SBehavEd language documentation

Nils Laurent, 2A Apprentice at Ensimag

May 31, 2018

1 Values

Regexp definition of values

1.1 Identifier

`[a-zA-Z]+[a-zA-Z0-9]*`

1.2 String

`"[a-zA-Z0-9]*"`

1.3 Number

`[0-9]+`

2 Variables

2.1 Strings

```
variable_name = "string"
```

2.2 Arrays

```
variable_name = ["array", "of", "strings"]
```

3 Identify BehavEd code and positions

3.1 Identify BehavEd code

Use of `«identifier»` in comments surrounding BehavEd source code

3.2 Identify positions in code

Use of `«@identifier»` to create a label

3.3 Example

Copy animation of the "jawa" entity to the "rax" entity
In the BehavEd source file `source.txt` :

```
affect ( "jawa", /*@AFFECT_TYPE*/ FLUSH )
{

        rem ( "<<jawa_dance>>" );

        task ( "back" )
        {
                set ( "SET_ANIM_BOTH", "BOTH_ATTACK_BACK" );
        }


        loop ( -1 )
        {

                loop ( 1 )
                {
                        do ( "back" );
                        wait ( 300.000 );
                }

        }

        rem ( "<<jawa_dance>>" );

}

affect ( "rax", /*@AFFECT_TYPE*/ FLUSH )
{

        rem ( "<<@rax_affect>>" );

}
```

In the SBehavEd source file `source.sb` :

```
behaved_factor_code(rax_affect , jawa_dance)
```

4 Functions

4.1 caffect_multiple

prototype

```
caffect_multiple(name_list, affect_type, destination_label)
```

```
caffect_multiple(name, affect_type, destination_label)
```

argument type

```
name : String
name_list : Array of strings
affect_type : Identifier
destination_label : Identifier
```

4.2 list_str

prototype

```
list_str(name, start_number, end_number)
```

argument type

```
name : String
start_number : Number
end_number : Number
```

return value

```
RETURN VALUE : Array of strings
```

4.3 behaved_factor_code

prototype

```
behaved_factor_code(destination_label, code_identifier)
```

argument type

```
destination_label : Identifier
code_identifier : Identifier
```