Maximilian Johann Aelfers

# A Manifesto of the Architectural Discourse in the Post-Digital Era

## Produktivität und Werteverhältnis im Kontext formalistischer Sprachen

Who can write a manifesto?
How can you write it?

Can one person write a manifesto?
Can I write a manifesto?

Do you have to frame it?
Do I have to frame it?
How can it be framed?

How can I explain my manifesto?
Do I have to explain it?
I will explain it

The manifesto's content represents the author's view and does not claim to represent the view of the Academy of Fine Arts Vienna and the Institute of Arts and Architecture.

But hopefully, it does a little bit...

## Writing a Post-Digital Manifesto I

Ausgangssituation meiner Arbeit war der zeitgenössische Architekturdiskurs, der mit der Frage, welche Beziehungen zwischen Bauhauslehre und Wiener-Kreis-Philosophie bestanden, korrespondierte. Durch die analytische Auseinandersetzung mit dem Bauhaus und dem Wiener Kreis, als Gruppe und als Einzelpersonen, fand ich zu meiner Arbeit *„A Manifesto of the Architectural Discourse in the Post-Digital."* Wie im Untertitel meiner Arbeit erkennbar, liegt mein Hauptinteresse im Wertverhältnis der Architektur. Damit kommen Fragen über das Formale und die Formsprachen in der Architektur auf. Fragen, die schon am Bauhaus und im Wiener Kreis diskutiert wurden. Mit dem Formalen und dessen Formsprachen stellt sich auch die Frage: Wie kann man über zeitgenössische Architektur sprechen? In englischer Sprache, um dem universalen Gedanken, um einer übergreifenden Verständigung gerecht zu werden? Oder doch in einer Programmiersprache, um den Posthumanismus zu reflektieren? Vielleicht auch in den Sprachen, die am Institut für Kunst und Architektur an der Akademie der bildenden Künste Wien gesprochen werden, um die Ausbildungsstätte unseres Studios zu repräsentieren? Oder doch in der jeweiligen Muttersprache der Autorenschaft? Es kann keine klare Antwort geben. Sprachen, auch formale Sprachen, sind geprägt von Paradoxien und sind zum Teil selbst voller Widersprüche.

## Writing a Post-Digital Manifesto II

Daher bedient sich meine Arbeit aus den Sprachen Deutsch – als meine Muttersprache –, Englisch als meine Zweitsprache und Processing (Java basierend) als meine Programmsprache. Mit diesen Widersprüchen in den verschiedenen Formsprachen und der kritischen Analyse der post-digitalen Ära stellt sich auch die Frage der Schlussfolgerung oder Stellungnahme, wie man es auch immer benennen möchte.

Was und welche Form hat diese Schlussfolgerung? Reflektierend, zum Thema bezogen und rückblickend auf die Epoche der Moderne, entscheide ich mich für die Wahl des Manifestes als Format meiner Reflexion und Analyse.

Das Manifest ist meine persönliche Anteilnahme an dem post-digitalen Diskurs. Wie der Diskurs, so hat auch das Manifestformat keine Einheitssprache, kein einheitliches Reglement, es ist in sich widersprüchlich.

Wie man ein Manifest verfasst und was ein Manifest darf und was es nicht darf oder sein kann, kann nicht klar definiert werden. Ein Manifest ist immer etwas schwer Definierbares, auch wenn es viele Manifestverfasserinnen wohl anders sehen. Mein post-digitales Manifest tritt am Ende dieses Beitrages zum Vorschein.

## Google Search Autocorrect Poems I+II

While writing a manifesto, I also asked the Google-Auto-correct-Algorithm about architecture and created some poems.

### [architecture is]

frozen music

architecture is
a human right
architecture is art

visual art and
the buildings speak
for themselves

architecture is
a good idea
quotes
architecture is computer
a style

a church
a concept

12|06|2019 17.45h

### [Architektur ist]

architektur ist
gefrorene musik
das großartige spiel
der körper im licht

im idealfall immer
direkte auseinandersetzung
mit den menschen

architektur ist
geiselnahme
die mutter aller künste

gefrorene kunst
gefrorene musik

kreuzworträtsel
für mich
kunst

erstarrte musik

12|06|2019 17.47h

## The Post-Digital I

Since the digital turn, the borders between digital and analog, between virtual and physical, have been expanded and are no longer at odds with each other. The contrast has become blurred. The digital is now fully embedded in everyday life. Nevertheless, this does not mean the end of the digital age. Rather, it signifies a state of constant contestation. The era of the post-digital has begun. Carin Schirmacher points out that the term "post-digital" comes from discourses in the music and art scenes.[1] The artist Kim Cascone established it in his 2000 article "The Aesthetics of Failure", in which Cascone refers to Negroponte's 1998 article "Beyond Digital."[2]

