

# Concept-, Software- and Net-Art

Contextualisation – project module „Patterns“

# The Project

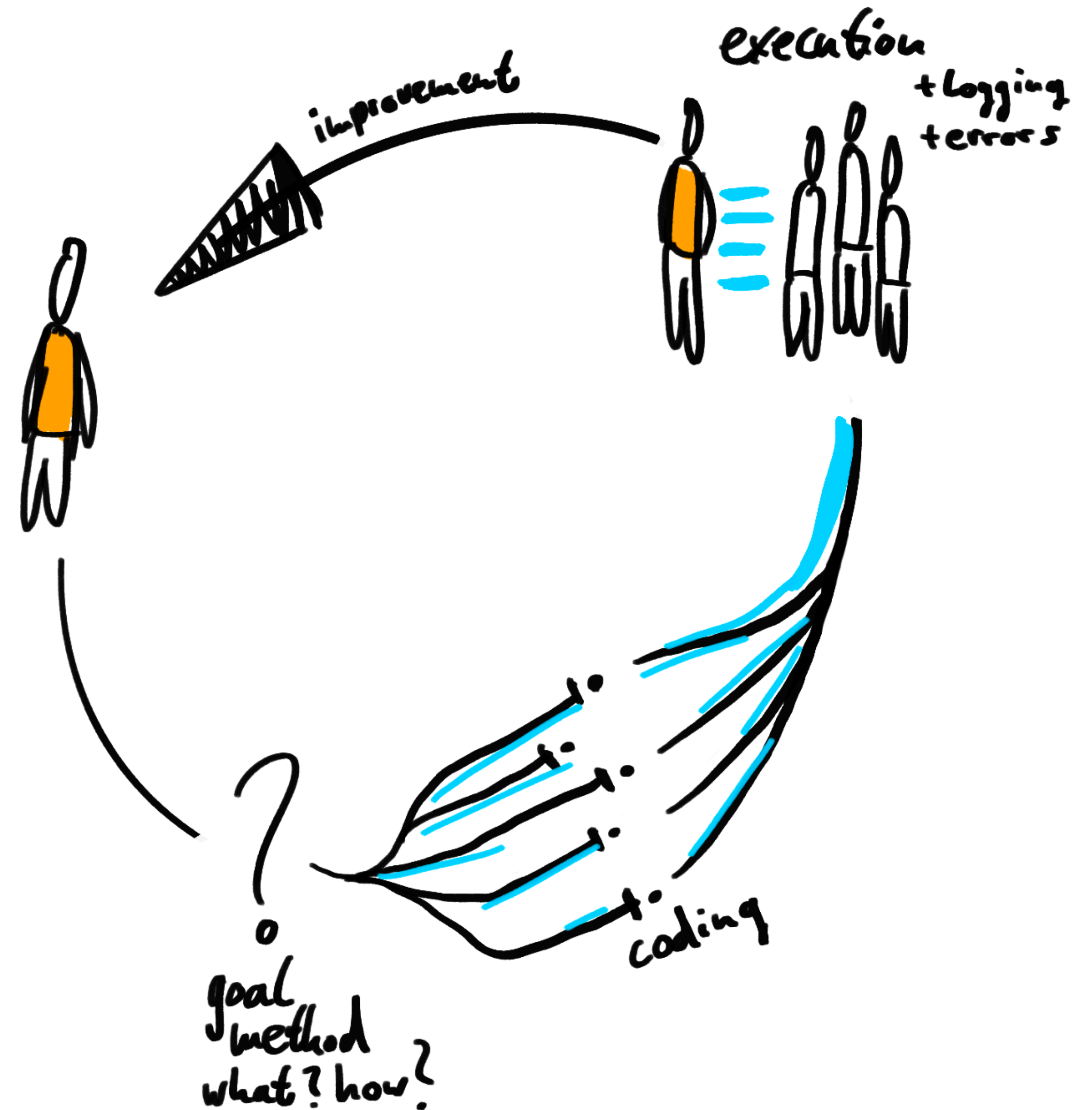
**Social interactions** can sometimes be genuinely difficult, bring uncertainty, sometimes even fear, and can be hard to understand or make sense of.

Rooted in personal experience, this project attempts to change that by applying the **methodology of software development** to social situations.

Software development typically begins with a problem, a goal, or a desire. The approach is pragmatic and structured: **define the goal, implement, execute, log, analyze errors, fix them**. This cycle repeats until a solution emerges that meets the original goal.

Now that goal is a social connection. The **human becomes the computer**, executing a social interaction according to a script that is revised and improved after each run.

