

UC Irvine

Plenaries: After Media — Embodiment and Context

Title

Platform Studies: Frequently Questioned Answers

Permalink

<https://escholarship.org/uc/item/01r0k9br>

Authors

Bogost, Ian
Montfort, Nick

Publication Date

2009-12-12

Peer reviewed

Platform Studies: Frequently Questioned Answers

Ian Bogost

The Georgia Institute of Technology
686 Cherry St.
Atlanta GA 30332 USA
ibogost@gatech.edu

Nick Montfort

Massachusetts Institute of Technology
77 Massachusetts Avenue 14N-233
Cambridge, MA 02139 USA
nickm@nickm.com

ABSTRACT

We describe six common misconceptions about platform studies, a family of approaches to digital media focused on the underlying computer systems that support creative work. We respond to these and clarify the platform studies concept.

Keywords

Platform studies, platforms, hardware design, technological determinism, social constructivism, pedagogy, new media.

1. INTRODUCTION

“Platform studies” is a new focus for the study of digital media, a set of approaches which investigate the underlying computer systems that support creative work. In 2009, the first platform-book that considered creative digital media from the perspective of the platform was published: our *Racing the Beam: The Atari Video Computer System* [9]. This is the first in the MIT Press Platform Studies series; we are the series editors for Platform Studies.

Although platform studies has only recently been introduced as a concept—at the 2007 Digital Arts and Cultures Conference [4]—it has already become popular enough to be misconstrued in a variety of ways. What we intended as an answer to a gap in digital media studies has often ended up being questioned in ways that seem to us to miss the point of the platform studies project. Detailed citations of these misconceptions, many of which have been communicated to us privately, are more likely to be offensive than helpful. In the interest of advancing platform studies, however, this paper reviews six misunderstandings about this new concept. We contrast the great potential of a new focus on the platform level with these misconceptions:

#1 Platform studies entails technological determinism.

Platform studies is opposed to “hard” determinism and invites us to continue to open the black box of technology in productive ways.

#2 Platform studies is all about hardware.

Platform studies includes software platforms as well.

#3 Platform studies is all about video games.

Platform studies extends to all computing platforms on which interesting creative work has been done.

#4 Everything these days is a platform.

We invite a focus on computational platforms, the basis for digital media work.

#5 Platform studies is about technical details, not culture.

Platform studies connects technical details to culture.

#6 Platform studies means that everyone in digital media will have to get computer science training or leave the field.

Platform studies shows how technical understanding can lead to new sorts of insights, but will not evict the many other important sorts of scholars from digital media.

By clarifying our concept of platform studies and explaining how it can contribute to the field, we hope to invite more scholars to do platform studies work and to make this approach more appealing. We also hope our comments will advance the discussion of the platform studies concept and will invite substantial, productive, and well-directed criticism of platform studies approaches, aiding in the development of work in this area.

2. MISCONCEPTION #1: PLATFORM STUDIES ENTAILS TECHNOLOGICAL DETERMINISM

Does technology follow a path of its own, influencing society directly, without social mediation? As an overall concept, “technological determinism” describes any theoretical or sociological approach which holds that technologies exert an effect on human society and behavior. Most contemporary usages make an important modification: technological determinism suggests that social change is *more* affected by technology than by other sources [10, p.2]. The popularity of this idea dates back at least to the eighteenth century, when science in general and technology in particular first became central in both intellectual and popular notions of social progress. Advances in invention through the 18th and 19th centuries, leading up to the industrial revolution, seemed to suggest that technology could, and would, solve all problems (even though it was starting to create new ones).

Based on Thomas J. Misa’s distinction [8] between different types of technological determinism, modern critics sometimes distinguish between “hard” and “soft” versions of determinism. The “soft” view holds that technological changes combine with social reception and discrimination, resulting in an impact subject to social malleability. The “hard” view holds that technological changes impact culture autonomously and without social intervention [5, pp 1-2]. This latter form may seem intuitively monstrous to us today, but even in the late 19th and early 20th centuries, hard determinism was still glinting with the idealism of the Enlightenment: invention and industry had made measurable improvements in ordinary life, including the notable changes brought about by the locomotive and electricity. In America in

particular, machinic innovation and “technocratic” thinking became figures of progress previously reserved for matters of state, nature, and religion.

