computer aided development corporation



Cadcorp SIS 9 Installer DLLs 2.0

Contents

CadcorpInstallerLicFileValidator.dll 2
CadcorpInstallerCustomLsInit.dll 5

November 2018

Some of the dlls described in this document are supplied as both 32-bit and 64-bit versions. 64-bit versions have '64' as the final part of the dll name. Please ensure that you use the correct version of the dll.

DLL Name: CadcorpInstallerLicFileValidator.dll

Function(s): BrowseAndValidateLicFile(MSIHANDLE hInstaller)

This function requires an 'MSIHANDLE' to be passed in from the host installer. It attempts to decode all entries in a licence file, and for each entry, it returns information about the Feature-Name, the result of the Licence-Check for that Feature/Entry, and the type of licence that entry represents.

First of all, the dll reads a "**LIC_FILE_PATH_CMD_LINE**" Msi-Property from the host installer – this "**LIC_FILE_PATH_CMD_LINE**" property MUST be present in the installer. If the Installer is being called from the command-line (as opposed to running in the normal User Interface mode), then the dll expects this "**LIC_FILE_PATH_CMD_LINE**" property to already contain a path to the licence file to decode. If, however, the "**LIC_FILE_PATH_CMD_LINE**" property is found to be empty (User Interface mode), then the dll will show a file-open dialog box, allowing the user to browse to a licence file. Once the user has selected a licence file, the dll will then store the file-path in a "**LIC_FILE_PATH**" property in the Installer, so that the Installer has the file-path of the selected licence file. Again, this "**LIC_FILE_PATH**" property MUST be present in the Installer.

The host installer must tell the CadcorpInstallerLicFileValidator.dll what licence features it wants the dll to process, by using a "**LIC_FEATS_TO_PROCESS**" property and setting this value to a comma-separated string of feature-names to be processed. This "**LIC_FEATS_TO_PROCESS**" property MUST be present in the host installer. The CadcorpInstallerLicFileValidator.dll will IGNORE any features found in the licence file that aren't listed in the "**LIC_FEATS_TO_PROCESS**" property! For example, if we set the property to:

"GeognoSIS|Webmap|Lyr,GeognoSIS|Webmap|Lyr|AMP,GeognoSIS|Webmap|Lyr|FMN"

…then the CadcorpInstallerLicFileValidator.dll will only process the "GeognoSIS|Webmap|Lyr", "GeognoSIS|Webmap|Lyr|AMP", and "GeognoSIS|Webmap|Lyr|FMN" features/entries in the licence file.

The use of a wildcard character (*) is allowed to provide a shorthand method of processing multiple features with one expression.

For example, the **LIC_FEATS_TO_PROCESS** property can be set to:

"GeognoSIS|Webmap|*,SIS|GTools|Viewer,SIS|Desk*"

This will return Licence check results for all products with feature names beginning with "GeognoSIS|Webmap|", the "SIS|GTools|Viewer" product, and all products with names beginning with "SIS|Desk".

WARNING: *The CadcorpInstallerLicFilevalidator.dll WILL IGNORE ALL LICENCE FEATURES IN THE LICENCE FILE THAT ARE NOT LISTED IN THIS "**LIC_FEATS_TO_PROCESS**" PROPERTY – THEREFORE IF THIS*

PROPERTY IS LEFT BLANK, THEN NO FEATURES WILL BE DECODED BY THE DLL.

Installers can also, optionally, include a property called **LIC_FEATS_VERSION_TO_PROCESS**. This property tells the CadcorpInstallerLicFileValidator.dll only to process licence features matching the specified version. This property is useful if Installers want to ensure that only licenses matching a specific version are deemed to be valid and can be used to install the application.
LIC_FEATS_VERSION_TO_PROCESS is optional – leaving this property out of the Installer, or including it but leaving its value 'blank' will result in no licence version checks being carried out.

Once the dll has a licence file-path, a list of features-to-process and optionally a licence version specified, it will attempt to parse the entire file, decoding all encoded entries it was told to, and returning information about each one. It stores this information in a string, and returns this to the installer by setting another Msi-Property, named "**LicFeatsAndResults**". This property MUST therefore also be present in the host Installer.