Feeling overwhelmed is an error, and **errors can be handled** within the script: they are not failures, but expected edge cases. This makes difficult situations tangible and manageable.



# The Project

**Social interactions** can sometimes be genuinely difficult, bring uncertainty, sometimes even fear, and can be hard to understand or make sense of.

Rooted in personal experience, this project attempts to change that by applying the **methodology of software development** to social situations.

Software development typically begins with a problem, a goal, or a desire. The approach is pragmatic and structured: **define the goal, implement, execute, log, analyze errors, fix them**. This cycle repeats until a solution emerges that meets the original goal.

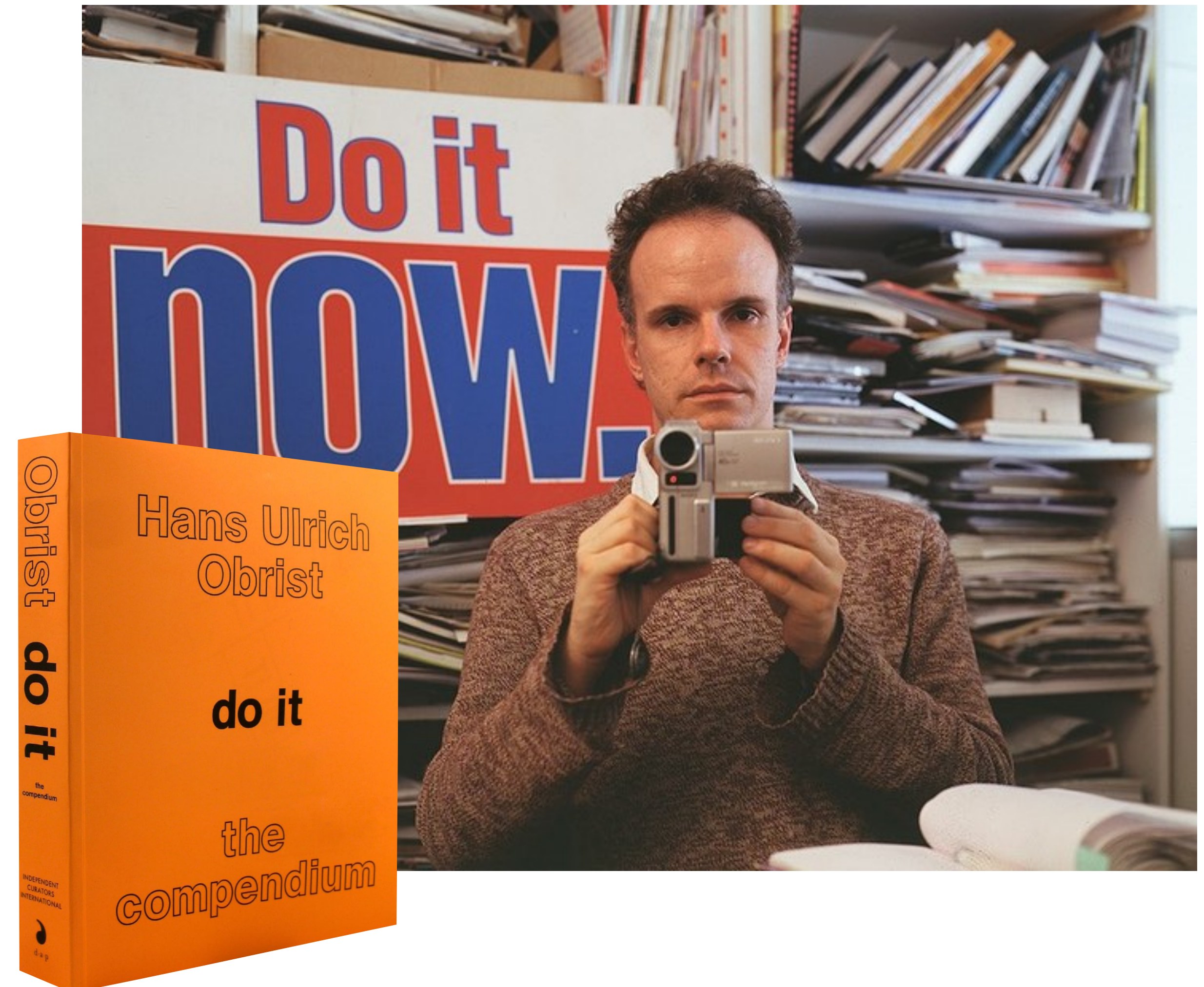
Now that goal is a social connection. The **human becomes the computer**, executing a social interaction according to a script that is revised and improved after each run.

