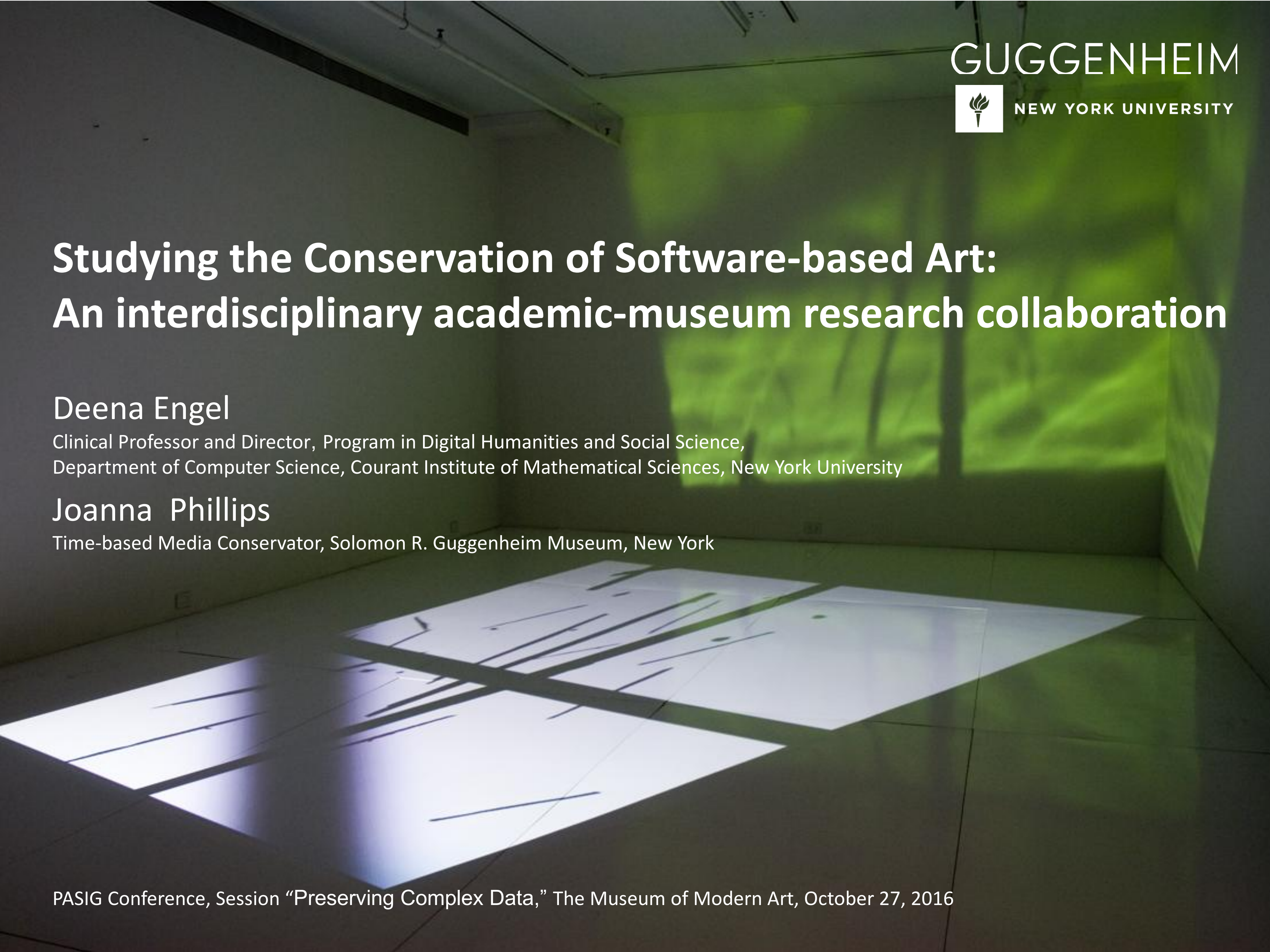


GUGGENHEIM

NEW YORK UNIVERSITY

Studying the Conservation of Software-based Art: An interdisciplinary academic-museum research collaboration

Deena Engel

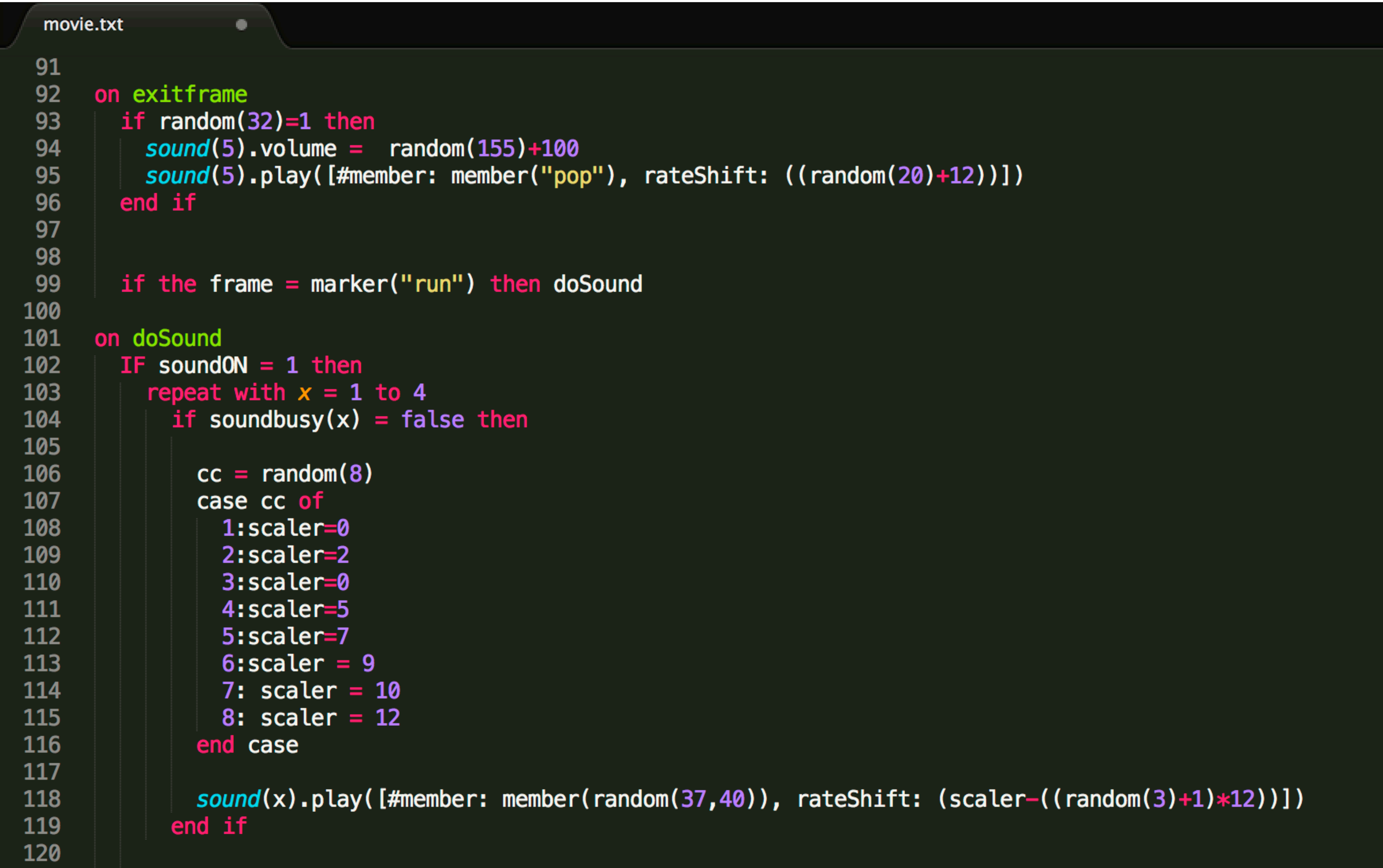
Clinical Professor and Director, Program in Digital Humanities and Social Science,
Department of Computer Science, Courant Institute of Mathematical Sciences, New York University

Joanna Phillips

Time-based Media Conservator, Solomon R. Guggenheim Museum, New York

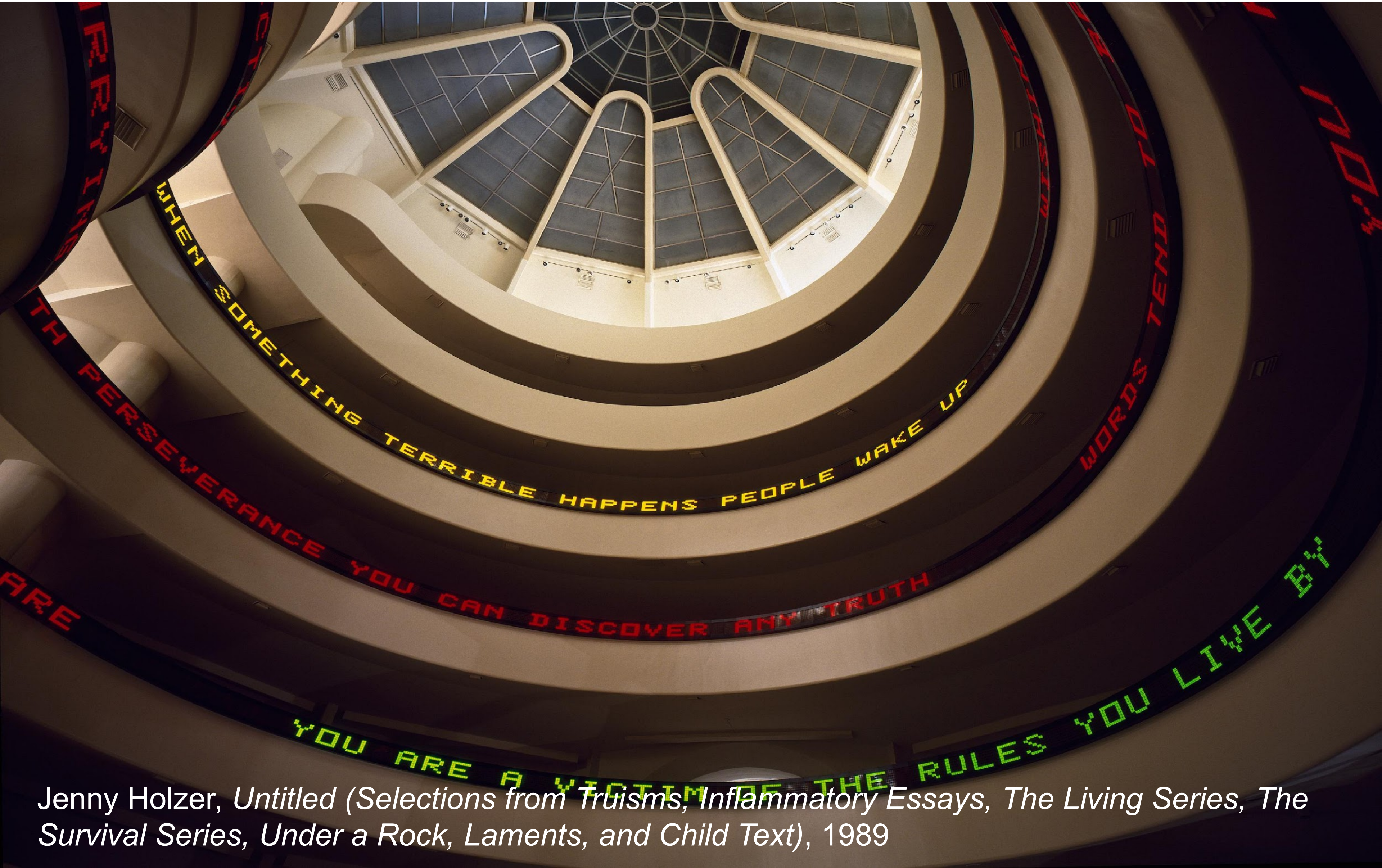
PASIG Conference, Session “Preserving Complex Data,” The Museum of Modern Art, October 27, 2016

The Conservation Object



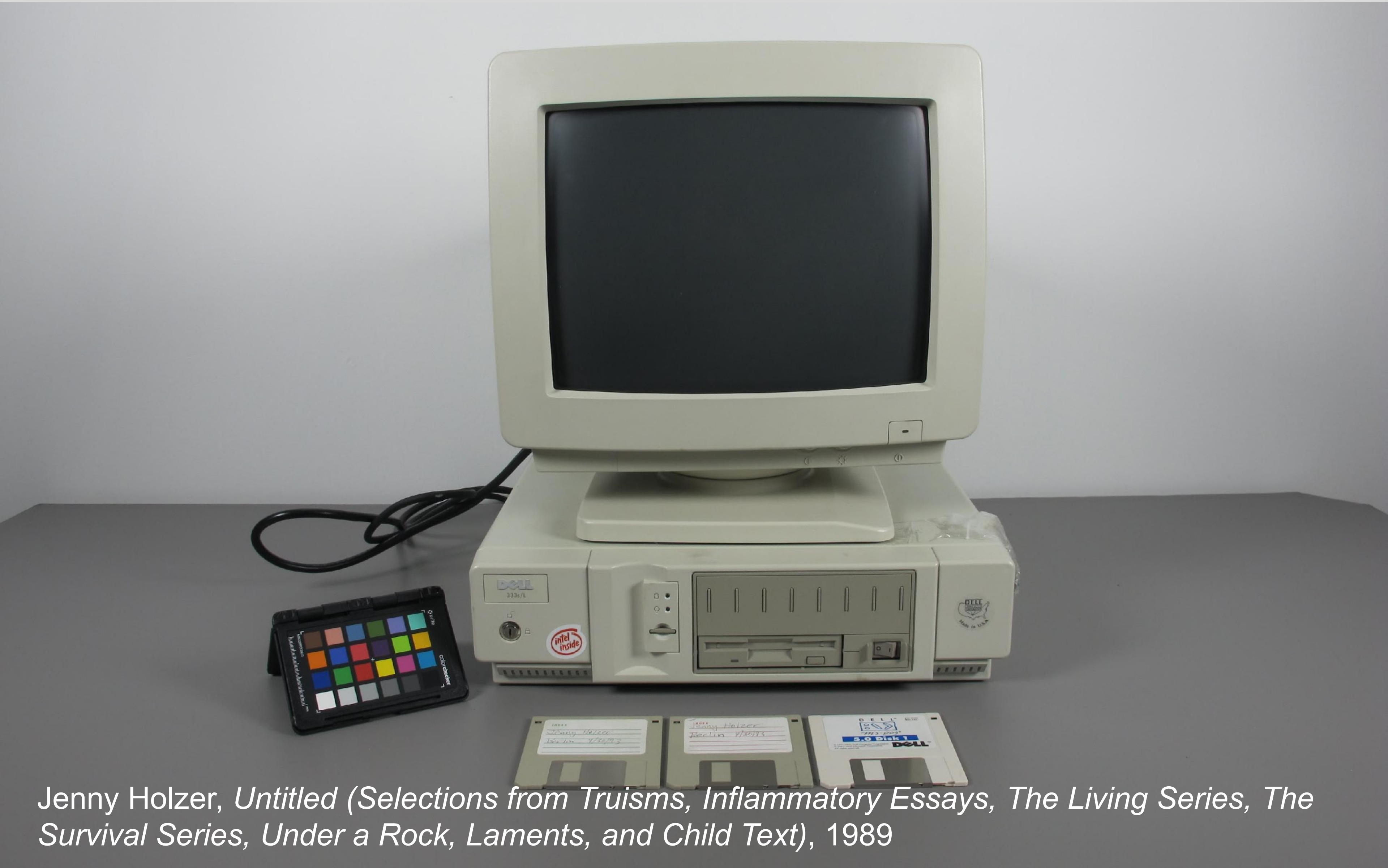
```

on exitframe
  if random(32)=1 then
    sound(5).volume =  random(155)+100
    sound(5).play([#member: member("pop"), rateShift: ((random(20)+12))])
  end if


  if the frame = marker("run") then doSound

on doSound
  IF soundON = 1 then
    repeat with x = 1 to 4
      if soundbusy(x) = false then

        cc = random(8)
        case cc of
          1:scaler=0
          2:scaler=2
          3:scaler=0
          4:scaler=5
          5:scaler=7
          6:scaler = 9
          7: scaler = 10
          8: scaler = 12
        end case

        sound(x).play([#member: member(random(37,40)), rateShift: (scaler-((random(3)+1)*12))])
      end if

