this data type is or'ed in the upper two bytes. See remarks for a sample.

mflDependency contains the dependency of the dynamic added entry. It can be have one or more of the following flags:

- *CM_GF_DATA* specifies that the parameter depends on the contents of the data of the Firm Item.
- *CM_GF_FUC* specifies that the parameter depends on the Firm Update Counter of the Firm Item.
- *CM_GF_SERIAL* specifies that the parameter depends on the Serial Number of the CmContainer.

mulReserve is reserved for future purposes.

mcbData contains the length of mabData, for text entries (*CM_GF_TEXT*) the number of characters, for all other types the number of stored bytes.

mabData[2 * CM_MAX_STRING_LEN] contains the data of the dynamic added entry with the length stored in **mcbData**. For text entries (*CM_GF_TEXT*) the byte array contains an UCS2-LE encoded null-terminated string.

Notes

To check the **mflCtrl** you can use the *CM_GF_PIO_MASK* to retrieve only the data type flag, e.g. `mflCtrl & CM_GF_PIO_MASK`. To retrieve the index of Extended Protected Data, Hidden Data or Secret Data you need to extract the upper two bytes of **mflCtrl**, e.g. by **mflCtrl >> 16**.

For an easier conversion of mabData it may be helpful to use *CmConvertString()*.

- *CM_GEI_INTERNALENTRYINFO* fills as many ⊟ CMINTERNALENTRYINFO structures as specified in **pvDest** that are necessary to store the information of all Product Items of same Product Code of a Firm Code.

Structure CMINTERNALENTRYINFO

CMINTERNALENTRYINFO stores internal data of an entry.

Syntax

```
typedef struct __CMINTERNALENTRYINFO {
    CMUSHORT musFirmItemReference;
    CMUSHORT musProductItemReference;
    CMULONG mulFirmItemType;
    CMULONG mulFirmUpdateCounter;
} CMINTERNALENTRYINFO;
```

Parameter

musFirmItemReference contains the Firm Item Reference in the CmContainer of the Firm Item.

© Copyright 2018 WIBU-SYSTEMS AG

musProductItemReference contains the Product Item Reference in the CmContainer of the Product Item.

mulFirmItemType specifies the type of the Firm Item. The following flags can be set:

- *CM_GF_BOXBASED* specifies that the Firm Item was programmed in the box based mode.
- *CM_GF_FILEBASED* specifies that the Firm Item was programmed in the file based mode.
- *CM_GF_TRANSFERBASED* specifies that the Firm Item has been created based on a license transfer.
- *CM_GF_FI_RESERVED* specifies that the Firm Item is a Reserved Firm Item.

mulFirmUpdateCounter contains the Firm Update Counter of the Firm Item.

mulReserve is reserved for future purposes.

- *CM_GEI_SYSTEMSTATUS* fills a CMSYSTEMSTATUS structure specified in **pvDest**.

Structure CMSYSTEMSTATUS

CMSYSTEMSTATUS stores system specific data.

Syntax

```
typedef struct __CMSYSTEMSTATUS {
  CMULONG mflStatusFlags;
  CMULONG mulServerStartTimeInSecondsSince01_01_2000;
} CMSYSTEMSTATUS;
```

Parameter

mflStatusFlags may contain the following bits set:

- *CM_SYSTEMSTATUS_RUNNING_AS_SERVICE* (0x00000001) specifies that CodeMeter runs as Windows service or, in the case of other operating systems, with respective privileges.
- *CM_SYSTEMSTATUS_CMACT_ENABLED* (0x00000002) specifies CmActLicense licenses are activated in the profiling and can be used.
- *CM_SYSTEMSTATUS_MSD_ENABLED* (0x00000004) specifies that CmDongles are activated in the profiling as Mass Storage Devices (MSD) and can be used.
- *CM_SYSTEMSTATUS_HID_ENABLED* (0x00000008) specifies that CmDongles are activated in the profiling as Human Interface Devices (HID) and can be used.
- *CM_SYSTEMSTATUS_CMACT_AVAILABLE* (0x00000010) specifies that CmActLicense can be used (it is activated and CodeMeter runs as service).

mulServerStartTimeInSecondsSince01_01_2000 contains the start time of CodeMeter server in seconds since 01.01.2000, 00:00 (UTC).

- *CM_GEI_SIGNEDLIST* fills byte buffer with a signed list of an entry passed in **pvDest**.
- *CM_GEI_VERSION* fills a CMVERSION structure specified in **pvDest**.

Structure CMVERSION

CMVERSION stores the complete version of a CodeMeter module.

Syntax

```
typedef struct __CMVERSION {
     CMUSHORT musVersion;
     CMUSHORT musSubVersion;
     CMUSHORT musBuild;
     CMUSHORT musCount;
     CMUSHORT musYear;
     CMUSHORT musMonth;
     CMUSHORT musDay;
} CMVERSION;
```

Parameter

musVersion contains the major version of the CodeMeter module.

© Copyright 2018 WIBU-SYSTEMS AG

musSubVersion contains the minor version of the CodeMeter module.
musBuild contains the build number of the CodeMeter module.
musCount contains the count of the CodeMeter module.
musYear contains the year of creation of the CodeMeter module.
musMonth contains the month of creation of the CodeMeter module.
musDay contains the day of creation of the CodeMeter module.

CM_GEI_CHIPINFO fills a CMCHIPINFO structure specified in **pvDest**.

Structure CMCHIPINFO

CMCHIPINFO stores the information about the Chip of the CmDongle.

Syntax

```
typedef struct __CMCHIPINFO {
     CMULONG mulFirmwareBuild;
     CMULONG mulFirmwareBuildTop;
     CMULONG mulMdfaLba;
     CMULONG mulReserved;
     CMUSHORT musChipType;
     CMUSHORT musSiliconRevision;
     CMUSHORT midFactory;
     CMUSHORT musProductYear;
     CMUSHORT musLotNumber;
     CMUSHORT midWafer;
     CMUSHORT midChipOnWafer;
     CMUSHORT musDowngradeCount;
} CMCHIPINFO;
```

Parameter

mulFirmwareBuild contains the build number of the currently stored Firmware version.
mulFirmwareBuildTop contains the highest build number of the Firmware version ever stored.
mulMdfaLba contains the start address of the File I/O communication.
mulReserved is reserved for future purposes.
musChipType contains the type of the chip.
musSiliconRevision contains the revision of the chip mask.
midFactory contains the identification of the chip fabric where it was manufactured.
musProductYear contains the production year in number of years since 2000.
musLotNumber contains the charge number of the produced wafer.
midWafer contains the identification of the wafer within the charge.
midChipOnWafer contains the identification of the chip within the wafer.
musDowngradeCount

Structure CMUSBCHIPINFO

CMUSBCHIPINFO stores the information about the USB chip of the CmDongle.

Syntax

```
typedef struct __CMUSBCHIPINFO {
     CMUSHORT musFirmwareMajor;
     CMUSHORT musFirmwareMinor;
     CMULONG  mulFlashSize;
     char mszNodeDescription[CM_MAX_STRING_LEN];
     CMBYTE mszNodes[CM_MAX_STRING_LEN];
} CMUSBCHIPINFO;
```

Parameter

© Copyright 2018 WIBU-SYSTEMS AG

musFirmWareMajor contains the major version of the USB chip Firmware.
musFirmWareMinor contains the minor version of the USB chip Firmware.
mulFlashSize contains the size of the flash memory part of the CmDongle.
mszNodeDescription contains the description of the current USB node of the connected CmDongle.
mszNodes contains a list of found CmDongle partitions. This feature requires a CodeMeter USB chip Firmware version >= 7.3.

- *CM_GEI_BOXSTATUS* fills a 32 bit value with following status flags.
  - *CM_BOXSTATUS_LOWMEMORY* specifies that the CmDongle has too low memory to replace damaged memory during normal usage.
  - *CM_BOXSTATUS_REPLUG* specifies that the CmDongle has initiated a Plug Out/In event for itself to force the operating system to remount the flash drive.
  - *CM_BOXSTATUS_HASFLASH* specifies that the CmDongle contains an additional flash disc besides the CodeMeter chip.
  - *CM_BOXSTATUS_ISCMACT* specifies that this is a CmActLicense.
  - *CM_BOXSTATUS_REMOVABLE* specifies the configuration of the CmDongle. This can be either be a fixed disk device or a removable disk device. The configuration of the CmDongle can be altered with *CmConfigDisk* on Windows.
  - *CM_BOXSTATUS_BATTERY_POWERED_CLOCK_USED* specifies that the hardware (CmStick/T) has a working battery powered clock inside.
  - *CM_BOXSTATUS_USES_HID_COMMUNICATION* specifies that the CmDongle is configured as Human Interface Device (HID).
  - *CM_BOXSTATUS_STATE_FUM* specifies that the CmDongle is in Field Update Mode (FUM). In this intermediate state an update of the Firmware version needs to be executed.
  - *CM_BOXSTATUS_STATE_LOCKED* specifies that the CmContainer is locked. This cannot be repaired. This CmContainer needs to be replaced.

  Enabling status:
  - *CM_BOXSTATUS_ENABLED* specifies that the CmContainer is enabled.
  - *CM_BOXSTATUS_DISABLED* specifies that the CmContainer is disabled.
  - *CM_BOXSTATUS_TEMPENABLED* specifies that the CmContainer is temporarily enabled.
  - *CM_BOXSTATUS_MASK* specifies all flags for the enabling status.

  CmActLicense status:
  - *CM_ACTSTATUS_FILE* specifies that a file is the base of this CmActLicense.
  - *CM_ACTSTATUS_LOADED* specifies that the CmActLicense is loaded and can be used. Used during the license creation on licensors side.
  - *CM_ACTSTATUS_ACTIVATE_BY_CODE* specifies that the CmActLicense can be activated by entering a code (phone activation).
  - *CM_ACTSTATUS_ACTIVATE_BY_FILE* specifies that the CmActLicense has to be activated by a file transmission of a Remote Update File.
  - *CM_ACTSTATUS_ACTIVE* specifies that the CmActLicense currently is valid and active.
  - *CM_ACTSTATUS_PROGRAMMABLE* specifies that this license can be programmed now. Used during the license creation on licensors side.
  - *CM_ACTSTATUS_INVALID* specifies that there occurred a problem with the security checks of the CmActLicense. The license cannot be used.
  - *CM_ACTSTATUS_BROKEN* specifies that the binding requirements are no longer fulfilled. The license cannot be used.
  - *CM_ACTSTATUS_PSN_REQUIRED* specifies that the licensee has to set a Product Serial Number (PSN) at the time of creating a remote context file.
  - *CM_ACTSTATUS_VM_DETECTED* specifies that the license is invalid since it is running in a virtual environment which is not permitted.
  - *CM_ACTSTATUS_BLACKLISTED* specifies that the CmActLicense is blacklisted by CodeMeter Timeserver and can no longer be used.

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_BOXSTATUS_CMACT_MASK* specifies all flags for CmActLicense.
- *CM_GEI_LISTALLPI* serves to faster return an array of Product Items and Module Items contents using internal lists and fills a CMPIENTRYBASE structure specified in **pvDest**.

Structure CMPIENTRYBASE

CMPIENTRYBASE returns all Product Items and Module Items of the Firm Item regardless whether the handle is a Firm Item or Product Item handle.

Syntax

```
typedef struct __CMPIENTRYBASE {
  CMULONG mulProductCode;
  CMULONG mulFeatureMap;
  CMULONG mflSetEffectivePios;
  CMULONG mflSetNativePios;
  CMULONG mulMaintenancePeriodStartSecondsSince01_01_2000;
  CMULONG mulMaintenancePeriodEndSecondsSince01_01_2000;
  CMULONG mulUnitCounter;
  CMULONG mulExpirationTimeSecondsSince01_01_2000;
  CMULONG mulActivationTimeSecondsSince01_01_2000;
  CMULONG mulUsagePeriodLifeTime;
  CMULONG mulUsagePeriodStartTimeSecondsSince01_01_2000;
  CMULONG mulLicenseQuantity;
  CMULONG mulMinimumRuntimeVersion;
  CMULONG mflLicenseQuantityFlags;
  CMNAMEDUSER mcmNamedUserEntry;
  CMUSHORT musProductItemRef;
  CMUSHORT musReserved1;
  CMULONG mflTransferStatus;
} CMPIENTRYBASE;
```

Parameter

mulProductCode contains the Product Code of the Product Item.

mulFeatureMap contains the Feature Map of the Product Item.

mflSetEffectivePios specifies in the context of used Module Items the effective representation, i.e including eventually inherited PIOs.

mflSetNativePios specifies in the context of used Module Items the native representation of the PIOs as programmed.

mulMaintenancePeriodStartSecondsSince01_01_2000 contains the start date of the Maintenance Period of the Product Item in seconds since 01_01_2000, if available.

mulMaintenancePeriodEndSecondsSince01_01_2000 contains the end date of the Maintenance Period of the Product Item in seconds since 01_01_2000, if available.

mulUnitCounter contains the Unit Counter of the Product Item, if available.

mulExpirationTimeSecondsSince01_01_2000 contains the start date of the Expiration Time of the Product Item in seconds since 01_01_2000, if available.

mulActivationTimeSecondsSince01_01_2000 contains the end date of the Expiration Time of the Product Item in seconds since 01_01_2000, if available

mulUsagePeriodLifeTime contains the period length of the Usage Period in seconds, if available.

mulUsagePeriodStartTimeSecondsSince01_01_2000 contains the start date of the Usage Period of the Product Item in seconds since 01_01_2000, if available.

mulLicenseQuantity contains the License Quantity of the Product Item, if available.

mulMinimumRuntimeVersion contains the Minimum Runtime Version of the Product Item, if available.

mflLicenseQuantityFlags contains the flags for license access options, i.e. local accesses.

- *CM_LQ_DENIEDUSERLIMIT*
- *CM_LQ_DENIEDNOUSERLIMIT*
- *CM_LQ_DENIEDEXCLUSIVE*
- *CM_LQ_DENIEDSTATIONSHARE*

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_LQ_ALLOWCMWAN*
- *CM_LQ_TRIPLEMODEREDUNDANCY*
- *CM_LQ_LOCALACCESSONLY*
- *CM_LQ_DENIEDALL*
- *CM_LQ_MAXVALUE*

mcmNamedUserEntry holds the data of the respective a CMNAMEDUSER structure.

Structure CMNAMEDUSER

CMNAMEDUSER stores the information about the PIO Named User. This PIO is used during *CmAccess2()* to verify the hashed `CMACCESS2::mcmCredential.msz***` parameters Username, DomainName and UserDefinedText supplied in CmAccess2 against the hash of the credentials stored here.

Syntax

```
typedef struct __CMNAMEDUSER
{
  CMUSHORT musPioType;
  CMUSHORT musIndex;
  CMULONG mflDependency;
  CMULONG mulReserved;
  CMULONG mcbEntryData;
  CMBYTE mabTag[4];
  CMBYTE mbVersion;
  CMBYTE mbNextIndex;
  CMBYTE mbfModeFlags;
  CMBYTE mbReserved2;
  CMULONG mulCrc32;
  CMBYTE mabHash[32];
  CMBYTE mabPlaintext[128];
} CMNAMEDUSER;
```

Parameter

musPioType and **mflDependency** refer to the same-named parameter of the CMENTRYDATA structure. For internal use only.

musIndex identifies the named user entry.

mulReserved is reserved for future purposes.

mcbEntryData contains the length of the following data.

mabTag[4] contains always the tag "cmnu" (CodeMeter Named User) to uniquely identify the PioType Extended Protected Data.

mbVersion refers to the version number of cmnu; currently set to version 1.

mbNextIndex currently holds a value of 0.
In future, if there should be more than a single Named User License for each license (Firm Code: Product Code), contains the **musIndex** of the next Named User license entry. Valid values will be 128-255.

mbfModeFlags contains one of the following flags:

- *CM_NMU_MODE_USERNAME* `0x01` (default) checks the login name only
- *CM_NMU_MODE_USERDOMAIN* `0x02` checks the login name and the domain name
- *CM_NMU_MODE_USERDEFINED* `0x03` checks the user defined text
- *CM_NMU_FLAG_PLAINTEXT* `0x10` specifies that the credentials used with the selected mode are additionally stored in **mabPlaintext**..

mbReserved2 is reserved for future purposes.

mulCrc32 holds the internal value of the CRC32 encryption of **mabTag** up to (incl.) **mbReserved2**.

mabHash[32] holds the hash value generated during license creation.

mabPlaintext[128] holds, if flag *CM_NMU_FLAG_PLAINTEXT* has been set, the credentials used with the selected mode. The format is UTF-8.
If flag *CM_NMU_FLAG_PLAINTEXT* is not set, this field is empty.

© Copyright 2018 WIBU-SYSTEMS AG

For *CM_NMU_MODE_USERDOMAIN* the information is saved according to the pattern "domain name/ login name", e.g. "ABC Company/John Doe".

Notes

If no entries exist, the error CMERROR_NO_DATA_AVAILABLE (209) is returned.
If the buffer is too small, the error CMERROR_BUFFER_OVERFLOW (112) is returned.
If the handle is no entry handle, the error CMERROR_WRONG_HANDLE_TYPE (115) is returned.

musProductItemRef holds the Product Item Reference.
musReserved1 is reserved for future purposes.
mflTransferStatus contains the flag of the transfer status. The following flags exist:.

- *CM_TRANSFER_STATUS_ENABLED*
- *CM_TRANSFER_STATUS_DISABLED*
- *CM_TRANSFER_STATUS_RETURNED*
- *CM_TRANSFER_STATUS_EXPIRED*

pvDest contains pointer to a structure dependant to the flag set in **flCtrl**.
cbDest contains the length of **pvDest** in bytes.

Return

The return value contains the real number of bytes returned in **pvDest**. If an error occurred, the return value is 0. If **pvDest** does not contain enough memory, the number of needed bytes will be returned. In this case, the return value is larger than the **cbDest** parameter and error code CMERROR_BUFFER_OVERFLOW is set.

