Snap! (by Berkeley)

Andrea Sterbini – sterbini@di.uniroma1.it

cc BY NC SA

Snap! (by Berkeley)

Evolution of Scratch

“Scratch for the Computer Scientist”

Object orientation

Many extensions/libraries

Support for code documentation

Support for debugging

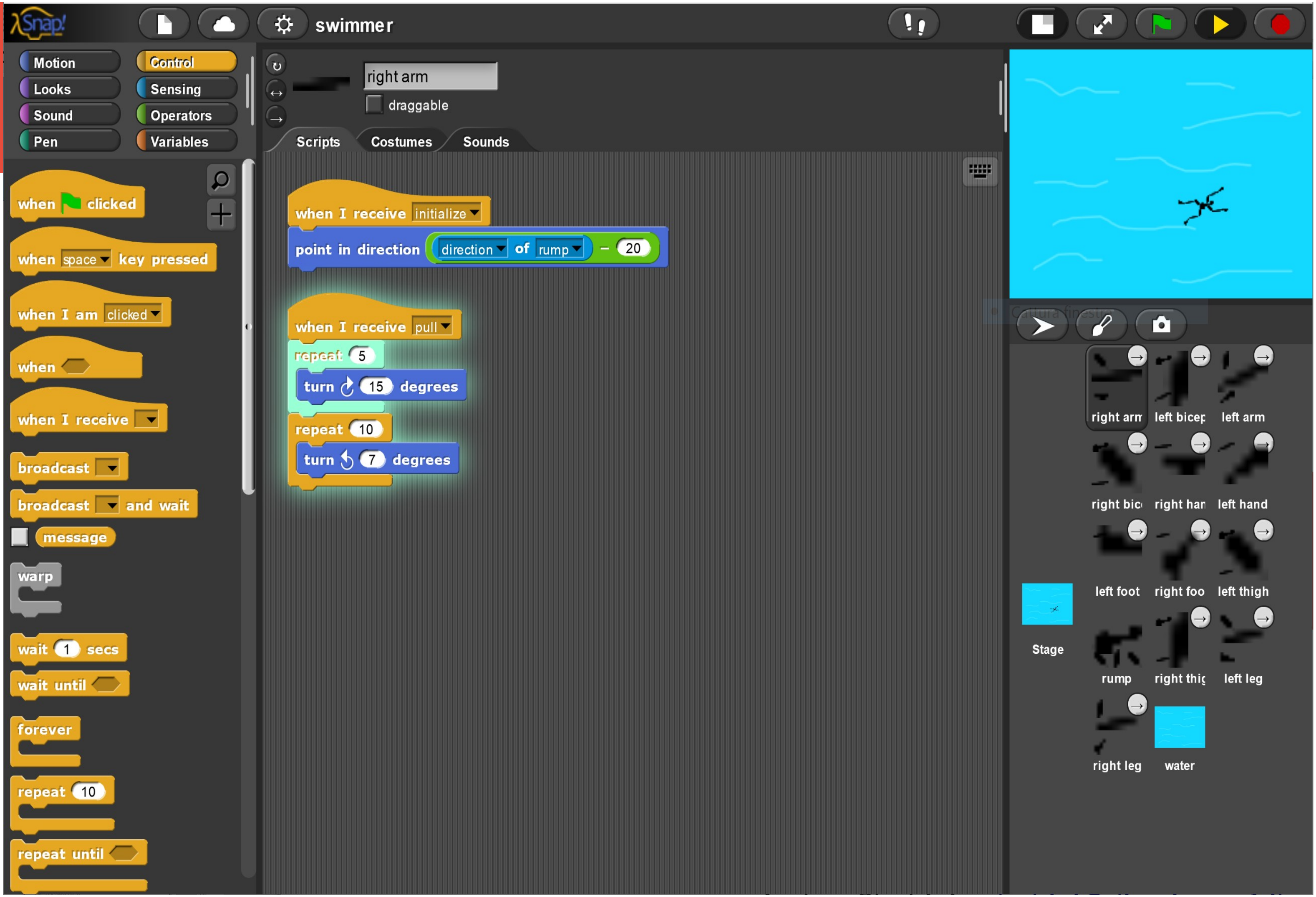
Concurrency

Coroutines

…

Music

Relative motion of sprites



```
when I receive initialize
point in direction ((direction of rump) - 20)

when I receive pull
repeat 5
  turn ↻ 15 degrees
repeat 10
  turn ↺ 7 degrees
```

Snap! improves many language constructs

Scratch

NO complex data

NO functions (only procedures)

NO local variables

NO references to clones

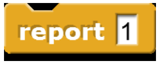
NO call methods

NO libraries

Snap!