1 Schirmacher 2017.

2 Negroponte 1998.

It is interesting to see that the post-digital was already part of the digital movement from the 1990s. Big Data Processes dominate Post-digital Design. The term big data intended, all kinds of different digital data structures that are "produced," saved, and distributed on the internet, mostly in cloud platforms. Besides the big data, other post-digital operations are taking over the design process. For example, crowdsourcing approaches are based on wikis and open-source technologies with mostly free accessibility. Inside all these data-mining and collecting data-cloud systems, a specific platform structure, a technology, a certain kind of system or mechanism is behind everything.

Benjamin Bratton calls it "The Stack." He describes the post-digital actions as one single phenomenon, almost like an accidental mega-machine. This megastructure is connected but divided into different layers, levels, or systems.[3] The name "The Stack" is referring to the layered architecture of specific software-hardware systems and is platform economic driven on a planetary scale.

3 Bratton 2015.

We can define it as a meta-structure. It is speculative and already existing in our physical and digital, in our post-digital world. With "The Stack" methodology, Bratton proposes that this planetary-scale computation happing in the post-digital time can be divided into different layers. He separates it into the Earth layer, Cloud layer, City layer, Address layer, Interface layer, and User layer.[4] Every layer is connected, subordinated, and related to all other layers. If one user wants to communicate with another user, the information goes all the way down to the earth layer and then back up again to correspond to the user.

4 Bratton 2015.

Who and what is the user, for "The Stack" it is not necessary. Human or non-human user are already mixing up and cannot be separated. It is a post-human era, a time of global computation and algorithmic governance. An example of post-humanism is *Shudu,* a digital model that is treated like a "real" model and was the face of the summer campaign 2019 of the fashion label Ellesse. Another example is *LilMiquela*, a digital influencer and California-based fashion blogger. She represents and corresponds in her totality to the contemporary youth culture.

## The Post-Digital II

Besides clouds and big-data structures, aleatoric processes also play a relevant role in post-digital design.[5] These processes look from the human perspective random and unpredictable, but they are still connected to a computation mechanism based on algorithmic calculations. Yet, these approaches on aleatoric methods seem from the human side often spontaneous and not well-thought through. That is why the potential of these "randomized" computational approaches are not yet fully estab-

5 Schirmacher 2017, S. 239-73.

lished in the design discourse and still imitated to a formal one. Nevertheless, the computer has an increasing part of design and form definition and decision-making processes. This transfer of specific tasks to the computer again creates the question of authorship. Similar to the architectural discourse of deconstructivism of the last century.

We have to discuss what role plays architecture in the post-digital. Is the new architecture limited to the formal, or is there more to get out of this new architeture to learn from it and develop meaningful architectural theory?

> Computers will be a sweeping yet invisible part of our everyday lives: We'll live in them, wear them, even eat them. A computer a day will keep the doctor away.[6]

6 Negroponte 1998.

> Now that we're in that future, of course, plastics are no big deal. Is digital destined for the same banality? Certainly, its literal form, the technology, is already beginning to be taken for granted, and its connotation will become tomorrow's commercial and cultural compost for new ideas. Like air and drinking water, being digital will be noticed only by its absence, not its presence.[7]

7 Negroponte 1998.

## The Post-Digital III

An example of non-human use of *The Stack* is Terra0. It is a self-possessing augmented forest. The forest acts as an organization in the Ethereum blockchain and sells its land for tokens, and later repurchases the tokens. It is generating an economic situation to preserve and proliferates itself. In the blockchain, nobody knows if you are human or not.

> I like to think of a cybernetic meadow where mammals and computers live together in mutually programming harmony like pure water touching clear sky.
>
> I like to think of a cybernetic forest, filled with pines and electronics where deer stroll peacefully past computers as if they were flowers with spinning blossoms.
>
> I like to think of a cybernetic ecology where we are free of our labors and joined back to nature, returned to our mammal brothers and sisters, and all watched over by machines of loving grace. Right now, please! It has to be![8]

8 terra 0 2019.