Notes

This function does not really scan all available or desired ports for connected CmContainers because of CodeMeter License Servers Plug&Play mechanism and the implementation of storing all CmContainers and their content in internal lists. The function call checks the internal lists only and returns the matching data.

Since CodeMeter 3.30 `CMNETINFOCLUSTER` and `CMNETINFOUSER` is also filled when CodeMeter is not running as a network server. In older versions the error CMERROR_NO_NETWORK_SERVER is set, if CodeMeter License Server is not running as a server.

The following flags (and the corresponding functionality) are only available since CodeMeter Version 4.30:

-
- *CM_BOXSTATUS_BATTERY_POWERED_CLOCK_USED*

The following flag (and the corresponding functionality) is only available since CodeMeter Version 5.00:

- *CM_BOXSTATUS_USES_HID_COMMUNICATION*

The following flags (and the corresponding functionality) are only available since CodeMeter Version 6.00:

- *CM_GEI_LTHISTORY*
- *CM_GEI_LTTRANSFEROPTIONS*

The following flag (and the corresponding functionality) is only available since CodeMeter Version 6.30:

- *CM_GEI_NAMEDUSER*
- *CM_GEI_LISTALLPI* and `CMPIENTRYBASE` structure

3.5.4 CmGetVersion

Description

CmGetVersion returns the major and minor version and Build of the requested CodeMeter modules. The itemized values are the Version and Subversion. To extract the version macros can be used.

© Copyright 2018 WIBU-SYSTEMS AG

C, C++ Programming Language

Syntax

```
unsigned long CMAPIENTRY CmGetVersion(HCMSysEntry hcmse)
```

Parameter

hcmse contains a handle opened by CmAccess2 (can be Box Access or Entry Access). NULL is also a possible value.

Return

- **hcmse == NULL**: The version of the local calling driver (on Windows: `WibuCm32.dll`) is returned.
- **hcmse == "Handle to an entry in the network"**: The version of the CodeMeter License Server running on the machine the handle is accessing.

The return value of the version has the format:

The itemized values are Version and Subversion. To extract the version macros for Version and Subversion can be used.

Notes

In CodeMeter Versions prior to 3.30 the return value also contained Build and Count. The itemized values in this older versions were Version, Subversion, Build and Count.

3.6 Programming operations

3.6.1 CmActLicenseControl

Description

CmActLicenseControl is a helper function that provides all activation essentials and handles CmActLicense files. Depending on the used flag a buffer or structure is used to describe the request data. If the command needs to return some data, it fills a given structure.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmActLicenseControl(HCMSysEntry hcmse, CMULONG flCtrl, const
void *pvData, CMUINT cbData, void *pvReturn, CMUINT cbReturn)
```

Parameter

hcmse contains a handle opened by CmAccess2. The required access mode depends on the operation set in **flCtrl**.

flCtrl contains one of the following flags to specify the action that is requested:

- *CM_GF_ACT_REQUEST* creates a remote Context File (`*.WibuCmRaC`) to request the activation of a CmActLicense. **pvData** holds a CMACTREQUEST structure. **pvReturn** is filled with a CMBYTE array containing the remote Context File (`*.WibuCmRaC`). **hcmse** must be Box Access, Firm Item Access or Entry Access.

Structure CMACTREQUEST

CMACTREQUEST holds information for creating a CmActLicense Context File.

Syntax

```
typedef struct __CMACTREQUEST {
    char mszProductSerialNumber[CMACT_MAX_LEN_VENDORTEXT];
    CMULONG mulProductSubType;
    CMULONG mulReserved1;
    CMBYTE mabLicensorData[CMACT_MAX_LEN_LICENSORDATA];
} CMACTREQUEST;
```

© Copyright 2018 WIBU-SYSTEMS AG

Parameter

tba.

Return

This function returns the number of bytes filled in **pvReturn**. If the command specified in **flCtrl** does not fill **pvReturn** by design, the function returns 1. If an error occurred, `0` will be returned.

Exception:

If **pvReturn** does not contain enough memory, the number of required bytes will be returned in **cbReturn** and error code CMERROR_BUFFER_OVERFLOW is set.

3.6.2 CmProgram

Description

CmProgram programs entries.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmProgram(HCMSysEntry hcmse, CMULONG flCtrl, void *pvCtrl,
CMUINT cbCtrl, void *pvVerify, CMUINT cbVerify)
```

Parameter

hcmse contains a handle opened by CmAccess2. The access mode can be *Box Access*, Firm Item Access or Entry Access. The required access mode depends on the operation set in **flCtrl**. The handle is connected to a CmLocal subsystem with a connected CmDongle. This CmDongle will be used for programming.

flCtrl contains one of the following commands:

- *CM_GF_ADD_FIRMITEM* adds a Firm Item into a CmDongle. **pvCtrl** is pointing to a CMPROGRAM_ADD_FIRMITEM structure. **hcmse** must be Box Access.

Structure CMPROGRAM_ADD_FIRMITEM

CMPROGRAM_ADD_FIRMITEM contains all needed information to add a Firm Item to a CmDongle. The following parameters of the structure are encrypted: musFirm ItemTypeEncrypted to mabSessionId (inclusive). This member defines a sequence of 64 bytes which are encrypted by AES-CBC. The key and the CBC initialization vector depends on the addressed Firm Code and of parameters of the addressed CmDongle.

Syntax

```
typedef struct __CMPROGRAM_ADD_FIRMITEM {
    /* plain members */
    CMUSHORT musFirmItemReference;
    CMUSHORT musFirmItemTypePlain;
    CMBYTE mabPublicLicensorKey[CM_PUBLIC_KEY_LEN];
    /* encrypted members */
    CMUSHORT musFirmItemTypeEncrypted;
    CMUSHORT musFirmAccessCounter;
    CMULONG mulFirmUpdateCounter;
    CMULONG mulFirmPreciseTime;
    CMULONG mulReserved;
    CMBYTE mabFirmKey[CM_BLOCKCIPHER_KEY_LEN];
    CMBYTE mabSessionId[2 * CM_SESSIONID_LEN];
    /* plain members */
    CMUSHORT mcchText;
    /* variable length, max = 256 */
    CMUSHORT mausText[CM_MAX_STRING_LEN];
} CMPROGRAM_ADD_FIRMITEM;
```


© Copyright 2018 WIBU-SYSTEMS AG

Parameter

musFirmItemReference contains the reserved Firm Item Reference received with *CmReserveFirmItem()*.

musFirmItemTypePlain specifies the type of the Firm Item. The following flags can be set:

- *CM_GF_BOXBASED* specifies that the Firm Item is programmed in the Box-Based-Mode.
- *CM_GF_FILEBASED* specifies that the Firm Item is programmed in the File-Based-Mode.
- *CM_GF_TRANSFERBASED* specifies that the Firm Item has been created based on a license transfer.

mabPublicLicensorKey contains 64 bytes of ECC Licensor Key which is required to decrypt the specified data via the private Box Key, Serial Key, Firm Common Key and Serial Number of the addressed CmDongle.

musFirmItemTypeEncrypted (encrypted) specifies the type of the Firm Item. The following flags can be set:

- *CM_GF_BOXBASED* specifies that the Firm Item was programmed in the Box-Based-Mode.
- *CM_GF_FILEBASED* specifies that the Firm Item was programmed in the File-Based-Mode.
- *CM_GF_TRANSFERBASED* specifies that the Firm Item has been created based on a license transfer.

This value must be equal to **musFirmItemTypePlain**!

musFirmAccessCounter (encrypted) contains the initial value of the Firm Access Counter. If this member is set to `0xFFFF` (the recommended value), this counter is not decremented by an encryption which targets this counter.

mulFirmUpdateCounter (encrypted) contains the initial value of the Firm Update Counter. The recommended value is 0.

mulFirmPreciseTime (encrypted) contains the new Firm Precise Time in seconds since 1st January 2000, 0:00:00. If the value is not used, 0 must be specified, which sets this field to the Running Box Clock value.

mulReserved (encrypted) should be set to 0. Reserved for future purposes.

mabFirmKey (encrypted) contains the secret Firm Key, which has a fixed length of 32 bytes.

mabSessionId (encrypted) contains the Session ID received with *CmReserveFirmItem()*.

mcchText contains the initial length of the Firm Item Text of the Firm Item to program.

mausText contains the initial Firm Item Text of the Firm Item to program.

- *CM_GF_UPDATE_FIRMITEM* updates a Firm Item in a CmDongle. **pvCtrl** is pointing to a CMPROGRAM_UPDATE_FIRMITEM structure. **hcmse** must be Firm Item Access or Entry Access.

Structure CMPROGRAM_UPDATE_FIRMITEM

CMPROGRAM_UPDATE_FIRMITEM contains all needed parameters to update a Firm Item.

Syntax

```
typedef struct __CMPROGRAM_UPDATE_FIRMITEM {
    CMUSHORT musFirmItemReference;
    CMUSHORT musCtrl;
    CMUSHORT musFirmAccessCounter;
    CMULONG mulFirmUpdateCounter;
    CMULONG mulFirmPreciseTime;
    CMUSHORT mcchText;
    /* variable length, max = 256, expand structure on the stack */
    CMUSHORT mausText[1];
} CMPROGRAM_UPDATE_FIRMITEM;
```

Parameter

musFirmItemReference contains the Firm Item Reference of the Firm Item.

musCtrl specifies the flags, which parameters should be updated.

- *CM_UFI_FAC* specifies that the Firm Access Counter should be updated.

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_UFI_FUC* specifies that the Firm Update Counter should be updated.
- *CM_UFI_FPT* specifies that the FirmPreciseTime should be updated.
- *CM_UFI_TEXT* specifies that the Firm Item Text should be updated.

musFirmAccessCounter (encrypted) contains the initial value of the Firm Access Counter. If this member is set to `0xFFFF` (the recommended value), this counter is not decremented by an encryption which targets this counter.

mulFirmUpdateCounter (encrypted) contains the initial value of the Firm Update Counter. The recommended value is 0.

mulFirmPreciseTime (encrypted) contains the new Firm Precise Time in seconds since 1st January 2000, 0:00:00. If the value is not used, 0 must be specified, which sets this field to the Running Box Clock value.

mcchText contains the initial length of the Firm Item Text of the Firm Item to program.

mausText contains the initial Firm Item Text of the Firm Item to program.

CM_GF_DELETE_FIRMITEM deletes a Firm Item in a CmDongle. **pvCtrl** is pointing to a CMPROGRAM_DELETE_FIRMITEM structure. **hcmse** must be Firm Item Access or Entry Access.

Structure CMPROGRAM_DELETE_FIRMITEM

CMPROGRAM_DELETE_FIRMITEM contains all parameters needed for deleting a complete Firm Item.

Syntax

```
typedef struct __CMPROGRAM_DELETE_FIRMITEM {
    CMUSHORT musFirmItemReference;
    CMBYTE mabTrailingValidationBlock[CM_TVB_LEN];
} CMPROGRAM_DELETE_FIRMITEM;
```

Parameter

musFirmItemReference contains the Reference of the Firm Item which has to be removed.

mabTrailingValidationBlock contains a 16-byte TVB (Trailing Validation Block), which must be valid to execute the operation. This TVB is ignored, if the addressed Firm Item was directly created with *CmReserveFirmItem()*.

CM_GF_ADD_PRODUCTITEM adds a Product Item into a CmDongle. *pvCtrl* is pointing to a CMPROGRAM_ADD_PRODUCTITEM structure. **hcmse** must be Entry Access.

Structure CMPROGRAM_ADD_PRODUCTITEM

CMPROGRAM_ADD_PRODUCTITEM contains all needed information to add a Product Item to a CmDongle. All parameters are encrypted.

Syntax

```
typedef struct __CMPROGRAM_ADD_PRODUCTITEM {
    CMUSHORT musFirmItemReference;
    CMUSHORT musProductItemSuccessorReference;
    /* variable length, expand structure on the stack */
    CMBYTE mabProductItemOptionBuffer[1];
} CMPROGRAM_ADD_PRODUCTITEM;
```

Parameter

musFirmItemReference (encrypted) contains the Reference of the Firm Item for which the Product Item is to be added. If this is 1, the IFI (Implicit Firm Item) is used. The addressed item must not have set the type. 0 is either not permitted.

musProductItemSuccessorReference (encrypted) contains the Reference of the Product Item, which will be the successor of the newly added Product Item. If this value is 0, the Product Item is created at the end of the Product Item list. If **musFirmItemReference** specifies 0 for the IFI, the Product Item may be individually addressed by setting 2 for User Product Item (UPI). 1 for the

© Copyright 2018 WIBU-SYSTEMS AG

Common Product Item (CPI) is not permitted because this must be always the first Product Item of the IFI.

mabProductItemOptionBuffer (encrypted) contains a byte sequence of variable length which specifies the added Product Code and all further Product Item Options. The sequence is defined by one or more subsequent `CMCPIO_...` structures. The parameter is filled with *CmCreateProductItemOption()* and finalized with *CmCreateSequence()*. The complete list is encrypted by AES-CBC. Finally, a 16-bytes TVB (Trailing Validation Block) is added, which is the encrypted SHA-256 hash of the preceding padded byte sequence.

CM_GF_UPDATE_PRODUCTITEM updates a Product Item in a CmDongle. **pvCtrl** is pointing to a CMPROGRAM_UPDATE_PRODUCTITEM structure. **hcmse** must be Entry Access.

Structure CMPROGRAM_UPDATE_PRODUCTITEM

CMPROGRAM_UPDATE_PRODUCTITEM contains all needed parameters to update a Product Item.

Syntax

```
typedef struct __CMPROGRAM_UPDATE_PRODUCTITEM {
    CMUSHORT musFirmItemReference;
    CMUSHORT musProductItemReference;
    CMUSHORT musCtrl;
    /* variable length, expand structure on the stack */
    CMBYTE mabProductItemOptionBuffer[1];
} CMPROGRAM_UPDATE_PRODUCTITEM;
```

Parameter

musFirmItemReference contains the reference of the Firm Item in which the Product Item should be updated.

musProduct ItemReference contains the reference of the Product Item which should be updated.

musCtrl contains flags set internally by the driver.

mabProductItemOptionBuffer contains a byte sequence of variable length which specifies the added Product Code and all further Product Item Options. The sequence is defined by one or more subsequent `CMCPIO_...` structures. The parameter is filled with *CmCreateProductItemOption()* and finialized with *CmCreateSequence()*. The complete list is encrypted by AES-CBC. Finally, a 16-bytes TVB (Trailing Validation Block) is added, which is the encrypted SHA-256 hash of the preceding padded byte sequence.

CM_GF_DELETE_PRODUCTITEM deletes a Product Item in a CmDongle. **pvCtrl** is pointing to a CMPROGRAM_DELETE_PRODUCTITEM structure. **hcmse** must be Entry Access.

Structure CMPROGRAM_DELETE_PRODUCTITEM

CMPROGRAM_DELETE_PRODUCTITEM contains the parameters for deleting a Product Item.

Syntax

```
typedef struct __CMPROGRAM_DELETE_PRODUCTITEM {
    CMUSHORT musFirmItemReference;
    CMUSHORT musProductItemReference;
    CMBYTE mabTrailingValidationBlock[CM_TVB_LEN];
} CMPROGRAM_DELETE_PRODUCTITEM;
```

Parameter

musFirmItemReference contains the reference of the Firm Item in which the Product Item is deleted.

musProductItemReference contains the reference of the Product Item which is to be deleted.

mabTrailingValidationBlock contains the Trailing Validation Block (TVB) which checks, if the delete operation is valid.

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_GF_SET_BOXCONTROL* sets the control flag of the CmDongle. **pvCtrl** is pointing to a CMPROGRAM_BOXCONTROL structure.

Structure CMPROGRAM_BOXCONTROL

CMPROGRAM_BOXCONTROL contains the parameters to program the CmDongle control registers.

Syntax

```
typedef struct __CMPROGRAM_BOXCONTROL {
    CMULONG mflCtrl;
    CMUSHORT musIndicatorFlags;
    CMUSHORT musReserve;
} CMPROGRAM_BOXCONTROL;
```

Parameter

mflCtrl specifies the kind of the programming of the CmDongle control registers.
- *CM_BC_ABSOLUTE* specifies that the control registers is programmed absolutely.
- *CM_BC_SWITCH* specifies that the control registers is programmed in the switch mode.

musIndicatorFlags contains the two bytes to program or switch.

musReserve is reserved for future purposes.

- *CM_GF_SET_FIRMKEY* programs a Firm Key into an already programmed FSB entry in a CmDongle. **pvCtrl** is pointing to a 32 element byte array. **hcmse** must be Entry Access.
- *CM_GF_ADD_ENABLEBLOCK* adds a new Enable Block Item to a Firm Item. The input parameters are stored in a CMENABLING_WRITEADD structure. The return value of the function is the index of the created Enable Block Item in the Enable Block Table. If no Enable Block Item is created, then 0 is returned and an error code is set.

Structure CMENABLING_WRITEADD

CMENABLING_WRITEADD describes the request sequence of the adding a Enable Block Item.

Syntax

```
typedef struct __CMENABLING_WRITEADD {
  CMULONG mcbLength;
  CMUSHORT musFirmItemReference;
  CMENABLING_ENABLEBLOCKITEM mcmEnableBlockItem;
} CMENABLING_WRITEADD;
```

Parameter

mcbLength contains the size of this structure.

musFirmItemReference contains the reference of the Firm Item in which the Enabling Block is added. If this is 0, the IFI (Implicit Firm Item) is used.

mcmEnableBlockItem specifies the Enabling Block Item which is added.
This is a variable length sequence which depends on the Enabling Block Item Type. The subsequent bytes are encrypted by AES-CBC. The sequence is padded with 0 bytes to a multiple of 16 bytes which is encrypted.
After this sequence a TVB (Trailing Validation Block) is added, which is the AES-encryption of the SHA-256 hash of the data before they are encrypted including the padded area (see structure CMENABLING_ENABLEBLOCKITEM).