```

The Conservation Object


WHEN SOMETHING TERRIBLE HAPPENS PEOPLE WAKE UP
TH PERSEVERANCE YOU CAN DISCOVER ANY TRUTH
ARE
WORDS TEND TO
YOU ARE A VICTIM OF THE RULES YOU LIVE BY

Jenny Holzer, *Untitled (Selections from Truisms, Inflammatory Essays, The Living Series, The Survival Series, Under a Rock, Laments, and Child Text)*, 1989

The Conservation Object

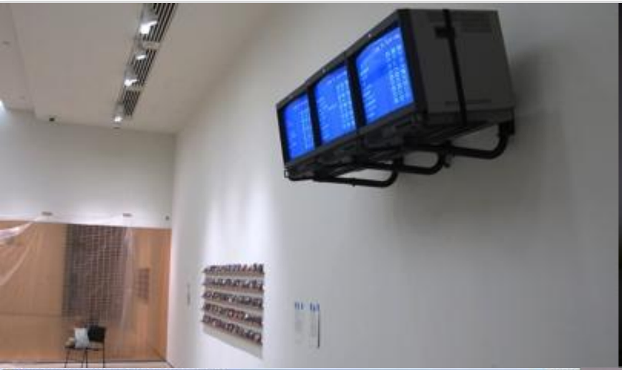


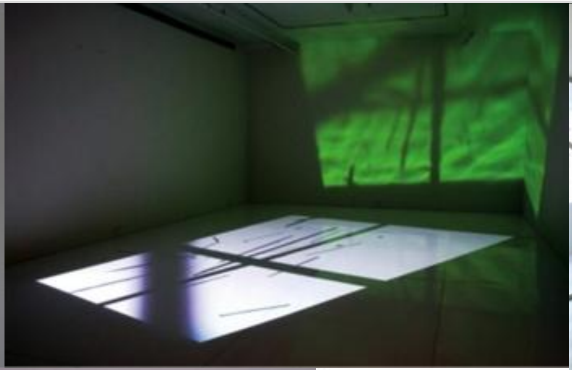
Jenny Holzer, *Untitled (Selections from Truisms, Inflammatory Essays, The Living Series, The Survival Series, Under a Rock, Laments, and Child Text)*, 1989

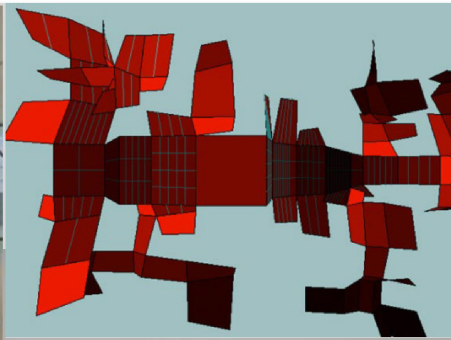
GUGGENHEIM

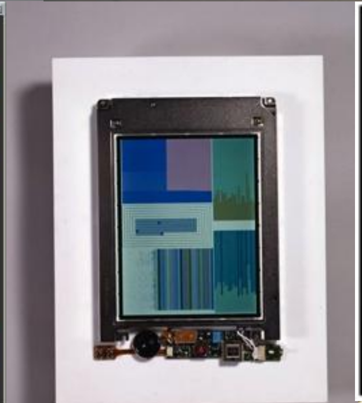
NEW YORK UNIVERSITY

The Conservation Object



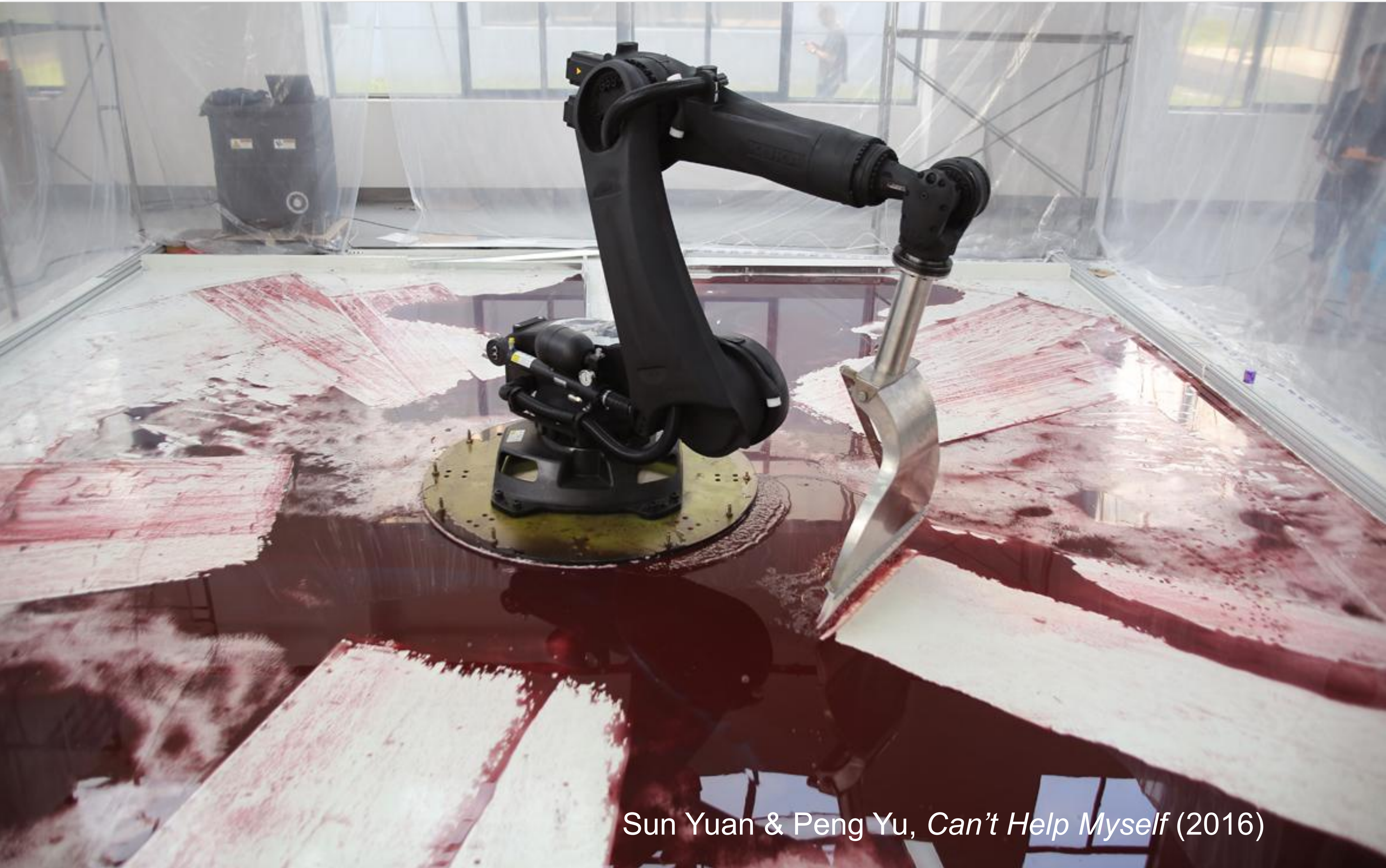






SOMETHING TERRIBLE HAPPENS PEOPLE WAKE UP
PERSEVERANCE YOU CAN DISCOVER ANY TRUTH
YOU ARE A VICTIM OF THE RULES YOU LIVE BY

The Conservation Object



Sun Yuan & Peng Yu, *Can't Help Myself* (2016)

Conservation Issues

SN21G5
AMD Athlon™ 64 / FX / X2 Processor Support (Socket 939)
NVIDIA® GeForce™ 6100 + nForce™ 410
1.0GHz HyperTransport™ Technology
Dual-Channel DDR 333/400 Memory Support (1GB per Slot, 2GB Max)
GeForce™ 6100 DirectX 9 Graphic Engine
PCI-E x16 Slot + PCI Slot
SATA2 Interface with RAID 0, 1, JBOD
SN21G50-L3GH00-
Green Product
Shuttle®
FRAG
FRAGILE
UP
ROSENQUIST
GUGGENHEIM
EXPECT
24 HOURS NOTICE PLEASE
New Acquisition - 2009

Conservation Issues

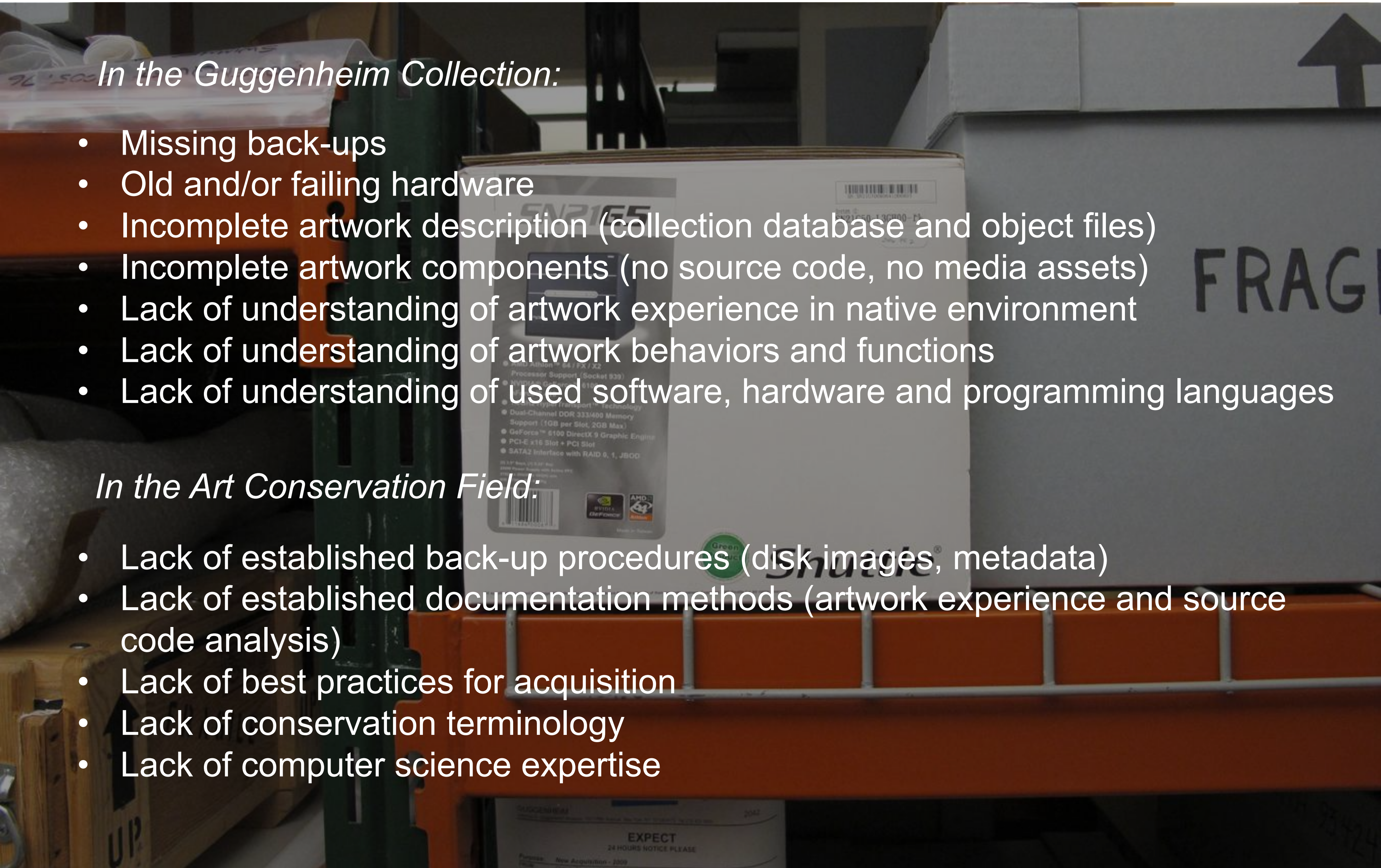
GUGGENHEIM

NEW YORK UNIVERSITY

In the Guggenheim Collection:

- Missing back-ups
- Old and/or failing hardware
- Incomplete artwork description (collection database and object files)
- Incomplete artwork components (no source code, no media assets)
- Lack of understanding of artwork experience in native environment
- Lack of understanding of artwork behaviors and functions
- Lack of understanding of used software, hardware and programming languages

Conservation Issues



In the Guggenheim Collection:

- Missing back-ups
- Old and/or failing hardware
- Incomplete artwork description (collection database and object files)
- Incomplete artwork components (no source code, no media assets)
- Lack of understanding of artwork experience in native environment
- Lack of understanding of artwork behaviors and functions
- Lack of understanding of used software, hardware and programming languages

