MapDef / MapFunc

MapDef maintains the registry of MapFunc and provides a global interface for creating and accessing them by name.

The MapFunc object will scale input and write it to the local Environment. Output of the MapFunc is calculated
when an input value is updated. MapFuncs can also generate output on a clock at a given rate.

The shape of the MapFunc scaling can be created using a Evelope style segment array pairs, a Spec or a Function.
The input to a MapFunc can be one or many. The output can also be one or many.

In the context of a ctrlBuss scenario, a Mapbuss will create the scaled and transport it to all relevant EventDef.

MapDef(name, input, output, recipe, parent)

name - a symbol
input - a symbol or an array of symbols containing the symbol names of MapBuss values.
output - a symbol or an array of symbols containing the symbol names of MapBuss values.
recipe - recipe can take a few forms:
  Array of array containing numbers. [output value array, input value array] ie. [[0,0.25,5,50], [0.5,0.3,0.2]]
  Function - a function that receives an argument. the last line of the function sets the value of the output name.
  ControlSpec - a ControlSpec
parent - internal use.

instance methods

.environment
references the local Environment in the mapItem which acts as the working environment for both. (MapFunc does not have an Environment)

.at(key)
retreive a value from the local environment

.put(key, value)
put a value into the local environment

.run(logical)

logical - set to true or false
will set the object to respond to input or not

.add(name, in, out, minVal, maxVal, envArgs, order)
NEED TO RE_EXAMINE

.out()
??

.start
start the internal clock driven data generator

.stop
stop the internal clock driven data generator

.setWait(interOnset)
interOnset - a number or a function that will return a number. time between re-calculations of output values
.toggle
start or stop the clock driven data generator

.addOutputResponder(name, chan, func)
name - a symbol, the name used to store the outputResponder locally
chan - a symbol, the name of the source data used as input.
func - a Function, used to calculate the response. The function is passed the value at chan.
This function is evaluated inthe local environment.

.removeOutputResponder(chan, name)
chan - a symbol, the name of the output channel
name - the name of the responder

.outputChannelTest(name)
name - a symbol
internal use, checks whether memory space exists, if not, create it

.update
resolves the state of the MapFunc

```
//
//
//

m = MapBuss.new;
m.addDef(\test, \input, \output, { |vv| [~input, ~output].postln; vv * 0.33 });
m.set(\input, 22);
MapDef(\test).environment
m.set(\input, rrand(10,15));
MapDef(\test).mapItem.environment === MapDef(\test).environment //WHAT?!


MapDef(\test).setWait({ rrand(0.1,0.25) });
MapDef(\test).addOutputResponder(\respOutput, \output, { |vv| [\respond, vv, ~prWait.
()].postln });
MapDef(\test).start;
m.set(\input, 22.rand);
MapDef(\test).at(\output)
MapDef(\test).stop;
```