Structure CMENABLING_ENABLEBLOCKITEM

CMENABLING_ENABLEBLOCKITEM contains data of a Enable Block Item (EBI) when they should be set into the CmContainer.

Syntax

© Copyright 2018 WIBU-SYSTEMS AG

```
typedef struct __CMENABLING_ENABLEBLOCKITEM {
  CMUSHORT musOption;
  CMBYTE mbIndex;
  CMBYTE mbType;
  union {
  /* variable length */
  CMENABLING_SIMPLEPIN mcmSimplePin;
  CMENABLING_TIMEPIN mcmTimePin;
  } mEnableType;
} CMENABLING_ENABLEBLOCKITEM;
```

Parameter

musOption specifies the following flags.

- *CM_EBI_VALID* specifies that the enabling block item is a valid structure.
- *CM_EBI_ENABLED* specifies that the block is permanently enabled.
- *CM_EBI_DISABLED* specifies that the Disabling Flag of the Enable Block Item is set and the block is permanently disabled.
- *CM_EBI_TEMPENABLED* specifies that the Enable Block Item is temporarily enabled, independently of the static Enabling Status of the Enable Block Item. For a Firmware version smaller than version 1.18 this option is only permitted, if the Enable Block Item is owned by the IFI (Implicit Firm Item = Firm Code 0).

mbIndex specifies the EBI Index inside a Firm Item (or IFI).

mbType specifies the type of the used enabling. This can be one of the following flags:

- *CM_EBI_SIMPLEPIN* specifies that SimplePin enabling is used.
- *CM_EBI_TIMEPIN* specifies that TimePin enabling is used.

mcmSimplePin contains the data when the EBI is using SimplePin enabling (see structure CMENABLING_SIMPLEPIN).

Structure CMENABLING_SIMPLEPIN

CMENABLING_SIMPLEPIN describes an Enabling Block of type "Simple PIN" in the Add Enable Block or Update Enable Block command request sequence.

Syntax

```
typedef struct __CMENABLING_SIMPLEPIN {
  CMBYTE mcbEnableAccessCode;
  /* variable length, max = 16 */
  CMBYTE mabEnableAccessCode[CM_CHALLENGE_LEN];
} CMENABLING_SIMPLEPIN;
```

Parameter

mcbEnableAccessCode contains the number of bytes which are defined in the Enable Access Code. This must be a value between 2 and 16.

mabEnableAccessCode contains the Enable Access Code bytes, the length is defined by the **mcbEnableAccessCode** member.

mcmTimePin contains the data when the EBI is using TimePin enabling (see structure CMENABLING_TIMEPIN).

Structure CMENABLING_TIMEPIN

CMENABLING_TIMEPIN describes an Enabling Block of type "Time PIN" in the Add Enable Block or Update Enable Block command request sequence.

Syntax

```
typedef struct __CMENABLING_TIMEPIN {
  CMULONG mulDisableTime;
  CMBYTE mcbEnableAccessCode;
```

© Copyright 2018 WIBU-SYSTEMS AG

```
  CMUSHORT mcchText;
  /* variable length, AccessCode + Text added, expand structure on the
  stack */
  CMBYTE mabBuffer[1];
} CMENABLING_TIMEPIN;
```

Parameter

mulDisableTime contains the Disable Time. This is number of seconds since the 1st January 2000 0:00:00 (UTC).

mcbEnableAccessCode contains the number of bytes which are defined in the Enable Access Code. This must be a value between 2 and 16.

mcchText contains the length of the description text.

mabBuffer specifies the text of the "Time PIN" and contains **mcbEnableAccessCode** bytes of Access Code and **mcchText** Unicode characters.

- *CM_GF_UPDATE_ENABLEBLOCK* updates the content of an existing Enable Block Item. The input parameters are stored in a byte array (pointer to unsigned char). The return value for this flag is 1 (TRUE), if no error occurred, else 0 (FALSE).
- *CM_GF_DELETE_ENABLEBLOCK* deletes an existing Enable Block Item in a Firm Item. The input parameters are stored in a CMENABLING_WRITEDELETE structure. The return value for this flag is 1 (TRUE), if no error occurred, else 0 (FALSE).

Structure CMENABLING_WRITEDELETE

CMENABLING_WRITEDELETE describes the request sequence for deleting a Enable Block Item (EBI). command.

Syntax

```
typedef struct __CMENABLING_WRITEDELETE {
  /* plain members */
  CMUSHORT musFirmItemReference;
  CMBYTE mbEnableBlockIndex;
  CMBYTE mabTrailingValidationBlock[CM_TVB_LEN];
} CMENABLING_WRITEDELETE;
```

Parameter

musFirmItemReference contains the reference of the Firm Item which defines the Enabling Block. If this is 0, the IFI (Implicit Firm Item) is used.

mbEnableBlockIndex contains the index of the Enabling Block which is deleted. This must be a value between 0 and 30.

mabTrailingValidationBlock contains the Trailing Validation Block (TVB), which checks whether the delete operation is valid. This is an AES encryption of the SHA-256 hash of the preceding bytes.

- *CM_GF_ATTACH_ENABLEBLOCK* attaches an existing Enable Block Item to a Firm Item or Product Item. The input parameters are stored in a CMENABLING_WRITEATTACHDETACH structure. The return value for this flag is 1 (TRUE), if if no error occurred, else 0 (FALSE).

Structure CMENABLING_WRITEATTACHDETACH

CMENABLING_WRITEATTACHDETACH describes the request sequence for attaching or detaching a Enable Block Item (EBI).

Syntax

```
typedef struct __CMENABLING_WRITEATTACHDETACH {
  CMBYTE mbEnableBlockIndex;
  CMUSHORT musFirmItemReference;
  CMUSHORT musProductItemReference;
```


© Copyright 2018 WIBU-SYSTEMS AG

```
  CMULONG mulFirmUpdateCounter;
  CMENABLING_ENABLELOOKUPENTRY mcmEnableLookupEntry;
  CMBYTE mabTrailingValidationBlock[CM_TVB_LEN];
} CMENABLING_WRITEATTACHDETACH;
```

Parameter

mbEnableBlockIndex contains the index of the Enabling Block Item which is attached.

The subsequent bytes are encrypted by AES-CBC or Triple-DES-CBC. They have a fixed length of 32 bytes.

musFirmItemReference contains the reference of the Firm Item in which the Enabling Block is attached. If this is 0, the IFI (Implicit Firm Item) is used.

musProductItemReference contains the Reference of the Product Item in which the Enabling Block is attached. If the Enabling Block should be attached to the Firm Item, this member must be set to 0. If **musFirmItemReference** specifies 0, the Product Item may be individually addressed by setting Common Product Item (CPI) or User product item (UPI).

mulFirmUpdateCounter contains the current value of the Firm Update Counter of the Firm Item which owns the Enabling Block.

mcmEnableLookupEntry contains the Enabling Lookup Entry which defines the attachment. If the **mbEnableLevel** member is set to **CM_ELUE_DETACH**, the Enabling Lookup Entry is deleted.

mabTrailingValidationBlock contains the TVB (Trailing Validation Block), which contains the current hash sequence (16 bytes) to validate that the sequence is correctly encrypted.

Notes

All parameters set by the user to this structure must be set in Little Endian format! Normally this must be noticed on Big Endian CPU's like the Apple Macintosh. On PC's (x86) the parameters (16 & 32 Bit) are stored in Little Endian so that the user has nothing to mentioned (Little/Big Endian).

- *CM_GF_DETACH_ENABLEBLOCK* detaches an existing Enable Block Item from a Firm Item or Product Item. The input parameters are stored in a CMENABLING_WRITEATTACHDETACH structure. The return value for this flag is 1 (TRUE), if no error occurred, else 0 (FALSE).
- *CM_GF_ENABLEITEM* enables the Enabling Block Item defined in CMENABLING_CONTEXT

Structure CMENABLING_CONTEXT

CMENABLING_CONTEXT contains enabling data. It can be collected using *CmGetInfo* with option CM_GEI_ENABLEINFO. It can be used for enabling using *CmProgram*.

Syntax

```
typedef struct __CMENABLING_CONTEXT {
    CMULONG mulFirmCode;
    CMULONG mulProductCode;
    CMUSHORT musOption;
    CMBYTE mbIndex;
    CMBYTE mbType;
    CMULONG mulDisableTime;
    CMUSHORT mausText[CM_MAX_STRING_LEN];
    CMUSHORT musFlags;
    CMBYTE mbItemType;
    CMBYTE mcbEnableAccessCode;
    CMBYTE mabEnableAccessCode[CM_CHALLENGE_LEN];
    CMUSHORT musEnablePassword[CM_MAX_PASSWORD_LEN];
    CMBYTE mbEnableLevel;
    CMBYTE mbDisableLevel;
} CMENABLING_CONTEXT;
```

Parameter

mulFirmCode contains the Firm Code the enabling block is attached to.

© Copyright 2018 WIBU-SYSTEMS AG

mulProductCode contains the Product Code the enabling block is attached to. In the case of a value of 0, check **mbItemType** to see, if it is attached to Product Item with Product Code 0 or to the Firm Item.

musOption specifies the following flags.

- *CM_EBI_VALID* specifies that the enabling block item is a valid structure.
- *CM_EBI_ENABLED* specifies that the block is permanently enabled.
- *CM_EBI_DISABLED* specifies that the Disabling Flag of the Enabling Block Item is set and the block is permanently disabled.
- *CM_EBI_TEMPENABLED* specifies that the Enabling Block Item is temporarily enabled, independently of the static Enabling Status of the Enable Block Item. For Firmware version smaller than version 1.18 this option is only permitted, if the Enable Block Item is owned by the IFI (Implicit Firm Item = Firm Code 0).

mbIndex specifies the EBI Index inside a Firm Item (or IFI).

mbType specifies the type of the used enabling. This can be one of the following flags:

- *CM_EBI_SIMPLEPIN* specifies that Simple Pin enabling is used.
- *CM_EBI_TIMEPIN* specifies that Time Pin enabling is used.

mulDisableTime contains disable time in seconds since the 1st January 2000 0:00:00 (UTC).

mausText contains the text of the Enabling Block Item. Only available for Time Pin and Finger Pin.

musFlags specifies the following states:

- *CM_ELUE_VALID* specifies that the Enabling Lookup Entry is a valid structure.
- *CM_ELUE_IFI* specifies that the Enabling Block Item of the IFI is used and not of the owning Firm Item.
- *CM_ELUE_REQUIRED* specifies that this lookup entry must permit the operation independent to other settings.

musFlags specifies the following states:

mbItemType specifies the following item types:

- *CM_EC_PRODUCTITEM* specifies that the Enabling Block Item is attached to the Product Item.
- *CM_EC_FIRMITEM* specifies that the Enabling Block Item is attached to the Firm Item.
- *CM_EC_GLOBALITEM* specifies that the Enabling Block Item is attached to the IFI (Implicit Firm Item = Firm Code 0)

mcbEnableAccessCode contains the number of bytes which are defined in the Enable Access Code. This must be a value between 2 and 16.

mabEnableAccessCode contains the Enable Access Code.

musEnablePassword contains the Enable Password.

mbEnableLevel specifies the Enable Level. This can be one of the following flags (see *mbDisableLevel*).

mbDisableLevel specifies the Disable Level. This can be one of the following flags:

- *CM_ELUE_LOCATE* only the reading of the Firm Code, the Product Code, the Enabling Block Table and the Enabling Lookup Table is permitted.
- *CM_ELUE_READ* the reading of the Firm Item or Product Item is permitted but no operations.
- *CM_ELUE_ENCRYPT* encryption, decryption and signature calculation is permitted but not with modification of the Unit Counter.
- *CM_ELUE_UNITUSE* encryption, decryption and signature calculation may change the Unit Counter. In a Firm Item, the Firm Update Counter and the Access Counter may be changed.
- *CM_ELUE_MODIFY* all operations are permitted including modification of the properties by update commands.

Note

You may either use **mabEnableAccessCode** or **musEnablePassword** for enabling / disabling the item depending on the value of **mcbEnableAccessCode**. If **mcbEnableAccessCode** is 0, the text in **musEnablePassword** is used, otherwise the code of **musEnableAccessCode** is used.
CMENABLING_CONTEXT is supported since CodeMeter Version 4.30.

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_GF_DISABLEITEM* disables the Enabling Block Item defined in CMENABLING_CONTEXT.
- *CM_GF_TEMPENABLEITEM* enables temporarily the Enabling Block Item defined in CMENABLING_CONTEXT.

pvCtrl contains a structure depending on the **flCtrl** flag (see above).

cbCtrl contains the number of bytes of the **pvCtrl** byte buffer/structure.

pvVerify contains a byte buffer receiving the verification sequence for the programmer. Except of the CM_GF_DELETE_FIRMITEM command this is a signed list command of the added/updated/deleted Firm Item *(Signed List of the whole Firm Item)*. CM_GF_DELETE_FIRMITEM handles the creation of a verification sequence in the CmDongle internally. A CMVALIDATE_DELETEFI structure must be passed here. If the parameter is NULL (==>> **cbVerify** = 0), no verification sequence is returned. If the buffer is to small for the verification sequence, the return value is larger than the **cbVerify** value and the error code CMERROR_BUFFER_OVERFLOW is set.

Structure CMVALIDATE_DELETEFI

CMVALIDATE_DELETEFI contains the data received by a CmDongle after deleting a Firm Item to validate, if a Firm Item is deleted correctly with a Firm Security Box (FSB).

Syntax

```
typedef struct __CMVALIDATE_DELETEFI {
    CMULONG mflFiCtrl;
    CMULONG mulFirmCode;
    CMULONG mulSystemTime;
    CMULONG mulFirmUpdateCounter;
    CMBYTE mabTrailingValidationBlock[CM_TVB_LEN];
} CMVALIDATE_DELETEFI;
```

Parameter

mflFiCtrl specifies the type of a Firm Item. This can be one of the following values:

- *CM_GF_BOXBASED* specifies that the Firm Item was programmed in the Box-Based-Mode.
- *CM_GF_FILEBASED* specifies that the Firm Item was programmed in the File-Based-Mode.
- *CM_GF_TRANSFERBASED* specifies that the Firm Item has been created based on a license transfer.
- *CM_GF_FI_RESERVED* specifies that the Firm Item is a Reserved Firm Item.

mulFirmCode contains the deleted Firm Code

mulSystemTime contains the current value of the System Running Box Clock. This is the number of seconds since 1st January 2000, 0:00:00. This value is used to build the TVB for an authentication that the Firm Item was deleted.

mulFirmUpdateCounter contains the final value of the Firm Update Counter. This value is used to build the TVB for an authentication that the Firm Item was deleted.

mabTrailingValidationBlock contains a 16-byte Trailing Validation Block (TVB), which may be recalculated by the sender of the Request Sequence that the operation was successfully executed.

cbVerify contains the number of bytes of the byte buffer/structure stored in **pvVerify**.

Return

If the programming task is executed successfully and if **pvVerify** is NULL, then TRUE is returned, else FALSE is returned. If **pvVerify** contains enough memory, the length of the signed list or a CMVALIDATE_DELETEFI structure is returned. If **pvVerify** does not contain enough memory, the number of required bytes will be returned. In this case the return value is larger than the **cbVerify** parameter and error code CMERROR_BUFFER_OVERFLOW is set.

The following flags (and the corresponding functionality) are only available since CodeMeter Version 4.30:

- CM_GF_ENABLEITEM
- CM_GF_DISABLEITEM
- CM_GF_TEMPENABLEITEM

© Copyright 2018 WIBU-SYSTEMS AG

3.7 Remote Programming operations

3.7.1 CmExecuteRemoteUpdate

Description

CmExecuteRemoteUpdate executes the given updates and applies them as specified. This function allows the same handling for CmDongles and CmActLicenses regarding the update process. It allows also to apply several updates in one step but getting feedback for each update.