## Programming Languages I

Talking about the post-digital, you have to mention also programming languages and the history behind them. Formal Languages like coding are becoming more critical in the contemporary architectural discourse. Coding and working with visual programming tools, like grasshopper, are standard in the post-digital design era. Programming languages are formal languages for conversion and inter-action between humans and non-humans. A language aims to imply a set of different instructions, a code. Coding is a way to implement algorithms and calculations within the computer. Usually, a programming language is defined through two components: syntax (form) and semantics (meanings). Additionally, some languages are also defined by a specification document (standard) and a dominant implementation (interpreter, compiler) as reference. We can differentiate by using an imperative or a declarative syntax. Different languages are being created constantly. There is no universal language emerging.[9]

9 Sammet 1972.

Chronology of Programming Languages

Pre-1950
1804 Jacquard Loom
1943–45 Plankalkül (concept)
1943–46 ENIAC coding system
1946 ENIAC Short Code
1946 Von Neumann and Goldstine
1947 ARC Assembly
1948 CPC Coding scheme
1948 Curry notation system
1948 Plankalkül (published)
1949 Short Code
1950s
1950 Short Code
1951 Sort Merge Generator
1952 Glennie Autocode
1952 A-0
1953 Speedcoding
1954 Mark I Autocode
1954 ARITH-MATIC
1954 MATH-MATIC
1954 IPL I (concept)
1954–55 Fortran (concept)
1955 FLOW-MATIC
1955 PACT I
1955 PRINT
1955–56 IT
1956–58 LISP (concept)
1957 Fortran I (implement.)
1957 COMTRAN
1957 COMIT (concept)
1957–58 UNICODE
1958 Fortran II
1958 IPL II (implement.)
1958 IPL V
1958 ALGOL 58 (IAL)
1959 COBOL (concept)
1959 JOVIAL
1959 LISP (implement.)
1959 FACT
1959 APT
1959 TRAC (concept)
1959 MAD
1960s
1960 ALGOL 60
1960 COBOL 61 (implement.)
1961 COMIT (implement.)
1962 FORTRAN IV
1962 APL (concept)
1962 Simula (concept)
1962 SNOBOL
1963 CPL
1963 SNOBOL3
1963 ALGOL 68 (concept)
1963 JOSS I
1964 MIMIC
1964 COWSEL
1964 PL/I (concept)
1964 BASIC
1964 IBM RPG
1964 Mark-IV
1964 Speakeasy-2
1964 TRAC (implement.)
1964 P"
1965 RPG II
1965 MAD/I (concept)
1965 TELCOMP
1965 Atlas Autocode
1966 JOSS II
1966 ALGOL W
1966 FORTRAN 66
1966 ISWIM (Concept)
1966 CORAL66
1966 APL (implement.)
1967 BCPL
1967 MUMPS
1967 Simula 67 (implement.)
1967 InterLisp
1967 EXAPT
1967 SNOBOL4
1967 XPL
1968 ALGOL 68
1968 POP-1
1968 DIBOL-8
1968 Forth (concept)
1968 LOGO
1968 MAPPER
1968 REFAL (implement.)
1968 TTM (implement.)
1968 PILOT
1969 PL/I (implement.)
1969 B
1969 Polymorphic (PPL)
1969 SETL
1969 TUTOR
1969 Edinburgh IMP
1970s
1970 Forth (implement.)
1970 POP-2
1970 Pascal
1970 BLISS
1971 KRL
1971 CDL
1972 Smalltalk-72
1972 PL/M
1972 K&R C
1972 INTERCAL
1972 Prolog
1972 SQL
1973 COMAL
1973 ML
1973 LIS
1973 Speakeasy-3
1974 CLU
1974 GRASS
1974 BASIC FOUR
1974 PROSE
1975 ABC
1975 Irvine Dataflow (con.)
1975 PROSE
1975 Scheme
1975 Altair BASIC
1975 CS-4
1975 Modula

