Navigation Paths

Navigation items in the WMC are identified by Navigation Paths. These case-insensitive paths describe the hierarchy of items in relation to other items of the same type. The root node of the Navigation Path (called the *domain*) indicates the type of navigation item.

Domain	Description
NAVIGATIONTREE	A Navigation Path that identifies an item in the navigation tree
HELP	A Navigation Path that identifies help pane content
TASKLIST	A Navigation Path that identifies a group or item in the task list.
MESSAGEBAR	A Navigation Path that identifies an item in the navigation bar at the top of the WMC

Examples:

NAVIGATIONTREE\ControlPanel\Events refers to the *Events* item in the navigation tree that sits under the *Control Panel* item.

TASKLIST\Data\ResetValue refers to the task *ResetValue* that lies under the *Data* task group.

HELP\NAVIGATIONTREE\ControlPanel\Events refers to a help item that is automatically associated with the navigation tree item *ControlPanel\Events*

NavInsertItem takes a Navigation Path and uses it to create an item at that location. The domain with which an item lives is defined by the root of the path and levels in the hierarchy are specified by the path separator '\'. Paths can effectively be any depth you wish.

Aliases

Navigation Paths can have aliases. By using aliases the underlying navigation path can change and the framework will still be able to find it via its alias. NavInsertItem allows you to specify an alias when you add the item. Use the NavItemHelpers function MakeAliasForName to easily create an alias from a name.

If you have a component that needs to allow others to know about its navigation items (To place tree node under it for instance) then publish the alias not the full navigation path. So long as you don't rename the alias then you can move/rename the navigation path within the hierarchy and client code will continue to find your item via its alias.

Like Navigation Paths, aliases are case-insensitive.

> Qbik highly recommends that you use aliases to find parent items when inserting your own navigation items. This ensures that your items continue to operate correctly even when the underlying navigation paths change.

> Make sure your alias is unique. Qbik Well known aliases are undecorated but you should ensure your alias can be differentiated from someone elses. E.g. Don't call your alias *File*, but use something that uniquely identifies your alias such as *MyCompanyScan nerComponentFile, mycompany.scanner.file or a string* UUID if you wish.
> MakeAliasFromName(L"703201E2-6ACF-463D-AA4B-038F99AD9414");