ModuleAdmin

A support library for cloneable modules

1. Introduction

Cloneable modules are stand-alone modules that can be cloned many times by a framework. Once a clone is created it stands alone and receives requests in the form of user events and “broadcasts” messages, also in the form of user events. Any application can register to receive broadcasts from a module type. Once registered, it will receive broadcasts from all clones of that module type. For further information about modules, see the Cloneable Module Template or the Pluggable Module Template documentation.

ModuleAdmin is a library to support cloneable modules. It consists of a ModuleAdminBase class, a child ModuleAdmin class and CloneRegistration class.

1. ModuleAdmin Class



* 1. ModuleAdminBase Class

Base Class for the ModuleAdmin

* + 1. Properties
       1. External Launch: bool

Boolean type that indicates if a module was launched by an external application such as a framework or if it was launched by itself. Externally launched modules cannot be closed except when a “Stop” request is received.

* + 1. Methods
       1. Reset (External Launch)

Sets the External Launch property

* 1. ModuleAdminClass
     1. Properties
        1. ModuleID :int32

When a module clone is created, the system assigns a unique Module ID which must be remembered by the clone so it can determine if an incoming request is for it. When ModuleID is set to -1, all module clones assume the request is for them.

* + - 1. First? :bool

The ModuleAdmin instance in the first clone of a module type maintains a registry of all other clones created. this means that it must be the last clone to be destroyed. If the first clone receives a stop command, all operations will stop but the clone will not be removed from memory until the module registry indicates that it is the only clone of its type remaining.

* + - 1. ModuleName :string

The ModuleName indicates what type it is. All modules of a particular type have the same name. modules are identified by the combination of ModuleName:ModuleID.

* + - 1. CloneRegRef :reference

A reference (pointer) to the CloneRegistration Class instance

* + 1. Methods
       1. New ()

Creates an new instance of ModuleAdmin class. If this is the first instance of a particular module type, then a CloneRegistration Class instance will also be created. The clone is added to to the clone registry.

* + - 1. AddressedToThisModule (ModuleID)

Returns AddressedToThisInstance? = TRUE if the ModuleID argument matches the ModuleID property or is -1. Returns AddressedToAll?=TRUE if the ModuleID is -1.

* + - 1. ModuleReset (ModuleName)

Resets all the properties of a Module instance. External Launch and First\_ are set to FALSE, Module ID is set to -1 and ModuleName is set to the input argument.

* + - 1. WaitOnModuleSync ()

Once a module clone is created, it needs some time to prepare to accept requests. This method is used to synchronize the launching application with the module so that no requests are sent until the module is ready.

* + - 1. WaitOnEventSync ()

Once a module clone is created, it needs some time to prepare to handle any events. This method is used to synchronize the launching application with the module so that no module events are generated until the clone is ready.

* + - 1. WaitOnStopSync ()

When a stop request is sent to a module, it takes a while for the module to stop. This method allows the launching app to wait until getting a sync signal that the module has stopped.

1. CloneRegistration Class

For all clones of a single module type, a single CloneRegistration class instance is created in the first clone of the type. the CloneRegistration retains a variant with a set of attributes names with the ModuleID with the value of the First? boolean. Methods allow for the addition of a new clone ModuleID, removal of the ModuleID, and a listing of all clone ModuleIDs and First? values in the registry.



* 1. Properties
     1. vRegistry :variant

This maintains the list of clone ModuleIDs each with a boolean value of TRUE is the clone is the first clone.

* 1. Methods
     1. Create ()

Creates a new data value reference for a CloneRegistratiion instance.

* + 1. Init ()

Initializes the vRegistry property with an empty variant.

* + 1. Add (ModuleID)

Adds a ModuleID to the list of clones. If there are no other clones in the registry, the First? value is set to TRUE, if there are clones, then First? is set to FALSE.

* + 1. isFirst (ModuleID)

Returns the ModuleID attribute First? value (TRUE or FALSE)

* + 1. List ()

Retuens arrays of ModuleID strings, First? values, and a boolean Found? value set TRUE if any modules were found in the registry or FALSE if no modules were found.

* + 1. Remove (ModuleID)

Removes the clone from the registry. Returns Found? TRUE if the ModuleId was found and removed or Found? False if the ModuleID was not found in the registry.

* + 1. Destroy ()

Destroys the data value reference to the CloneRegistration instance.