1.1 Media Ecology

All of this changed by the mid-twentieth century, when World War II and the emergence of nuclear weaponry and energy led to a near-immediate skepticism regarding the inherent social benefits of technology, even as governmental and industrial investments in science and technology continued to accelerate. Unfortunately for us, this is also precisely the moment that the century’s most well-known technological determinist, Marshall McLuhan, began publishing his research on the way media shape human sense and consciousness.

McLuhan’s work is complex and difficult to read, and thus remains more cultural meme than media philosophy, a situation exacerbated by the thinker’s own media celebrity in the 1960s, which included both a cameo in Woody Allen’s *Annie Hall* and an interview in *Playboy* magazine. But his philosophy is quite simple to understand at its core: McLuhan argues that media are “extensions of the physical human body or the mind,” that is, they affect the ways that people perceive, understand, and relate to the world [7, p. 93.] This idea is best known through the aphorism “the medium is the message,” a phrase meant to clarify that the *properties* of a medium, not its “payload,” ought to be the object of study. McLuhan’s position is an extreme one: for him, the content of a television program or a newspaper story is far less important than the logics of these media, the ways they transform human sensation and experience.

McLuhan’s approach, which he calls *media ecology*, can be most clearly understood when applied to technologies that might not normally be described by the term “media,” at least not in the way we use it in contemporary culture. A classic example is the lightbulb, which McLuhan considers a medium because it enhances vision at night, at the same time as it diminishes other senses.

Often misunderstood as a “hard” determinist, McLuhan actually embodies an immensely pliant version of “soft” determinism. Under this account, as David Kaplan puts it, “Technology mediates and steers a society, but it does not quite drive it” [5, p. 2].

1.2 Social Constructivism

While it remains familiar, at least, in media and cultural studies, media ecology never really caught on in sociology, anthropology, and other fields that study how human cultures manipulate technology. In the field of Sciences and Technology Studies (STS), an alternative known as social constructivism, or more popularly as SCOT—for Social Construction of Technology, prefers historical and ethnographic methods to account for how people come to develop and then use particular technologies instead of others [3]. SCOT advocates point out that technologies do not—and cannot—arise from thin air; rather, humans create them in response to social and cultural effects, ideals, goals, problems or related factors.

The strong influence of STS as a hybrid of humanistic and social scientific study notwithstanding, social constructivism is an example of a much broader trend in cultural studies of the last century: a strong, even overwhelming focus on matters of intangible human behavior, and a weak, even dismissive attitude

toward tangible objects, including the machines and devices of creativity themselves. For example, while the popular fields of literary and film studies occasionally consider the material construction of books and film and the like, scholars working in these fields more frequently ask questions about the ways such media represent human ideas, either intentionally or through the expression of unseen ideologies. In McLuhan’s terms, cultural studies is only concerned with content, and not with media properties.

1.3 A Hollow Black Box

To be fair, studies in STS do make frequent inquiries into scientific and technological apparatuses. An originary work in STS, for example, is Bruno Latour’s 1984 study of Louis Pasteur’s discovery of microbes, published in English as *The Pasteurization of France* [6]. But almost without exception, such studies focus on the ways technologies come in to being, rather than how people respond to them after the fact. This observation forms a criticism of SCOT advanced by Langdon Winner, and a counterpoint in favor of a certain kind of “soft” technological determinism [12]. As the title of his critique (“Upon Opening the Black Box and Finding it Empty”) suggests, social constructivist approaches fail to take into account the operation and use of specific technologies from the inside out—the black box remains, in Winner’s words, hollow.

Unfortunately, the popularity of cultural studies and STS approaches to media and technology have led to naive and unreasoned accusations of technological determinism anytime critics begin opening and discussing the black boxes of specific technologies. Winner argues that this reaction comes partly from a threat to the ethnographic and interpretive methods such critics prefer. But another, even simpler explanation is that the technological determinism objection has become fashionable or even old hat, a stock answer anytime the lid comes off the box.

It is not entirely inaccurate to say that many objections to determinism thus arise from a misperception that *any* attention paid to the material construction and use of a technology automatically amounts to “hard” determinism, an extreme position that technology arises and evolves of its own volition, carrying humans away like the ebbing tide. Indeed, we agree with such objections, and embrace them, because they actually support a platform studies approach: people make negotiations with technologies as they develop cultural ideas and artifacts, and people *themselves* create technologies in response to myriad social, cultural, material, and historical issues. If we were to believe that technology manifests itself and unidirectionally influences the course of human history, we would be cutting off at least half of platform studies: The study of how our technologies, our computer platforms, embody particular cultural concepts and ideals, how they too are created in a cultural context.