With *CmExecuteRemoteUpdate* you can execute the contents of Update Files (`*.WibuCmRaU`) and of license information files (`*.wbb`). If you have files you need to read, the files by yourself and store the contents in the memory buffer.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmExecuteRemoteUpdate(HCMSysEntry hcmse, CMULONG flCtrl, void*
pMemBuffer, CMUINT cbMemBuffer, CMFAS *pCmFasSelect, CMUINT cbCmFasSelect,
CMFAS *pCmFasResult, CMUINT cbCmFasResult, CMUINT *pcbCmFasResultCount)
```

Parameter

hcmse contains the handle opened by CmAccess2 (must be Subsystem Access).

flCtrl contains one of the following commands:

- *CMFAS_CTL_CMD_LIST* lists all updates and simulates the execution without any change on the licenses. The input structure **pCmFasSelect** has to be null. This flag cannot be combined with any other flag. The simulated result can be found in **pCmFasResult.mulResult**.
- *CMFAS_CTL_CMD_UPDATE* all updates are executed. Any set restriction in **pCmFasSelect** is respected.
- *CMFAS_CTL_CMD_UPDATE_EXISTING* only updates for existing CmContainer are executed. Any set restriction in **pCmFasSelect** is respected. If the additional option *CMFAS_CTL_OPT_ABORT_ON_ERROR* is set, the command only aborts, if an update of an existing CmContainer fails.
- *CMFAS_CTL_CMD_GET_COUNT* returns the count of needed structures of structure CMFAS for **pCmFasResult**. All other parameters can be null resp. `0`.

Structure CMFAS

CMFAS specifies additional data for *CmExecuteRemoteUpdate* or retrieves detailed results of this command.

Syntax

```
typedef struct __CMFAS{
    CMULONG mflSelect;
    CMULONG mulItemType;
    CMULONG mulSerial;
    CMUSHORT musMask;
    CMUSHORT musReserved1;
    CMULONG mulFirmCode;
    CMULONG maulReserved3[20];
    CMULONG mulResult;
    CMULONG maulReserved4[10];
} CMFAS;
```

Parameter

mflSelect specifies the restrictions of updates by setting one of the following flags:

- *CMFAS_SELECT_WILDCARD* specifies that no restriction regarding the update is set.

© Copyright 2018 WIBU-SYSTEMS AG

- *CMFAS_SELECT_SERIALNUMBER* specifies that a specific serial number is set in **mulSerial** and **musMask**. The update is only applied, if the serial number matches.
- *CMFAS_SELECT_FIRMCODE* specifies that a Firm Code is set in **mulFirmCode**. The update is only applied, if the Firm Code matches.
- To retrieve the select options you can use *CMFAS_SELECT_MASK*.
- *CMFAS_OPT_IGNORE* specifies that this update is ignored. This update will not be listed in the CMFAS array that retrieves the result of this operation.
- To retrieve the additional options you can use *CMFAS_OPT_MASK*.

mulItemType retrieves the type of update as a result of the update execution:

- *CMFAS_ITEMTYPE_LICENSE* specifies that the update was a license update.
- *CMFAS_ITEMTYPE_CMACT_LICENSE* specifies that the license was a CmActLicense update.
- *CMFAS_ITEMTYPE_CMHW_LICENSE* specifies that the update type was CmDongle.
- *CMFAS_ITEMTYPE_CMFIRM_WBC* specifies that the update was a `CmFirm.wbc` (only for developer).
- *CMFAS_ITEMTYPE_CMACT_FI_TEMPLATE* specifies that the update was a CmActLicense template file (`*.wbb`, only for developer).

mulSerial contains the serial number of the update either as restriction (input) or as result (list or update).

musMask contains the serial mask of the update either as restriction (input) or as result (list or update).

musReserved1 is reserved for future purposes.

mulFirmCode contains the Firm Code of the update either as restriction (input) or as result (list or update).

maulReserved3[20] is reserved for future purposes.

mulResult contains the error codes of the single updates. If the error code CMERROR_UPDATE_FAILED is set, more than one update was specified and at least one failed; detailed information can be found here.

maulReserved4[10] is reserved for future purposes.

To retrieve all the command options you can use `CMFAS_CTL_CMD_MASK`.

These flags can be or'ed with one of the following flags:

- *CMFAS_CTL_OPT_ABORT_ON_ERROR*: If more than one update is specified, the command aborts at the first error.

To retrieve all the command options you can use `CMFAS_CTL_OPT_MASK` specifies all flags of the optional command options.

pMemBuffer contains the data of the update.

cbMemBuffer contains the length of **pMemBuffer**.

pCmFasSelect contains an array of structures with input data, e.g. the definition which updates should be applied.

cbCmFasSelect contains the count of CMFAS structures contained in **pCmFasSelect**.

pCmFasResult contains an array of CMFAS structures with output data, e.g. which updates were successful.

cbCmFasResult contains the count of CMFAS structures contained in **pCmFasResult**.

pcbCmFasResultCount retrieves the count of filled CMFAS structures in **pCmFasResult**.

Return

Returns TRUE, if all specified updates were successful, FALSE otherwise.

If the error code CMERROR_BUFFER_OVERFLOW is set, the count of CMFAS structures in **pCmFasResult** was too small; the needed count can be found in **pcbCmFasResultCount**.

If the option **CMFAS_CTL_OPT_ABORT_ON_ERROR** was set and the command aborted due to an error, the error code is set to the specific error code of the update that has failed.

© Copyright 2018 WIBU-SYSTEMS AG

If the error code CMERROR_UPDATE_FAILED is set, more than one update was specified and at least one failed; detailed information can be found in **pCmFasResult**.

Notes

Detailed information about the results of the update execution can be found in **pCmFasResult**.

3.7.2 CmGetRemoteContextBuffer

Description

This function operates like *CmGetRemoteContext()* but uses a pre-allocated memory buffer instead of a file pointer.


For CmDongle only a local access is allowed. CmActLicense cover accesses locally or via LAN.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmGetRemoteContextBuffer(HCMSysEntry hcmBox, CMUINT flCtrl,
CMUINT *pulFirmCodes, CMUINT cbFirmCodes, void* pvMemBuffer, CMUINT
cbMemBuffer, CMUINT* pcbWritten);
```

Parameter

hcmBox contains a handle opened by CmAccess2.

flCtrl reserved for future purposes.

pulFirmCodes points to a set of Firm Codes to reserve.

cbFirmCodes holds the number of Firm Codes in **pulFirmCodes**.

pvMemBuffer pointer to memory to hold the data.

cbMemBuffer holds the size of **pvMemBuffer**.

pcbWritten receives the number of bytes written to **pvMemBuffer**. If the number of bytes given in **cbMemBuffer** is too small to receive the complete context, the number of necessary bytes is written to **pcbWritten**.

Return

Returns the number of Firm Items stored in the Context File `*.WibuCmRaC` memory buffer. If **pvDest** does not contain enough memory, the number of needed bytes will be written to **pcbWritten**. In this case the return value is larger than the **cbDest** parameter and error code CMERROR_BUFFER_OVERFLOW is set.

Notes

If a Firm Item with the specified Firm Code does not exist in the CmDongle, *CmReserveFirmItem()* will be called implicitly. The result will then be stored in the Context File `*.WibuCmRaC` memory buffer.

3.8 Secure Disk operations

3.8.1 CmSecureDiscRead

Description

CmSecureDiscRead is used to read one or several sectors of a Secure Disk partition of a CmContainer.


For more details on the Secure Disk feature, please contact Wibu-Systems support.

C, C++ Programming Language

© Copyright 2018 WIBU-SYSTEMS AG

Syntax

```
int CMAPIENTRY CmSecureDiscRead(HCMSysEntry hcmse, CMULONG flCtrl, const
CM_DISC_SECTOR *sector, const CMBYTE *data, CMUINT cbData);
```

Parameter

hcmse contains a handle opened by *CmAccess2* (can be Box Access, Entry Access or Firm Item access).
flCtrl contains the flags available for reading operation of a secure disk partition:

- *CM_SECURE_DISC_READ* reads sector(s) of a secure disk partition (0x00000201).

sector contains the sector specification of the secure disk partition (see structure CM_DISC_SECTOR).

Structure CM_DISC_SECTOR

CM_DISC_SECTOR contains sector information on the secure disk partition.

Syntax

```
typedef struct __CM_DISC_SECTOR {
  CMULONG mulNumberOfRequestedSectors;
  CMULONG mulSectorOffset;
  CMBYTE mabReserved[56];
} CM_DISC_SECTOR;
```

Parameter

mulNumberOfRequestedSectors contains the number of sectors (at least 1).
mulSectorOffset contains the start sector.
mabReserved[56] must be NULLed and is reserved for future purposes.

data contains the sector information.
cbData contains the length of **data** in bytes .

Return

If the functions fails, a value of 0 is returned.

Notes

3.8.2 CmSecureDiscWrite

Description

CmSecureDiscWrite is used to write one or several sectors of a Secure Disk partition of a CmContainer.

For more details on the Secure Disk feature, please contact Wibu-Systems support.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmSecureDiscWrite(HCMSysEntry hcmse, CMULONG flCtrl,
CM_DISC_STATUS *status, const CM_DISC_SECTOR *sector, const CMBYTE *data,
CMUINT cbData);
```

Parameter

hcmse contains a handle opened by *CmAccess2* (can be Box Access, Entry Access or Firm Item access).
flCtrl contains the flags available for reading operation of a secure disk partition:

- *CM_SECURE_DISC_WRITE* writes sector(s) of a secure disk partition (0x00000202).

This flag can be or'ed with *CM_SECURE_DISC_WRITE*. 'Auto' is the default:

- *CM_SECURE_DISC_VERIFY_ON* (0x00010000).

© Copyright 2018 WIBU-SYSTEMS AG

- *CM_SECURE_DISC_VERIFY_OFF* (0x00020000).
- *CM_SECURE_DISC_VERIFY_AUTO* (0x00000000).

These flags control whether after writing the sectors are read and compared with written sectors. Default is *CM_SECURE_DISC_VERIFY_AUTO*, i.e depending on technical conditions a verify is performed. Currently, this is the case, if the file system is not FAT.
If the verify is not successful, the error CMERROR_VERIFY_FAILED (308) returns.

status contains contains information on the secure disk partition (see structure CM_DISC_STATUS).

Structure CM_DISC_STATUS

CM_DISC_STATUS contains information on the secure disk partition.

Syntax

```
typedef struct __CM_DISC_STATUS {
  CMULONG mulErrorCode;
  CMULONG mulEnablingBits;
  CMULONG mulDiscSectorsTotal;
  CMULONG mulWriteOnceCount;
  CMBYTE  mabReserved[64];
} CM_DISC_STATUS;
```

Parameter

mulErrorCode contains the error code after a read or write operation.
mulEnablingBits contains the enabling bits set for the Secure Disk partition.
mulDiscSectorsTotal contains the total number of sectors (each 512 byte).
mulWriteOnceCount contains the number of the block following a block which is one-time written.
mabReserved[64] is reserved for future purpose.

sector contains the sector specification of the secure disk partition (see structure CM_DISC_SECTOR).

Structure CM_DISC_SECTOR

CM_DISC_SECTOR contains sector information on the secure disk partition.

Syntax

```
typedef struct __CM_DISC_SECTOR {
  CMULONG mulNumberOfRequestedSectors;
  CMULONG mulSectorOffset;
  CMBYTE mabReserved[56];
} CM_DISC_SECTOR;
```

Parameter

mulNumberOfRequestedSectors contains the number of sectors (at least 1).
mulSectorOffset contains the start sector.
mabReserved[56] must be NULLed and is reserved for future purposes.

data contains the sector information.
cbData contains the length of **data** in bytes.

Return

If the functions fails, a value of 0 is returned.

Notes

© Copyright 2018 WIBU-SYSTEMS AG

3.8.3 CmExtendedDiscControl

Description

CmExtendedDiscControl is used to obtain information of a Secure Disk partition of a CmContainer.


For more details on the Secure Disk feature, please contact Wibu-Systems support.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmExtendedDiscControl(HCMSysEntry hcmse, CMULONG flCtrl, const
CMBYTE *data, CMUINT cbData);
```

Parameter

hcmse contains a handle opened by *CmAccess2* (can be Box Access, Entry Access or Firm Item access).
flCtrl contains the flags available for obtaining information of a secure disk partition.

- *CM_SECURE_DISC_STATUS* contains information on the secure disc status (0x00000200).
- *CM_GET_DISC_CONFIGURATION* contains information of the disk configuration (0x00000203).

data points to the structures:
CM_DISC_STATUS.

Structure CM_DISC_STATUS

CM_DISC_STATUS contains information on the secure disk partition.

Syntax

```
typedef struct __CM_DISC_STATUS {
  CMULONG mulErrorCode;
  CMULONG mulEnablingBits;
  CMULONG mulDiscSectorsTotal;
  CMULONG mulWriteOnceCount;
  CMBYTE  mabReserved[64];
} CM_DISC_STATUS;
```

Parameter

mulErrorCode contains the error code after a read or write operation.
mulEnablingBits contains the enabling bits set for the Secure Disk partition.
mulDiscSectorsTotal contains the total number of sectors (each 512 byte).
mulWriteOnceCount contains the number of the block following a block which is one-time written.
mabReserved[64] is reserved for future purpose.

CM_DISC_CONFIGURATION)

Structure CM_DISC_CONFIGURATION

CM_DISC_CONFIGURATION contains information of the disk configuration.

Syntax

```
typedef struct __CM_DISC_CONFIGURATION {
  CMBYTE mbMajorFirmwareRevision;
  CMBYTE mbMinorFirmwareRevision;
  CMUSHORT musBytesPerSector;
  CMULONG mulLengthOfBlock0;
  CMULONG mulLengthOfBlock1;
  CMULONG mulLengthOfBlock2;
  CMULONG mulLengthOfBlock3;

  CMULONG mulFlashSize;
```

© Copyright 2018 WIBU-SYSTEMS AG

```
  CMULONG mulSerialNumber;
  CMULONG mulEnablingStatus;

  CMBYTE[12] mabReserved;

} CM_DISC_CONFIGURATION;
```

Parameter

mbMajorFirmwareRevision contains major Firmware version number of the flash controller, e.g. 9.
mbMinorFirmwareRevision contains the minor Firmware version number of the flash controller, e.g. 3.
musBytesPerSector contains the fixed size of the sector. Currently 512.
mulLengthOfBlock0 contains block length of public disk.
mulLengthOfBlock1 contains block length of private disk.
mulLengthOfBlock2 contains block length of cd-rom disk.
mulLengthOfBlock3 contains block length of secure disk.
mulFlashSize contains size of flash memory.
mulSerialNumber contains CodeMeter chip serial number without mask code.
mulEnablingStatus contains the various enabling stati in the IFI (Implicit Firm Item).
mabReserved is reserved for future purpose.
mulArticleMinor contains.
mulArticleSubcase contains.

cbData contains the length of **data** in bytes.

Return

If the functions fails, a value of `0` is returned.

Notes

3.9 Time Update operations

3.9.1 CmSetSystemTime

Description

Using time specifications on implementing time-limited license models

Since embedded systems may often have no real-time clock, *CmEmbedded* features the function ***CmSetSystemTime*** to implement time-based license models. If time-limited license options, e.g. Expiration Time or Usage Period, are to be considered, previous to calling API functions, e.g. *CmCrypt2*, the current time must be set using the API function *CmSetSystemTime*.
CmSetSystemTime sets the internal clock of a CmContainer to the time transferred by the structure CMTIME.
This internal time cannot be set back, i.e. in the case of such an attempt, the call is ignored.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmSetSystemTime( HCMSysEntry hcmse, CMTIME * newSystemTime )
```

Parameter

hcmse contains a handle opened by CmAccess2.
newSystemTime holds the new time. The member '**mulSecondsSince01_01_2000**' of the structure CMTIME is used.

© Copyright 2018 WIBU-SYSTEMS AG

Structure CMTIME

CMTIME sets or gets the time to or from a CmContainer.

Syntax

```
typedef struct __CMTIME {
  CMUSHORT musYear;
  CMUSHORT musMonth;
  CMUSHORT musDay;
  CMUSHORT musHours;
  CMUSHORT musMinutes;
  CMUSHORT musSeconds;
  CMULONG mulSecondsSince01_01_2000;
} CMTIME;
```

Parameter

musYear specifies the value of the year (e.g. `2003`).
musMonth specifies the value of the month (`1..12`).
musDay specifies the value of the day (`1..31`).
musHours specifies the value of the hours (`0..23`).
musMinutes specifies the value of the minutes (`0..59`).
musSeconds specifies the value of the seconds (`0..59`).
mulSecondsSince01_01_2000 specifies the time in seconds since `01/01/2000`.

Notes

When using time-dependent license options, e.g. *Usage Period*, you must call this function before using other essential functions, e.g. *CmCrypt2*.

If you use *CmDongle*, it is sufficient to repeat this call every hour.
If you use *CmActLicense*, Wibu-Systems recommends to call this function more often, e.g. before each *CmCrypt2* call, if the time is not forwarded by internal mechanisms. You test internal forwarding on a platform by repeated detecting the time every other minute using the function *CmGetInfo*(`CM_GEI_BOXTIME`) to prove the correct time.

3.10 License Transfer operations

3.10.1 CmLtCreateContext

Description

This function represents the first step of a return or push license transfer but also on programming a usual entry using the Firm Security Box (FSB). Here all necessary data of the target side is collected (including the required, eventually signed, public key certificates) and is established in the format of a Context File `*.WibuCmRaC`.
For more details see the respective knowledge base article on license transfer.

C, C++ Programming Language

Syntax

```
int CMAPIENTRY CmLtCreateContext(
HCMSysEntry hcmse, CMULONG flCtrl, const CMLTREQUEST *pcmltRequest, const void
*pContextForPull, CMUINT cbContextForPull, void *pContext, CMUINT cbContext);
```

Parameter

hcmse contains a handle opened by *CmAccess2* (can be Box Access, Entry Access or Firm Item Access). Principally, a Box Access handle is set.

© Copyright 2018 WIBU-SYSTEMS AG

flCtrl must contain one of the following flags describing the programming / type of the transfer. The corresponding data is further processed and the transfer is performed.



The transfer or programming types are stored in the output data and remain across the protocol. Thus the flag must not be specified for the following API functions.

- *CM_LT_FSB* specifies programming a license entry using a Firm Security Box (FSB).

pcmltRequest contains a pointer to a CMLTREQUEST structure. In this structure, only the value of **mulFirmCode** is set. This will create the context of this Firm Code and not for the complete CmContainer content. All other values of this structure are ignored for now (see structure CMLTREQUEST).

Structure CMLTREQUEST

CMLTREQUEST contains all needed information for creating the desired license transfer context.

Syntax

```
typedef struct __CMLTREQUEST {
    CMULONG mflCtrl;
            CM_LTR_NORECEIPT
    CMULONG mulFirmCode;
    CMBYTE mabReserved[248];
} CMLTREQUEST;
```

Parameter

mflCtrl currently contains a value of `0`.

If no license transfer context is required, for example on initial programming using a Firm Security Box (FSB), the following flag must be set:

- *CM_LTR_NORECEIPT* specifies that no receipt is created.

mulFirmCode contains the Firm Code for creating the license transfer request.

mabReserved[248] is reserved for future purposes.

pContext points to a byte buffer to which the return data is copied to.

Here the created data is one-to-one copied in format Context File `*.WibuCmRaC` to the memory.

cbContext holds the size of the allocated memory in **pContext**.

Return

The exact number of bytes copied to **pContext** is returned.

- If the number of bytes is smaller or equal to **cbContext**, no error occurred and the data has been copied to **pContext**.
- If the number is larger than **cbContext** no data has been returned and the error code CMERROR_BUFFER_OVERFLOW (--> *CmGetLastErrorCode ()* or *CmGetLastErrorText ()* / *CmGetLastErrorText2 ()*) is set, because the buffer size was too small. Then the call can be repeated using a buffer **pContext** having enough memory allocated.
- If a value of `0` is returned, another error has occurred. The exact reason can be detected using *CmGetLastErrorCode ()* or *CmGetLastErrorText ()* / *CmGetLastErrorText2 ()*.

Notes

Basically, it is appropriate for the first call to use a buffer size which has been defined huge enough before.

Available with CodeMeter Version 6.00 and higher.

3.11 CmExecuteFieldUpdate()

Description

CmExecuteRemoteUpdate executes a firmware update of the CodeMeter Chip, as specified in the given Wibu Binary file, for the given box.

C, C++ Programming Language

© Copyright 2018 WIBU-SYSTEMS AG

Syntax