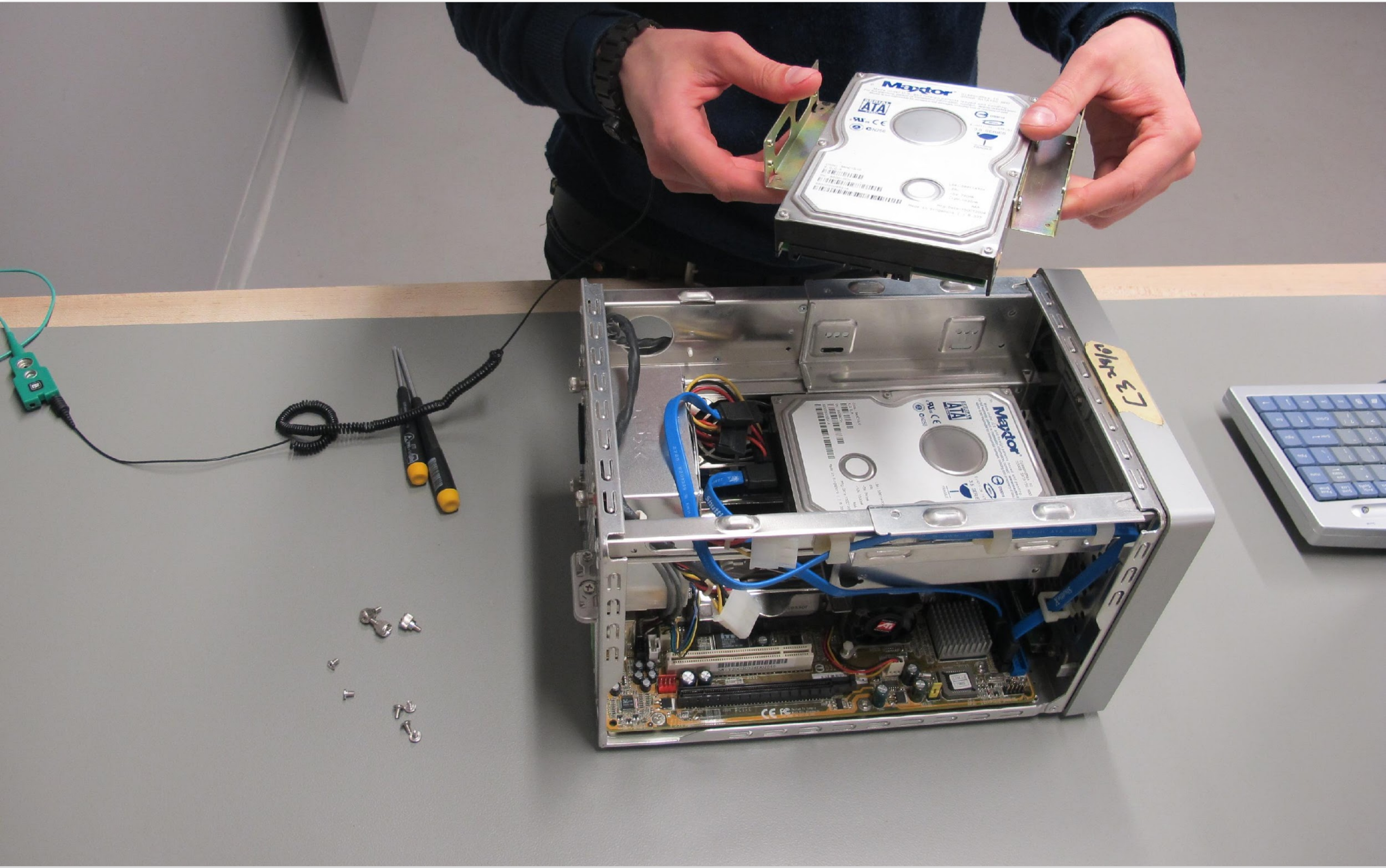
In the Art Conservation Field:

- Lack of established back-up procedures (disk images, metadata)
- Lack of established documentation methods (artwork experience and source code analysis)
- Lack of best practices for acquisition
- Lack of conservation terminology
- Lack of computer science expertise

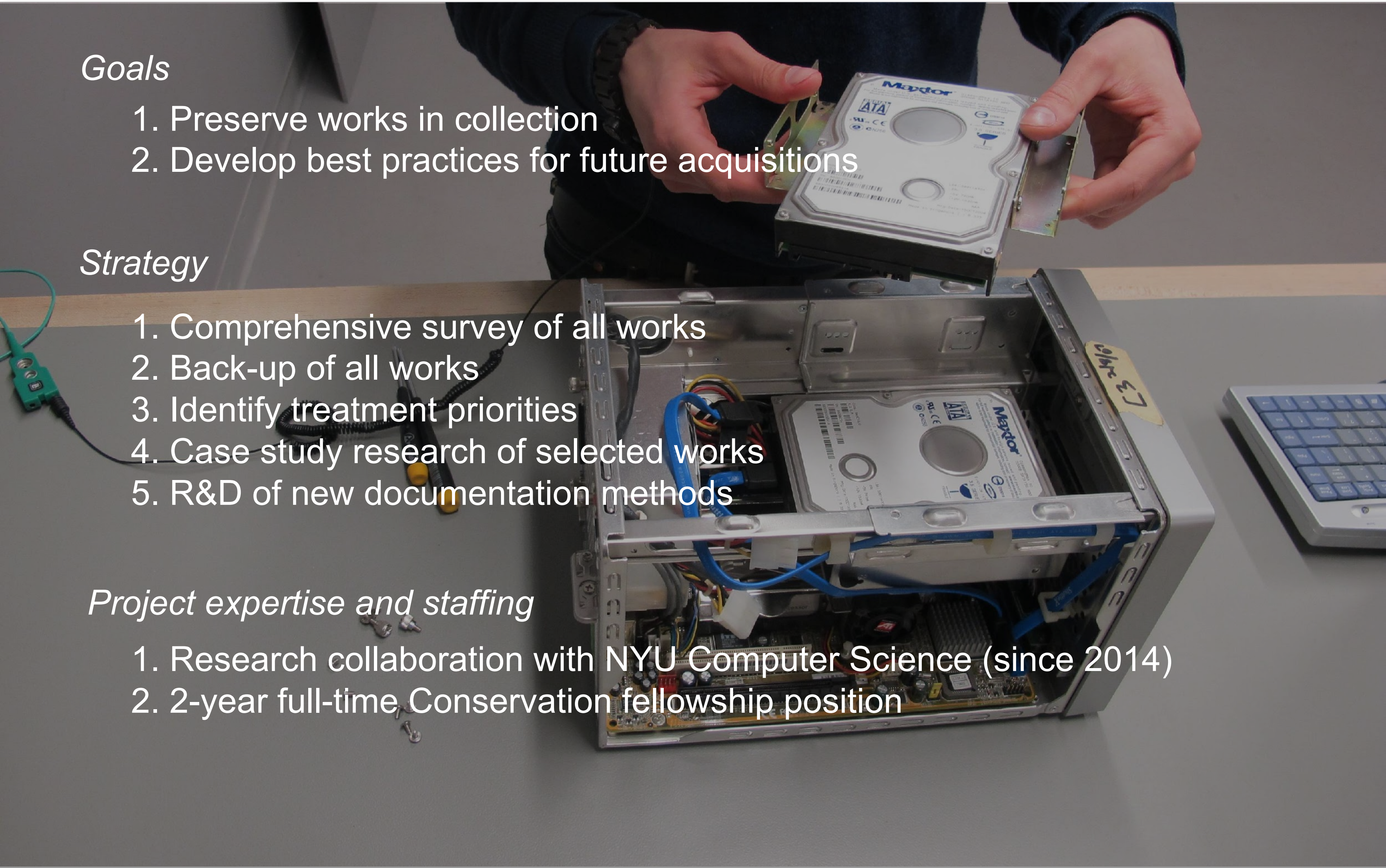
The CCBA Initiative

GUGGENHEIM

NEW YORK UNIVERSITY



The CCBA Initiative



Goals

1. Preserve works in collection
2. Develop best practices for future acquisitions

Strategy

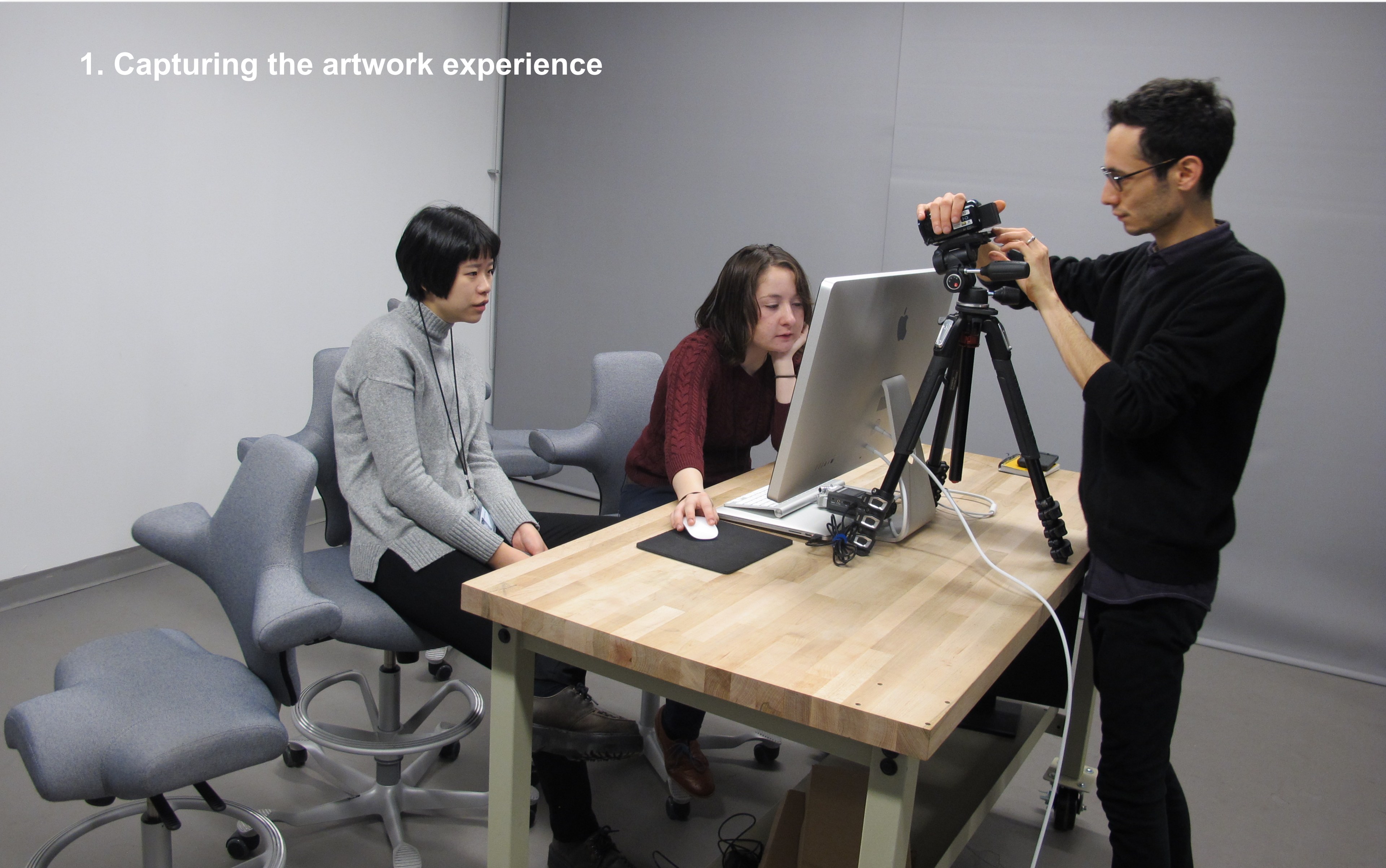
1. Comprehensive survey of all works
2. Back-up of all works
3. Identify treatment priorities
4. Case study research of selected works
5. R&D of new documentation methods

Project expertise and staffing

1. Research collaboration with NYU Computer Science (since 2014)
2. 2-year full-time Conservation fellowship position

CCBA Research and Development

1. Capturing the artwork experience

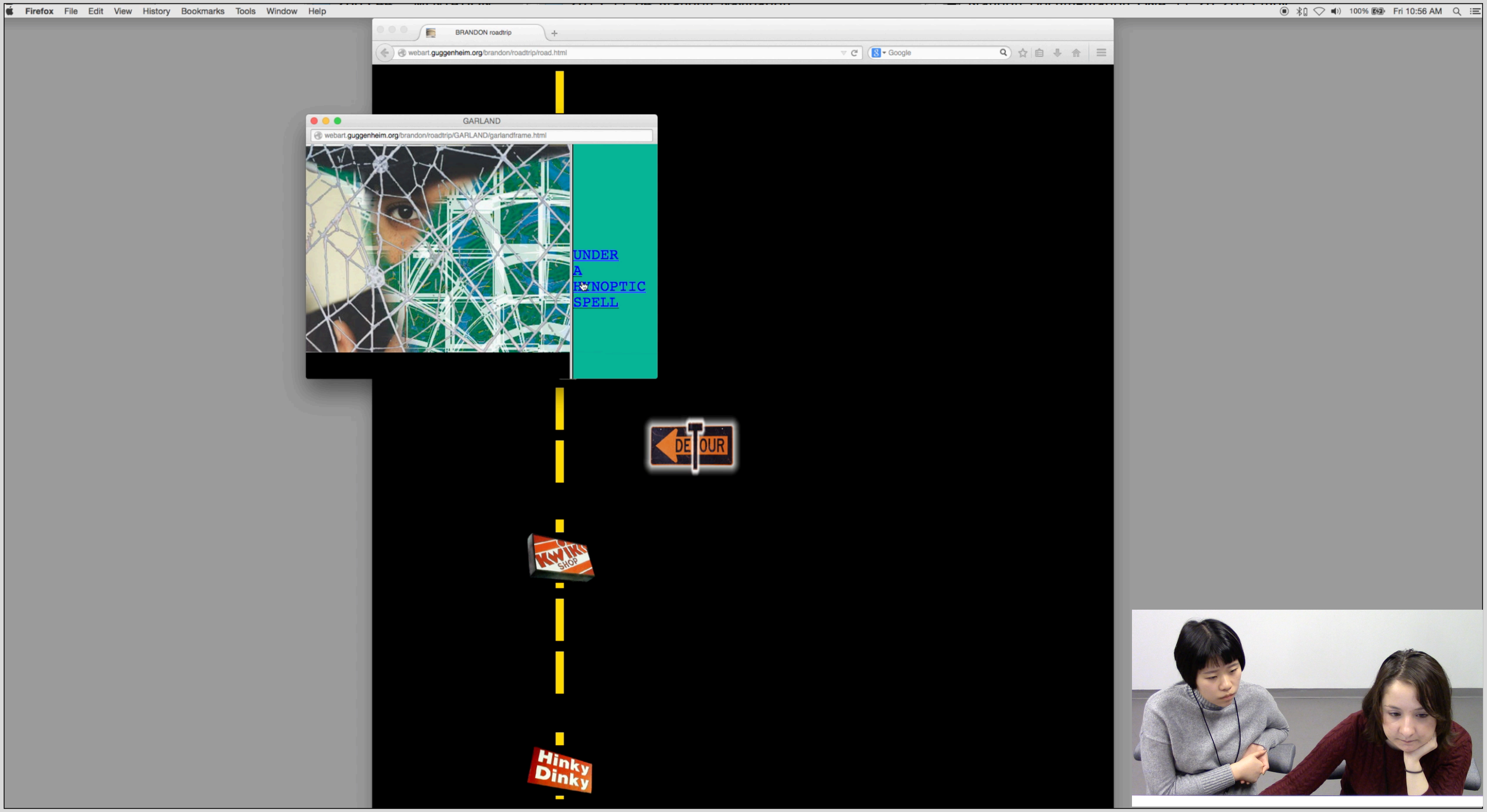


CCBA Research and Development

GUGGENHEIM

NEW YORK UNIVERSITY

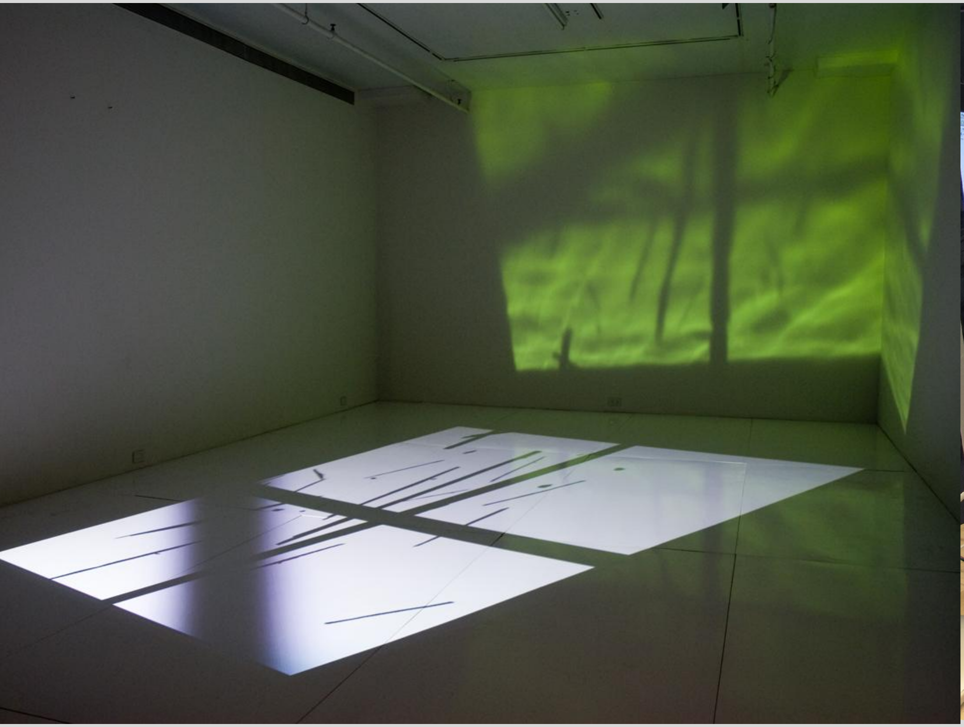
1. Capturing the artwork experience


Firefox File Edit View History Bookmarks Tools Window Help
100% Fri 10:56 AM
BRANDON roadtrip
webart.guggenheim.org/brandon/roadtrip/road.html
Google
GARLAND
webart.guggenheim.org/brandon/roadtrip/GARLAND/garlandframe.html
UNDER
A
HYNOPTIC
SPELL
DETOUR
KWIK SHOP
Hinky Dinky