Objects, Lists, Lists of Objects

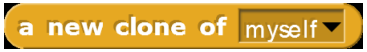
Functions



```
report 1
```

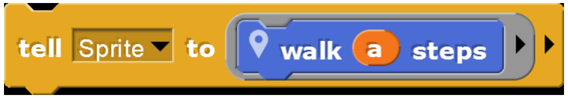
Local variables (easy recursion)

References to clones



```
a new clone of myself
```

Call methods

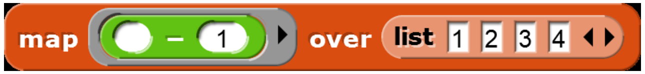


```
tell Sprite to walk a steps
```

Global blocks (library of functions)

Inheritance of clone properties

“Lambda” functions



```
map ( ( ) − 1 ) over list 1 2 3 4
```

Relative motion of Sprites/Agents

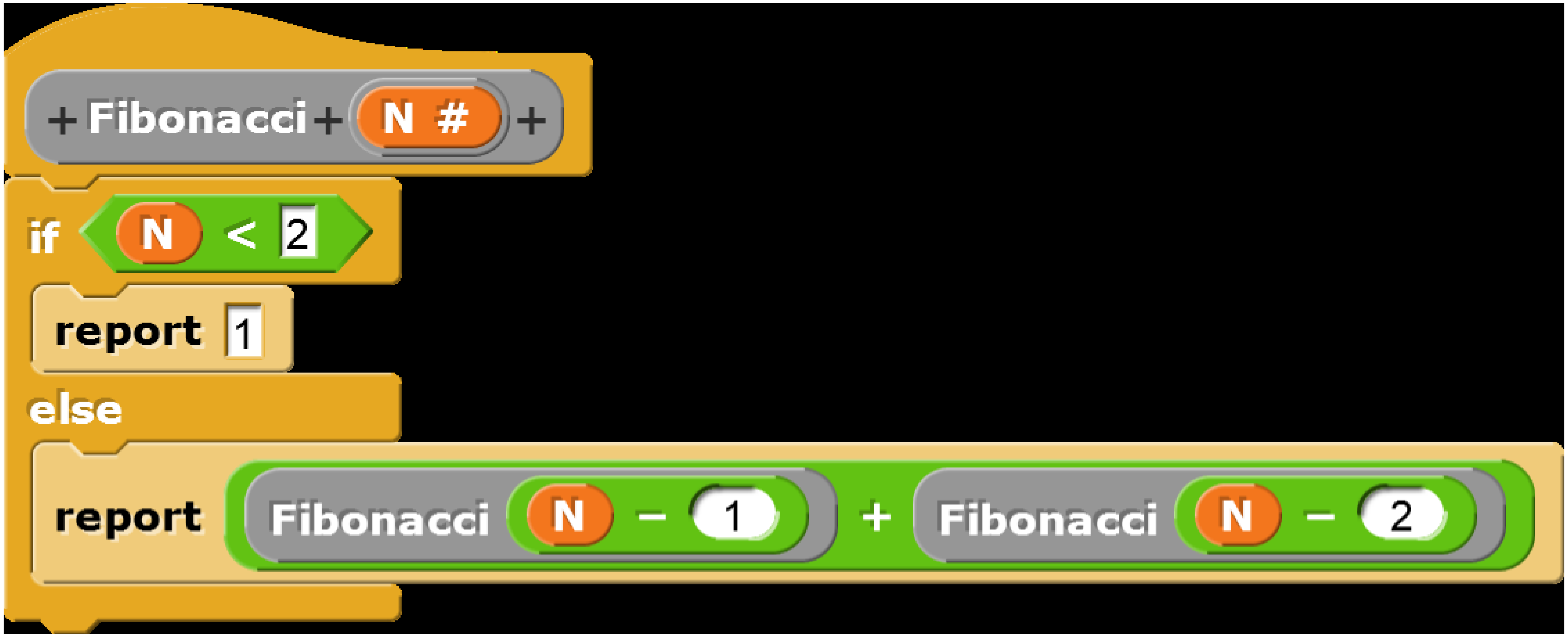
Relatively easy to build

- **collective motion of many clones (fireworks, snow, birds, …)**
- **coordinated motion of agent with many parts (man walking)**

Example: Swimmer

- Main motion: body trunk and head (bumping to the walls)
- Attached to body: thighs and biceps (rotating w.r.t. the body)
- Attached to thighs and biceps: arms and legs (staying in the same direction as the body)
- Attached to arms and legs: hands and feet (rotating w.r.t. the arm and leg)

Easy recursion



```
+ Fibonacci + (N #) +
if ⟨(N) < 2⟩
    report 1
else
    report ((Fibonacci ((N) − 1)) + (Fibonacci ((N) − 2)))
```

Standard Libraries/Extensions

Loops and compositions

List operations

Generators (lazy lists)

Multiple args operators

Web access

Words manipulation

Switch/case

RGB/HSV colors

Handle big lists

Frequency distribution analysis

Try/catch

Multiline input

GUI settings

Bignum, rational, complex

Text to speech

Animations

Image manipulation

Audio generation

Json

Parallelization

Other extensions

SOFTWARE:

Cellular automata (Cellular)

Graphs (Edgy)

NLP (NLTK wrapper)

HARDWARE:

Orbotix Sphero

Lego NXT (but not EV3 yet)

Wiimote

Arduino

Raspberry Pi

Speech synthesis

LEAP

Finch, Hummingbird

Styles of programming

Functional

Lists, filters, map, coroutines

Procedural

Concurrent

Concurrent execution

Message events

Object-oriented/Agent based

Agent properties, Agent methods

Clones: references to created clones, inherited properties