The "**LicFeatsAndResults**" property will be set to a string containing, for each entry, a comma separated triple-set of (3) values denoting the Feature-Name, Licence-Check-Result, and Licence-Type. If there is more than one entry in the licence file, then each set of triple (3) values for each entry, will be separated by the '#' character. For example, we may have:

"FeatureName1,LIC_RET_VAL_SUCCESS,DONGLE#FeatureName2, LIC_RET_VAL_FAIL_EXPIRED, MACHINE", etc…

The list of possible values that can be returned for the 'Licence-Check-Result' part of the string are:

- LIC_RET_VAL_SUCCESS (the licence is valid)
- LIC_RET_VAL_FAIL_EXPIRED (the licence is out of date and has expired)
- LIC_RET_VAL_FAIL_LOCK_CODE_MISMATCH (the licence is locked to a different machine/dongle/key than this one)
- LIC_RET_VAL_FAIL_GET_MACH_ID (error - the dll failed to obtain the id of the machine or dongle or key)
- LIC_RET_VAL_FAIL_MACH_ID_TO_LOCKCODE (error – the dll failed to convert the machine-id into a lock-code)
- LIC_RET_VAL_FAIL_UNRECOG_LOCK_TYPE (the dll encountered an unrecognised locking-type)

The list of possible values that can be returned for the 'Licence-Type' part of the string are:

- DONGLE (the licence is locked to a dongle)
- MACHINE (the licence is locked to a machine (MAC Address or combination of MAC and IP address))
- KEY (the licence is locked to an id-key)
- UNLOCKED (the licence is unlocked (not locked to any hardware))

Using this information returned by the CadcorpInstallerLicFilevalidator.dll, the host installer can tell whether the features in a licence file are valid, and if they are not, why not (e.g. they have expired, or they are locked to another dongle or machine, etc.). It can then show a message to the user so that they know they are using the incorrect licence file.

NOTE: Before trying to read the "**LicFeatsAndResults**" property, the host installer should first check the overall return value from the dll (see the use of the "**LicFileProcessResults**" property, below):

Return values: The actual "**BrowseAndValidateLicFile**" function returns a 'UINT' (unsigned integer), but this value is always "ERROR_SUCCESS". This is so that the host Installer doesn't abort, but can continue the process. Instead, the dll communicates errors back to the Installer, by using a "**LicFileProcessResult**" property. This "**LicFileProcessResult**" property MUST therefore be present in the installer, and will be set to one of the following values:

- LIC_FILE_PROC_RET_VAL_ERR_GET_CMD_LINE_LIC_FILE (An error occurred whilst reading the 'LIC_FILE_PATH_CMD_LINE' property)
- LIC_FILE_PROC_RET_VAL_ERR_GET_FEATS_TO_PROCESS (An error occurred whilst reading the 'LIC_FEATS_TO_PROCESS' property)
- LIC_FILE_PROC_RET_VAL_ERR_NO_LIC_STRINGS_DECODED (No licence strings to decode were found in the licence file (remember that the dll will only try and decode licence strings that relate to the features listed in the "**LIC_FEATS_TO_PROCESS**" property)
- LIC_FILE_PROC_RET_VAL_ERR_GEN_EXCEPTION (A general exception occurred)
- LIC_FILE_PROC_RET_VAL_ERR_LIC_LIB_INIT (A Dll initialisation error occurred)
- LIC_FILE_PROC_RET_VAL_ERR_LIC_FILE_READ (A licence file could not be found in the specified path)
- LIC_FILE_PROC_RET_VAL_ERR_INVALID_FILE (The licence file in the path specified is not valid (e.g. it has an invalid name, etc))
- LIC_FILE_PROC_RET_VAL_ERR_SET_HOST_ID_FUNC (Unable to set the Host ID Custom Lock function)
- LIC_FILE_PROC_RET_VAL_CANCELLED (The user hit 'cancel' or closed the open-file dialog)
- LIC_FILE_PROC_RET_VAL_SUCCESS (Successfully decoded one or more licence strings)

DLL Name: CadcorpInstallerCustomLsInit.dll

The "CadcorpInstallerCustomLsInit" dll can be used in an msi-type installer project (such as those created by InstallShield, Wise, and Visual Studio Setup and Deployment projects), to write necessary licensing-related information into the host computer's registry. This licensing information must be written to the registry in order for the application's licensing to work correctly.

