FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

PHILOSOPHISCHE FAKULTÄT
UND FACHBEREICH THEOLOGIE

Prof. Dr. Benjamin Jörissen
http://joerissen.name
benjamin@joerissen.name

Code, Design, Subjekt

Kritischer Designdiskurs und Critical Software Studies in bildungstheoretischer Perspektive

Otto-von-Guericke-Universität Magdeburg, 3.7.2015

I.

Software, Gesetz, Subjekt

„computationalism“

David Golumbia: The Cultural Logic of Computation. Cambridge, MA, Harvard Univ. Press 2009.

code/law

Lawrence Lessig (2000). Code Is Law. On Liberty in Cyberspace. http://harvardmagazine.com/2000/01/code-is-law-html [20.6.2015]

software vs. code

Wendy Hui Kyong Chun: Programmed Visions: Software and Memory. Cambridge, MA: MIT Press 2011.

performative
prozedurale
Ideologeme

performative
prozedurale
Ideologeme

(Entwicklungslinien einer Kulturgeschichte der Informationslisierung seit der Neuzeit)

- Informationalisierung der "res extensa“
- Informationalisierung des Ökonomischen
- Informationalisierung der Person (quantitative Leistungsmessung)
- Informationalisierung des Körpers (Medizin; Bioinformatik)
- Informationalisierung des Medialen:

performative
prozedurale
Ideologeme

II.

Raum, Subjekt, Artikulation: exemplarische Perspektiven der Software Studies

„software studies“ - „critical code studies“

Matthew Fuller (ed.), Software Studies: A Lexicon. Cambridge, MA: MIT Press, 2008.

Software Studies (Series, MIT Press), Edited by Matthew Fuller, Lev Manovich, and Noah Wardrip-Fruin.

computationalculture.net (seit Nov. 2011) 2011-

code/space

Rob Kitchin/Martin Dodge: Code/Space. Software and Everyday Life. MA: MIT Press, 2011.

API und Emotion

Taina Bucher (2013). Objects of Intense Feeling: The Case of the Twitter API. Computational Culture 4/2013.

„datasexuals“

Evgeny Morozov: Smarte neue Welt. Digitale Technik und die Freiheit des Menschen. Zürich: Blessing, 2013.

Algorithmen des Tentativen

Frank Pasquale: The Black Box Society. The Secret Algorithms That Control Money and Information. Cambridge, MA: Harvard UP, 2015

Algorithmisierungen des Verweisens

Anne Helmond (2013). The Algorithmization of the Hyperlink: Computational Culture 4/2013

„lock-in“:
Protokolle, Algorithmen,
Datenformate
ästhetischer Artikulation

Jaron Lanier: Gadget. Warum die Zukunft uns noch braucht.
Frankfurt/M.: Suhrkamp 2010.

III.

Design

Design als Wissenskultur

Design als Wissenskultur

Design als implizite Pädagogik

Design als relationaler Prozess

Design als diskursiver Prozess

Design-Þing

Software-Þing

Pelle Ehn: Design Things. Cambridge, MA: MIT Press, 2011

Affordanz und Subjektivation

Affordanz und Subjektivation

Spielräume digitaler Artefakte

IV.

Reflexionsangebote der Software Studies

Software als hegemoniale Artikulation

Wendy Hui Kyong Chun: Programmed Visions. Software and Memory. Cambridge, MA: MIT Press 2011.

Software als subversive Artikulation

Alexander R. Galloway/Eugene Thacker: The Exploit. A Theory of Networks. Minneapolis: Univ. of Minnesota Press 2007

flip — an entity that oscillates between two other data types

full — contains the complete universe of all information and matter

gateway — an associative entity connecting two or more other entities

glossolalia — an entity that is readable, writable, or executable only on a hypothetical machine

incontinent — an entity that involuntarily expresses itself as any other data type

infinity — an entity that is unbounded in quality and magnitude

palimpsest — an entity that contains the traces of a previous entity

poltergeist — an entity whose sole function is to invoke another entity

putrefaction — an entity that only produces its data when deleted

qualitative — an entity with contents that are not numerical in value

random — expresses a random entity from a random type

topology — an arrangement of interrelated, constituent parts

unknown — an entity that cannot be specified, identified, or evaluated in any intelligible way

vector — a compound type representing intensity and direction, consisting of an origin type and a destination type

whatever — an entity that always matters. It is not defined as part of any set (including the set of "whatevers") and cannot be identified through reference to either the particular or the general.

zombie — a process that is inactive but cannot be killed

Operators

() cast — transform an entity from its current type to a new type

-- debase — decreases the political or social standing of the predicate

!= disassignment — assigns any other entity except for the one specified in the predicate

