zSpace Licensing Library 2.0 - Developer Documentation

Overview

This document describes version 2.0 of the zSpace Licensing Library, which allows zSpace-enabled applications to be licensed using the zSpace Licensing System. The document covers various related nomenclature, the characteristics of the licensing model used by the zSpace Licensing Library, how to integrate the zSpace Licensing Library into an application, and the API for the zSpace Licensing Library.

zSpace is working to transition all zSpace-enabled applications to use the zSpace Licensing Library in order to reduce the burden on customers with respect to managing software licenses. Transitioning all zSpace-enabled applications to use the zSpace Licensing Library will give customers a consistent licensing experience across all applications and will allow zSpace to build a license management system that will make it quicker and easier for customers to manage licenses across all of their zSpace devices and applications.

Nomenclature

- Application
  - A program that is intended to be launched by end users.
- Application mode
  - A specific way that an application can be launched which determines the type of content it presents.
  - For the purposes of licensing, an application's modes are only relevant if they correspond to separately purchasable entities.
  - Most applications only have a single mode.
- Application supplement
  - Functionality, content, etc. of an application or application mode that can be licensed independently of the application/application mode itself.
- Licensing product group
  - A group of licensing products that can be licensed as a single unit (e.g. by activating a single license key).
- Licensing product
  - An individually licensable entity within a licensing product group.
  - Corresponds to either an application mode or an application supplement.

Licensing Model

The zSpace Licensing Library will initially use a device-locked, license-key-based licensing model with the following characteristics:

- Application functionality/content is organized into modes and supplements that map to individual licensing products.
  - Each licensing product may be licensed independently.
- A license can cover multiple licensing products in the form of a licensing product group.
  - Multiple licensing products can be bundled into licensing product groups that can be activated with a single license.
- When a license is purchased, a corresponding license key is issued and can be used to activate associated licensing products on devices.
  - Customers that purchase a license for a licensing product group will be issued a corresponding license key.
  - Entering the license key into any of the applications hosting the licensing products in the licensing product group associated with the license key will activate all licensing products (and corresponding application modes and supplements) in the group on the device that the license key was entered on.
- Licenses may have multiple seats that allow a license to be used concurrently on multiple devices.
  - Licenses may have multiple license seats and may be used to concurrently activate the associated licensing products on a number of different devices that is less than or equal to the number of seats included in the license.
- License seats used on a device can be deactivated and transferred to other devices.
  - Customers are able to deactivate licensing products on a device, freeing up the seats that the device was using in the corresponding licenses and allowing those seats to be used on other devices. If a license covers a group of licensing products, all licensing products (and corresponding application modes and supplements) in the group will be deactivated on a device at the same time when any licensing product in the group is deactivated on the device.
- Both expiring and perpetual licenses are supported.
  - Licenses may have an expiration date associated with them. Licensing products activated on a device with a license that has an expiration date will no longer be usable on the device once the expiration date plus an expiration grace period has passed.
  - A license without an associated expiration date is a perpetual license and can be used for an unlimited period of time.
- Applications periodically perform enhanced license checks that use an Internet connection but are able to run offline for a configurable amount of time between successful enhanced checks.
  - Periodically, when running an application hosting a licensing product that has been activated on a device, the application will attempt to perform an enhanced license check that uses an Internet connection. Licenses allow a licensing product that has been activated on a device to be used offline (i.e. without an

Internet connection and without performing enhanced license checks) for up to a configurable amount of time. When running an application hosting a licensing product that has been activated on a device, if an enhanced license check has not been successfully performed for more than the configured maximum amount of offline time, then the licensing product will not be usable until an Internet connection is established and an enhanced license check is successfully performed.

Unity Integration Steps

For Unity-based applications, it is recommended to use the following steps to integrate the zSpace Licensing Library. These steps use the higher-level `zSpace.Licensing.Unity.ApplicationLicensingConfiguration` and `zSpace.Licensing.Unity.LicensingUtility` classes, which makes it easier to integrate the zSpace Licensing Library into Unity-based applications.

1. Acquire the zSpace Licensing Library files from zSpace.
2. Acquire the application-specific licensing data from zSpace.
   a. This will include:
      i. Application licensing secret.
      ii. Application licensing details file.
      iii. One or more application mode IDs.
         1. More than one application mode ID will be used in cases such as the following:
            a. The application has more than one mode and access to each mode can be purchased separately.
            b. The application's mode ID has changed over time or varies by licensing product group.
      iv. Zero or more application supplement IDs.
3. Place the application licensing details file into the application's `StreamingAssets` directory.
4. Add code to initialize the zSpace Licensing Library and to perform a license check.
   a. This should be done before the user can gain access to the main content of the application.
      i. It is recommended to do this while the application's splash screen is being displayed.
   b. The code to do this should perform the following steps:
      i. Prevent the user from interacting with the application in any way that would allow them to access or use the main content of the application.
      ii. Create an instance of the `ApplicationLicensingConfiguration` class and call the various setter methods on it to create the licensing configuration for the application.
         1. All applications will generally only need to call the following setter methods:

a. `SetApplicationModeId()`
b. `SetApplicationSupplementIds()`
    i. Only needed if the application has supplements.
c. `SetApplicationVersion()`
    i. Pass the application's version number to this method in the form of a string containing a sequence of version number components (each containing only digits) separated by dots (e.g. "1.2.3").
d. `SetApplicationLicensingSecret()`
e. `SetApplicationLicensingDetailsFromStreamingAssetsFile()`
    i. Pass the relative path to the application licensing details file with the application's `StreamingAssets` directory.
f. `SetLocale`
    i. Pass the locale code that the application is currently using as well as the localized, user-friendly application name for that locale.
    ii. The locale code should be of the form `<language code>-<region code>` where `<language code>` is the lower-case, two letter language code (e.g. `en` for English, `zh` for Chinese, etc.) and `<region code>` is the upper-case, two letter region code (e.g. `US` for the United States, `CN` for China, etc.).
iii. Initialize the zSpace Licensing Library and perform a license check by calling either the `LicensingUtility.InitializeLicensingAndCheckLicense()` static method or the `InitializeLicensingAndCheckLicenseCoroutineEnumerator()` static method.
    1. Once zSpace Licensing Library initialization and licensing check completes, do one of the following:
        a. If the initialization and license check process indicated that the application should exit or that an error occurred, exit immediately.
        b. Otherwise, allow the user to interact with the application normally and proceed with letting the user access the main content of the application.
    2. Note: All licensing related UI that needs to be displayed during the license check process (such as license key prompts and error/warning/notification messages) will be displayed internally by the zSpace Licensing Library using an application-agnostic UI system.

5. Add UI and code that provides access to the license management UI.
    a. The in-application UI should consist of a "manage license" button that is only accessible after the application's license has been successfully checked.
        i. It is recommended to put the "manage license" button somewhere in the application's existing settings/configuration UI.
    b. The code for this should perform the following steps when the in-application "manage license" button is pressed:
        i. Prevent the user from interacting with the application in any way that would allow them to access or use the main content of the application.
        ii. Display the license management UI by calling the `LicensingUtility.ShowLicenseManagementUi()` method.
            1. Once the license management process completes, do one of the following:
                a. If the result of the `LicensingUtility.ShowLicenseManagementUi()` method indicates that the application should exit or that an error occurred, exit immediately.
                b. Otherwise, allow the user to interact with the application normally.