3. MISCONCEPTION #2: PLATFORM STUDIES IS ALL ABOUT HARDWARE

While hardware platforms are convenient for and familiar to many people—scholars, companies, and videogame players, for instance—they are not the only kind of platform. A platform is a computing system of any sort upon which further computing development can be done. It can be implemented entirely in hardware, entirely in software (which runs on any of several hardware platforms), or in some combination of the two.

Important cultural production has been done on both hardware and software platforms. The Atari VCS is entirely hardware, consisting of a circuit that lacks even built-in ROM. Java, on the other hand, offers a virtual computer, implemented in software and capable of running on many different hardware systems. It is referred to by Sun as a platform and is useful to consider as such:

A platform is the hardware or software environment in which a program runs [...] like Microsoft Windows, Linux, Solaris OS, and Mac OS. Most platforms can be described as a combination of the operating system and underlying hardware. The Java platform differs from many other platforms in that it's a software-only platform that runs on top of other hardware-based platforms [11].

In many cases, platforms contain other platforms, just as McLuhan's notion of a medium contains other media.

If it's useful to think of a software environment independently of the hardware it runs on, it makes at least some sense to think of that environment as a platform. This is certainly the case when digital media work is done in Java and Flash, systems that were designed to work similarly across different hardware platforms. Saying something runs on "Windows" takes the perspective that the software is the important aspect of the platform, while using the term "Wintel" (a portmanteau of "Windows" and "Intel") includes the hardware level as well. When publishers put together books full of BASIC programs in the early 1980s and didn't specify a particular microcomputer, they were treating BASIC as if it were a platform—even though it wasn't even really a single language, but a family of a related languages, implemented differently. A platform studies approach shouldn't neglect these sorts of differences, of course, but these factors indicate that it may be useful to consider BASIC and other software systems as platforms.

4. MISCONCEPTION #3: PLATFORM STUDIES IS ALL ABOUT VIDEO GAMES

The home videogame console is an influential and important type of platform, and one of the most easily identifiable platforms, because manufacturers have standardized the design and features of such systems and have spent a great deal of time in advertising them and making them distinctive. But platforms are pervasive in all sorts of computing. Personal computers like the Apple II are platforms. Programming languages such as BASIC can be thought of as platforms. Culturally important systems from decades past, such as the PLATO systems of the 1960s and '70s, are platforms. Platforms support digital art, hypertext, interactive fiction, chatterbots, recreational programs that aren't standard games, and other sorts of new media production.

This misconception no doubt stems in part from the choice of the Atari VCS, the first massively successful video game platform, as the first object of study in the platform studies book series. But perhaps the general separation between videogame studies and other digital media studies, enforced by separate conferences and journals and different spaces on book store shelves, is also partly to blame. We hope the platform studies series will afford opportunities for writers and readers to cross these boundaries; it is certainly not meant in any way to help solidify them.

5. MISCONCEPTION #4: EVERYTHING THESE DAYS IS A PLATFORM

While we see the concept of "platform" as covering more than just video game systems, and more than just hardware, we have focused our series on the study of *computational* or *computing* systems that allow developers to work creatively on them. We consider computational platforms without denying that there are other sorts of platforms as well—oil-drilling platforms, railroad platforms, political platforms, and communications platforms, to name a few. Computational platforms, unlike these others, are the (so far very neglected) specific basis for digital media work. In taking "platform" to mean this, we agree with Marc Andreessen, co-creator of the Mosaic browser, founder of Netscape, and co-founder of social network website Ning:

Definitionally, a "platform" is a system that can be reprogrammed and therefore customized by outside developers—users—and in that way, adapted to countless needs and niches that the platform's original developers could not have possibly contemplated, much less had time to accommodate [1].

Not only does this definition explain the platform concept concisely; it also explains how the flexibility of platforms provides them with creative potential. In a later blog post [2], Andreessen, who focuses on software platforms, emphasizes (with boldface in the original) that "[T]he key term in the definition of platform is 'programmed'. **If you can program it, then it's a platform. If you can't, then it's not.**"

1.4 Opposition to the Computational Sense

In an analysis of "how online content providers such as YouTube are positioning themselves," communication scholar Tarleton Gillespie considers the Web 2.0-era use of the term "platform" in detail [Gillespie 2009]. He begins "by highlighting four semantic territories that the word 'platform' has signified in the past, as its emergence as a descriptive term for digital intermediaries depends on all four." These territories are the computational, architectural, figurative, and political. Of the first, he writes:

In a technical context like this, the use of the term "platform" certainly harkens back specifically to its computational meaning: an infrastructure that supports the design and use of particular applications, be it computer hardware, operating systems, gaming devices, mobile devices, and digital disc formats.