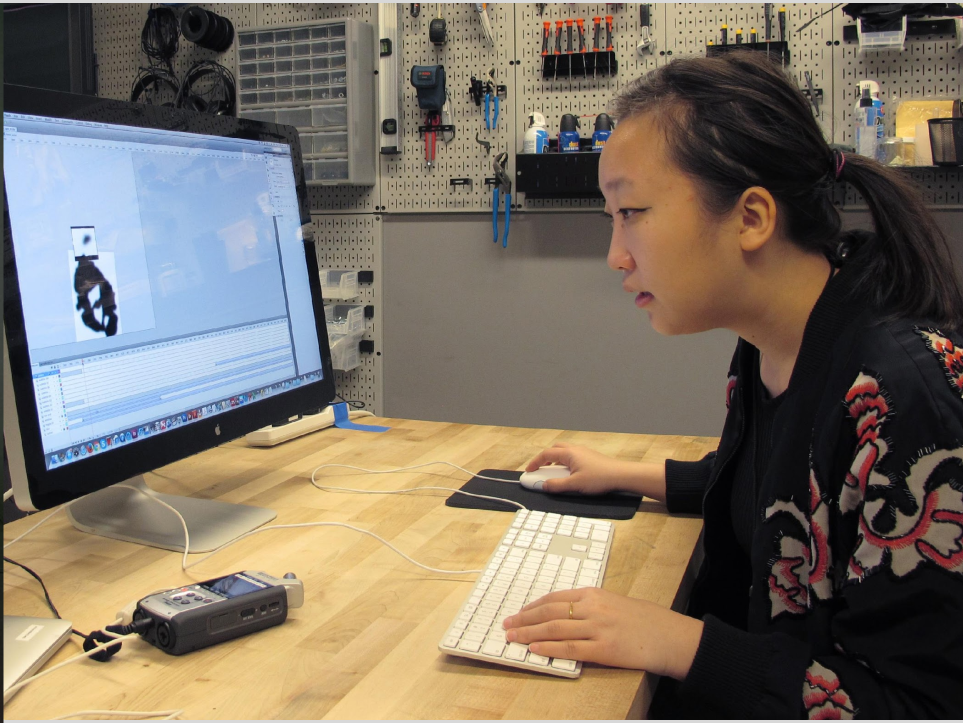
Narrated screen navigation of Shu Lea Cheang's *Brandon* (1998-1999)

CCBA Research and Development

2. Examination of native production environment

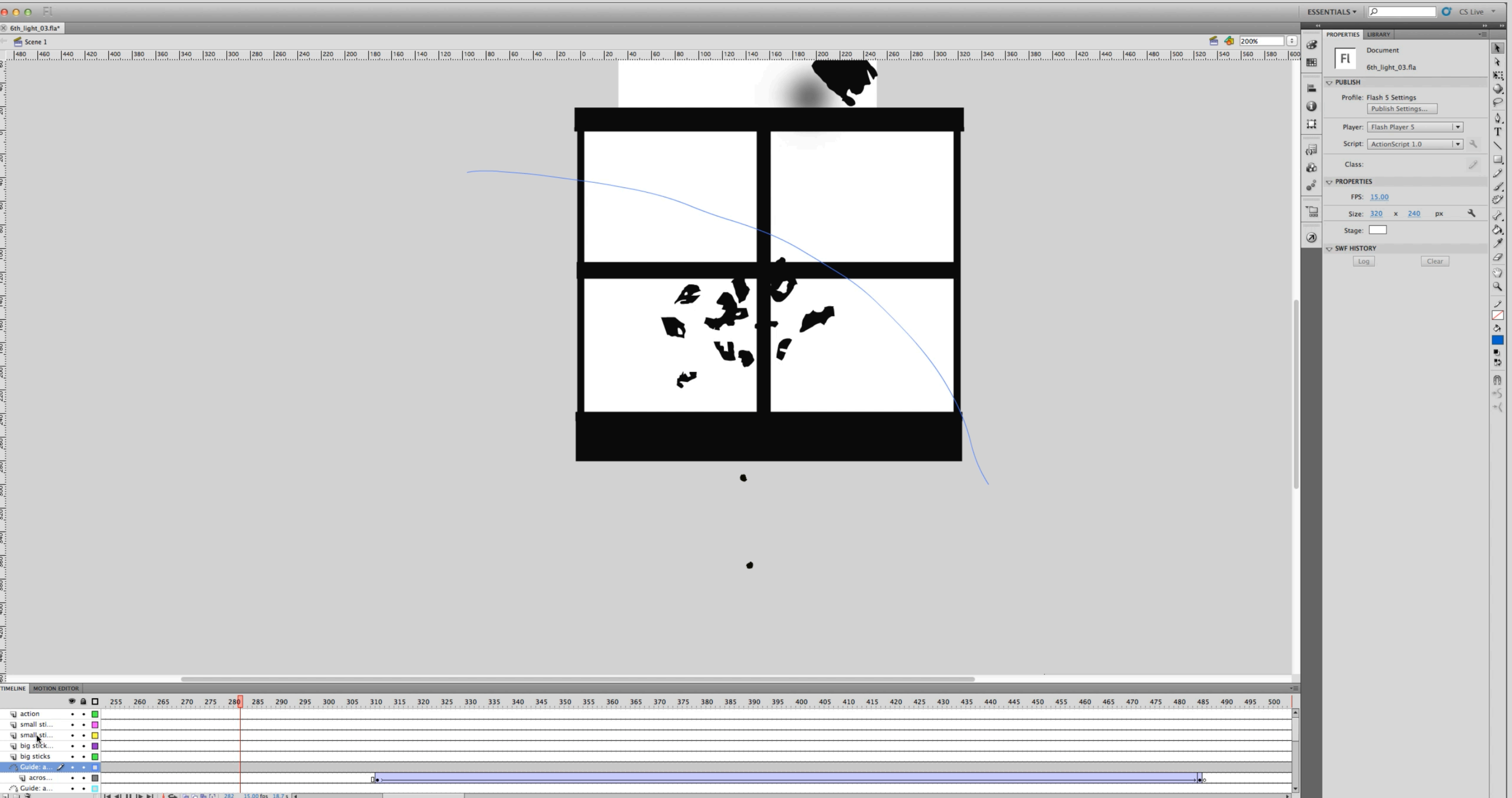


Paul Chan, *6th ~~Light~~* (2007)
Projected Flash Animation



Analysis and documentation of the artwork in its native production environment Adobe Flash

2. Examination of native production environment



6th_light_03.fla*

Scene 1

Document

6th_light_03.fla

PUBLISH

Profile: Flash 5 Settings

Player: Flash Player 5

Script: ActionScript 1.0

Class:

PROPERTIES

FPS: 15.00

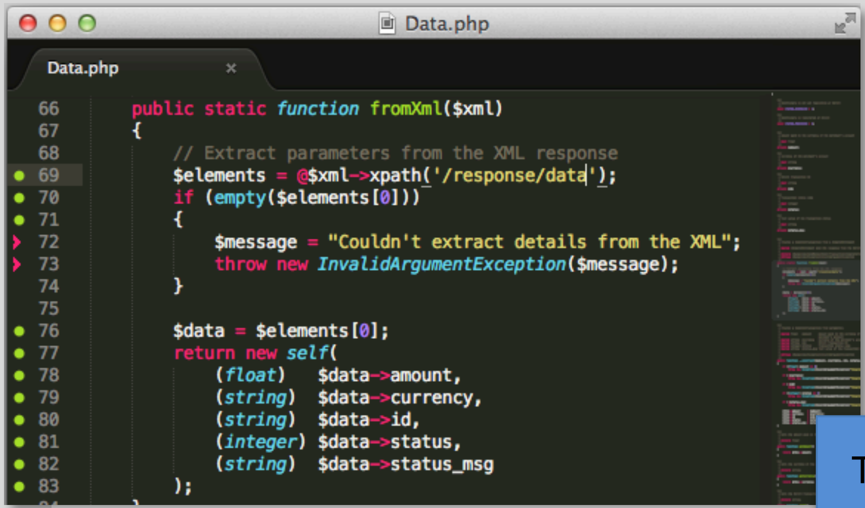
Size: 320 x 240 px

Stage:

SWF HISTORY

CCBA Research and Development

3. Source code analysis → Conservation documentation



```
    public static function fromXml($xml)
    {
        // Extract parameters from the XML response
        $elements = @$xml->xpath('/response/data');
        if (empty($elements[0]))
        {
            $message = "Couldn't extract details from the XML";
            throw new InvalidArgumentException($message);
        }

        $data = $elements[0];
        return new self(
            (float)   $data->amount,
            (string)  $data->currency,
            (string)  $data->id,
            (integer) $data->status,
            (string)  $data->status_msg
        );
```

Translation

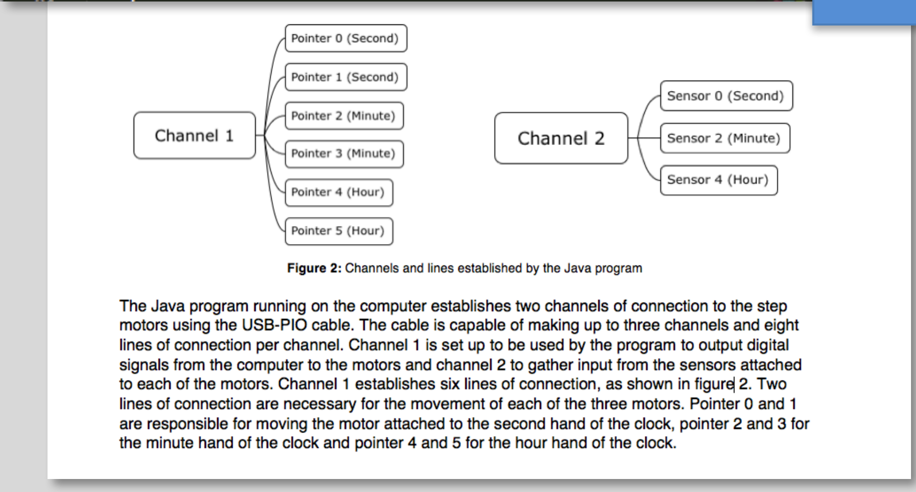
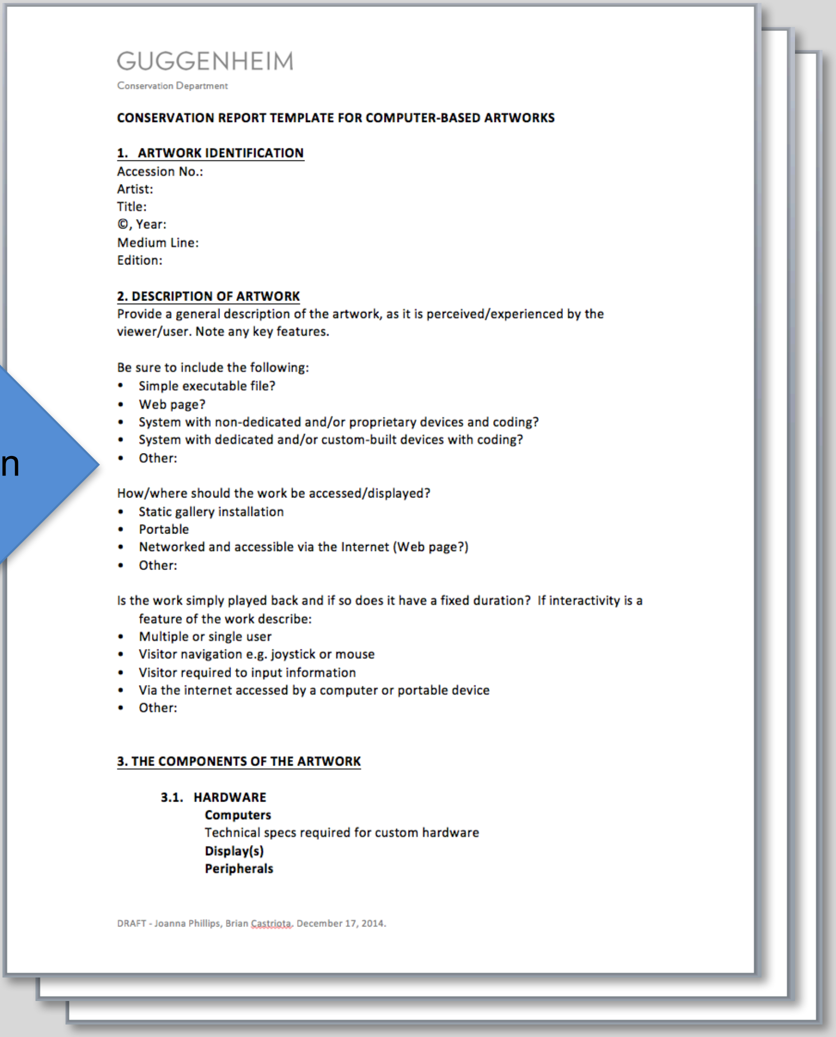

Channel 1
Pointer 0 (Second)
Pointer 1 (Second)
Pointer 2 (Minute)
Pointer 3 (Minute)
Pointer 4 (Hour)
Pointer 5 (Hour)
Channel 2
Sensor 0 (Second)
Sensor 2 (Minute)
Sensor 4 (Hour)

Figure 2: Channels and lines established by the Java program

The Java program running on the computer establishes two channels of connection to the step motors using the USB-PIO cable. The cable is capable of making up to three channels and eight lines of connection per channel. Channel 1 is set up to be used by the program to output digital signals from the computer to the motors and channel 2 to gather input from the sensors attached to each of the motors. Channel 1 establishes six lines of connection, as shown in figure 2. Two lines of connection are necessary for the movement of each of the three motors. Pointer 0 and 1 are responsible for moving the motor attached to the second hand of the clock, pointer 2 and 3 for the minute hand of the clock and pointer 4 and 5 for the hour hand of the clock.



