

Destroying Artworks

ASA Annual Meeting, Denver, Oct 22, 2009

Marcus Rossberg, marcus.rossberg@uconn.edu

1. DIY Guide

Some examples, in order of increasing difficulty (be thorough – restorers are dedicated these days):

- Painting burn painted canvas (or dissolve it completely in acid)
- Marble Statue pulverise sculpture
- Bronze Statue melt down statue & pulverise mould
- Woodcuts, Etchings, Prints burn print & burn wood-block, melt plate, burn silk-screen, etc.
- Architecture blow up building & burn blue-prints, etc. (?)
- Literature burn copies & ... ?
- Music burn sheet music & melt records, CDs, etc. & ... ?

2. “The” Ontology of Art

Some options for *the* ontology of art (not exhaustive):

- Original (first, physical) work plausible for painting and marble sculpture, not so much for multiple artworks (but defended by Timmerman)
- Universals, platonic forms implausible at least for painting and marble sculpture (but defended by Schmücker)
- Action types (Curry) implausible (at least) for painting, marble sculpture
- Action tokens, “performances” (D. Davies) ostensive work mere trace of token performance
- Abstract objects Huh? Can’t we see a painting? Listen to music? Watch a ballet? Read a novel? Enter a building? Kick a statue?

Conjecture: there is no common ontological category for all of the *fine arts* – an idea which seems to be an invention of the 18th century anyway, largely to be blamed on Abbé Batteaux. Such ontological “pluralism” is argued for by, e.g., Wollheim, Goodman, more recently Lopes, Thomasson.

Suggestion: painting, sculpture, prints, architecture(?) — concrete: originals
music, literature, drama(?), dance(?) — abstract: reproducible, “unfakeable”

[Regarding reproducibility note the well-known Cervantes/Menard-problem, cf. Tolhurst & Wheeler. The problem is not exclusive to artwork-abstracta, but appears more pressing here.]

Traditional Problem for Artwork-Abstracta

- Artists do not create artworks but merely *discover* and *draw attention* to them
- Ostensive artworks are a mere heuristic for appreciation, but not objects of appreciation
- Alarm when ostensive artwork is destroyed appears inexplicable

Traditional Response to these Problems

- Discovery can be just as difficult, praiseworthy & valuable as creation (cf. science, Michelangelo)
- Ostensive works are heuristic and the *only* way of appreciating the work, thus valuable
- Alarm regarding destruction is explained by loss of access to the real work (the abstractum)

Regarding the last point: if the *Mona Lisa* was burnt, we would still have all the reproductions and thus access, but we would still be alarmed

Comeback: that’s because the *Mona Lisa* is concrete and an original;
no analogous case for music or literature is conceivable: copies are instances of the work

3. The Case for Artwork-Abstracta

Music & Literature

Perhaps best understood as *types*? (Wollheim, also for prints and other concrete multiple forms)

Or *equivalence classes*? (Aren't types equivalence classes anyway?)

— appears unnecessary: types and equivalence classes are easily eliminated from art discourse (cf. Goodman)

Structures (Kivy, Levison, for music), akin to mathematical structures

Instantiation problem: not all instantiations of the structure count as performances (e.g. visualisations, possibly “wrong” instrumentations).

Kivy: instrumentation etc. part of the structure (mathematical model appears to fail in this is followed).

Conceptual Art

- Early forms, e.g., readymades — concrete
- *What Roy had for breakfast this morning* — concrete (or non-existent)
- *What happened in Denver between 9 and 9:10 a.m. on October 22, 2009* — event, so arguable concrete
- *All prime numbers identified prior to 1999* — only coincidentally abstract
- Radical conceptual art, mere instruction — perhaps: (merely) possible artworks; what if instruction explicitly *precludes* an execution? *Necessarily* merely possibly?

Designs as opposed to mere abstract structure or pattern – designs require intentional creation. (Compare Thomasson's “abstract artifacts”.)

Computer and Internet Art

Check, e.g., www.rhizome.org, a website that links to and also hosts Internet art (bring time).

The apparent similarity between computer games (or other computer programmes) and works of computer art may not be coincidental as regards ontology. Computer scientists distinguish between an *algorithm* and its *implementation*. (Note: implementation problem analogous to instantiation problem.)

Think *multiply realizable*, but in fact it is more complicated than that:

- an algorithm can be implemented in different ways in the same programming language or in different programming languages;
- the programme can be run on different interpreters running on, or can be compiled for, different operating systems (Windows, MacOS, Linux, ...); cross-platform, Flash-style applets are yet another possibility (usually understood to differ from interpreters);
- different versions of operating systems run on different types of hardware (not just Mac vs PC, desktop vs laptop, but also different architectures of the same type of computer).

What *actually happens* in a computer when the implemented algorithm is executed differs significantly, depending on hardware, operating system, interpreter (compiler, applet) and specific implementation. (Updates of Flash applets, for instance, may change the processing of the programme code of some artwork or game, while the specific code need not be changed in any way and will still produce the same audio-visual result.)

An internet or computer artwork is perhaps best conceived of as an algorithm, and thus an abstract, mathematical object. (At least in some cases: the rhizome.org questionnaire suggests that some computer art may not be multiply implementable – or at least that some artists may think so.)