Chronology of Programming Languages

1976 Plus
1976 Mesa
1976 SAM76
1976 Ratfor
1976 S
1976 SAS
1976 Integer BASIC
1977 FP
1977 Bourne Shell (sh)
1977 Commodore BASIC
1977 IDL
1977 Standard MUMPS
1977 Icon (concept)
1977 Euclid
1977 Applesoft BASIC
1978 RAPT
1978 C shell
1978 RPG III
1978 HAL/S
1978 Applesoft II BASIC
1975 Irvine Dataflow (imp.)
1978 MATLAB
1978 SMALL
1978 VisiCalc
1979 Modula-2
1979 REXX
1979 AWK
1979 Icon (implement.)
1979 Vulcan dBase-II
1980s
1980 Ada 80 (MIL-STD-1815)
1980 C with classes
1980 Applesoft III
1980 Apple III Microsoft BASIC
1980–81 C BASIC
1981 BBC BASIC
1981 IBM BASICA
1982 Speakeasy-IV
1982 Draco
1982 PostScript
1982 Turing
1983 GW-BASIC
1983 Turbo Pascal
1983 Ada 83
1983 Objective-C
1983 C++
1983 True BASIC
1983 occam
1983 ABAP
1983 Korn Shell (ksh)
1983 Clascal
1984 CLIPPER
1985 PARADOX
1985 QuickBASIC
1986 Clarion
1986 CorVision
1986 Eiffel
1986 GFA BASIC
1986 Informix-4GL
1986 LabVIEW
1986 Miranda
1986 Object Pascal
1986 Erlang
1987 Ada ISO 8652:1987
1987 Self (concept)
1987 occam 2
1987 HyperTalk
1987 Perl
1987 Oberon
1987 Mathematica
1987 Turbo Basic
1987 Clean
1988 Octave
1988 Tcl
1988 STOS BASIC
1988 Actor
1988 Object REXX
1988 SPARK
1988 A+
1988 Hamilton C shell
1988-89 C (ISO/IEC 9899:1990)
1989 Turbo Pascal OOP
1989 Modula-3
1989 PowerBASIC
1989 VisSim
1989 LPC
1989 Bash
1989 Magik
1989 Python
1990s
1990 AMOS BASIC
1990 AMPL
1990 Object Oberon
1990 J
1990 Haskell
1990 EuLisp
1990 Z Shell (zsh)
1991 GNU E
1991 Oberon-2
1991 Python
1991 Oz
1991 Q
1991 Visual Basic
1992 Borland Pascal
1992 Dylan
1992 S-Lang
1993 Self (implement.)
1993 Amiga E
1993 Brainfuck
1993 LiveCode Transcript
1993 AppleScript
1993 K
1993 Lua
1993 R
1993 ZPL
1993 NewtonScript
1993 Euphoria
1994 Claire
1994 ANSI Common Lisp
1994 RAPID
1994 Pike
1994 ANS Forth
1995 Ada 95

Chronology of Programming Languages

1995 Borland Delphi
1995 ColdFusion (CFML)
1995 Java
1995 LiveScript
1995 Mercury
1995 PHP
1995 Ruby
1995 JavaScript
1995 Racket
1996 Curl
1996 Perl Data Language
1996 VBScript
1996 OCaml
1996 NetRexx
1997 Component Pascal
1997 E
1997 Pico
1997 Squeak
1997 ECMA Script
1997 F-Script
1997 ISLISP
1997 Tea
1997 REBOL
1998 Logtalk
1998 ActionScript
1998 Standard C++
1998 M2001
1998 PureBasic
1998 UnrealScript
1998 XSLT (+ XPath)
1998 Xojo (REALbasic)
1999 C99(ISO/IEC 9899:1999)
1999 Game Maker Language
1999 Harbour
2000s
2000 Join Java
2000 DarkBasic
2000 C#
2001 Joy
2001 AspectJ
2001 D
2001 Processing
2001 Visual Basic.NET
2001 GDScript (GDS)
2002 Io
2002 Gosu
2002 Scratch
2003 Nemerle
2003 Factor
2003 Scala
2003 C++03
2003 Squirrel
2003 Boo
2004 Subtext
2004 Alma-0
2004 FreeBASIC
2004 Groovy
2004 Little b
2005 Fantom
2005 F#
2005 Haxe
2005 Oxygene
2005 Seed7
2006 Cobra
2006 Windows PowerShell
2006 OptimJ
2006 Fortress
2006 Vala
2007 Ada 2005
2007 Agda
2007 QB64
2007 Clojure
2007 LOLCODE
2007 Oberon-07
2007 Swift
2008 Nim
2008 Genie
2008 Pure
2009 Chapel
2009 Go
2009 CoffeeScript
2009 Idris
2009 Parasail
2009 Whiley
2009 Dafny
2010s
2010 Rust
2011 C11
2011 Ceylon
2011 Dart
2011 C++11
2011 Kotlin
2011 Red
2011 MonkeyX
2011 Opa
2012 Elixir
2012 Elm
2012 TypeScript
2012 Julia
2012 P
2012 Ada 2012
2013 P4
2013 Cuneiform
2014 Crystal
2014 Hack
2014 Swift
2014 C++14
2015 Perl 6
2016 Reason
2016 Ring
2017 C++17
2017 Ballerina
2018 C18 (ISO/IEC9899)
2018 Fortran 2018[10]