GUGGENHEIM
Conservation Department

CONSERVATION REPORT TEMPLATE FOR COMPUTER-BASED ARTWORKS

1. ARTWORK IDENTIFICATION

Accession No.:
Artist:
Title:
©, Year:
Medium Line:
Edition:

2. DESCRIPTION OF ARTWORK

Provide a general description of the artwork, as it is perceived/experienced by the viewer/user. Note any key features.

Be sure to include the following:

- Simple executable file?
- Web page?
- System with non-dedicated and/or proprietary devices and coding?
- System with dedicated and/or custom-built devices with coding?
- Other:

How/where should the work be accessed/displayed?

- Static gallery installation
- Portable
- Networked and accessible via the Internet (Web page?)
- Other:

Is the work simply played back and if so does it have a fixed duration? If interactivity is a feature of the work describe:

- Multiple or single user
- Visitor navigation e.g. joystick or mouse
- Visitor required to input information
- Via the internet accessed by a computer or portable device
- Other:

3. THE COMPONENTS OF THE ARTWORK

3.1. HARDWARE

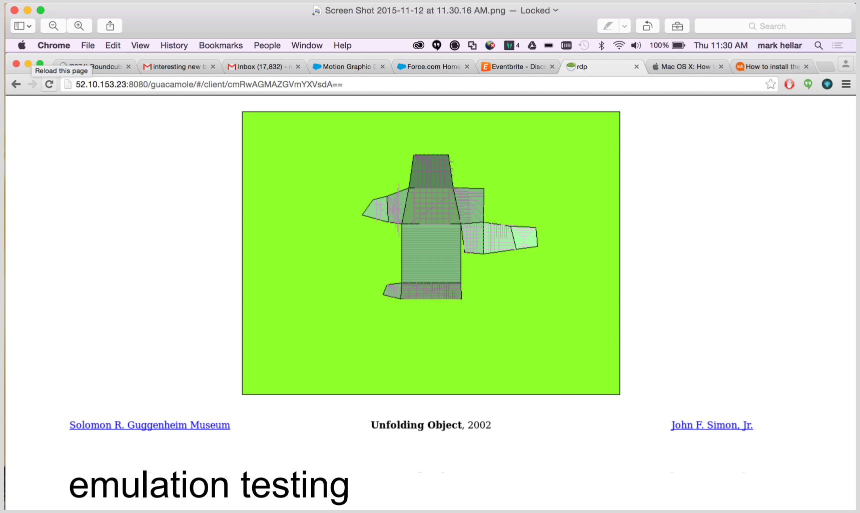
Computers
Technical specs required for custom hardware
Display(s)
Peripherals

DRAFT - Joanna Phillips, Brian Castriota, December 17, 2014.

CCBA Research and Development

4. Conservation treatment and accessibility

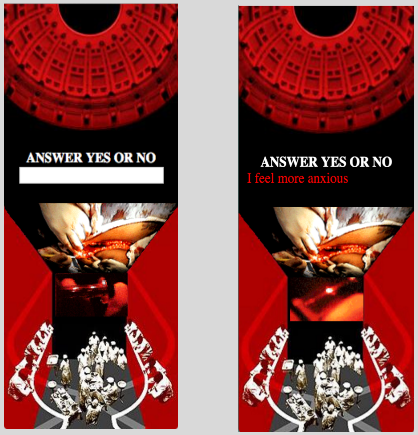
disc imaging and back-up


Screen Shot 2015-11-12 at 11.30.16 AM.png — Locked
Chrome File Edit View History Bookmarks People Window Help
Thu 11:30 AM mark hellar
Reload this page
52.10.153.23:8080/guacamole/#/client/cmRwAGMAZGVmYXVsdA==

Solomon R. Guggenheim Museum **Unfolding Object**, 2002 John F. Simon, Jr.

emulation testing

creation and testing of replica computers



migration prototyping, here Java to javascript

Academic Requirements

GUGGENHEIM

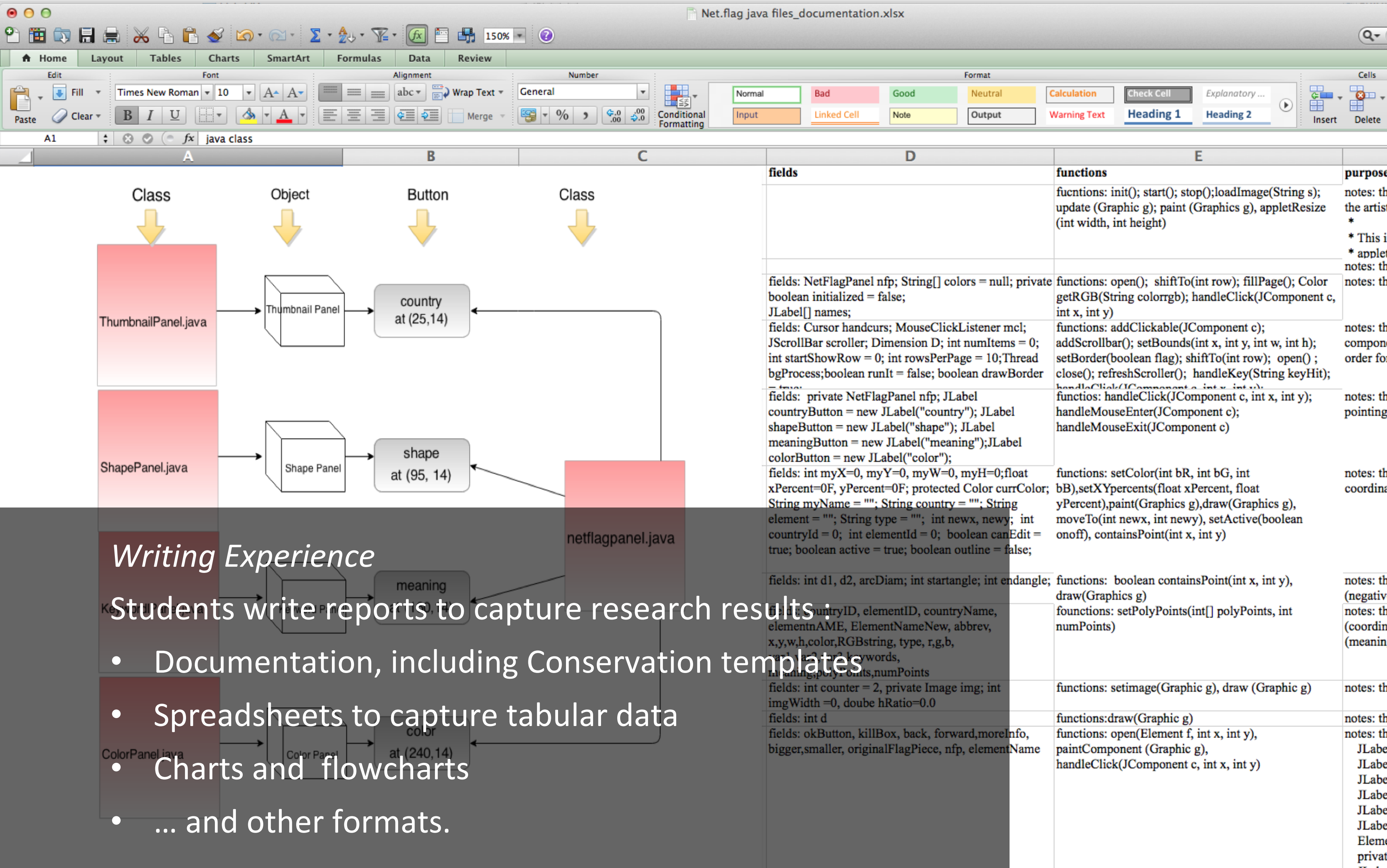
NEW YORK UNIVERSITY

Recruitment

- Students receive academic credit for research projects
- Students are carefully selected among a pool of applicants

Academic Requirements

GUGGENHEIM
NEW YORK UNIVERSITY



fields	functions	purpose
	fucntions: init(); start(); stop();loadImage(String s); update (Graphic g); paint (Graphics g), appletResize (int width, int height)	notes: th the artis * * This i * applet
		notes: th

fields: NetFlagPanel nfp; String[] colors = null; private boolean initialized = false;
JLabel[] names;
	functions: open(); shiftTo(int row); fillPage(); Color getRGB(String colorrgb); handleClick(JComponent c, int x, int y)	notes: th

fields: Cursor handcurs; MouseClickListener mcl; JScrollBar scroller; Dimension D; int numItems = 0; int startShowRow = 0; int rowsPerPage = 10;Thread bgProcess;boolean runIt = false; boolean drawBorder	functions: addClickable(JComponent c); addScrollbar(); setBounds(int x, int y, int w, int h); setBorder(boolean flag); shiftTo(int row); open() ; close(); refreshScroller(); handleKey(String keyHit);	notes: th compon order fo
fields: private NetFlagPanel nfp; JLabel countryButton = new JLabel("country"); JLabel shapeButton = new JLabel("shape"); JLabel meaningButton = new JLabel("meaning");JLabel colorButton = new JLabel("color");	functios: handleClick(JComponent c, int x, int y); handleMouseEnter(JComponent c); handleMouseExit(JComponent c)	notes: th pointing
fields: int myX=0, myY=0, myW=0, myH=0;float xPercent=0F, yPercent=0F; protected Color currColor; String myName = ""; String country = ""; String element = ""; String type = ""; int newx, newy; int countryId = 0; int elementId = 0; boolean canEdit = true; boolean active = true; boolean outline = false;	functions: setColor(int bR, int bG, int bB),setXYpercents(float xPercent, float yPercent),paint(Graphics g),draw(Graphics g), moveTo(int newx, int newy), setActive(boolean onoff), containsPoint(int x, int y)	notes: th coordina
fields: int d1, d2, arcDiam; int startangle; int endangle;	functions: boolean containsPoint(int x, int y), draw(Graphics g)	notes: th (negativ
fields: countryID, elementID, countryName, elementnAME, ElementNameNew, abbrev, x,y,w,h,color,RGBstring, type, r,g,b, var1,var2,var3,keywords, meaning,polyPoints,numPoints	founctions: setPolyPoints(int[] polyPoints, int numPoints)	notes: th (coordin (meanin
fields: int counter = 2, private Image img; int imgWidth =0, doube hRatio=0.0	functions: setimage(Graphic g), draw (Graphic g)	notes: th
fields: int d	functions:draw(Graphic g)	notes: th
fields: okButton, killBox, back, forward,moreInfo, bigger,smaller, originalFlagPiece, nfp, elementName	functions: open(Element f, int x, int y), paintComponent (Graphic g), handleClick(JComponent c, int x, int y)	notes: th JLabe JLabe JLabe JLabe JLabe JLabe Eleme privat

Writing Experience

Students write reports to capture research results :

- Documentation, including Conservation templates
- Spreadsheets to capture tabular data
- Charts and flowcharts
- … and other formats.

Academic Requirements

Public Speaking

Case study presentations in the media conservation lab are open to Guggenheim staff and their guests.

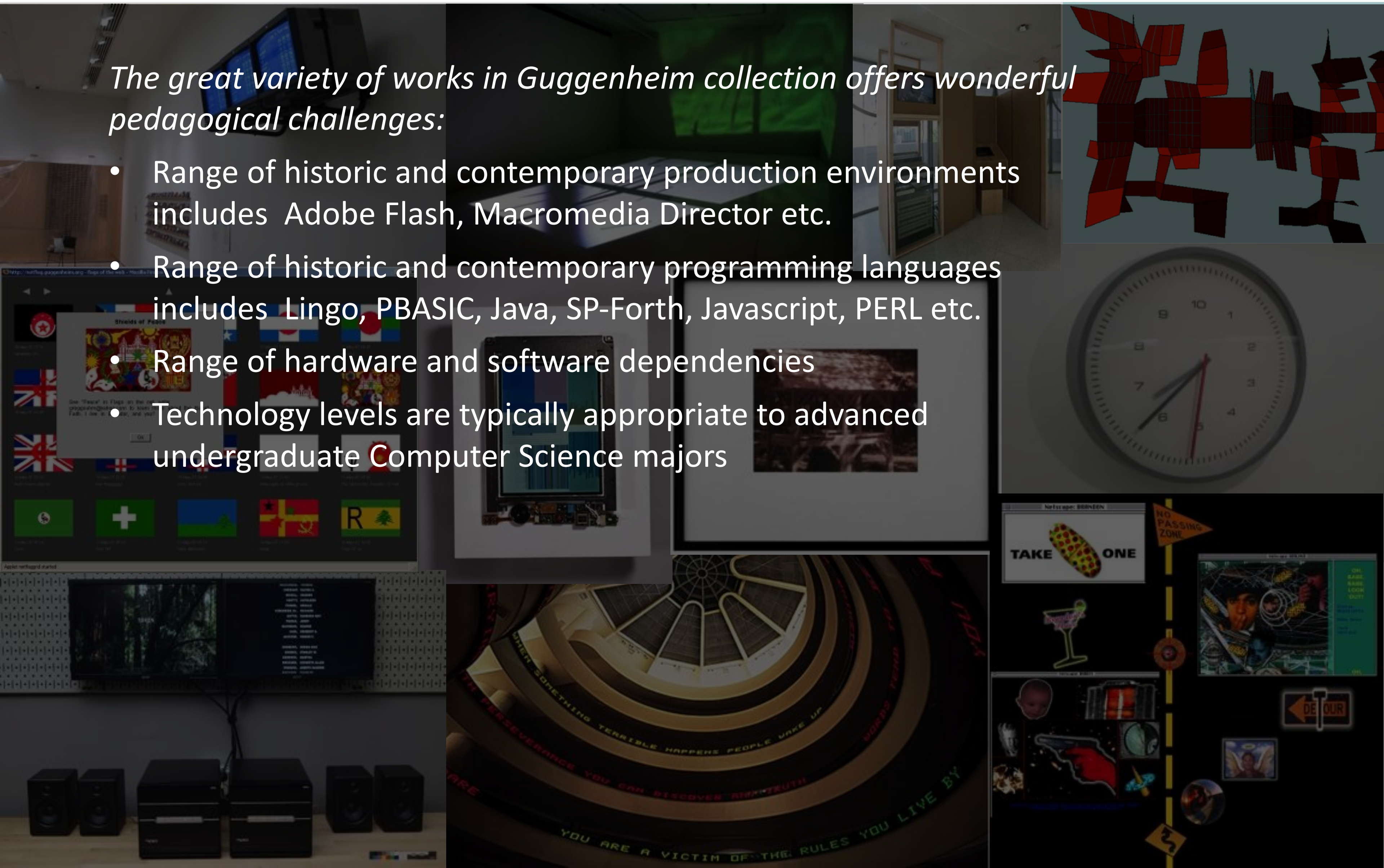
- Students do oral presentations at the end of each semester.
- For many students: first public speaking experience

Academic Requirements

Weekly meetings with faculty member or mentor:

- Students meet weekly with a faculty member or mentor
- Variety of research approaches and methodologies
- Joanna and I role-model inter-disciplinary collaboration