While looser than the definition offered by Andreessen (whom Gillespie quotes) in that it includes data formats, this take does at least characterize the computational sense of platform. Gillespie describes Andreessen's position as one that tries "to tie the word back to its computational specifics" before concluding: "Platforms are platforms not necessarily because they allow code to be written or run, but because they afford an opportunity to communicate, interact, or sell."

Gillespie states that the current meaning of "platform" is broad, and he rhetorically buttresses this position by characterizing a 2007 blog post by a major industry figure as retrograde: trying to "tie the word back" just as others "harken back" to something a term "has signified in the past."

Gillespie's view of the term "platform" is perhaps reasonable when we consider only the public rhetoric of "online content

providers,” but if he reads the computational sense of “platform” as outdated, this view is not at all tenable. No matter how the “platform” has been applied, its sense as a computational infrastructure very much remains. Current video game developers, for example, have a very clear idea of what “platform” means, and they use the term in the same way that we do and that Andreessen does. Like any software developers, the ones at YouTube would also have to be familiar with this definition of platform and to use the term in this sense when they write software. The sense of a platform as a computational platform is just as current as any other in use today, and is certainly, overall, the most relevant one in the history of digital media.

1.5 Types of Platforms

The clearest case of a computing platform is a foundational system that supports general-purpose computing: a mainframe, minicomputer, or microcomputer running a particular operating system; a programming language; or a video game console or handheld. Some systems may be communication platforms or simply very large-scale computing systems, but may not be best understood as computing or new media platforms because they do not mainly support the development of general digital artifacts, including computer programs.

Andreessen, speaking specifically about software platforms (and mainly online software platforms), identifies [2] three levels of platform.

Level 1, the “Access API,” is the easiest kind to create and is “the approach taken by eBay, Paypal, the Google Search API (before they killed it), Flickr, Delicious, etc.” It requires a lot of work from third-party developers, who have to develop and run their own applications and provide everything necessary for doing so.

Level 2 is that of the “Plug-In API,” in which a greater integration with the platform is allowed. Photoshop and Firefox are applications that showcase this type of capability; on the Web, Facebook was the first major example.

Finally, Level 3 indicates a runtime environment in which developers upload code and run it directly on the platform, as they would on a computer’s operating system. Andreessen’s Ning is an example in the online world. Andreessen sees Level 3 platforms as having tremendous potential, as lowering the barriers to development and opening up new software ecosystems; because of this, they are worth the great effort and many resources that they require.

In turning to video game systems and computers, we see other differentiations between platforms: Some can be programmed only at the register level in machine code (the Atari VCS), some have a small operating system which can be used by assembly programmers (the Intellivision), and some have high-level languages built into ROM (the Apple][, the Commodore 64, and many other home computers). Some are built around specific input and output technologies, or for particular purposes, or for more general use. There are many ways to slice platforms, but certainly, the ones that are most likely to be culturally important are those that are most accessible to people, that have interesting capabilities, and that specifically welcome developers.

1.6 Instead of “Is It a Platform?” ...

The question of whether something is or isn’t a platform may not ever have a useful answer, by itself. We could ask whether the Web is a platform—it certainly is, if we don’t limit ourselves to

thinking about HTML and static documents that are somehow delivered. Is *World of Warcraft*? *Second Life*? *LambdaMOO*? Certainly we can think of all of these as platforms, since they have APIs. But the real question should be whether a particular system is influential and important *as a platform*. Something is a platform when a developers consider it as such and use it; that activity can be more or less culturally interesting. Rather than asking “Is it a platform?” we might ask “What interesting or influential things have been developed on the system?” and “Does the system have unique or innovative features as a platform?” If the system is really most interesting as a large ongoing game, as code and performance by players, platform-level discussion may reveal little. But if an API allows the creative and unexpected use of a multiplayer system, to produce machinima or to allow new, modded instances of the large-scale game to appear, for instance, then considering the platform level may be very interesting and may shed light on these creative practices.