Licensing is based on 'features' (i.e. a 'feature' is licensed). A licensing 'feature' consists of a 'feature-name' and a 'version' together as a pair (for example, feature-name = "MySoftware", version = "1.3"). Cadcorp will provide the values for a given feature-name and version. The "CadcorpInstallerCustomLsInit" dll can be used to write the licensing information for one or more features, to the registry (i.e. it can process just one, or several features in one go, if required).

To utilise the "CadcorpInstallerCustomLsInit" dll as part of your install project, perform the following tasks:

- Create a property in the installation project, named "**FEATURES_AND_VERS**". The property must have this exact name. Then, set this property to the list of (one or more) 'features' that you want the licensing information set for, by using the "*" character to separate a 'feature-name' from its 'version'. If specifying more than one feature, use the "#" character to separate one feature from another. For example, to specify one feature you can set the "**FEATURES_AND_VERS**" property to this: "MyFeature1*1.1". The "CadcorpInstallerCustomLsInit" dll will break this down into:
    - Feature = "MyFeature1", version number = "1.1"

  To specify three features, you can set the "**FEATURES_AND_VERS**" property to this: "MyFeature1*1.1#MyFeature2*1.2#MyFeature3*1.3. The "CadcorpInstallerCustomLsInit" dll will break this down into:
    - First feature = "MyFeature1", version number = "1.1"
    - Second feature = "MyFeature2", version number = "1.2"
    - Third feature = "MyFeature3", version number = "1.3"
- Create a 'Property' in the installation project, named "LS_INIT_RETURN". The Property must have this exact name. The "CadcorpInstallerCustomLsInit" dll will write a value to (i.e. set) this Property, to communicate back to the installer whether or not the license initialisation it performed was successful.
- Create a 'Custom Action' in the installer project, to call the "CadcorpInstallerCustomLsInit" dll – the 'type' of dll for this Custom Action should be set to a 'Windows Installer' or 'Standard' dll, as opposed to a 'Custom' dll. Set the dll 'Entry Point' (i.e. the function to call within the dll) to "LsInit".
- After calling the "CadcorpInstallerCustomLsInit" dll with a Custom Action, the installer can check the value of the "LS_INIT_RETURN" Property (using InstallShield or VB Script, etc.), in order to ascertain whether or not the license information was successfully written to the

registry. The installer can then take any appropriate action deemed necessary. The “CadcorpInstallerCustomLsInit” dll will return (i.e. set the value of the “LS_INIT_RETURN” Property to) “LS_INIT_SUCCESS” if it successfully wrote the license information to the registry. Any other value indicates a failure of some description. The complete list of possible values returned by the “CadcorpInstallerCustomLsInit” dll are:

- **LS_INIT_INVALID_INPUT_PARAM_FAIL**
- **LS_INIT_LIB_FAIL**
- **LS_INIT_SECURITY_DESCRIP_FAIL**
- **LS_INIT_CREATE_REG_KEY_FAIL**
- **LS_INIT_SET_REG_KEY_SECURITY_FAIL**
- **LS_INIT_REG_KEY_OPEN_FAIL**
- **LS_INIT_SET_VALUE_FAIL**
- **LS_INIT_QUERY_FAIL**
- **LS_INIT_CREATE_FILE_OR_SET_SECURITY_FAIL**
- **LS_INIT_OPEN_FILE_FAIL**
- **LS_INIT_WRITE_INTO_FILE_FAIL**
- **LS_INIT_SET_FILE_SECURITY_FAIL**
- **LS_INIT_EXCEPTION_OCCURED_FAIL**
- **LS_INIT_INTERNAL_FAIL**
- **LS_INIT_PERMISSIONS_FAIL**
- **LS_INIT_TAMPER_DETECTED_FAIL**
- **LS_INIT_GET_PERSIST_DB_PATH_FAIL**
- **LS_INIT_FILE_READ_FAIL**
- **LS_INIT_NO_RESOURCES_FAIL**
- **LS_INIT_CANT_GET_LOCK_FAIL**
- **LS_INIT_SET_PERSIST_FILE_LOCATION_FAIL**
- **LS_INIT_UNDEFINED_FAIL**
- **LS_INIT_CAUGHT_EXCEPTION_FAIL**