4. Destruction — technical report

Equivalence classes come into and go out of existence with their members (they are impure sets), but this seems connected to their ontological dispensability

Types defined by their first token which is often thought also to create the type

- (i) Types are equivalence classes: see above.
- (ii) Types are *sui generis* and can be destroyed by destroying all tokens and all means (including memory) for creating new tokens.
But what if a token comes into existence coincidentally thereafter? No-one recognizing it as such appears to be a mere epistemic mishap.
(Compare Thomasson's abstract artifacts; contrast Goodman on survival of artworks beyond the destruction of the resp. world-version)
- (iii) Types are *sui generis* and cannot be destroyed (despite being able to be created). *Three Blind Mice* was created but will last forever. (Is this coherent?)

Structures eternal (mathematical) objects and thus indestructible

The destruction of instances and means for reproduction of instances is not sufficient for destruction. Structures are pure abstract objects *par excellence*, independent from the physical and from minds.

In the case of music, the destruction of all scores, recordings, memories and all other means of reproducing a new faithful score (except coincidentally) leaves the work proper untouched.

Three Blind Mice was discovered, not created, and can only be forgotten, not destroyed.

Designs need to be intentionally created

- (i) What is created is a pattern or structure, akin to type-(iii) types.
- (ii) Designs can be destroyed akin to type-(ii) types: see there.

Note that some works of conceptual art appear to be indestructible even if they are identified as concrete objects, e.g., *What happened in Denver between 9 and 9:10 a.m. on October 22, 2009*.

Algorithms eternal (mathematical) objects and thus indestructible: see structure.

Computer art can be discovered and forgotten, but not created and destroyed. Internet art, I submit (and will argue elsewhere), is not essentially different from computer art and can survive disconnection from the internet.

Works of internet (or computer) art survive the (joint) destruction of:

- the internet
- all computers
- all implementations of the algorithm
- all means (including memory) to re-implement the algorithm
- implementation instructions

While it seems just about possible to create artworks if they are abstract objects, their destruction seems harder: they tend to linger on.

<http://www.rhizome.org/artbase/policy.htm>

[...]

APPENDIX D: ARTIST QUESTIONNAIRE

TECHNICAL PROFILE

Technical Profile—Platforms

1. Which of the following computer platforms are required to view or run your project?

- Any platform
- Macintosh OS9
- Macintosh OSX
- Windows 95/98
- Windows 2000/NT
- Windows XP
- Linux or Unix (specify below)
- PDA (specify below)
- I don't know
- Other—please specify/explain:

Technical Profile—Browsers

1. Is your work or documentation optimized for a specific browser?

- Not optimized
- Internet Explorer for Mac
- Internet Explorer for PC
- Netscape for Mac
- Netscape for PC
- I don't know
- Other—please specify/explain:

2. Are there any browsers in which you know your work or documentation does not function correctly?

- Internet Explorer for Mac
- Internet Explorer for PC
- Netscape for Mac
- Netscape for PC
- I don't know
- Other—please specify/explain:

[...]

Technical Profile—Server-Side Tech and Databases

[...]

3. Please check all server-side scripting languages, software or other programs involved in your work or documentation (excluding databases):

- | | | | |
|---------------|--------------|----------------------|-------|
| Perl | PHP | Cold Fusion | |
| Java Servlets | I don't know | Other—please specify | [...] |

ARTIST'S INTENT

As technology continues to evolve and change, it is highly likely that some or all of the technologies used in your work or documentation will become obsolete. For example, web browsers may not be able to read HTML someday. If that happens, we would like to find a way to make your work or documentation accessible. The main options appear to be:

1. documentation (e.g. screen shots)
2. migration (e.g. updating code)
3. emulation (a way to run old software on new platforms)
4. reinterpretation (re-creating your work or documentation in a new technological environment)

It's up to you to decide how far you want us to go to preserve your work or documentation. Often, a combination of approaches makes the most sense. For example, we might want to combine documentation with reinterpretation.

If you don't want us to do anything, we won't, and your work or documentation will eventually become inaccessible. If you do want us to take measures to preserve your work or documentation, you need to give us permission to do so. We will make our best efforts to honor your intentions.

All of the preservation measures contemplated in this section of the questionnaire involve compromises. We hope that you will give us wide latitude to ensure that your work or documentation remains accessible in some form. We will do our best to remain true to your intent.

[...]

Artist's Intent – Migration

Migration involves translating data written in an obsolete format to a format that is currently supported. For example, future web browsers may not support current HTML code. Migration would involve replacing old tags with new tags. Migration may be a viable option for obsolete code, animation formats, plug-ins, etc. On the other hand, it may not make sense for some projects.

Migration may result in changes in the way your project looks, sounds, moves, etc. If you give us permission to use migration as a preservation strategy, we will do our best to make the changes as minimal as possible

[...]

Artist's Intent – Reinterpretation

Reinterpretation involves recreating your project, or certain parts of it, while keeping its conceptual framework intact. Reinterpretation may be appropriate for projects that utilize external content or are performative in nature.

One example of a project that might benefit from reinterpretation is Mark Napier's "Shredder." This project, written in HTML and Javascript, pulls down an external web page, recombines its elements, and spits out a "shredded" version of the page. If HTML is no longer used to publish content on the web, does the Shredder have to cease to exist as well? The symbolic barriers that Shredder assaults may still exist in this new environment. Napier has suggested that his project could be reinterpreted in Flash, for example, if it replaces HTML as the dominant web publishing platform. Another option might be to run "Shredder" in an emulated environment, but instruct users to input the URL's of other projects in the ArtBase.

[...]