10 Wikimedia, 2019

# Google Search Autocorrect Poems III+IV

## [architecture is bad]

architecture is bad

clean architecture
is bad

a bad career

brutalist architecture
is bad
postmodern architecture
is bad
modern architecture
is bad reddit

viper architecture is bad

bad career choice
architecture badge

bad for environment

12|06|2019 17.49h

## [architecture is good]

good career
architecture is good
or not

good idea
architecture is good
good idea youtube

architecture is good for you

microservice architecture
is good for
good idea ksiazka

a good idea blog
a good idea instagramm

12.06.2019 17:45h

## Writing a Post-Digital Manifesto IV

The last and most important questions of this work are: What kind of values are in the post-digital discourse inhabited? What ethics and morals do we want to apply in architecture? As architects, we have to start positioning ourselves more in values, rights, and policy. In other words, human rights and dignity must become crucial in the post-digital.

We have to force a paradigm shift in the post-digital era. Our focus as architects must change. How we train and teach young architects must change. We have to find the correct answers for the problems of our time, not just in architecture, in general. As human beings, we have to start protecting our planet and all beings and not focusing on destroying it and them. We as architects need to find a way to contribute positively to that cause. But the post-digital time does not just bring up much more significant problems and more difficulties. We can also benefit from the mechanisms of the Stack, the Big Data, Cloud platforms, to change and transform these inherited structures for the better.

Most of the relationships are generated aleatoric, but with deleting and adding new entries, the user adapts to the system. How I said in my manifesto, I am an optimist. We are able to transform our world for the better. I am sure!

# A Post-Digital Manifesto

What is wrong with architecture?
What is not wrong with architecture?

Wir leben in einer Zeit der post-digitalen und post-humanen Phänomene. Digitales Entwerfen ist dominiert von Big Data Strukturen, Cloud Plattformen, Open Source und aleatorischen Prozessen. Diese diffusen, aus dem Internet generierten Datenstrukturen und diese aleatorischen, für den Mensch zufällig wirkenden algorithmischen Prozesse, bilden die neuen Paradigmen der Architektur.

The post-digital age is the era of new geopolitical challenges. A time of global computation and algorithmic governance. Non-human users, planetary systems and mega machines are creating a new ductus. Borders and territories are emerging and shifting. Diffusion is enhancing a new set of rules.

The question is, can the architectural discourse deal with such a set of phenomena? Can it take up the challenges of the post-digital and the so-called Anthropocene era?

Die Grenzen zwischen digital und analog, virtuell und physisch sind nicht mehr klar trennbar. Digitaler und physischer Raum verschmelzen ineinander kontinuierlich. Genaue Trennlinien können nicht mehr widerspruchsfrei definiert werden.

But in this sphere of the post-digital, it seems like the contemporary architectural discourse has shifted to a just simplified discussion about these new set of dogmas. A critical contention is missing.

Today if we talk about computational architecture and the digitalization of the formal, some could argue it is based on rational and socio-economic decisions, decisions with cultural and social significance and depth. Yes, the post-digital design indeed has its roots in the cybernetic theories, modernistic ideas of architecture and philosophy of the last century, but the emphasis from a ratio-functional approach has shifted to a mainly aesthetic.

As we look back in the modernist era, it is true, that also architectural theory taught for example at the Bauhaus in Germany, dealt a lot with formal languages, yet there were a fluid discourse and critique. Criticism about the formal and the social responsibilities of architects in that time and how they gave certain weight and values to topics of that time.

Nowadays this fluid critique is mostly lost. The architecture of the post-digital is without a clear social and ecological responsibility. Questions of aesthetic and spatial consciousness are mostly unanswered. This path is leading to dull, generic and very often just simply meaningless architecture.

Ihre Anwender beziehen sich auf pseudo-theoretische Sätze und aleatorische Prozesse, die beliebig und ohne relevante Kontextualisierung und Reflexion bis zur Unendlichkeit wiederholt und wiederverwertet werden können.