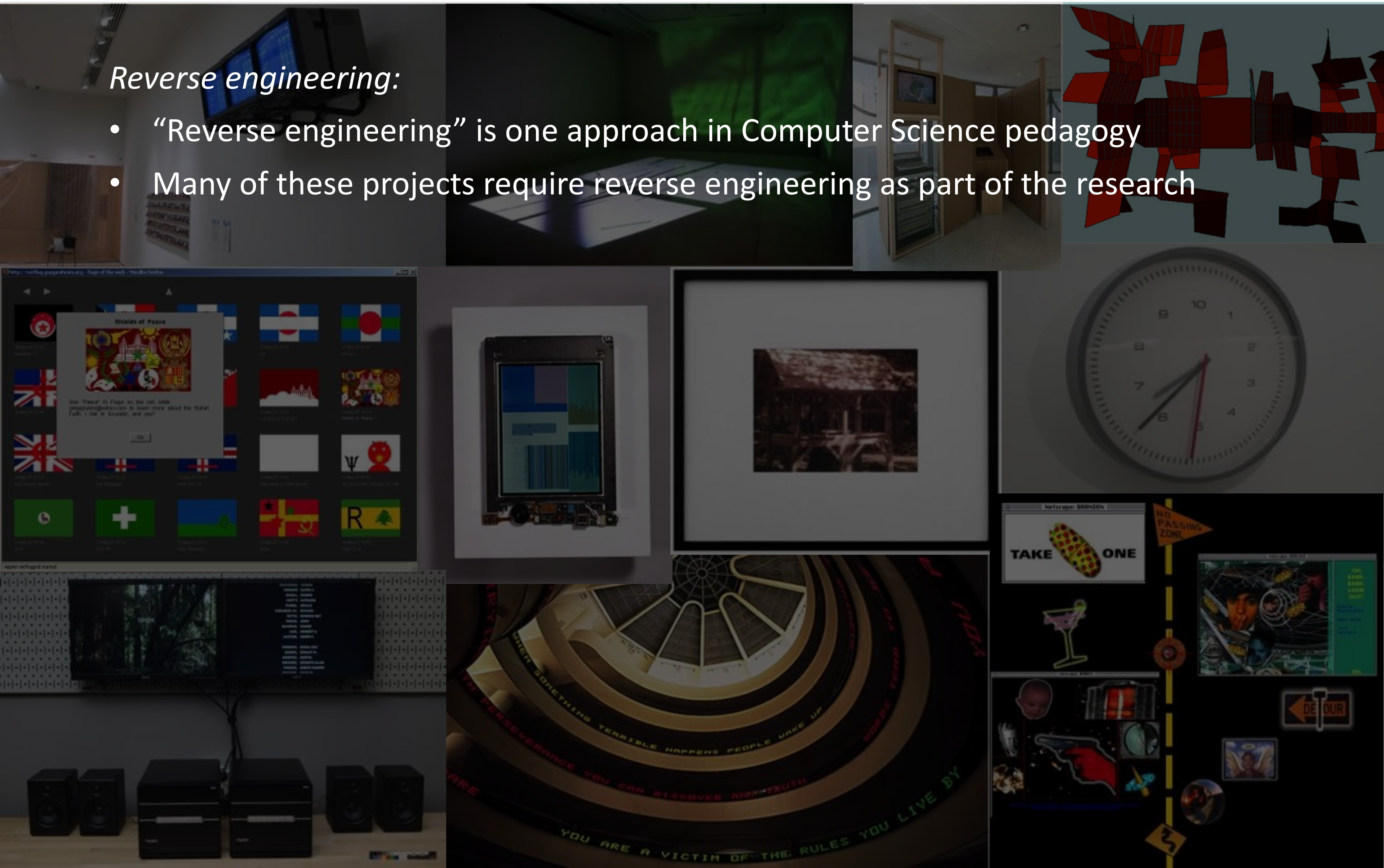
Computer Science Pedagogy


TAKE ONE
NO PASSING ZONE
DETOUR
SOMETHING TERRIBLE HAPPENS PEOPLE WAKE UP
PERSEVERANCE YOU CAN DISCOVER ANY TRUTH
YOU ARE A VICTIM OF THE RULES YOU LIVE BY

The great variety of works in Guggenheim collection offers wonderful pedagogical challenges:

- Range of historic and contemporary production environments includes Adobe Flash, Macromedia Director etc.
- Range of historic and contemporary programming languages includes Lingo, PBASIC, Java, SP-Forth, Javascript, PERL etc.
- Range of hardware and software dependencies
- Technology levels are typically appropriate to advanced undergraduate Computer Science majors

Computer Science Pedagogy



TAKE ONE
NO PASSING ZONE
DETOUR
SOMETHING TERRIBLE HAPPENS PEOPLE WAKE UP
PERSEVERANCE YOU CAN DISCOVER ... TRUTH
YOU ARE A VICTIM OF THE RULES YOU LIVE BY

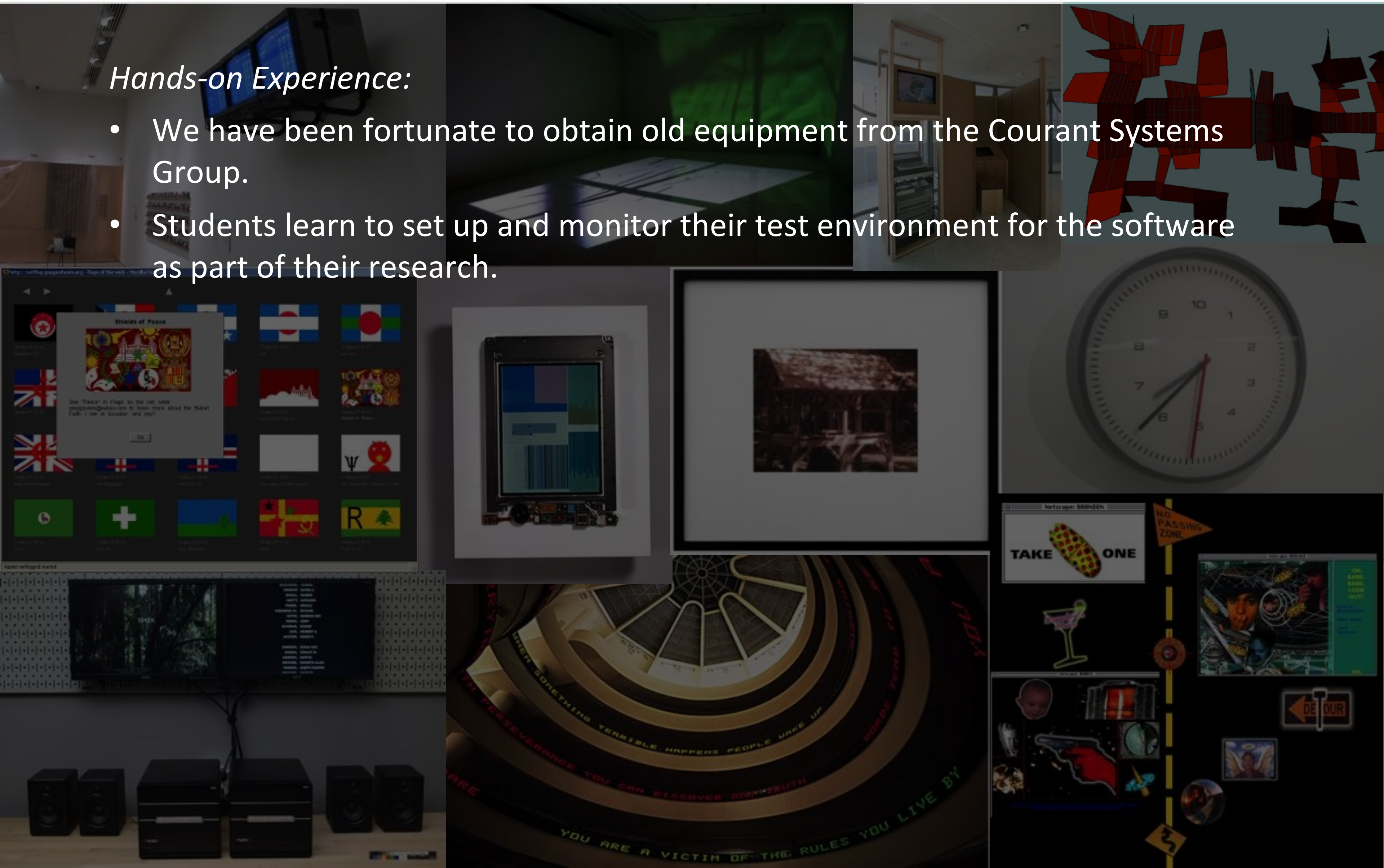
Reverse engineering:

- “Reverse engineering” is one approach in Computer Science pedagogy
- Many of these projects require reverse engineering as part of the research

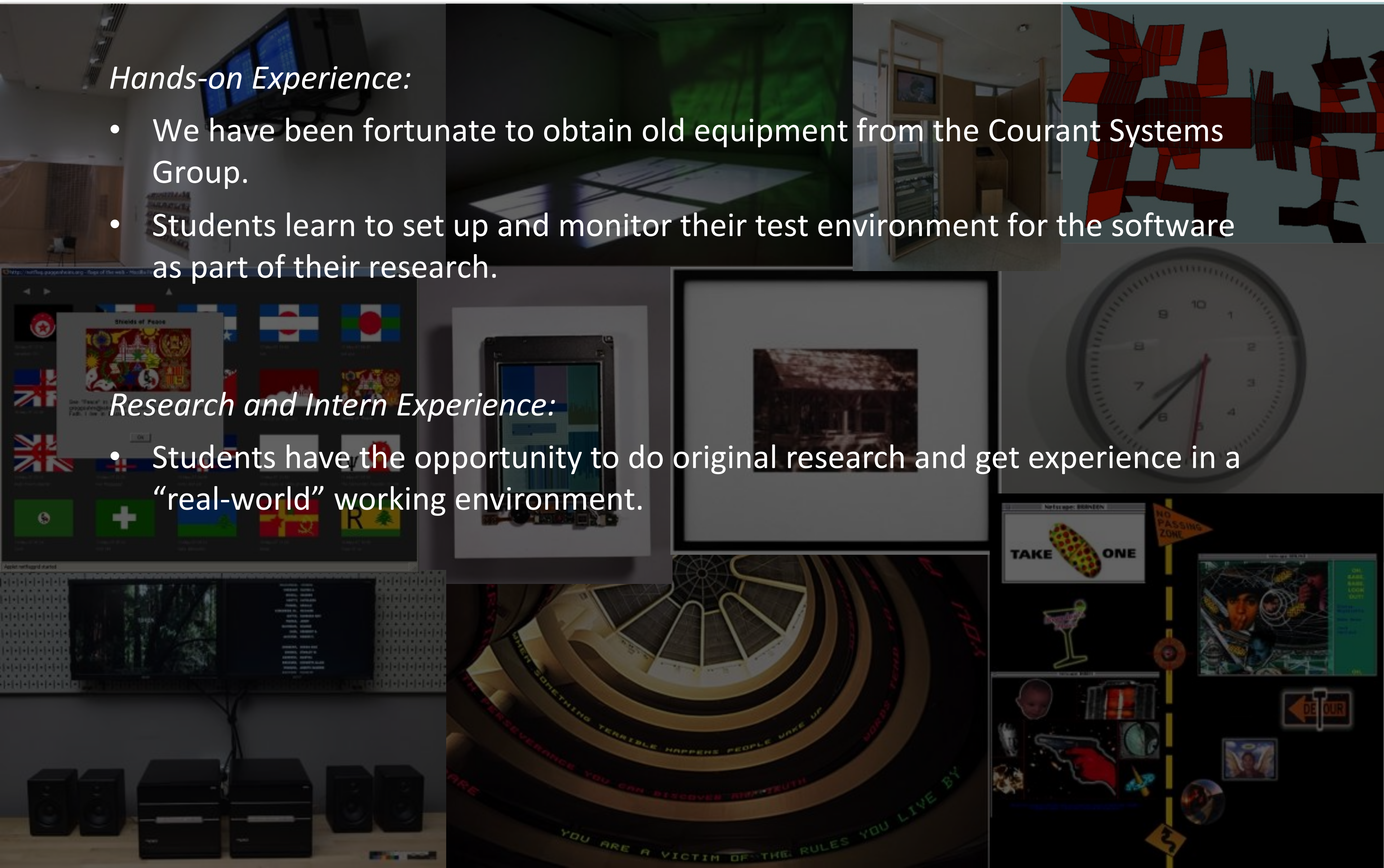
Computer Science Pedagogy

Hands-on Experience:

- We have been fortunate to obtain old equipment from the Courant Systems Group.
- Students learn to set up and monitor their test environment for the software as part of their research.


TAKE ONE
NO PASSING ZONE
DETOUR
SOMETHING TERRIBLE HAPPENS PEOPLE WAKE UP
YOU ARE A VICTIM OF THE RULES YOU LIVE BY

Computer Science Pedagogy



Hands-on Experience:

- We have been fortunate to obtain old equipment from the Courant Systems Group.
- Students learn to set up and monitor their test environment for the software as part of their research.

Research and Intern Experience:

- Students have the opportunity to do original research and get experience in a “real-world” working environment.

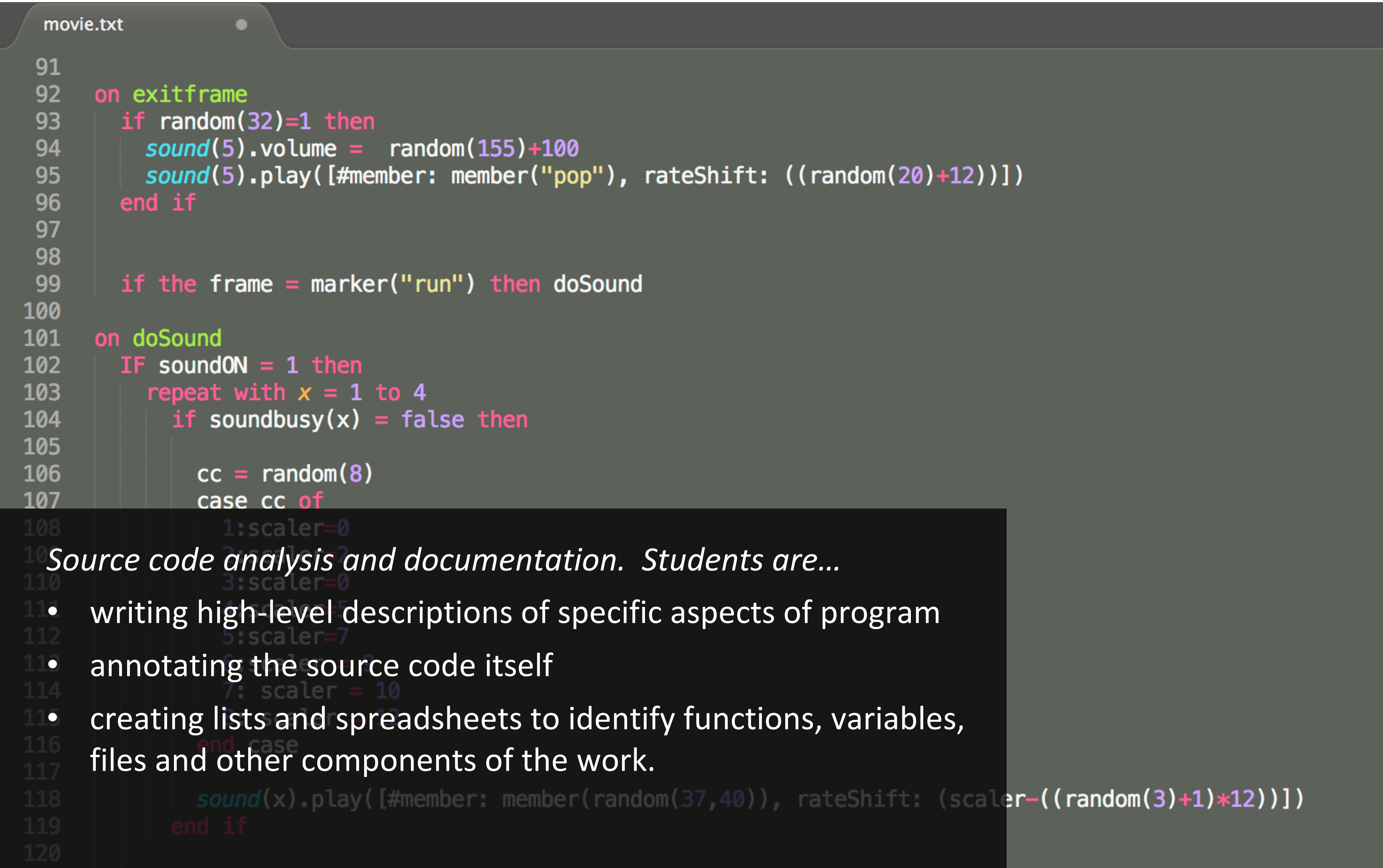
Research Approach

The Source code:

- Guggenheim supplies NYU faculty and students with artwork source code
- If source code is not available, students learn how to decompile executable file

Research Approach

GUGGENHEIM
NEW YORK UNIVERSITY

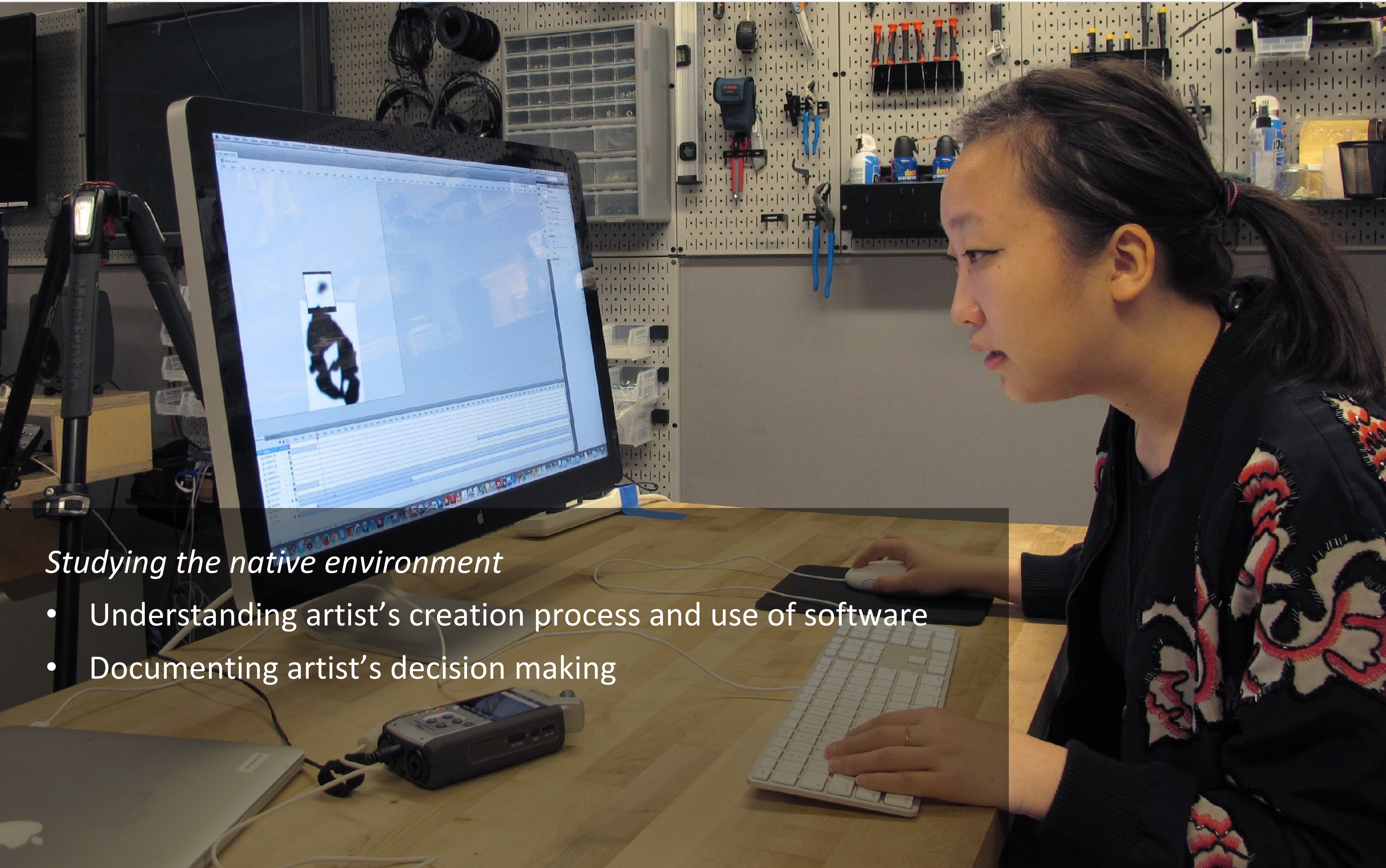


```
on exitframe
  if random(32)=1 then
    sound(5).volume =  random(155)+100
    sound(5).play([#member: member("pop"), rateShift: ((random(20)+12))])
  end if


  if the frame = marker("run") then doSound

on doSound
  IF soundON = 1 then
    repeat with x = 1 to 4
      if soundbusy(x) = false then

        cc = random(8)
        case cc of
          1:scaler=0
          2:scaler=2
          3:scaler=0
          4:scaler=5
          5:scaler=7
          6:scaler = [illegible]
          7: scaler = 10
          8:scaler=[illegible]
        end case


        sound(x).play([#member: member(random(37,40)), rateShift: (scaler-((random(3)+1)*12))])
      end if

```

Source code analysis and documentation. Students are…

- writing high-level descriptions of specific aspects of program
- annotating the source code itself
- creating lists and spreadsheets to identify functions, variables, files and other components of the work.

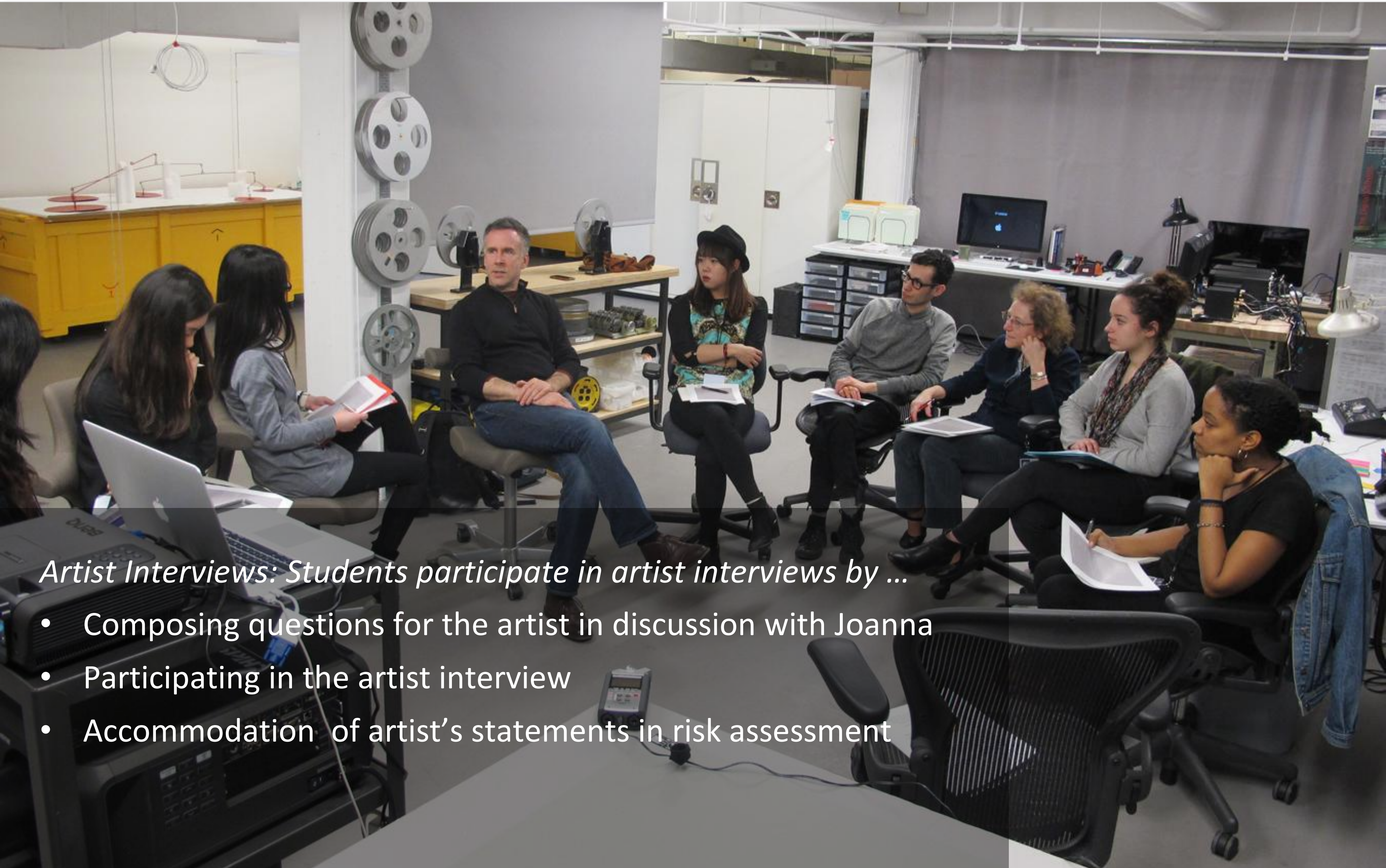
Research Approach



Studying the native environment

- Understanding artist’s creation process and use of software
- Documenting artist’s decision making

Research Approach

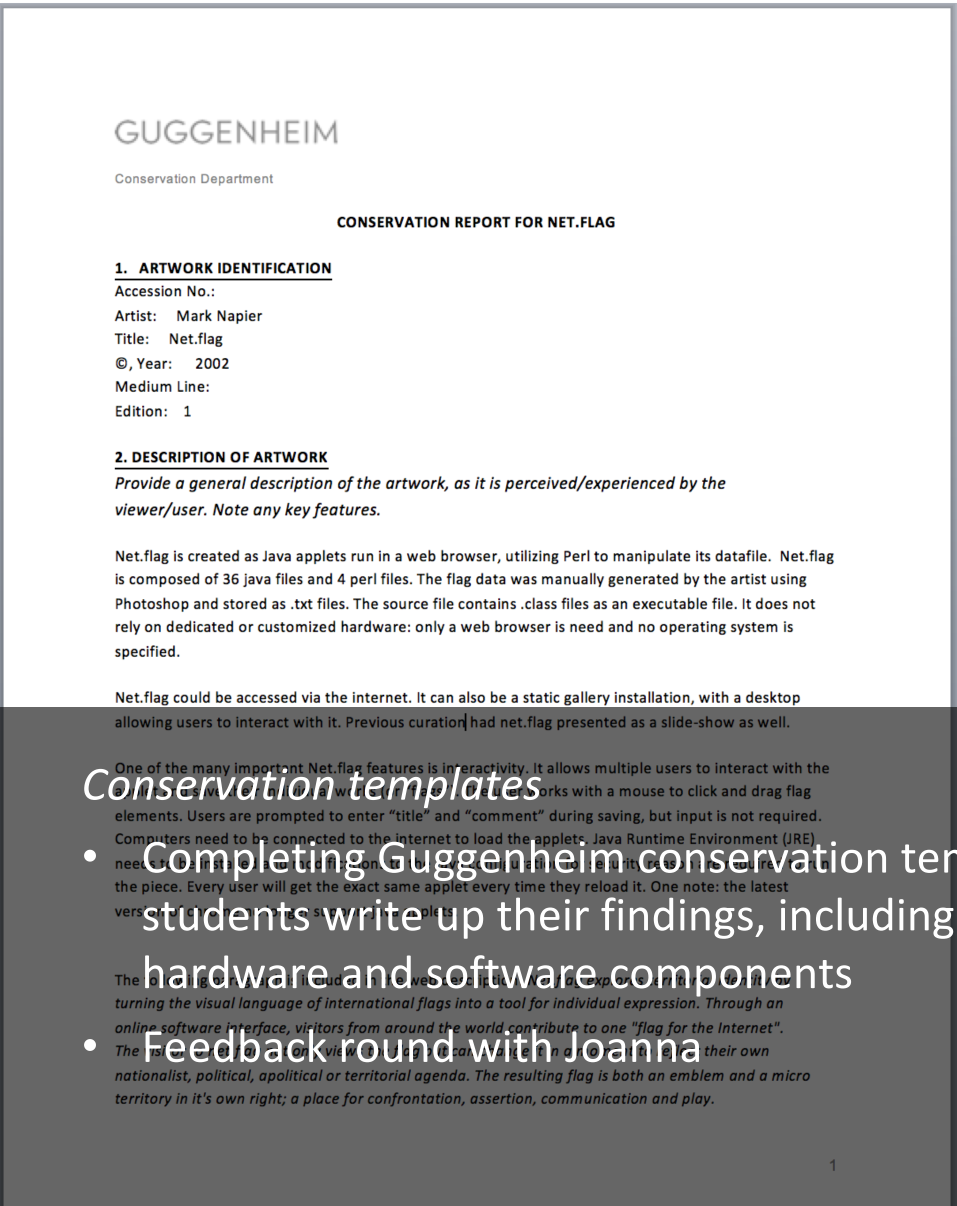


Artist Interviews: Students participate in artist interviews by …

- Composing questions for the artist in discussion with Joanna
- Participating in the artist interview
- Accommodation of artist’s statements in risk assessment

Research Approach

GUGGENHEIM
NEW YORK UNIVERSITY



GUGGENHEIM

Conservation Department

CONSERVATION REPORT FOR NET.FLAG

1. ARTWORK IDENTIFICATION

Accession No.:
Artist: Mark Napier
Title: Net.flag
©, Year: 2002
Medium Line:
Edition: 1