Feeling overwhelmed is an error, and **errors can be handled** within the script: they are not failures, but expected edge cases. This makes difficult situations tangible and manageable.

```
14 // Execution
15
16 // — phase 1: arrive —————
17 arrive("the space")
18 breathe(3) // before doing anything else
19 observe("the room", 2) // 2 minutes, no interaction yet
20 LOG "arrival" { time, location, your mood on entry }
21
22 // — phase 2: approach —————
23 choose(["one person not already in a conversation"])
24 approach("within speaking distance, not directly facing")
25 make_eye_contact() // wait for it – do not force it
26 IF eye_contact_returned == false
27 LOG "approach" { "no eye contact – choosing different person" }
28 choose("a different person") // restart phase 2
29 CATCH "too anxious to approach anyone"
30 breathe(4)
31 sit("somewhere visible") // let someone approach you instead
32 wait("until anxiety has a shape you can name")
33 LOG "error" { "what specifically felt impossible about approaching" }
34
35 // — phase 3: open —————
36 greet() // say hello. nothing more yet.
37 offer("one honest observation about the shared situation")
38 // not a compliment. not a question. a remark.
39 listen()
40 LOG "opening" { your exact words, their response, their affect }
41 IF they_responded == false
42 LOG "opening" { "no response – run ends here. not an error." }
43 RETURN goal_met = false
44 CATCH "mind went blank"
45 say("what you are actually noticing right now") // honesty is a valid opening
46 breathe(2)
47 LOG "error" { "what you were going to say before the blank" }
48
49 // — phase 4: exchange —————
50 ask("one question that cannot be answered with yes or no")
51 // about the situation, not about them personally yet
52 listen() // do not plan your next sentence while they speak
53 respond("to what they actually said, not what you expected")
54 IF they_ask_something_back == true
```

# do it – Hans Ulrich Obrist

- Comes up 1993 in a talk between artists Christian Boltanski and Bertrand Lavier and curator Hans Ulrich Obrist
- Experimenting with how exhibition formats could be rendered more flexible and open ended
- Takes „scores“, written instructions, as departure point
- 13 artists send instructions which were then distributed internationally as a book
- Exhibitions happened all over the world
- New book in 2013, new instructions in 2020



# do it – Hans Ulrich Obrist

Jonas Mekas (1996)

**MEKAS, Jonas**  
*Instruction* (1996)

Do it...  
move your finger  
up and down for  
one minute every morning



# do it – Hans Ulrich Obrist

James Bridle (2020)

Open any web page on a computer. (If you're already on the web, use this one).

Right-Click anywhere on the page, and select "Inspect".

(Or press Command-Option-I or Control-Shift-I).

In the new window, look for a button marked "Console" and click it (if you can't find the console, ask the internet for help).

In the Console space (marked by '>'), type the following:

```
y = document.getElementsByTagName('*'); function z() { for (i=0; i < y.length; i++)  
{ y[i].style.fontSize = Math.random()*100+"px"; y[i].style.backgroundColor='#'+  
(Math.random()*0xFFFFFFFF<<0).toString(16); } } setInterval(z, 0);
```

Press enter.

Next, write your own.

# runme.org

- Collaborative, open-access database dedicated to "software art"
- By international group of artists, theorists, and programmers
- Launched in 2003, highly active early 2000's, active maintaining stopped around 2012
- Now historical archive of early digital and net art and the software art movement
- Interesting: people thought really deeply about code and the web

The screenshot shows the runme.org website interface. At the top, there is a navigation bar with the site name and tagline "runme.org - say it with software art!". Below this is a search bar and a menu with links for "front", "latest", "featured", "categories", "keywords", "news archive", "submit a project", "read\_me festival", "about", "faq", and "feedback".

The main content area is divided into several sections:

- category list**: A list of categories with their respective counts, such as "algorithmic appreciation (2)", "artificial intelligence (11)", "code art (26)", and "digital aesthetics r&d (12)".
- archive mode**: A notice stating that the site is in archive mode and that users should continue to browse projects here.
- latest projects**: A list of recent projects with their titles and dates, including "Pellow [13 Oct 2020]" and "100.000.000 stolen pixels [16 Jun 2011]".
- featured projects**: A list of featured projects, including "LYCAY (Let Your Code pLAY)" and "The Invisible Hand Machine".
- news**: A section titled "Social Bits!" containing news about new projects and a "Sneak Preview of Runme's New Website Design".
- keyword cloud**: A cloud of words representing various themes and topics, such as "sexy", "pragmatic", "symbiosis", "mindware", "subversive", "post-typing", "backwards", "anachronistic", "realtime", "meme", "algorithmic", "subjectivity", and "philosophical".