Platform studies in an opportunity to connect computation (at a fundamental level) with culture and creativity. By some reckoning, communication studies have existed for a century. In any case, communications systems, which remain important today, have been studied for a long while. While digital communications systems deserve attention, platform studies is focused on computational platforms, the overlooked basis of a half-century of computational, digital work. The approach (and the book series that invites examples of it) was founded to invite this new type of study into a kind of platform that has been and remains central to the creative use of computing.

6. MISCONCEPTION #5: PLATFORM STUDIES IS ABOUT TECHNICAL DETAILS, NOT CULTURE

Often, discussions of computer technology revert to technofetishism: a celebration of the technical details of a computer system as an end in itself, rather than as a means to understanding the historical, cultural, or expressive relevance of that system. Contemporary culture is replete with gadgeteering of this sort, from celebrations of a new digital camera for its sensor megapixel count to the anticipation of the latest mobile phone for its slick appearance as much as (or more than) for the actual novelty of its features. Such an attitude is short-sighted, as it fails to ask either practical or critical questions about technology. And for this reason, many scholars rightly denigrate discussions of technical details as uncritical idol-worship.

But often, those who embrace the geekery of such discussion do so from a thorough and well-researched technical perspective, one that includes far more nuanced understanding of the operation of such systems than is usually found in more scholarly discussions of the cultural aspects of digital media, which can sometimes gloss over or even misstate the way that digital media systems function. We contend that both the technically adept gadgeteer and the technically ignorant critic represent extreme positions. Another option exists, one which incorporates the positive aspects of both.

Platform studies is about the connection between technical specifics and culture. In one direction, it allows investigation of how particular aspects of a platform’s design influenced the work done on that platform—for instance, how the presence of a particular graphics mode enabled games of a certain sort to be made and made these games appealing to developers. In the other

direction, it looks at how social, economic, cultural, and other factors led platform designers to put together systems in particular ways. The approach recognizes that not only the user's experience, but also interface, form and function, code, and platform, are fully embedded in culture. When undertaking such a task, interrogations of technical detail become important indeed. Such questions are posed not for their geek value alone, but in order to shed light on the relationships between technology and culture.

7. MISCONCEPTION #6: PLATFORM STUDIES MEANS THAT EVERYONE IN DIGITAL MEDIA WILL HAVE TO GET COMPUTER SCIENCE TRAINING OR LEAVE THE FIELD

Understanding platforms does involve technical investigation, which can be undertaken individually or collaboratively. However, there is nothing about the platform studies focus that would run any new media scholars out of the field. Platform studies is not an attempt to take over cultural studies, literary criticism, critical approaches to interface, game studies, code studies, and certainly not software studies, with which it is highly compatible and consistent. It is an attempt to understand a layer of new media that has been neglected using appropriate methods.

Nevertheless, we think that the time has come for digital media scholars, and particularly the ones still undertaking their formal education, to learn more about the ways computer hardware and software are designed and programmed. Such a knowledge need not be of the same order as that of a computer scientist or electrical engineer; the new media scholar is aiming to understand technologies well enough to connect them to culture, but not to invent new algorithms, computer architectures, or hardware and software techniques. To greatly emphasize such training could even be detrimental to the particular interests of the digital media scholar, who also needs a deep engagement with the humanities.

But just as the serious scholar of film might choose to learn about film production in order to understand the methods by which his chosen medium is created, and a serious scholar of the book might study bibliography, printing processes and technologies, and how binding and paper-making is done, so the serious scholar of digital media might need to delve deeper into the material construction of software and hardware. An appropriate education in these areas would not be focused on creating new computer platforms or on becoming an expert developer upon them, but on knowing the best questions to ask about existing ones and how to go about answering them.

As we wrote previously, SCOT advocates, according to Langdon Winner, have feared the opening of the black box because the use of new methods, particularly those that are difficult to understand and come from other disciplinary traditions, is seen as a challenge to those who use the current, dominant methods. It may be that methodological change is needed to uncover new complexities and to be required to make progress in a field. The training necessary to address what we call the code and platform layers of new media does not require a computer science degree, and some are quite capable in these approaches without having completed any formal coursework. But investigation of these levels does require interest, commitment, and follow-through, and a

willingness to use new and challenging methods of thinking and investigation.

