# Where to put dependent dlls

When deciding where to put a dependent dll, think of Entity Anywhere in two parts:

1. Core EAF
2. Plugins

Now, deciding where the dependency should go becomes easy. Ask yourself these three questions:

1. Is the dependency only used by 1 plugin?  
   Yes – Then the dependent dll should copied (using the build action) to the plugin’s *bin* directory.
2. Is the dependency used only by plugins, but used by more than one plugin?  
   Yes – Then the dependent dll should be copied to the *SharedBin* directory. Each plugin that references the dll should have a build action to do the copy. Why? In case one plugin is deleted or replaced or no longer depends on it.
3. Is the dependency used by the core of EAF, the part that loads the plugins?  
   Yes –Then the dependent dll should be copied to the root *bin* directly for the website.

# Where to put Autofac Registrations

When deciding where to put a register a class, think of Entity Anywhere in two parts:

1. Core EAF
2. Plugins

Now, deciding where the dependency should go becomes easy. Ask yourself these three questions:

1. Is the class only used by 1 plugin?  
   Yes – Then the registration should be in the plugin’s registration module.
2. Is the class used only by plugins, but used by more than one plugin?  
   Yes – Then the registration should be in each plugin’s registration module
3. Is the class used by the core of EAF, the part that loads the plugins?  
   Yes – Then the registration should be in Starter.cs or a module it loads, either in the root container or the wcf container, which is also created in Starter.cs.