```
int CmExecuteFieldUpdate (const CMBOXINFO * pcmBoxInfo, const unsigned char *
pbWbb, unsigned int cbSize)
```

The FieldUpdate WBB File in **pbWbb** has to contain a single staged or a double staged FieldUpdate. It has to correspond with the **mulFirmwareBuild** and the **musSiliconRevision** of the specified box, as returned by *CmGetInfo (..., CM_GEI_CHIPINFO, ...)*.

The update's serial number range has to cover the **mulSerialNumber** of **pcmBoxInfo**. Only CodeMeter Chips of type `0x97` or larger (see **musChipType** in *CMCHIPINFO*) can be updated with this function.

Parameter

pcmBoxInfo contains the CMBOXINFO structure that specifies the dongle which is to be updated.

Structure CMBOXINFO

This structure contains information on a CmContainer.

Syntax

```
typedef struct __CMBOXINFO {
     CMBYTE mbMajorVersion;
     CMBYTE mbMinorVersion;
     CMUSHORT musBoxMask;
     CMULONG mulSerialNumber;
     CMUSHORT musBoxKeyId;
     CMUSHORT musUserKeyId;
     CMBYTE mabBoxPublicKey[CM_PUBLIC_KEY_LEN];
     CMBYTE mabSerialPublicKey[CM_PUBLIC_KEY_LEN];
     CMULONG mulReserve;
} CMBOXINFO;
```

Parameter

mbMajorVersion contains the major version of the CmContainer.
mbMinorVersion contains the minor version of the CmContainer.
musBoxMask contains the box mask of the CmContainer.
mulSerialNumber contains the serial number of the CmContainer.
musBoxKeyId contains the ID of the used BoxKey of the CmContainer.
musUserKeyId contains the ID of the used UserKey of the CmContainer.
mabBoxPublicKey[**CM_PUBLIC_KEY_LEN**] contains the public Box Key of the CmContainer.
mabSerialPublicKey[**CM_PUBLIC_KEY_LEN**] contains the public Serial Key of the CmContainer.
mulReserve is reserved for future purposes.

Notes

The length of the **mabBoxPublicKey** and **mabSerialPublicKey** is 64 byte long. The used public key are currently 224 bits = 28 bytes long. In Elliptic Curve Cryptography the public key is a 2 dimensional point, so the x-value is stored in the first 28 bytes (`0..27`) and the y-value is store in byte `32..59`.

pbWbb contains the pointer to the buffer with the WBB file.
cbSize contains the length of the WBB file.

Code

```
CMBOXINFO *cmbxinfo;
CMBYTE *wbb;
HCMSysEntry hsub;
```

© Copyright 2018 WIBU-SYSTEMS AG

```
CMACCESS2 cmacc;
int nb_boxes, success;

memset(&cmacc, 0, sizeof(cmacc));
cmacc.mflCtrl = CM_ACCESS_SUBSYSTEM;
hsub = CmAccess2(CM_ACCESS_LOCAL, &cmacc);

if ((hsub != NULL) && (CmGetBoxes(hsub, CM_GB_DONGLE, cmbxinfo, 0) > 0)) {
   cmbxinfo = malloc(sizeof(CMBOXINFO) * (size_t)nb_boxes);
   CmGetBoxes(hsub, CM_GB_DONGLE, cmbxinfo, (size_t)nb_boxes);
   wbb = read_file_to_buffer();
   success = CmExecuteFieldUpdate(&cmbxinfo[0], wbb, sizeof(*wbb));
   if (success == 0) {
     printf("Error: %d\n", CmGetLastErrorCode());
   }
   free(wbb);
   free(cmbxinfo);
   CmRelease(hsub);
}
```


© Copyright 2018 WIBU-SYSTEMS AG

4 Error Codes

C, C++ Programming Language

Error Name	Error Code
CMERROR_NO_ERROR	0
CMERROR_COMMANDUNDEFINED	1
CMERROR_COMMANDINVALID	2
CMERROR_COMMANDTOOLARGE	3
CMERROR_COMMANDWRONG	4
CMERROR_COMMANDTVBWRONG	5
CMERROR_COMMANDINCOMPLETE	6
CMERROR_AWAITENCRYPTIONRESTART	7
CMERROR_ANSWERBUFFERFULL	8
CMERROR_VALUETOOLARGE	9
CMERROR_OPTIONUNDEFINED	10
CMERROR_OPTIONINVALID	11
CMERROR_TYPEUNDEFINED	12
CMERROR_TYPEWRONG	13
CMERROR_KEYSOURCEUNDEFINED	14
CMERROR_KEYSOURCEINVALID	15
CMERROR_KEYSOURCEMISSED	16
CMERROR_KEYSOURCEWRONG	17
CMERROR_DATATOOSHORT	18
CMERROR_DATATOOLONG	19
CMERROR_DATAWRONG	20
CMERROR_FIRMITEMINVALID	21
CMERROR_FIRMCODEEXISTS	22
CMERROR_PRODUCTITEMINVALID	23
CMERROR_PRODUCTITEMOPTIONUNDEFINED	24
CMERROR_PRODUCTITEMOPTIONINVALID	25
CMERROR_PRODUCTITEMOPTIONMISSED	26
CMERROR_FIRMCODEWRONG	27
CMERROR_PRODUCTCODEWRONG	28
CMERROR_FEATUREMAPMISSED	29
CMERROR_FEATUREMAPINCOMPATIBLE	30
CMERROR_UNITCOUNTERMISSED	31
CMERROR_UNITCOUNTERZERO	32
CMERROR_UNITCOUNTERUNDERRUN	33
CMERROR_EXPIRATIONTIMEMISSED	34
CMERROR_EXPIRATIONTIMEOVERRUN	35
CMERROR_ACTIVATIONTIMEMISSED	36
CMERROR_ACTIVATIONTIMEUNDERRUN	37
CMERROR_FIRMACCESSCOUNTERZERO	38
CMERROR_CERTIFIEDTIMEOBSOLETE	39
CMERROR_ENCRYPTIONINVALID	40
CMERROR_DIFFIEHELLMANFAILED	41
CMERROR_ECIESFAILED	42

© Copyright 2018 WIBU-SYSTEMS AG

Error Name	Error Code
CMERROR_SUBCOMMANDUNDEFINED	43
CMERROR_SUBCOMMANDINVALID	44
CMERROR_SUBCOMMANDINCOMPLETE	45
CMERROR_SUBCOMMANDSTRUCTUREWRONG	46
CMERROR_TVBWRONG	47
CMERROR_FIRMUPDATECOUNTERWRONG	48
CMERROR_PITVBFLAGINVALID	49
CMERROR_PITVBFLAGINCOMPATIBLE	50
CMERROR_ENABLEBLOCKUSED	51
CMERROR_BOXDISABLED	52
CMERROR_FIRMITEMDISABLED	53
CMERROR_PRODUCTITEMDISABLED	54
CMERROR_INDEXINVALID	55
CMERROR_LEVELINVALID	56
CMERROR_TABLEFULL	57
CMERROR_MEMORYFULL	58
CMERROR_MEMORYDAMAGED	59
CMERROR_QUEENMEMORYDAMAGED	60
CMERROR_FIRMITEMDAMAGED	61
CMERROR_CRTOKENOBSOLETE	62
CMERROR_MEMORYWRITEFAILED	63
CMERROR_BOXBUSY	64
CMERROR_FEATURECODEINVALID	65
CMERROR_COMMANDUNSUPPORTED	66
CMERROR_FIELDUPDATELIMITEXCEEDED	67
CMERROR_FLASHWRITEFAILED	68
CMERROR_ACCESSMODEWRONG	69
CMERROR_BOXLOST	70
CMERROR_USAGEPERIODOVERRUN	71
CMERROR_USAGEPERIODUNDERRUN	72
CMERROR_BOXLOCKED	73
CMERROR_BOX_DAMAGED	74
CMERROR_BOX_LOCKED_TO	75
CMERROR_MAINTENANCEPERIODMISSING	76
CMERROR_MAINTENANCEPERIODUNDERRUN	77
CMERROR_MAINTENANCEPERIODOVERRUN	78
CMERROR_MAINTENANCEPERIODINVALID	79
CMERROR_BOXMEMORYDAMAGED	80
CMERROR_BOSSMEMORY_DAMAGED	81
CMERROR_FIRMWAREMEMORY_DAMAGED	82
CMERROR_BOSS_FIRMWAREMEMORY_DAMAGED	83
CMERROR_FUMMEMORY_DAMAGED	84
CMERROR_BOSS_FUMMEMORY_DAMAGED	85

© Copyright 2018 WIBU-SYSTEMS AG

Error Name	Error Code
CMERROR_FIRMWARE_FUMMEMORY_DAMAGED	86
CMERROR_BOSS_FIRMWARE_FUMMEMORY_DAMAGED	87
CMERROR_SYSTIMEOBSOLETE	88
CMERROR_BAD_IO_MODE	89
CMERROR_FULA_UPDATEEB_REACHED	90
CMERROR_FULA_ENCRCOMM_REACHED	91
CMERROR_UNFRAMED_COMMAND_REJECTED	92
CMERROR_COMMAND_PARTIALLY_REJECTED	93
CMERROR_COMMAND_PARTIALLY_UNSUPPORTED	94
CMERROR_BOX_COMMUNICATION_95	95
CMERROR_BOX_COMMUNICATION_96	96
CMERROR_BOX_COMMUNICATION_97	97
CMERROR_BOX_COMMUNICATION_98	98
CMERROR_BOX_LICENSEMEMORY_NOT_FORMATTED	99
CMERROR_NETWORK_FAULT	100
CMERROR_SERVER_NOT_FOUND	101
CMERROR_SEND_FAULT	102
CMERROR_RECEIVE_FAULT	103
CMERROR_INTERNAL_ORGANISATION	104
CMERROR_INVALID_PARAMETER	105
CMERROR_INVALID_HANDLE	106
CMERROR_NO_MORE_HANDLES	107
CMERROR_NO_MORE_MEMORY	108
CMERROR_SECURITY_PROBLEM	109
CMERROR_NO_LOCAL_SERVER_STARTED	110
CMERROR_NO_NETWORK_SERVER	111
CMERROR_BUFFER_OVERFLOW	112
CMERROR_BAD_ADDRESS	113
CMERROR_BAD_HANDLE	114
CMERROR_WRONG_HANDLE_TYPE	115
CMERROR_UNDEFINED_SYSTEM_HANDLE	116
CMERROR_NO_EVENT_OCCURRED	117
CMERROR_CTCS_FAULT	118
CMERROR_UNKNOWN_OS	119
CMERROR_NO_SECURITY_OBJECT	120
CMERROR_WRONG_SECURITY_OBJECT	121
CMERROR_NO_FSB_FUNCTIONALITY	122
CMERROR_NO_CTSB_FUNCTIONALITY	123
CMERROR_WRONG_OEMID	124
CMERROR_SERVER_VERSION_TOO_OLD	125
CMERROR_SUBSYSTEM_NOT_AVAILABLE (<4.0)	126
CMERROR_LIBRARY_NOT_FOUND (since 4.0)	126

© Copyright 2018 WIBU-SYSTEMS AG

Error Name	Error Code
CMERROR_HANDLE_INCONSISTENCY	127
CMERROR_LICENSE_TYPE_CMACT_DEACTIVATED	128
CMERROR_USER_RUNTIME_VERSION_TOO_OLD	129
CMERROR_REDUNDANCY_SERVER_MISSING	130
CMERROR_TRANSFER_BUFFER_OVERFLOW	131
CMERROR_NAMEDUSERCREDENTIALSWRONG	132
CMERROR_BOX_COMMUNICATION	176
CMERROR_BOX_COMMUNICATION	177
CMERROR_BOX_COMMUNICATION	178
CMERROR_BOX_COMMUNICATION	179
CMERROR_BOX_COMMUNICATION	180
CMERROR_BOX_COMMUNICATION	181
CMERROR_BOX_COMMUNICATION	182
CMERROR_BOX_COMMUNICATION	183
CMERROR_BOX_COMMUNICATION	184
CMERROR_BOX_COMMUNICATION	185
CMERROR_BOX_COMMUNICATION	186
CMERROR_BOX_COMMUNICATION	187
CMERROR_BOX_COMMUNICATION	188
CMERROR_BOX_COMMUNICATION	189
CMERROR_BOX_COMMUNICATION	190
CMERROR_BOX_COMMUNICATION	191
CMERROR_I2C_TIMEOUT_REQUEST	192
CMERROR_I2C_TIMEOUT_ANSWER	193
CMERROR_I2C_ANSWER_CRC	194
CMERROR_I2C_TWCOMMAND_INVALID	195
CMERROR_I2C_PROTOCOL_VERSION	196
CMERROR_I2C_COMMUNICATION	197
CMERROR_I2C_REQUEST_CRC	198
CMERROR_I2C_ANSWER_BEFORE_REQUEST	199
CMERROR_ENTRY_NOT_FOUND	200
CMERROR_BOX_NOT_FOUND	201
CMERROR_CRYPTION_FAILED	202
CMERROR_CRC_VERIFY_FAILED	203
CMERROR_NO_ENABLING_EXISTS	204
CMERROR_NO_PIO_SET	205
CMERROR_FIRMKEY_TOO_SHORT	206
CMERROR_NO_BOXLOCK_NEEDED	207
CMERROR_BOXLOCK_NOT_CHANGED	208
CMERROR_NO_DATA_AVAILABLE	209
CMERROR_SIGNEDLIST_FAILED	210
CMERROR_VALIDATION_FAILED	211

© Copyright 2018 WIBU-SYSTEMS AG

Error Name	Error Code
CMERROR_NO_MORE_LICENSES	212
CMERROR_EXCLUSIVE_MODE_CONFLICT	213
CMERROR_RESERVEDFI_MISSED	214
CMERROR_WRONG_CONTENT	215
CMERROR_FIELDUPDATE_FAILED	216
CMERROR_LICENSEFILE_CREATION_FAILED	217
CMERROR_NO_LICENSE_AVAILABLE	218
CMERROR_CONTEXT_FILE_WRITING_FAILED	219
CMERROR_UPDATE_FILE_READING_FAILED	220
CMERROR_NETINFO_FAILED	221
CMERROR_FIELDUPDATE_RESET_FAILED	222
CMERROR_WRONG_ACCESS_MODE	223
CMERROR_WRONG_BOX_VERSION	224
CMERROR_INVALID_LICENSE_PARAMETER	225
CMERROR_BORROW_LICENSE_FAILED	226
CMERROR_BORROW_LICENSE_ISENABLED	227
CMERROR_UPDATE_TOO_NEW	228
CMERROR_UPDATE_TOO_OLD	229
CMERROR_SEVERAL_REASONS	230
CMERROR_ACCESS_DENIED	231
CMERROR_BORROW_TIMEDIFFTOOLARGE	232
CMERROR_UPDATE_FAILED	233
CMERROR_UPDATE_ABORTED	234
CMERROR_COMMAND_PENDING	235
CMERROR_NO_WAN_LICENSE	236
CMERROR_CMWAN_CREDENTIALS_CHANGED	237
CMERROR_SERVER_START_PENDING	238
CMERROR_REMOTE_ACCESS_DENIED	239
CMERROR_FILE_NOT_FOUND	240
CMERROR_CHIP_INCOMPATIBLE	241
CMERROR_FIELDUPDATE_DONGLE_FILE_MISFIT	242
CMERROR_FIELDUPDATE_INCOMPLETE	243
CMERROR_INTERNAL	254
CMERROR_UNKNOWN	255
CMERROR_CMACT_SUBSYSTEM_FAILED	260
CMERROR_CMACT_LICENSE_NOT_ACTIVATED	261
CMERROR_CMACT_LICENSE_ALREADY_ACTIVATED	262
CMERROR_CMACT_LICENSE_REACTIVATION_REQUIRED	263
CMERROR_CMACT_LICENSE_INVALID	264
CMERROR_CMACT_COMMAND_NOT_SUPPORTED	265

© Copyright 2018 WIBU-SYSTEMS AG

Error Name	Error Code
CMERROR_CMACT_LICENSE_ALREADY_EXISTS	266
CMERROR_CMACT_RUN_AS_SERVICE_REQUIRED	267
CMERROR_CMACT_BINDING_NOT_POSSIBLE	268
CMERROR_CMACT_OPEN_FIRMITEM_TEMPLATE_FAILED	269
CMERROR_CMACT_OSVERSION_NOT_PERMITTED	270
CMERROR_CMACT_HOST_ID_NOT_AVAILABLE	271
CMERROR_CMACT_PLUGIN_MALFUNCTION	272
CMERROR_CMACT_VIRTUAL_MACHINE_NOT_ALLOWED	273
CMERROR_INCOMPATIBLE_SYSTEM_SOFTWARE_FOUND	274
CMERROR_ERRORTEXT_NOT_FOUND	300
CMERROR_SEND_PROTECTION_FAILED	301
CMERROR_ANALYSING_DETECTED	302
CMERROR_COMMAND_EXECUTION_FAILED	303
CMERROR_FUNCTION_NOT_IMPLEMENTED	304
CMERROR_OBSOLETE_FUNCTION	305
CMERROR_FUNCTION_NOT_SUPPORTED	306
CMERROR_EXCEPTION_CAUGHT	307
CMERROR_VERIFY_FAILED	308
CMERROR_REQUEST_OVERLOAD_REJECTED	309
CMERROR_TIME_CONVERSION	310
CMERROR_OPERATION_NOT_SUPPORTED_FOR_UNIVERSAL_LICENSE	311
CMERROR_INTERNAL_COMMAND_FAILED	312
CMERROR_CM_LIB_NOT_INITIALIZED	313
CMERROR_BOX_ASN1_DECODER_FAILED	700
CMERROR_BOX_ASN1_SIGNATURE_FAILED	701
CMERROR_BOX_ASN1_ENCRYPT_DECRYPT_FAILED	702
CMERROR_BOX_ASN1_INPUT_BUFFER_MISMATCH	703
CMERROR_BOX_LTUC_WRONG	704
CMERROR_BOX_LTSM_ERROR	705
CMERROR_BOX_LT_ONLY_RESTRICTED_BCIS_ALLOWED	706
CMERROR_BOX_MAX_ENC_RATE_REACHED	711
CMERROR_BOX_ENC_RATE_NO_RESOURCES	712

© Copyright 2018 WIBU-SYSTEMS AG

Error Name	Error Code
CMERROR_DONGLE_FUNCTIONS_RESTRICTED	713
CMERROR_BOX_INTEGRITY_ERROR	730
CMERROR_BOX_INTEGRITY_WARNING	731
CMERROR_BOX_REPORT1_TYPE2_UNSUPPORTED	780
CMERROR_LT_DOTRANSFER_INSUFFICIENT_NUMBER_OF_UNITS	3491496034
CMERROR_LT_DOTRANSFER_INSUFFICIENT_NUMBER_OF_LICENSES	3491496035
CMERROR_LT_DOTRANSFER_TRANSFER_DEPTH	3491496038
CMERROR_LT_DOTRANSFER_TARGET_CMACT_ID_NOT_ALLOWED	3491496039
CMERROR_LT_DOTRANSFER_TRANSFER_TYPE_NOT_ALLOWED	3491496040
CMERROR_LT_DOTRANSFER_FI_TRANSFER_NOT_ALLOWED	3491496044
CMERROR_LT_DOTRANSFER_BORROW_EXPIRATION_TIME_TOO_LARGE	3491496045
CMERROR_LT_DOTRANSFER_BORROW_EXPIRATION_TIME_TOO_SMALL	3491496046
CMERROR_LT_DOTRANSFER_ALREADY_COMPLETED	3491496047
CMERROR_LT_DOTRANSFER_CREATE_RAU_FAILED	3491496051
CMERROR_LT_DOTRANSFER_INVALID_RAC_CONTENT	3491496052
CMERROR_LT_RAR_ACT_RECEIPT_CREATION_FAILED	3491496194
CMERROR_LT_STATUS_DISABLED	3491496209
CMERROR_LT_STATUS_DELETED	3491496210
CMERROR_LT_LICENSEQUANTITY_TOO_SMALL	3491496212
CMERROR_LT_NO_PERMISSION_RETURN	3491496273

© Copyright 2018 WIBU-SYSTEMS AG

5 Knowledge Base

C, C++ Programming Language
Access Modes
Encryption Modes
Little/Big Endian Conversion
License Quantity
Usage Period

5.1 Access Modes

A protected application can access the CodeMeter Embedded using one of the hierarchically structured types of access:

- computer (*Sub System Access*)
- CmContainer (*Box Access*)
- single Firm Item (FC, *Firm Item Access*) or a
- CmContainer entry (FC:PC, *Entry Access*) in a given subsystem.

Please note the existing hierarchy: a *Sub System Access* represents a superset of *Box Access*. A *Box Access* is a superset of *Firm Item Access*. A *Firm Item Access* is a superset of an *Entry Access*. Anything that can be done with a *Sub System Access* can also be done with *Box Access*, but not vice versa!

C, C++ Programming Language

Access Mode	Description
Subsystem Access	*Subsystem Access* accesses a single Subsystem. No CmContainer has to be connected to this subsystem. The only requirement is that CodeMeter License Server (e.g. CodeMeter.exe) is running at this subsystem. *Subsystem Access* requires the following settings: • Firm Code must be set to 0 and the CM_ACCESS_SUBSYSTEM flag must be set at the or CMACCESS2 Ctrl parameter.
Box Access	*Box Access* accesses a single CmContainer. In this case only CmContainer specific tasks can be performed. *Box Access* requires the following settings: • If Firm Code = 0 and Product Code = 0, and the box mask or the serial number are not set, then the access to the first found CmContainer at the given subsystem is done. • If Firm Code = 0 and Product Code = 0, and the box mask and the serial number are set in the or CMACCESS2 BoxInfo parameter, the specified CmContainer is accessed. **Important Note**: Accessing Firm Item/Product Item 0:0 fails over the network subsystem due to security reasons. However, due to the hierarchical access structure, you may alternatively use a *Firm Item Access* or *Entry Access*.
Firm Item Access	*Firm Item Access* accesses a single Firm Item. This access mode is, for example, needed for programming a new Product Item into the CmContainer. Access to Firm Item 0 can only be performed locally. *Firm Item Access* requires the following settings: • If Product Code = 0, the access to the first found Firm Item in all connected CmContainer at the given subsystem is performed. Additionally please set CM_ACCESS_FIRMITEM in the CMACCESS2 Ctrl parameter to force Firm Item access. • If Product Code = 0 and a serial number is set in the or CMACCESS2 BoxInfo parameter, then the entry is searched in the specified CmContainer only. Additionally, please set CM_ACCESS_FIRMITEM in the CMACCESS Ctrl/ CMACCESS2 Ctrl parameter to force Firm Item access.

© Copyright 2018 WIBU-SYSTEMS AG

Access Mode	Description
Entry Access	*Entry Access* accesses a single CmContainer Entry. In this case all entry specific tasks can be performed. Access to entries in Firm Item 0 can only be performed locally. *Entry Access* can be done using the following settings: • If Product Code != 0, the access to the first found matching entry in all connected CmContainer at the given subsystem is done. • If Product Code != 0 and a serial number is set in the / CMACCESS2 BoxInfo parameter, the entry is searched in the specified CmContainer.

5.2 Encryption Modes

The following encryption modes exist:

C, C++ Programming Language

Encryption mode	Description
Direct encryption	The direct encryption encrypts or decrypts a sequence with a fixed length (usually 16 bytes) directly in the CmContainer. Use *CmCrypt2* with the flags `CM_CRYPT_DIRECT_`...
Indirect encryption	Use *CmCrypt2* with the flags `CM_CRYPT_STREAM`, `CM_CRYPT_AES_`... but not `CM_CRYPT_AES_ENC_CBC_DIRECT` or `CM_CRYPT_AES_DEC_CBC_DIRECT`
Software encryption	The software encryption encrypts or decrypts sequence **without** using any CmContainer. This implements standard AES encryption with cipher block chaining (CBC) mode. Two parameters influence the encryption result: the AES key and the initialization vector (IV) (both see CMCRYPT2). Use *CmCrypt2* with the flags `CM_CRYPT_AES_ENC_CBC_DIRECT` or `CM_CRYPT_AES_DEC_CBC_DIRECT`. The software encryption is supported since 4.30 and is implemented in the library part of CodeMeter (dll).

5.3 Little/Big Endian Conversion

C, C++ Programming Language

Description
To swap the byte order from Little Endian to Big Endian and vice versa this can be done like this: • 16 Bit value (unsigned short): `x = ((y & 0xff) << 8)	((y & 0xff00)>> 8)` • 32 Bit value (unsigned int): `x = ((y & 0xff) << 24)	((y & 0xff00) << (8 0xff0000) >> 8)	((y & 0xff000000) >> 24)`

5.4 License Quantity

Description
The License Quantity is an option of a Product Item. It defines the number of licenses that are available for this Product Item.

© Copyright 2018 WIBU-SYSTEMS AG

Description

The License Quantity is stored in the Extended Protected Data entry with Extended Type 128. The length of data has to be 16 bytes:

- **Byte 0..3:** The first 4 bytes contain the signature **"cmnw"** (ASCII: 99,109,110,119).
- **Byte 4..7:** These 4 bytes contain the License Quantity value in little endian order.
- **Byte 8..11:** These 4 bytes contain the access restrictions. For local access these flags don't have any effect. Bit definition:
  - Bit 0 (Byte 8): If this flag is set, no UserLimit access via network is allowed to this entry.
  - Bit 1 (Byte 8): If this flag is set, no NoUserLimit access via network is allowed to this entry.
  - Bit 2 (Byte 8): If this flag is set, no Exclusive access via network is allowed to this entry.
  - Bit 3 (Byte 8): If this flag is set, no StationShare access via network is allowed to this entry.
- **Byte 12..15:** These 4 bytes are reserved for future purposes.

Further description:

- If no License Quantity is set (the Extended Protected Data of type 128 does not exist), CodeMeter License Server automatically provides one license.
- If the bits in Byte 8 are not set, there is no restriction in the access to this entry via network.
- If the License Quantity is set to a value of 0, the access is only allowed from the local system. Any access over network is prohibited.

5.5 Usage Period

Description

The *Usage Period* is an option of a Product Item. It defines a time period, for example, 30 days, for which this Product Item can be used. The Usage Period is activated when the first encryption is done using this entry. From this moment Usage Period behaves as an Expiration Time.

Usage Period is available for CmContainer Firmware 1.11 and newer.

C, C++ Programming Language

Programming of the Usage Period:

For programming CmContainer please use High Level Programming API (HIP) or CmBoxPgm.

Activating the Usage Period:

The Usage Period is activated by calling *CmCrypt2()*. The Usage Period will not be activated, if `CM_CRYPT_ETIGNORE` was set.

Reading the Usage Period:

The Usage Period can be read by calling `CmGetInfo/CM_GEI_ENTRYDATA` which will fill a `CMENTRYDATA` structure. To detect an Usage Period use `CM_GF_USAGEPERIOD`. For easier handling the `CMENTRYDATA` content can be assigned to a `CMUSAGEPERIOD` structure that contains all necessary information.

© Copyright 2018 WIBU-SYSTEMS AG

6 Types, macros and helper functions

C, C++ Programming Language

Base types

CMUINT

Description

This type represents an unsigned 32-bit integer value.

CMULONG

Description

This type represents an unsigned 32-bit integer value.

CMUSHORT

Description

This type represents an unsigned 16-bit value.

CMBYTE

Description

This type represents an unsigned 8-bit value. It is declared as an unsigned char.

Handles

HCMSysEntry

Description

HCMSysEntry is a 32-bit value which can be seen as a handle that points to a Product Item, a Firm Item, a CmContainer or a subsystem. Such a handle can be created by *CmAccess2*.

Macros

CM_SET_VERSION

Description

`CM_SET_VERSION(a, b, c)` sets the complete version in a `CMULONG` value.

- `a`: sets the major version.
- `b`: sets the minor version.
- `c`: sets the build version.

Syntax