The scholars we need most in digital media are those who bring nuanced cultural analysis to bear on computer systems. And for such scholars, learning about logic gates, computer architecture, assembler, or high-level languages—essential aspects of computing systems—is no more complex than was learning the difficult and intricate theories of psychoanalysis or poststructuralism. It is simply a different mode of learning that allows different levels of insight into digital media. At this stage in the development of the digital media field, we should be beyond the point of wondering whether education in digital technologies is useful. Instead, scholars at different levels should be trying to select courses to take, to develop new courses, and to pursue independent studies and projects that expand our ability to understand computation as it exists in and relates to culture.

8. OUR CONCEPT

Platform studies investigates the relationships between the hardware and software design of computing systems (platforms) and the creative works produced on those systems, which include but are not limited to video games—digital art, electronic literature, recreational and playful programs, and virtual environments are all built upon platforms, too.

By choosing a platform, new media creators simplify development and delivery in many ways. Their work is supported and constrained by what this platform can do. Sometimes the influence is obvious: A monochrome platform can't display color, a video game console without a keyboard can't accept typed input. But there are more subtle ways that platforms interact with creative production, due to the idioms of programming that a language supports or due to transistor-level decisions made in video and audio hardware. In addition to allowing certain developments and precluding others, platforms also encourage and discourage different sorts of expressive new media work, as the Shockwave and Flash platforms show. (Platforms need not be hardware.) In drawing raster graphics, the difference between setting up one scan line at a time, having video RAM with support for tiles and sprites, or having a native 3D model can end up being much more important than resolution or color depth. And, of course, as we mentioned in our discussion of technological determinism, the nature of the platform is itself culturally situated, influenced by business, economic, social, and other factors; a full platform study will also consider how the platform came about in its particular shape, and how that particular shape later influenced how and what later things were brought about.

The platform's influence as experienced by a user is mediated through code, the formal behavior of the program, and the interface. Because the platform is "deep" or "far away" from the user experience, reaching it though these several layers, its influence can easily be overlooked, even in an otherwise careful analysis of a game, artwork, or other program. And, though this influence is often profound, a platform can be unconsciously factored out by someone who comes to understand and assume what a platform is like.

Particular platform studies may emphasize different technical or cultural aspects and draw on different critical and theoretical approaches, but they will be united in being technically rigorous and in deeply investigating computing systems in their interactions with creativity, expression, and culture. While

Platform Studies books will not have to be written by computer scientists, and should be addressed to readers without a computer science background, these books will drive deep into the workings of computers, opening an exciting and productive new level for readers and for the field.

9.ACKNOWLEDGMENTS

Portions of this paper are adapted from the text of the Platform Studies site, Bogost, I. and N. Montfort, platformstudies.org.

10.REFERENCES

- [1] Andreessen, M. 2007. Analyzing the Facebook Platform, three weeks in, http://web.archive.org/web/20071021003047/blog.pmarca.com/2007/06/analyzing_the_f.html
- [2] Andreessen, M. 2007. The three kinds of platforms you meet on the Internet, <http://web.archive.org/web/20071018161644/http://blog.pmarca.com/2007/09/the-three-kinds.html>
- [3] Barnes, B., Bloor, D., and Henry, J. 1996. Scientific Knowledge: A Sociological Analysis..University of Chicago Press.
- [4] Bogost, I and Montfort, N. 2007. Platform Studies: Computing and Creativity on the VCS, MPC, and Wii. In Proceedings of the Digital Arts and Cultures Conference (Melbourne, Australia September 14–18).
- [5] Kaplan, D.M. 2004. Readings in the Philosophy of Technology. Rowman & Littlefield.
- [6] Latour, B. 1988. The Pasteurization of France. Harvard University Press.
- [7] McLuhan, M. and McLuhan, E. 1988. Laws of Media: The New Science. University of Toronto Press.
- [8] Misa, T.J. 1988. How Machines Make History, and How Historians (and Others) Help Them Do So. Science, Technology, and Human Values 13: 308-331.
- [9] Montfort, N. and Bogost, I. 2009. Racing the Beam: The Atari Video Computer System. MIT Press.
- [10] Smith, M.R. 1994. Technological Determinism in American Culture, in Does Technology Drive History?: The Dilemma of Technological Determinism, Smith and Marx eds. MIT Press.
- [11] Sun Microsystems. 1995. About the Java Technology, <http://java.sun.com/docs/books/tutorial/getStarted/intro/definition.html>
- [12] Winner, L. 1993. Upon Opening the Black Box and Finding it Empty: Social Constructivism and the Philosophy of Technology. Science Technology & Human Values 18:3, 362-378.