Still, this kind of architecture is seen as an avant-garde style, a style for the future. Interestingly if we look back to modernistic architecture, we treat it historically as well as an avant-garde style.

The difference is, there was always a social and political agenda embedded into the style, that formed a field of discussion and critique.

Der Umgang heutzutage mit neuer Technologie und dessen Anwendung, wird fast schon als mystischer Prozess, zwischen Mensch und Maschine betrachtet. Unsere Verantwortung ist es, Technologien zu de-mystifizieren und Prozesse mit rationaler Anwendung, Analyse und Selbstreflexion zu implizieren und Maschinenstürmerei entgegenzuwirken.

Es muss hier erwähnt werden, dass auch der Minimalismus kein adäquates Gegenmodell zum post-digitalen Design bildet. Das analoge, prä-digitale Konzept des Minimalismus ist eine noch viel stärkere Simplifizierung von Ästhetik und Formsprache. Das rein Formale steht über allen anderen Werten. Soziale und Umweltbedürfnisse werden ebenso wenig im Diskurs berücksichtigt wie die Beantwortung relevanter gesellschaftlicher und politischer Fragen in der Architektur.

But the most alarming process are the failures in the agendas of elementary social needs and environmental issues. Social and environmental problems should not be neglected in the contemporary discourse any longer.

The main focus of architects supposed to be and always should be, preserving and improving our internal and external environments. Although aesthetical questions must be answered and perceptibility of space in all senses and emotions must be the highest dogma of all.

That is why we also have to question ourselves and our productivity as architects and as human beings.

What is our purpose?
Where are our values?
Is it meaningful or even necessary what we do?

Das Werteverhältnis im Diskurs scheint verschoben und verzerrt. Unsere Egos, unser Drang zur Selbstverwirklichung wird über die ursprünglichen gesellschaftskollektiven Aufgaben der Architektinnen gestellt.

But was it ever different?

Haben wir als Architektinnen, im Gravitationsfeld des post-digitalen generischen Entwerfens und des gelebten fast, schon fetischisiert anmutenden Minimalismus, überhaupt eine Chance diese neu entstandenen formalen Grenzen zu durchschreiten, geschweige denn zu durchbrechen?

In the recent architectural discourse, a critical dialog is missing. A clear lack of responsibility in social and ecological agendas is diagnosed.

As architects, we are not as powerful and important as we think we are.

That is why we have to push our egos aside, work on trans-disciplinary bases and start focusing on the important matters of our time. Because as a collective force, we can become more powerful and our decisions and actions become meaningful.

We also have to accept that we are part of the system. We are not chosen from above. We cannot behave as priest-like architects and think we are separated or different from other human beings. But to be part of the system does not mean we cannot change it.

As architects, we need to politically position ourselves and do not let others abuse us for their decision making.

But I am an optimist.

I believe the post-digital design can and will improve the architectural theory and transform it for the better. There is a lot of potential in the post-digital and post-human era to respond to socio-economic questions of our time and future.

On the path to do so, we have to push social and environmental values to significance, no matter how difficult it will be. We have to take action and act as social and political collectives.

Now, does a post-digital manifesto make sense?
Does it matter?
I mean seriously?!
Does anybody care?
I do...

What is wrong with architects?
What is wrong with me?
What is not wrong with me?

## Literaturverzeichnis

Carin M. Schirmacher, *Paradoxien des Digitalen Turn in der Architektur 1990-2015. Von den Verlockungen der Organischen: digitales Entwerfen zwischen informellen Denken und biomorphen Resultat,* Dissertation der Staatlichen Akademie der Bildenden Künste Stuttgart zur Erlangung des Grades eines Doktor-Ingenieurs, 2017.

Nicholas Negroponte, *WIRED Ventures Ltd. Beyond Digital*, Issue 6/12, December 1998, URL: https://web.media.mit.edu/~nicholas/Wired/WIRED6-12.html [Stand 14.06.2019]

Benjamin H. Bratton, *The Stack. On Software and Sovereignty,* MIT Press Cambridge Massachusetts, London England, 2015

Terra 0 Project. URL: https://terra0.org/ [Stand 14.06.2019]

Jean E. Sammet, *Programming Languages: History and Future,* IBM Corporation. Association for Computing Machinery Inc., 1972

Wikipedia Foundation, *History of programming languages,* URL: https://en.wikipedia.org/wiki/History_of_programming_languages [Stand 14.06.2019].