**Software Art incorporates projects in which self-written algorithmic computer software is not merely a functional tool, but is itself an artistic creation.**

**The development of computer hardware and the global network infrastructure provide ever-extending possibilities for social and creative interactions between people. These possibilities have to be designed and coded in software in order to be realised. Thus, software becomes a crucial catalyst for the emergence of the Information Society and its social and cultural structures.**

*– [runme.org](http://runme.org) (*Transmediale 2001*)*

# runme.org

- [iSee](#) – Mapping software to give route with the fewest surveillance cameras in Manhattan
- [Evil](#) – Displays phone numbers of random Facebook users – Shows that data was publicly visible in various groups throughout Facebook
- [CodeSounding](#) – Translates code into sound by assigning instruments and pitches to coding statements (like if, for, function etc.) and plays them as program is executed
- [flexplorer](#) – Displays any website as a mind map
- [GoogleSynth](#) – Merges two images via different parameters to make a new ,artwork‘
- [brainfuck](#) – "Esoteric" programming language consisting of only 8 characters (<>+-[] ,.) – very much into theoretic computer science

# runme.org

## Drunk Eliza

- Eliza by Joseph Weizenbaum but written in Entropy
- Entropy: All data decays as the program runs (all numbers and characters vary more and more the longer the program runs)
- Addresses compulsive thinking being reinforced by programming
- Therefore thinks about the human behind the code and using the code

RUNK ELIZA \*\*\*

ON ELIZA CODE BY JOHN SCHUGG, JANVARY 1985 ORIG  
HELP YOU !

ELIZA: HI! I'M ELIYA. WHAT'S YOUR PROBLEM?  
I don't have a problem

EMIZA: WHY DON'T YOU HAVE A PROBLEM?

I dont know why, do you?

EMIZA: DON'T YOU SEALLY KNOW WHY, DO I?

Do you=

EMHZA; OH, I =.

I meant ? instead of =

DMHZA; SAY, DO XOU HAVE ANY PTYCHOLPGICAL PROBL

Oh that's a very personal question

DLHZA; OH, I ESTION.

Who is Estion?

DLHZA; WHAT ANSVER WOULD PLEASE ZOU THE MOST?

I dont get you

DLIZA; XHY DON'T YOT GET I?

Because you are drunk

DLHZB; DOES IT PLEASE YOT TO BELIEVE I AM DRUNK

No not really, makes me uncomfotable

DLHZA< WE WERE DISCUSSING YOU-- NOT ME.

Who said that?

DLIZB; WHAT ANSWER WOULD PLEBSE YNU THE NOST?

# runme.org

Drunk Eliza

Entropy addresses the compulsive thinking reinforced by programming, the rigidity of logic, and the way we must compromise with the computer in order to get it to understand us. It is something that has become more familiar to non-programmers as people manage more of their social interactions and other messy human stuff through computers.

– *Esolang Wiki*

RUNK ELIZA \*\*\*

ON ELIZA CODE BY JOHN SCHUGG, JANVARY 1985 ORIG  
HELP YOU !

ELIZA: HI! I'M ELIYA. WHAT'S YOUR PROBLEM?

I don't have a problem

EMIZA: WHY DON'T YOU HAVE A PROBLEM?

I dont know why, do you?

EMIZA: DON'T YOU SEALLY KNOW WHY, DO I?

Do you=

EMHZA; OH, I =.

I meant ? instead of =

DMHZA; SAY, DO XOU HAVE ANY PTYCHOLPGICAL PROBL

Oh that's a very personal question

DLHZA; OH, I ESTION.

Who is Estion?

DLHZA; WHAT ANSVER WOULD PLEASE ZOU THE MOST?

I dont get you

DLIZA; XHY DON'T YOT GET I?

Because you are drunk

DLHZB; DOES IT PLEASE YOT TO BELIEVE I AM DRUNK

No not really, makes me uncomfotable

DLHZA< WE WERE DISCUSSING YOU-- NOT ME.

Who said that?

DLIZB; WHAT ANSWER WOULD PLEBSE YNU THE NOST?

# runme.org

- Went pretty deep into playing around with the code world itself
- Seems deprecated because technology is far more evolved now
- Still very relevant because it questions the basics of the web and coding
- Art like this is still happening
- E.g. Jmail by Luke Igel and [Riley Walz](#)
  - Replica of Gmail client of Jeffrey Epsteins inbox
  - Riley Walz: [IMG\\_0001](#), [Panama Playlists](#), [BOP Spotter](#)