2. DESCRIPTION OF ARTWORK

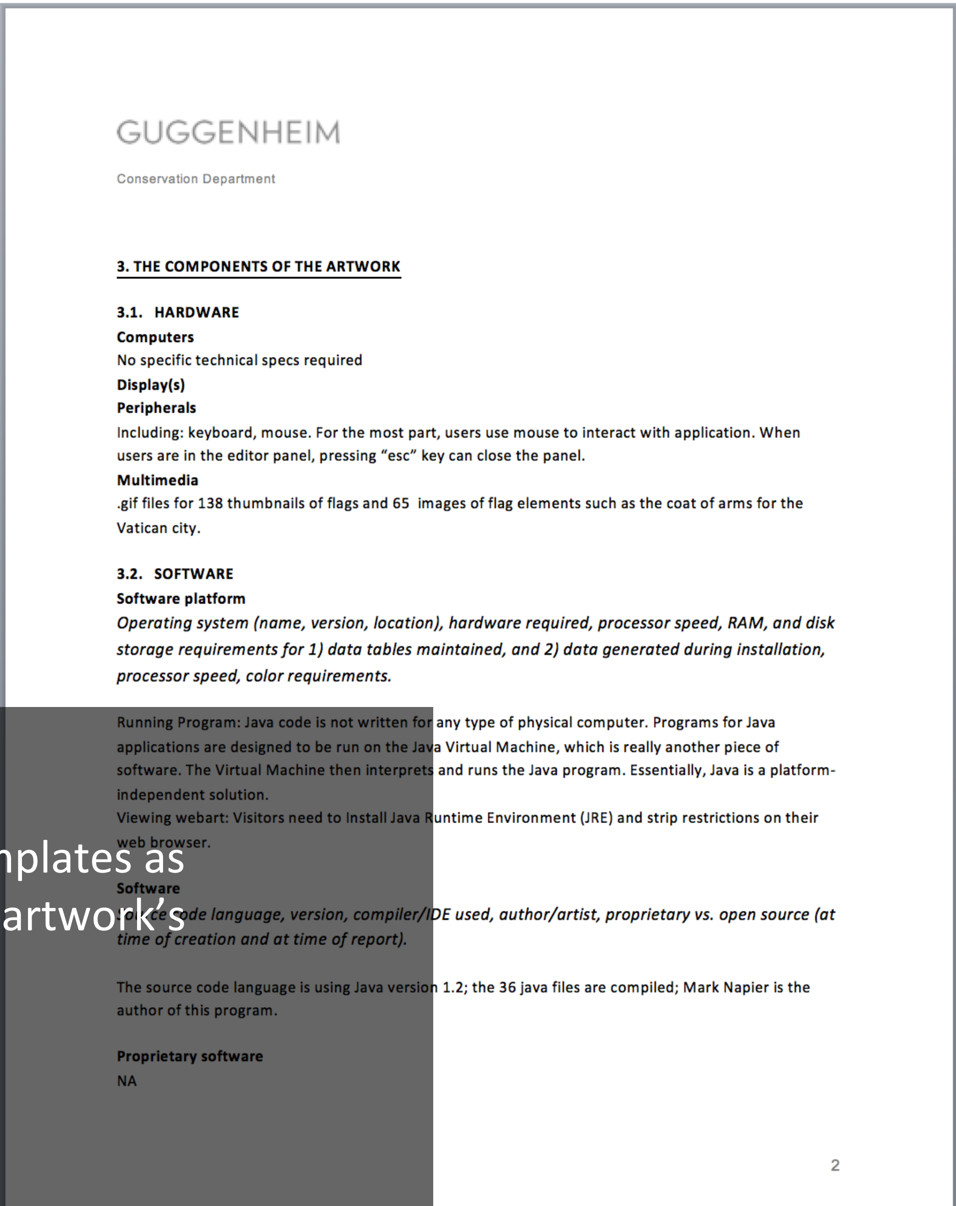
Provide a general description of the artwork, as it is perceived/experienced by the viewer/user. Note any key features.

Net.flag is created as Java applets run in a web browser, utilizing Perl to manipulate its datafile. Net.flag is composed of 36 java files and 4 perl files. The flag data was manually generated by the artist using Photoshop and stored as .txt files. The source file contains .class files as an executable file. It does not rely on dedicated or customized hardware: only a web browser is need and no operating system is specified.

Net.flag could be accessed via the internet. It can also be a static gallery installation, with a desktop allowing users to interact with it. Previous curation had net.flag presented as a slide-show as well.

One of the many important Net.flag features is interactivity. It allows multiple users to interact with the applet and save their individual works (or "flags"). The user works with a mouse to click and drag flag elements. Users are prompted to enter "title" and "comment" during saving, but input is not required. Computers need to be connected to the internet to load the applets. Java Runtime Environment (JRE) needs to be installed and modifications to the Java configuration for security reason are required to run the piece. Every user will get the exact same applet every time they reload it. One note: the latest version of chrome no longer support java applets.

The following paragraph is included in the web description: *Net.flag explores territorial identity by turning the visual language of international flags into a tool for individual expression. Through an online software interface, visitors from around the world contribute to one "flag for the Internet". The visitor to net.flag not only views the flag but can change it in an instant to reflect their own nationalist, political, apolitical or territorial agenda. The resulting flag is both an emblem and a micro territory in it's own right; a place for confrontation, assertion, communication and play.*



GUGGENHEIM

Conservation Department

3. THE COMPONENTS OF THE ARTWORK

3.1. HARDWARE

Computers

No specific technical specs required

Display(s)

Peripherals

Including: keyboard, mouse. For the most part, users use mouse to interact with application. When users are in the editor panel, pressing "esc" key can close the panel.

Multimedia

.gif files for 138 thumbnails of flags and 65 images of flag elements such as the coat of arms for the Vatican city.

3.2. SOFTWARE

Software platform

Operating system (name, version, location), hardware required, processor speed, RAM, and disk storage requirements for 1) data tables maintained, and 2) data generated during installation, processor speed, color requirements.

Running Program: Java code is not written for any type of physical computer. Programs for Java applications are designed to be run on the Java Virtual Machine, which is really another piece of software. The Virtual Machine then interprets and runs the Java program. Essentially, Java is a platform-independent solution.
Viewing webart: Visitors need to Install Java Runtime Environment (JRE) and strip restrictions on their web browser.

Software

Source code language, version, compiler/IDE used, author/artist, proprietary vs. open source (at time of creation and at time of report).

The source code language is using Java version 1.2; the 36 java files are compiled; Mark Napier is the author of this program.

Proprietary software

NA

Conservation templates

- Completing Guggenheim conservation templates as students write up their findings, including artwork's hardware and software components
- Feedback round with Joanna

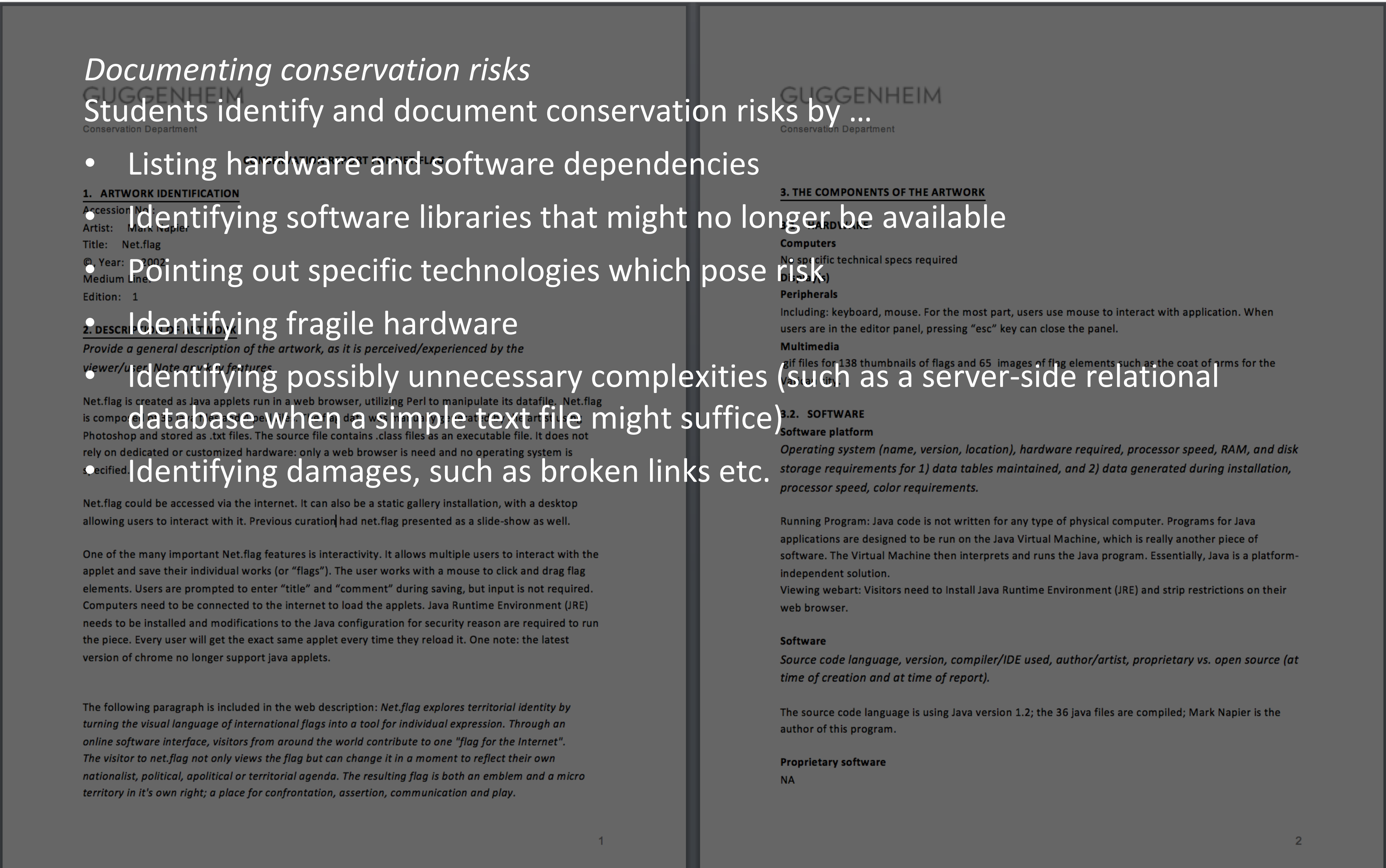
Guggenheim-NYU Research Approach

GUGGENHEIM
NEW YORK UNIVERSITY

Documenting conservation risks

Students identify and document conservation risks by …

- Listing hardware and software dependencies
- Identifying software libraries that might no longer be available
- Pointing out specific technologies which pose risk
- Identifying fragile hardware
- Identifying possibly unnecessary complexities (such as a server-side relational database when a simple text file might suffice)
- Identifying damages, such as broken links etc.



GUGGENHEIM
Conservation Department

CONSERVATION REPORT FOR NET.FLAG

1. ARTWORK IDENTIFICATION

Accession No.:
Artist: Mark Napier
Title: Net.flag
©, Year: 2002
Medium Line:
Edition: 1

2. DESCRIPTION OF ARTWORK

Provide a general description of the artwork, as it is perceived/experienced by the viewer/user. Note any key features.

Net.flag is created as Java applets run in a web browser, utilizing Perl to manipulate its datafile. Net.flag is compo[illegible] perl files. The flag data was manually generated by the artist using Photoshop and stored as .txt files. The source file contains .class files as an executable file. It does not rely on dedicated or customized hardware: only a web browser is need and no operating system is specified.

Net.flag could be accessed via the internet. It can also be a static gallery installation, with a desktop allowing users to interact with it. Previous curation had net.flag presented as a slide-show as well.

One of the many important Net.flag features is interactivity. It allows multiple users to interact with the applet and save their individual works (or "flags"). The user works with a mouse to click and drag flag elements. Users are prompted to enter "title" and "comment" during saving, but input is not required. Computers need to be connected to the internet to load the applets. Java Runtime Environment (JRE) needs to be installed and modifications to the Java configuration for security reason are required to run the piece. Every user will get the exact same applet every time they reload it. One note: the latest version of chrome no longer support java applets.

The following paragraph is included in the web description: *Net.flag explores territorial identity by turning the visual language of international flags into a tool for individual expression. Through an online software interface, visitors from around the world contribute to one "flag for the Internet". The visitor to net.flag not only views the flag but can change it in a moment to reflect their own nationalist, political, apolitical or territorial agenda. The resulting flag is both an emblem and a micro territory in it's own right; a place for confrontation, assertion, communication and play.*

1

GUGGENHEIM
Conservation Department

3. THE COMPONENTS OF THE ARTWORK

3.1. HARDWARE

Computers

No specific technical specs required

Display(s)

Peripherals

Including: keyboard, mouse. For the most part, users use mouse to interact with application. When users are in the editor panel, pressing "esc" key can close the panel.

Multimedia

.gif files for 138 thumbnails of flags and 65 images of flag elements such as the coat of arms for the Vatican City.

3.2. SOFTWARE

Software platform

Operating system (name, version, location), hardware required, processor speed, RAM, and disk storage requirements for 1) data tables maintained, and 2) data generated during installation, processor speed, color requirements.

Running Program: Java code is not written for any type of physical computer. Programs for Java applications are designed to be run on the Java Virtual Machine, which is really another piece of software. The Virtual Machine then interprets and runs the Java program. Essentially, Java is a platform-independent solution.
Viewing webart: Visitors need to install Java Runtime Environment (JRE) and strip restrictions on their web browser.

Software

Source code language, version, compiler/IDE used, author/artist, proprietary vs. open source (at time of creation and at time of report).

The source code language is using Java version 1.2; the 36 java files are compiled; Mark Napier is the author of this program.

Proprietary software

NA

2

Guggenheim-NYU Research Approach

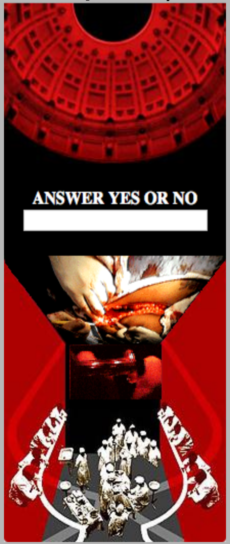
Conservation treatment

Building prototypes for conservation intervention, by…

- Writing software to render the artwork and allow it to run.
- Setting up test environments to test an approach such as emulation for a specific work.

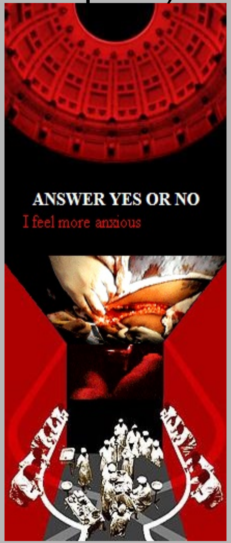
Before:

(w/o java applet exception)



ANSWER YES OR NO

(w/ java applet exception)

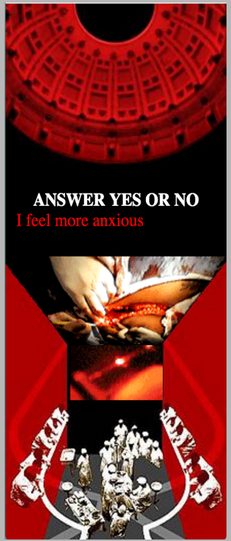


ANSWER YES OR NO

I feel more anxious

After:

javascript implementation



ANSWER YES OR NO

I feel more anxious

Migration from Java to javascript, Shu Lea Cheang's *Brandon* (1998-1999)

Research Approach

GUGGENHEIM
NEW YORK UNIVERSITY

Paul Chan's
6th Light
Kaitlin Gu
Java Applets
application written in Java that can be embedded in a web page
dynamic and interactive
client-side
.class file compiled from a .java file
no longer supported by Google Chrome
unsigned applications blocked by default by Java Security

Research Presentation and Discussion:
Students give a formal presentation to Guggenheim staff and the NYC conservation community.

Thank you!

Solomon R. Guggenheim Museum

Brian Castriota, former Samuel H. Kress Fellow in Time-based Media Conservation
Amy Brost, former Andrew W. Mellon Graduate Intern for Time-based Media Conservation
Lia Kramer, former Polonsky Intern for Digital Humanities
Jiwon Shin, former summer intern for Time-based Media Conservation

New York University

Fall Semester 2014
Jiwon Shin
Aarti Chandrakant Bagul
Shan Shao

Spring Semester 2015
Jiwon Shin
Michelle Liu
Vivian Peng
Emily Hua
Caroline Slason
Mia Matthias

Fall Semester 2015
Emma Dickson
Jillian Zhong

Spring Semester 2016
Emma Dickson
Jillian Zhong
Kaitlin Gu