```
#define CM_SET_VERSION(a,b,c) (((a) & 0xff) \
 (((a) & 0xff) << 24) | (((b) & 0xff) << 16) | ((c) & 0xffff)
```

CM_GET_MAJOR_VERSION

Description

`CM_GET_MAJOR_VERSION(v)` gets major version from a complete version value.

- `v`: complete version value.

Syntax

```
#define CM_GET_MAJOR_VERSION(v) (((v) >> 24) & 0xff)
```

CM_GET_MINOR_VERSION

© Copyright 2018 WIBU-SYSTEMS AG

Macros

Description

`CM_GET_MINOR_VERSION(v)` gets minor version from a complete version value.

- `v`: complete version value.

Syntax

```
#define CM_GET_MINOR_VERSION(v) (((v) >> 16) & 0xff)
```

CM_GET_BUILD_VERSION

Description

`CM_GET_BUILD_VERSION(v)` gets the build version from a complete version value.

- `v`: complete version value.

Syntax

```
#define CM_GET_BUILD_VERSION(v) ((v) & 0xffff)
```

Helper functions

CmConvertTime

Description

CmConvertTime as a helper function converts calendar date specifications into seconds since 2000-01-01T00:00 and vice versa.

Syntax

```
CODEMETER_API int CMAPIENTRY CmConvertTime(CMULONG flCtrl, CMTIME *pcmtime);
```

Parameter

flCtrl specifies the following flags.

- *CM_CONVERT_SECONDS_TO_CALENDAR* specifies that the time is converted from "seconds since 2000" to a calender date.
- *CM_CONVERT_CALENDAR_TO_SECONDS* specifies that the the time is converted from a calender date to "seconds since 2000".

CMTIME contains a given CMTIME structure as part of API calls, for example, *CmAccess2*, etc.

Structure CMTIME

CMTIME sets or gets the time to or from a CmContainer.

Syntax

```
typedef struct __CMTIME {
    CMUSHORT musYear;
    CMUSHORT musMonth;
    CMUSHORT musDay;
    CMUSHORT musHours;
    CMUSHORT musMinutes;
    CMUSHORT musSeconds;
    CMULONG mulSecondsSince01_01_2000;
} CMTIME;
```

Parameter

musYear specifies the value of the year (e.g. `2003`).
musMonth specifies the value of the month (`1..12`).
musDay specifies the value of the day (`1..31`).

© Copyright 2018 WIBU-SYSTEMS AG

Helper functions

musHours specifies the value of the hours (`0..23`).
musMinutes specifies the value of the minutes (`0..59`).
musSeconds specifies the value of the seconds (`0..59`).
mulSecondsSince01_01_2000 specifies the time in seconds since `01/01/2000`.

pcmtime contains the conversion time result according to the specified flag.

Return

If *`CM_CONVERT_SECONDS_TO_CALENDAR`* is specified, *pcmtime holds valid amount of seconds (unequal 0) since 2000-01-01T00:00.

If *`CM_CONVERT_CALENDAR_TO_SECONDS`* is specified, *pcmtime holds valid calender date.

© Copyright 2018 WIBU-SYSTEMS AG

7 Version / Contact / Open Source Components

***CodeMeter* CmEmbedded API Help Version 2.30, 10/30/2018**
Module *CmActLicense* Module *CmDongle* Module LAN Module Runtime Bridge
©Copyright 2003-2018 by WIBU-SYSTEMS AG, Karlsruhe / Germany
Wibu-Systems contact information:

Address:	WIBU-SYSTEMS AG Rueppurrer Strasse 52-54 D-76137 Karlsruhe / Germany
Phone:	+49 (0) 721 93172-0
Fax:	+49 (0) 721 93172-22
Internet:	http://www.wibu.com
Email:	support@wibu.com

Flot

Project	Runtime component CodeMeter WebAdmin (CodeMeter License Server)
Version	0.8.1 (flot)
Operating system	Windows, macOS, Linux

```
Copyright (c) 2007-2013 IOLA and Ole Laursen

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.
```

JQuery

Project	Runtime component CodeMeter WebAdmin (CodeMeter License Server)
Version	v1.11.3 (jQuery)
Operating system	Windows, macOS, Linux

```
Copyright jQuery Foundation and other contributors, https://jquery.org/

This software consists of voluntary contributions made by many
individuals. For exact contribution history, see the revision history
available at https://github.com/jquery/jquery-ui

The following license applies to all parts of this software except as
documented below:

====
```


© Copyright 2018 WIBU-SYSTEMS AG

```
Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample
code is defined as all source code contained within the demos directory.

CC0: http://creativecommons.org/publicdomain/zero/1.0/

====
All files located in the node_modules and external directories are
externally maintained libraries used by this software which have their
own licenses; we recommend you read them, as their terms may differ from
the terms above.
```

go-macaron framework

Project	Runtime component CodeMeter WebAdmin (CodeMeter License Server)
Version	1.8.3 (go)
Operating System	Windows, macOS, Linux

```
go-macaron/macaron is licensed under Apache License 2.0

How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate
notice, with the fields enclosed by brackets "[]" replaced with your own
identifying information. (Don't include the brackets!) The text should be
enclosed in the appropriate comment syntax for the file format. We also
recommend that a file or class name and description of purpose be included on
the same "printed page" as the copyright notice for easier identification within
third-party archives.

