Decker hacking

This page documents the modifications I've been making to Decker and to lil/lilt, the programming/scripting language it has.

Patches

Patch to change the prompt to '%>' and make it return a newline on exit

C implementation notes

It seems like the norm is for interfaces to be defined in dom.h within the constants block. In order to avoid dom.h containing all the functionality implementations, it uses forward refs.

```
lv *interface_x(lv *self, lv *i, lv *x);
```

And then in the constants block, it's set as

```
dset(env, lmistr("x"), lmi(interface_x, lmistr("x"),NULL));
```

Later on, we can define the interface:

```
lv *interface_x(lv *self, lv *i, lv *x) {
  return x?x:NONE;
}
```

Methods within the interface can be defined by first defining the actual function:

```
lv *n_x_method(lv *self, lv *a) {
  if(a || !a) (void)self; return NONE;
}
```

followed by adding it in the interface definition with

```
ikey("method") return lmnat(n_x_method, self);
```

I think that if the last argument to lnmat isn't self then it would be a global method rather than part of the 'x' interface.

Notes on the types

to turn a char* into a lil string:

```
lv *r = lmutf8(chrstr)
```

don't forget to free() the chrstr

to get the first arg of the lv *a inside a method, use l_first(a)

methods i don't understand the function of yet:

lis(lv*) lil(lv*) lmistr(lv*) lmnat(lv*,lv*) - i think this is signalling that the method in question is a native function?

also what are x and i

API docs now that i understand it a bit better

`lv` is a variable that's a struct. the actual `char *` is member `sv`.

When defining an interface with `ikey`, the second arg can be lmnat if it's supposed to call a function with arguments (a native method) or it can be just a function call that returns an lv* or it can be just an lv* itself, if it should be a member and not a method.

invoking a method via lmnat will cause the first arg to be lv *self and the second to be lv *a. i'm unsure what self is, but a is a list of arguments.

lmnat, lms and so on are all declared via the lm(str, int) macro

From:
https://wiki.pup.casa/ - **pup.casa Docs**

Permanent link:
https://wiki.pup.casa/notes:decker_hacking

Last update: **2024/04/15 20:43**