DNNrocket Module Interfaces for DNN

# Introduction

Exporting and importing content is used to move data and to created backups and website templates in DNN. The exporting and importing of modules, in DNN is done via the IPortable interface. The upgrade uses the IUpgradeable interface and the search uses an inherited function from the “ModuleSearchBase” class.

In DNNrocket this is applied to the “DNNrocketModuleController” class in the DNNrocketAPI project.

# DNNrocketModuleController

This class is used as a generic interface for DNNrocket to link to the Export/Import/Upgrade and search interfaces in DNN.

It calls the DNNrocket API using “provider based” code. The provider is defined in the DNNrocket system data as an interface.

This class must be defined as the business controller for the DNN module.

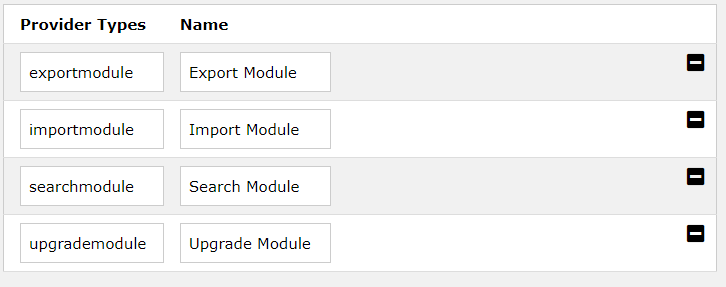
<businessControllerClass>  
DNNrocketAPI.Componants.DNNrocketModuleController,DNNrocketAPI  
</businessControllerClass>

# System Data

“/desktopmodules/dnnrocket/adminsystem.html”

An interface must be created for each DNN interface that needs to be call. This interface will define the correct “Namespace and Class” and “Assembly” and the command to be actioned.

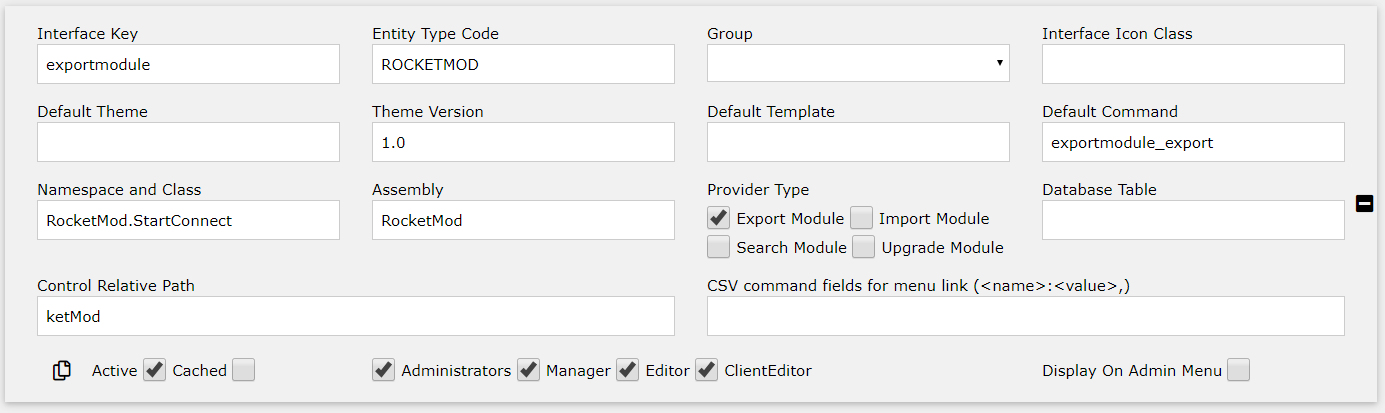
Providers need to be setup in the DNNrocket system data. Under the “Provider Type” tab.



These MUST match the above “Provider Type” values in lowercase. These relative provider type must be selected when the interface is created under the interface tab. Multiple interfaces per provider type can be created. All will be executed, and all data generated will be actioned.

# Export Function

## Interface



## Code

The code can be found in the “DNNrocketModuleControler” class.

# XML structure

<export>

{XML data generated by each provider interface}

</export>

# Export Provider

The provider that is called must return an XML string with ALL data, including images and files. The images and file are usually created using the c# “ToBase64String” function.

Any XML returned by each provider type will be placed in the export XML, therefore if multiple interfaces are used, each XML section should have an identification node or a root node which is unique for that data.

# Import Provider

The import interface for DNN is setup in the same way. But the provider called must take the export XML and import it into the site. The import data is passed via the postInfo variable which is a SimplisityInfo class. The data is saved in XMLData, without the standard “genxml” node. The exact XML exported will be assigned.