attach(type, objpublic)

Description
This is the only function that most programmers will have to call. You call it from the constructor(s) of your class. Its arguments are the class name (type), and the tool you are about to return to the user (that is, you pass it "public" if you are using the usual closure-tool idiom).
If necessary, it loads all the plugin (`.gp`) files for the requested types, and runs any plugin `init()` functions it finds. It then calls all the `attach()` functions on the loaded plugins. Those attach functions add functions to the tool.

Arguments

type	type (class) for which we want to load plugins Allowed:string
objpublic	the tool the constructor will return to the caller Allowed:record

Returns
reference to objpublic (so you can return it directly)

1.8.11 asynceval - Tool

Package utility
Module tasking

Asynchronous eval

Description
`asynceval` is a simple tool for asynchronous evals. It is mainly of use in Glish programming. You give a command that is to be eval()'ed and then you can wait for the result. It's particularly useful for commands that may fail for some unexpected reason. The Glish interpreter is not then blocked.
The events recognized are:

run (accepted) the argument is a command string to be eval()'ed.

result (emitted) The argument is the result of the eval().

Constructors

asynceval	Construct asynceval tool

Functions

done	Kill this asynceval

asynceval - Constructor

Package	**utility**
Module	**tasking**
Tool	**asynceval**

Construct asynceval tool

Synopsis
asynceval()

Description
This makes an asynceval tool. It is a good idea to make one of these when needed and to kill it (via done) when finished.

---

asynceval.done - Function

Package	**utility**
Module	**tasking**
Tool	**asynceval**

Kill this asynceval

Synopsis
done()

Description
The asynceval is terminated. All existing whenevers are deactivated and the subsequence itself is destroyed.

Returns
Bool