   Copyright 2014 The Macaron Authors

   Licensed under the Apache License, Version 2.0 (the "License");
   you may not use this file except in compliance with the License.
   You may obtain a copy of the License at

     http://www.apache.org/licenses/LICENSE-2.0

   Unless required by applicable law or agreed to in writing, software
   distributed under the License is distributed on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
   See the License for the specific language governing permissions and
limitations under the License.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
```

© Copyright 2018 WIBU-SYSTEMS AG

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory

© Copyright 2018 WIBU-SYSTEMS AG

patent infringement, then any patent licenses granted to You under this License
for that Work shall terminate as of the date such litigation is filed.

4. Redistribution.

You may reproduce and distribute copies of the Work or Derivative Works thereof
in any medium, with or without modifications, and in Source or Object form,
provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of
this License; and
You must cause any modified files to carry prominent notices stating that You
changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute,
all copyright, patent, trademark, and attribution notices from the Source form
of the Work, excluding those notices that do not pertain to any part of the
Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any
Derivative Works that You distribute must include a readable copy of the
attribution notices contained within such NOTICE file, excluding those notices
that do not pertain to any part of the Derivative Works, in at least one of the
following places: within a NOTICE text file distributed as part of the
Derivative Works; within the Source form or documentation, if provided along
with the Derivative Works; or, within a display generated by the Derivative
Works, if and wherever such third-party notices normally appear. The contents of
the NOTICE file are for informational purposes only and do not modify the
License. You may add Your own attribution notices within Derivative Works that
You distribute, alongside or as an addendum to the NOTICE text from the Work,
provided that such additional attribution notices cannot be construed as
modifying the License.
You may add Your own copyright statement to Your modifications and may provide
additional or different license terms and conditions for use, reproduction, or
distribution of Your modifications, or for any such Derivative Works as a whole,
provided Your use, reproduction, and distribution of the Work otherwise complies
with the conditions stated in this License.

5. Submission of Contributions.

Unless You explicitly state otherwise, any Contribution intentionally submitted
for inclusion in the Work by You to the Licensor shall be under the terms and
conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of
any separate license agreement you may have executed with Licensor regarding
such Contributions.

6. Trademarks.

This License does not grant permission to use the trade names, trademarks,
service marks, or product names of the Licensor, except as required for
reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.

7. Disclaimer of Warranty.

Unless required by applicable law or agreed to in writing, Licensor provides the
Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied,
including, without limitation, any warranties or conditions of TITLE,
NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are
solely responsible for determining the appropriateness of using or
redistributing the Work and assume any risks associated with Your exercise of
permissions under this License.

8. Limitation of Liability.

In no event and under no legal theory, whether in tort (including negligence),
contract, or otherwise, unless required by applicable law (such as deliberate
and grossly negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special, incidental,
or consequential damages of any character arising as a result of this License or
out of the use or inability to use the Work (including but not limited to
damages for loss of goodwill, work stoppage, computer failure or malfunction, or
any and all other commercial damages or losses), even if such Contributor has
been advised of the possibility of such damages.

© Copyright 2018 WIBU-SYSTEMS AG

```
9. Accepting Warranty or Additional Liability.

While redistributing the Work or Derivative Works thereof, You may choose to
offer, and charge a fee for, acceptance of support, warranty, indemnity, or
other liability obligations and/or rights consistent with this License. However,
in accepting such obligations, You may act only on Your own behalf and on Your
sole responsibility, not on behalf of any other Contributor, and only if You
agree to indemnify, defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason of your
accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS
```

golang.org (/x/sys / and /x/crypto)

Project	Runtime component CodeMeter WebAdmin (CodeMeter License Server)
Version	1.8.3 (go)
Operating System	Windows, macOS, Linux

```
Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

libusb Linux

Project	Runtime component Linux - CodeMeter License Server
Version	1.0-1.0.20
Operating system	Linux

The programming library serves to read and write USB devices on Linux.

```
                  GNU LESSER GENERAL PUBLIC LICENSE
                       Version 2.1, February 1999

 Copyright (C) 1991, 1999 Free Software Foundation, Inc.
 51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA
 Everyone is permitted to copy and distribute verbatim copies
 of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL.  It also counts
 as the successor of the GNU Library Public License, version 2, hence
 the version number 2.1.]

                            Preamble
```


© Copyright 2018 WIBU-SYSTEMS AG

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to

© Copyright 2018 WIBU-SYSTEMS AG

encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1

© Copyright 2018 WIBU-SYSTEMS AG

above, provided that you also meet all of these conditions:

    a) The modified work must itself be a software library.

    b) You must cause the files modified to carry prominent notices
    stating that you changed the files and the date of any change.

    c) You must cause the whole of the work to be licensed at no
    charge to all third parties under the terms of this License.

    d) If a facility in the modified Library refers to a function or a
    table of data to be supplied by an application program that uses
    the facility, other than as an argument passed when the facility
    is invoked, then you must make a good faith effort to ensure that,
    in the event an application does not supply such function or
    table, the facility still operates, and performs whatever part of
    its purpose remains meaningful.

    (For example, a function in a library to compute square roots has
    a purpose that is entirely well-defined independent of the
    application. Therefore, Subsection 2d requires that any
    application-supplied function or table used by this function must
    be optional: if the application does not supply it, the square
    root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

  3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

  Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

  This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

  4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

  If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

© Copyright 2018 WIBU-SYSTEMS AG

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

© Copyright 2018 WIBU-SYSTEMS AG

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

© Copyright 2018 WIBU-SYSTEMS AG

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the

© Copyright 2018 WIBU-SYSTEMS AG

```
ordinary General Public License).

  To apply these terms, attach the following notices to the library.  It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
"copyright" line and a pointer to where the full notice is found.

    <one line to give the library's name and a brief idea of what it does.>
    Copyright (C) <year>  <name of author>

    This library is free software; you can redistribute it and/or
    modify it under the terms of the GNU Lesser General Public
    License as published by the Free Software Foundation; either
    version 2.1 of the License, or (at your option) any later version.

    This library is distributed in the hope that it will be useful,
    but WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
    Lesser General Public License for more details.

    You should have received a copy of the GNU Lesser General Public
    License along with this library; if not, write to the Free Software
    Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary.  Here is a sample; alter the names:
  Yoyodyne, Inc., hereby disclaims all copyright interest in the
  library `Frob' (a library for tweaking knobs) written by James Random Hacker.

  <signature of Ty Coon>, 1 April 1990
  Ty Coon, President of Vice

That's all there is to it!
```

The code of the programming library **libusb** can be downloaded at the project website http://www.libusb.org/. On request the source can also be electronically obtained free of charge contacting Wibu-Systems (support@wibu.com).

libcurl

Project	Runtime component CodeMeter License Server
Version	7.28.0
Operating system	Windows, macOS, Linux

```
COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2017, Daniel Stenberg, daniel@haxx.se, and many contributors, see the THANKS
file.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee
is hereby granted, provided that the above copyright notice and this permission notice appear in
all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT
NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE
FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising
or otherwise to promote the sale, use or other dealings in this Software without prior written
authorization of the copyright holder.
```

© Copyright 2018 WIBU-SYSTEMS AG

Google Protocol Buffers

Project	Runtime component	SDK component AxProtector
Version	2.6.1	2.6.1
Operating system	Windows, macOS, Linux	Windows, macOS, Linux

```
This license applies to all parts of Protocol Buffers except the following:

  - Atomicops support for generic gcc, located in
    src/google/protobuf/stubs/atomicops_internals_generic_gcc.h.
    This file is copyrighted by Red Hat Inc.

  - Atomicops support for AIX/POWER, located in
    src/google/protobuf/stubs/atomicops_internals_power.h.
    This file is copyrighted by Bloomberg Finance LP.

Copyright 2014, Google Inc.  All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

    * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
    * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
    * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it.  This code is not
standalone and requires a support library to be linked with it.  This
support library is itself covered by the above license.
```

ASN.1 Compiler

Project	Runtime component CodeMeter License Server	SDK component Programming API (HIP)
Version	1.0	1.0
Operating system	Windows, macOS, Linux	Windows, macOS, Linux

```
Copyright (c) 2003-2016  Lev Walkin <vlm@lionet.info> and contributors.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
```


© Copyright 2018 WIBU-SYSTEMS AG

```
   documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED.  IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
```

Bootstrapper dotnetInstaller

Project	Runtime component dotnetInstaller	SDK component dotnetInstaller
Version	2.3	2.3
Operating system	Windows	Windows

```
MIT License (MIT)

Copyright (c) 2009-2012 Davide Icardi, Daniel Doubrovkine and Contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and
associated documentation files (the "Software"), to deal in the Software without restriction,
including without limitation the rights to use, copy, modify, merge, publish, distribute,
sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or
substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT
NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT
OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

LLVM compiler and toolchain technology

Project	SDK component AxProtector
Version	3.8.0
Operating system	Windows, macOS, Linux

```
==============================================================================
LLVM Release License
==============================================================================
University of Illinois/NCSA
Open Source License

Copyright (c) 2003-2015 University of Illinois at Urbana-Champaign.
All rights reserved.

Developed by:

    LLVM Team

    University of Illinois at Urbana-Champaign

    http://llvm.org

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal with
the Software without restriction, including without limitation the rights to
```

© Copyright 2018 WIBU-SYSTEMS AG

```
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

    * Redistributions of source code must retain the above copyright notice,
      this list of conditions and the following disclaimers.

    * Redistributions in binary form must reproduce the above copyright notice,
      this list of conditions and the following disclaimers in the
      documentation and/or other materials provided with the distribution.

    * Neither the names of the LLVM Team, University of Illinois at
      Urbana-Champaign, nor the names of its contributors may be used to
      endorse or promote products derived from this Software without specific
      prior written permission.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.  IN NO EVENT SHALL THE
CONTRIBUTORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS WITH THE
SOFTWARE.

==============================================================================
Copyrights and Licenses for Third Party Software Distributed with LLVM:
==============================================================================
The LLVM software contains code written by third parties.  Such software will
have its own individual LICENSE.TXT file in the directory in which it appears.
This file will describe the copyrights, license, and restrictions which apply
to that code.

The disclaimer of warranty in the University of Illinois Open Source License
applies to all code in the LLVM Distribution, and nothing in any of the
other licenses gives permission to use the names of the LLVM Team or the
University of Illinois to endorse or promote products derived from this
Software.

The following pieces of software have additional or alternate copyrights,
licenses, and/or restrictions:

Program             Directory
-------             ---------
Autoconf            llvm/autoconf
                    llvm/projects/ModuleMaker/autoconf
Google Test         llvm/utils/unittest/googletest
OpenBSD regex       llvm/lib/Support/{reg*, COPYRIGHT.regex}
pyyaml tests        llvm/test/YAMLParser/{*.data, LICENSE.TXT}
ARM contributions   llvm/lib/Target/ARM/LICENSE.TXT
md5 contributions   llvm/lib/Support/MD5.cpp llvm/include/llvm/Support/MD5.h
```

ASM Java Program Library

Project	SDK component AxProtector
Version	5.1
Operating system	Windows, macOS, Linux

```
Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
```


© Copyright 2018 WIBU-SYSTEMS AG

```
3. Neither the name of the copyright holders nor the names of its
   contributors may be used to endorse or promote products derived from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
THE POSSIBILITY OF SUCH DAMAGE.
```

Apache Commons

Project	SDK component AxProtector
Version	cli 1.3.1, io 2.4, lang 3-3.4
Operating system	Windows, macOS, Linux

```
Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

   1. Definitions.

      "License" shall mean the terms and conditions for use, reproduction,
      and distribution as defined by Sections 1 through 9 of this document.

      "Licensor" shall mean the copyright owner or entity authorized by
      the copyright owner that is granting the License.

      "Legal Entity" shall mean the union of the acting entity and all
      other entities that control, are controlled by, or are under common
      control with that entity. For the purposes of this definition,
      "control" means (i) the power, direct or indirect, to cause the
      direction or management of such entity, whether by contract or
      otherwise, or (ii) ownership of fifty percent (50%) or more of the
      outstanding shares, or (iii) beneficial ownership of such entity.

      "You" (or "Your") shall mean an individual or Legal Entity
      exercising permissions granted by this License.

      "Source" form shall mean the preferred form for making modifications,
      including but not limited to software source code, documentation
      source, and configuration files.

      "Object" form shall mean any form resulting from mechanical
      transformation or translation of a Source form, including but
      not limited to compiled object code, generated documentation,
      and conversions to other media types.

      "Work" shall mean the work of authorship, whether in Source or
      Object form, made available under the License, as indicated by a
      copyright notice that is included in or attached to the work
      (an example is provided in the Appendix below).

      "Derivative Works" shall mean any work, whether in Source or Object
      form, that is based on (or derived from) the Work and for which the
      editorial revisions, annotations, elaborations, or other modifications
      represent, as a whole, an original work of authorship. For the purposes
```

© Copyright 2018 WIBU-SYSTEMS AG

of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed

© Copyright 2018 WIBU-SYSTEMS AG

as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

© Copyright 2018 WIBU-SYSTEMS AG

```
      To apply the Apache License to your work, attach the following
      boilerplate notice, with the fields enclosed by brackets "[]"
      replaced with your own identifying information. (Don't include
      the brackets!)  The text should be enclosed in the appropriate
      comment syntax for the file format. We also recommend that a
      file or class name and description of purpose be included on the
      same "printed page" as the copyright notice for easier
      identification within third-party archives.

   Copyright [yyyy] [name of copyright owner]

   Licensed under the Apache License, Version 2.0 (the "License");
   you may not use this file except in compliance with the License.
   You may obtain a copy of the License at

       http://www.apache.org/licenses/LICENSE-2.0

   Unless required by applicable law or agreed to in writing, software
   distributed under the License is distributed on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
   See the License for the specific language governing permissions and
   limitations under the License.
```

Public Suffix List

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Version	https://publicsuffix.org/
Operating System	Windows, macOS, Linux

```
Mozilla Public License Version 2.0
==================================

1. Definitions
--------------

1.1. "Contributor"
    means each individual or legal entity that creates, contributes to
    the creation of, or owns Covered Software.

1.2. "Contributor Version"
    means the combination of the Contributions of others (if any) used
    by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
    means Covered Software of a particular Contributor.

1.4. "Covered Software"
    means Source Code Form to which the initial Contributor has attached
    the notice in Exhibit A, the Executable Form of such Source Code
    Form, and Modifications of such Source Code Form, in each case
    including portions thereof.

1.5. "Incompatible With Secondary Licenses"
    means

    (a) that the initial Contributor has attached the notice described
        in Exhibit B to the Covered Software; or

    (b) that the Covered Software was made available under the terms of
        version 1.1 or earlier of the License, but not also under the
        terms of a Secondary License.

1.6. "Executable Form"
    means any form of the work other than Source Code Form.

1.7. "Larger Work"
    means a work that combines Covered Software with other material, in
    a separate file or files, that is not Covered Software.
```


© Copyright 2018 WIBU-SYSTEMS AG

1.8. "License"
    means this document.

1.9. "Licensable"
    means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"
    means any of the following:

    (a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

    (b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor
    means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"
    means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"
    means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
    means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License.

© Copyright 2018 WIBU-SYSTEMS AG

Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice,

© Copyright 2018 WIBU-SYSTEMS AG

provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

****
*   *
****

© Copyright 2018 WIBU-SYSTEMS AG

```
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is"       *
*  basis, without warranty of any kind, either expressed, implied, or  *
*  statutory, including, without limitation, warranties that the       *
*  Covered Software is free of defects, merchantable, fit for a        *
*  particular purpose or non-infringing. The entire risk as to the     *
*  quality and performance of the Covered Software is with You.        *
*  Should any Covered Software prove defective in any respect, You     *
*  (not any Contributor) assume the cost of any necessary servicing,   *
*  repair, or correction. This disclaimer of warranty constitutes an   *
*  essential part of this License. No use of any Covered Software is   *
*  authorized under this License except under this disclaimer.         *
*                                                                      *
************************************************************************

************************************************************************
*                                                                      *
*  7. Limitation of Liability                                          *
*  --------------------------                                          *
*                                                                      *
*  Under no circumstances and under no legal theory, whether tort      *
*  (including negligence), contract, or otherwise, shall any           *
*  Contributor, or anyone who distributes Covered Software as          *
*  permitted above, be liable to You for any direct, indirect,         *
*  special, incidental, or consequential damages of any character      *
*  including, without limitation, damages for lost profits, loss of    *
*  goodwill, work stoppage, computer failure or malfunction, or any    *
*  and all other commercial damages or losses, even if such party      *
*  shall have been informed of the possibility of such damages. This   *
*  limitation of liability shall not apply to liability for death or   *
*  personal injury resulting from such party's negligence to the       *
*  extent applicable law prohibits such limitation. Some               *
*  jurisdictions do not allow the exclusion or limitation of           *
*  incidental or consequential damages, so this exclusion and          *
*  limitation may not apply to You.                                    *
*                                                                      *
************************************************************************

8. Litigation
-------------

Any litigation relating to this License may be brought only in the
courts of a jurisdiction where the defendant maintains its principal
place of business and such litigation shall be governed by laws of that
jurisdiction, without reference to its conflict-of-law provisions.
Nothing in this Section shall prevent a party's ability to bring
cross-claims or counter-claims.

9. Miscellaneous
----------------

This License represents the complete agreement concerning the subject
matter hereof. If any provision of this License is held to be
unenforceable, such provision shall be reformed only to the extent
necessary to make it enforceable. Any law or regulation which provides
that the language of a contract shall be construed against the drafter
shall not be used to construe this License against a Contributor.

10. Versions of the License
---------------------------

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section
10.3, no one other than the license steward has the right to modify or
publish new versions of this License. Each version will be given a
distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version
of the License under which You originally received the Covered Software,
or under the terms of any subsequent version published by the license
steward.
```


© Copyright 2018 WIBU-SYSTEMS AG

```
10.3. Modified Versions

If you create software not governed by this License, and you want to
create a new license for such software, you may create and use a
modified version of this License if you rename the license and remove
any references to the name of the license steward (except to note that
such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary
Licenses

If You choose to distribute Source Code Form that is Incompatible With
Secondary Licenses under the terms of this version of the License, the
notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice
-------------------------------------------

  This Source Code Form is subject to the terms of the Mozilla Public
  License, v. 2.0. If a copy of the MPL was not distributed with this
  file, You can obtain one at https://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular
file, then You may include the notice in a location (such as a LICENSE
file in a relevant directory) where a recipient would be likely to look
for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice
---------------------------------------------------------

  This Source Code Form is "Incompatible With Secondary Licenses", as
  defined by the Mozilla Public License, v. 2.0.
```

Licenses in Runtime and SDK indirect via Qt

The listed components are directly or indirectly required via using Qt. Only the 'lite' installer on Linux and the 'reduced' installer on Windows require no Qt.

Text Codecs: EUC-JP, ISO 2022-JP (JIS), Shift-JIS

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtcore-attribution-qeucjpcodec.html http://doc.qt.io/qt-5/qtcore-attribution-qjiscodec.html http://doc.qt.io/qt-5/qtcore-attribution-qsjiscodec.html
Operating System	Windows, macOS, Linux

```
Copyright (C) 1999 Serika Kurusugawa, All rights reserved.


Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.


THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
```

```
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED.  IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
```

Text Codec: GBK

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtcore-attribution-qbkcodec.html
Operating System	Windows, macOS, Linux

```
Copyright (C) 2000 TurboLinux, Inc.  Written by Justin Yu and Sean Chen.
Copyright (C) 2001, 2002 Turbolinux, Inc.  Written by James Su.
Copyright (C) 2001, 2002 ThizLinux Laboratory Ltd.  Written by Anthony Fok.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED.  IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
```

PCRE2

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtcore-attribution-qbkcodec.html

© Copyright 2018 WIBU-SYSTEMS AG

Operating System	Windows, macOS, Linux

```
PCRE2 LICENCE
-------------
PCRE2 is a library of functions to support regular expressions whose syntax
and semantics are as close as possible to those of the Perl 5 language.

Release 10 of PCRE2 is distributed under the terms of the "BSD" licence, as
specified below, with one exemption for certain binary redistributions. The
documentation for PCRE2, supplied in the "doc" directory, is distributed under
the same terms as the software itself. The data in the testdata directory is
not copyrighted and is in the public domain.

The basic library functions are written in C and are freestanding. Also
included in the distribution is a just-in-time compiler that can be used to
optimize pattern matching. This is an optional feature that can be omitted when
the library is built.

THE BASIC LIBRARY FUNCTIONS
---------------------------

Written by:       Philip Hazel
Email local part: ph10
Email domain:     cam.ac.uk

University of Cambridge Computing Service,
Cambridge, England.

Copyright (c) 1997-2017 University of Cambridge
All rights reserved.

PCRE2 JUST-IN-TIME COMPILATION SUPPORT
--------------------------------------

Written by:       Zoltan Herczeg
Email local part: hzmester
Emain domain:     freemail.hu

Copyright(c) 2010-2017 Zoltan Herczeg
All rights reserved.

STACK-LESS JUST-IN-TIME COMPILER
--------------------------------

Written by:       Zoltan Herczeg
```

```
Email local part: hzmester
Emain domain:     freemail.hu

Copyright(c) 2009-2017 Zoltan Herczeg
All rights reserved.

THE "BSD" LICENCE
-----------------

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

    * Redistributions of source code must retain the above copyright notices,
      this list of conditions and the following disclaimer.

    * Redistributions in binary form must reproduce the above copyright
      notices, this list of conditions and the following disclaimer in the
      documentation and/or other materials provided with the distribution.

    * Neither the name of the University of Cambridge nor the names of any
      contributors may be used to endorse or promote products derived from this
      software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

EXEMPTION FOR BINARY LIBRARY-LIKE PACKAGES
------------------------------------------

The second condition in the BSD licence (covering binary redistributions) does
not apply all the way down a chain of software. If binary package A includes
PCRE2, it must respect the condition, but if package B is software that
includes package A, the condition is not imposed on package B unless it uses
PCRE2 independently.
```

Parts of QTemporaryFile

© Copyright 2018 WIBU-SYSTEMS AG

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtcore-attribution-qtemporaryfile.html)
Operating System	Windows, macOS, Linux

```
Copyright (c) 1987, 1993
     The Regents of the University of California.  All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED.  IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
```

QEventDispatcher on macOS

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtcore-attribution-qeventdispatcher-cf.html)
Operating System	macOS

```
Copyright (c) 2007-2008, Apple, Inc.

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

  * Redistributions of source code must retain the above copyright notice,
```

```
    this list of conditions and the following disclaimer.

  * Redistributions in binary form must reproduce the above copyright notice,
    this list of conditions and the following disclaimer in the documentation
    and/or other materials provided with the distribution.

  * Neither the name of Apple, Inc. nor the names of its contributors
    may be used to endorse or promote products derived from this software
    without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

Cocoa Platform Plugin

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-cocoa-platform-plugin.html
Operating System	macOS

```
Copyright (c) 2007-2008, Apple, Inc.

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

  * Redistributions of source code must retain the above copyright notice,
    this list of conditions and the following disclaimer.

  * Redistributions in binary form must reproduce the above copyright notice,
    this list of conditions and the following disclaimer in the documentation
    and/or other materials provided with the distribution.

  * Neither the name of Apple, Inc. nor the names of its contributors
    may be used to endorse or promote products derived from this software
    without specific prior written permission.
```


© Copyright 2018 WIBU-SYSTEMS AG

```
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

IAccessible2 IDL Specification

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-iaccessible2.html
Operating System	macOS

```
Copyright (c) 2013 Linux Foundation
All rights reserved.


Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

 1. Redistributions of source code must retain the above copyright
    notice, this list of conditions and the following disclaimer.

 2. Redistributions in binary form must reproduce the above
    copyright notice, this list of conditions and the following
    disclaimer in the documentation and/or other materials
    provided with the distribution.

 3. Neither the name of the Linux Foundation nor the names of its
    contributors may be used to endorse or promote products
    derived from this software without specific prior written
    permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR
```

```
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE,
EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This BSD License conforms to the Open Source Initiative "Simplified
BSD License" as published at:
http://www.opensource.org/licenses/bsd-license.php
```

qtmain Library

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtmain.html
Operating System	Windows, macOS, Linux

```
Copyright (C) 2016 The Qt Company Ltd.

Commercial License Usage
Licensees holding valid commercial Qt licenses may use this file in
accordance with the commercial license agreement provided with the
Software or, alternatively, in accordance with the terms contained in
a written agreement between you and The Qt Company. For licensing terms
and conditions see https://www.qt.io/terms-conditions. For further
information use the contact form at https://www.qt.io/contact-us.

BSD License Usage
Alternatively, you may use this file under the terms of the BSD license
as follows:

"Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:
  * Redistributions of source code must retain the above copyright
    notice, this list of conditions and the following disclaimer.
  * Redistributions in binary form must reproduce the above copyright
    notice, this list of conditions and the following disclaimer in
    the documentation and/or other materials provided with the
    distribution.
  * Neither the name of The Qt Company Ltd nor the names of its
    contributors may be used to endorse or promote products derived
    from this software without specific prior written permission.
```


© Copyright 2018 WIBU-SYSTEMS AG

```
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE."
```

Anti-aliasing rasterizer from FreeType 2

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-freetype.html
Operating System	Windows, macOS, Linux

```
Portions of this software are copyright <2006>.
The FreeType Project (www.freetype.org).
All rights reserved.
```

FreeType 2

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-freetype.html
Operating System	Windows, macOS, Linux

```
Portions of this software are copyright <2006>.
The FreeType Project (www.freetype.org).
All rights reserved.
```

LibJPEG-turbo

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-libjpeg.html
Operating System	Windows, macOS, Linux

```
This software is based in part on the work of the Independent JPEG Group.
libjpeg-turbo Licenses
======================
...
The Modified (3-clause) BSD License
===================================

Copyright (C)2009-2017 D.R.Commander.  All Rights Reserved.
```

© Copyright 2018 WIBU-SYSTEMS AG

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.
- Neither the name of the libjpeg-turbo Project nor the names of its
  contributors may be used to endorse or promote products derived from this
  software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS",
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

Contributions to the Cocoa Platform Plugin Files

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/mac-licensing.html
Operating System	macOS

Copyright (C) 2007-2008, Apple, Inc.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of Apple, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

© Copyright 2018 WIBU-SYSTEMS AG

Freetype 2 - Bitmap Distribution Format (BDF) support

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-freetype-bdf.html
Operating System	Windows, macOS, Linux

```
Copyright (C) 2001-2002 by Francesco Zappa Nardelli

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

*** Portions of the driver (that is, bdflib.c and bdf.h):

Copyright 2000 Computing Research Labs, New Mexico State University
Copyright 2001-2002, 2011 Francesco Zappa Nardelli

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
```

```
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.  IN NO EVENT SHALL
THE COMPUTING RESEARCH LAB OR NEW MEXICO STATE UNIVERSITY BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT
OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR
THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

Freetype 2 - Portable Compiled Format (PCF) support

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-freetype-bdf.html
Operating System	Windows, macOS, Linux

```
Copyright (C) 2000 by Francesco Zappa Nardelli

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

HarfBuzz

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-harfbuzz.html
Operating System	Windows, macOS, Linux

```
HarfBuzz was previously licensed under different licenses.  This was
changed in January 2008.  If you need to relicense your old copies,
consult the announcement of the license change on the internet.
Other than that, each copy of HarfBuzz is licensed under the COPYING
file included with it.  The actual license follows:
```


© Copyright 2018 WIBU-SYSTEMS AG

```
Permission is hereby granted, without written agreement and without
license or royalty fees, to use, copy, modify, and distribute this
software and its documentation for any purpose, provided that the
above copyright notice and the following two paragraphs appear in
all copies of this software.

IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE TO ANY PARTY FOR
DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES
ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN
IF THE COPYRIGHT HOLDER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

THE COPYRIGHT HOLDER SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING,
BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE.  THE SOFTWARE PROVIDED HEREUNDER IS
ON AN "AS IS" BASIS, AND THE COPYRIGHT HOLDER HAS NO OBLIGATION TO
PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.
```

HarfBuzz-NG

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-harfbuzz-ng.html
Operating System	Windows, macOS, Linux

```
HarfBuzz is licensed under the so-called "Old MIT" license.  Details follow.
For parts of HarfBuzz that are licensed under different licenses see individual
files names COPYING in subdirectories where applicable.

Copyright © 2010,2011,2012  Google, Inc.
Copyright © 2012  Mozilla Foundation
Copyright © 2011  Codethink Limited
Copyright © 2008,2010  Nokia Corporation and/or its subsidiary(-ies)
Copyright © 2009  Keith Stribley
Copyright © 2009  Martin Hosken and SIL International
Copyright © 2007  Chris Wilson
Copyright © 2006  Behdad Esfahbod
Copyright © 2005  David Turner
Copyright © 2004,2007,2008,2009,2010  Red Hat, Inc.
Copyright © 1998-2004  David Turner and Werner Lemberg

For full copyright notices consult the individual files in the package.


Permission is hereby granted, without written agreement and without
```

```
license or royalty fees, to use, copy, modify, and distribute this
software and its documentation for any purpose, provided that the
above copyright notice and the following two paragraphs appear in
all copies of this software.

IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE TO ANY PARTY FOR
DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES
ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN
IF THE COPYRIGHT HOLDER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

THE COPYRIGHT HOLDER SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING,
BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE.  THE SOFTWARE PROVIDED HEREUNDER IS
ON AN "AS IS" BASIS, AND THE COPYRIGHT HOLDER HAS NO OBLIGATION TO
PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.
```

HarfBuzz-NG

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-harfbuzz.html
Operating System	Windows, macOS, Linux

```
HarfBuzz was previously licensed under different licenses.  This was
changed in January 2008.  If you need to relicense your old copies,
consult the announcement of the license change on the internet.
Other than that, each copy of HarfBuzz is licensed under the COPYING
file included with it.  The actual license follows:

Permission is hereby granted, without written agreement and without
license or royalty fees, to use, copy, modify, and distribute this
software and its documentation for any purpose, provided that the
above copyright notice and the following two paragraphs appear in
all copies of this software.

IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE TO ANY PARTY FOR
DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES
ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN
IF THE COPYRIGHT HOLDER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

THE COPYRIGHT HOLDER SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING,
BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE.  THE SOFTWARE PROVIDED HEREUNDER IS
```


© Copyright 2018 WIBU-SYSTEMS AG

```
ON AN "AS IS" BASIS, AND THE COPYRIGHT HOLDER HAS NO OBLIGATION TO
PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.
```

OpenGL ES 2 Headers

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-opengl-es2-headers.html
Operating System	Windows, macOS, Linux

```
Copyright (c) 2013-2014 The Khronos Group Inc.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and/or associated documentation files (the
"Materials"), to deal in the Materials without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Materials, and to
permit persons to whom the Materials are furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Materials.

THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.
```

OpenGL Headers

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-opengl-headers.html
Operating System	Windows, macOS, Linux

```
Copyright (c) 2013-2014 The Khronos Group Inc.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and/or associated documentation files (the
"Materials"), to deal in the Materials without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Materials, and to
permit persons to whom the Materials are furnished to do so, subject to
the following conditions:
```

```
The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Materials.

THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.
```

XCB

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-xcb.html
Operating System	Windows, macOS, Linux

```
Copyright © 2000 Keith Packard
Copyright © 2006 Jamey Sharp
Copyright © 2007-2008 Vincent Torri <vtorri@univ-evry.fr>
Copyright © 2007 Bart Massey
Copyright © 2008-2009 Julien Danjou <julien@danjou.info>
Copyright © 2008 Arnaud Fontaine <arnau@debian.org>
Copyright © 2008 Bart Massey <bart@cs.pdx.edu>
Copyright © 2008 Ian Osgood <iano@quirkster.com>
Copyright © 2008 Jamey Sharp <jamey@minilop.net>
Copyright © 2008 Josh Triplett <josh@freedesktop.org>
Copyright © 2008 Ulrich Eckhardt doomster@knuut.de

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use, copy,
modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY
```


© Copyright 2018 WIBU-SYSTEMS AG

```
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF
CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the names of the authors or
their institutions shall not be used in advertising or otherwise to
promote the sale, use or other dealings in this Software without
prior written authorization from the authors.
```

xkbcommon

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-xkbcommon.html
Operating System	Windows, macOS, Linux

```
The following is a list of all copyright notices and license statements which
appear in the xkbcommon source tree.

If making new contributions, the first form (i.e. Daniel Stone, Ran Benita,
etc) is vastly preferred.

All licenses are derivative of the MIT/X11 license, mostly identical other
than no-endorsement clauses (e.g. paragraph 4 of The Open Group's license).

These statements are split into two sections: one for the code compiled and
distributed as part of the libxkbcommon shared library and the code
component of all tests (i.e. everything under src/ and xkbcommon/, plus the
.c and .h files under test/), and another for the test data under test/data,
which is distributed with the xkbcommon source tarball, but not installed to
the system.


BEGINNING OF SOFTWARE COPYRIGHT/LICENSE STATEMENTS:


--------------------------------------------------------------------------------

Copyright © 2009-2012 Daniel Stone
Copyright © 2012 Ran Benita <ran234@gmail.com>
Copyright © 2010, 2012 Intel Corporation
Copyright © 2008, 2009 Dan Nicholson
Copyright © 2010 Francisco Jerez <currojerez@riseup.net>

Permission is hereby granted, free of charge, to any person obtaining a
```

copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice (including the next
paragraph) shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

Copyright 1985, 1987, 1988, 1990, 1998 The Open Group

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the names of the authors or their
institutions shall not be used in advertising or otherwise to promote the
sale, use or other dealings in this Software without prior written

© Copyright 2018 WIBU-SYSTEMS AG

authorization from the authors.

Copyright (c) 1993, 1994, 1995, 1996 by Silicon Graphics Computer Systems, Inc.

Permission to use, copy, modify, and distribute this
software and its documentation for any purpose and without
fee is hereby granted, provided that the above copyright
notice appear in all copies and that both that copyright
notice and this permission notice appear in supporting
documentation, and that the name of Silicon Graphics not be
used in advertising or publicity pertaining to distribution
of the software without specific prior written permission.
Silicon Graphics makes no representation about the suitability
of this software for any purpose. It is provided "as is"
without any express or implied warranty.

SILICON GRAPHICS DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS
SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY
AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL SILICON
GRAPHICS BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL
DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE
OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH
THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright 1987, 1988 by Digital Equipment Corporation, Maynard, Massachusetts.

All Rights Reserved

Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted,
provided that the above copyright notice appear in all copies and that
both that copyright notice and this permission notice appear in
supporting documentation, and that the name of Digital not be
used in advertising or publicity pertaining to distribution of the
software without specific, written prior permission.

DIGITAL DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING
ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL
DIGITAL BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR
ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS,
WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION,
ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS
SOFTWARE.

Copyright (C) 2011 Joseph Adams <joeyadams3.14159@gmail.com>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

END OF SOFTWARE COPYRIGHT/LICENSE STATEMENTS

BEGINNING OF LICENSE STATEMENTS FOR UNDISTRIBUTED DATA FILES IN test/data,
derived from xkeyboard-config:

© Copyright 2018 WIBU-SYSTEMS AG

Copyright 1996 by Joseph Moss
Copyright (C) 2002-2007 Free Software Foundation, Inc.
Copyright (C) Dmitry Golubev <lastguru@mail.ru>, 2003-2004
Copyright (C) 2004, Gregory Mokhin <mokhin@bog.msu.ru>
Copyright (C) 2006 Erdal Ronahî

Permission to use, copy, modify, distribute, and sell this software and its
documentation for any purpose is hereby granted without fee, provided that
the above copyright notice appear in all copies and that both that
copyright notice and this permission notice appear in supporting
documentation, and that the name of the copyright holder(s) not be used in
advertising or publicity pertaining to distribution of the software without
specific, written prior permission. The copyright holder(s) makes no
representations about the suitability of this software for any purpose. It
is provided "as is" without express or implied warranty.

THE COPYRIGHT HOLDER(S) DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE,
INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO
EVENT SHALL THE COPYRIGHT HOLDER(S) BE LIABLE FOR ANY SPECIAL, INDIRECT OR
CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.

Freetype 2 - zlib

Project	Runtime component CodeMeter Control Center (CodeMeter License Server)
Origin	http://doc.qt.io/qt-5/qtgui-attribution-freetype-zlib.html
Operating System	Windows, macOS, Linux

Copyright (C) 1995-2002 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied
warranty. In no event will the authors be held liable for any damages
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
   claim that you wrote the original software. If you use this software
   in a product, an acknowledgment in the product documentation would be
   appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
   misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Adler
jloup@gzip.org madler@alumni.caltech.edu

musl

Project	CmEmbedded
Version	2.3
Operating system	Android

musl as a whole is licensed under the following standard MIT license:

Copyright © 2005-2014 Rich Felker, et al.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

© Copyright 2018 WIBU-SYSTEMS AG