=/ disputatio — spontaneously makes further, perhaps useless, distinctions within a given entity

:: figuration — establishes a figurative relationship between two or more entities

– hybrid — combines two or more entities into a new hybrid entity cast from the "empty" type

?+ manna — assigns new values to preexisting entities, the meaning of which is known only to the machine

<> negotiate — reassigns the predicate entity based on a negotiation between it and the subject entity

" " normative — attaches a political evaluation to a type or code block

^ parasite — establishes a parasitical relationship between two or more entities

++ privilege — increases the political or social standing of the predicate

Control Structures

boredom — executes a code block in a trivial, meaningless way that often results in dull, tedious, or sometimes unexpected output

exceptional — designates an abnormal flow of program execution and guarantees that it will never be handled as an error

flee — a branching construct that moves flow control from the current instruction to a stray position in the program

historic — executes a code block by evaluating an entity according to its current value as well as all previous values

maybe — allows for possible, but not guaranteed, execution of code blocks

never — guarantees that a block of code will never be executed. This is similar to block quotes in other languages, except that "never" blocks are not removed during compilation.

Software als ästhetische Artikulation

Noah Wardrip-Fruin: Expressive Processing. Digital Fictions, Computer Games, and Software Studies. MIT Press 2009.
Geoff Cox: Speaking Code. Coding as Aesthetic and Political Expression. MIT Press 2013.

Software als kulturelle Artikulation

Philip/Irani/Dourish: Postcolonial Computing: A Tactical Survey. Science, Technology & Human Value 37(1), 3-29.

Our Mission

Mukurtu (MOOK-oo-too) is a grassroots project aiming to empower communities to manage, share, preserve, and exchange their digital heritage in culturally relevant and ethically-minded ways. We are committed to maintaining an open, community-driven approach to Mukurtu's continued development. Our first priority is to help build a platform that fosters relationships of respect and trust.

Plateau Peoples' Web Portal

This portal is a gateway to the cultural materials of Plateau peoples that are held in Washington State University's Libraries, Manuscripts, Archives and Special Collections (MASC), the Museum of Anthropology and by national donors. The collections represented here have been chosen and curated by tribal consultants working in cooperation with University and Museum staff. Click on the "About" tab for more details.

Coeur d'Alene

hu, ku ch-n'ułkhw. The ancestral territory includes almost 5,000,000 acres of what is now north Idaho, eastern Washington and western Montana. Coeur d'Alene Indian villages were numerous and permanent, each village and the people there had a distinct name in the ancestral language. Collectively, members today call themselves, "Schitsu'umsh," meaning "Those Who Are Found Here."

Listen to Coeur d'Alene welcome

Explore Coeur d'Alene tribal path

Tribal Links Coeur d'Alene Tribe

Colvville

The Colville Confederated Tribes are comprised of 12 bands which include, the Moses-Columbia, San poil, Nespelem, Methow, Entiat, Colville, Lakes, Wenatchee (Wenatchi), Chief Joseph's Band of Nez Perce, Palus, Southern Okanogan, and Chelan. Our ancestral lands incorporated approximately thirty nine million acres in Central Washington and Southern British Columbia. Our ancestors made their living off the land, following the seasons and the resources. We govern our people in unity, maintain our ancestral ways, and strive to document our history.

Explore Colville tribal path

Tribal Links Confederated Tribes of the Colville Reservation

V.

Fazit

These 1

Software und ihre (praktischen, ästhetischen, kommunikativen, sozialen, ökonomischen, politischen) Logiken sind für Subjektivations- und Orientierungsprozesse **konstitutiv**.

These 2

„Software“ steht in diesem Sinne nicht für die Anwesenheit von Medienappaturen, sondern vielmehr für die **Appräsenz** von Strukturen, innerhalb derer Sozialität und Selbstbezüge sich entfalten.

These 3

Die Bildungspotenziale digitaler Architekturen – und damit der heutigen digitalisierten Welt – werden beschränkt durch hegemoniale Effekte auf der Ebene des **Designs** von Software (Code, Datenformate, Daten und begrenzte Kreise von Akteuren).

These 4

Designtheorie stellt dabei bildungstheoretische anregende Perspektiven bereit, da sie Wissen, Dinge, Programmatiken, Performanzen und Akteure systematisch aufeinander bezieht.

These 5

Aus Perspektive der Medienbildung stellt Software einen wohl unhintergehbaren Forschungsgegenstand dar, der allerdings – im bildungstheoretischen Interesse – medien- und kulturtheoretisch integriert werden muss.

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

PHILOSOPHISCHE FAKULTÄT
UND FACHBEREICH THEOLOGIE

Prof. Dr. Benjamin Jörissen
http://joerissen.name
benjamin@joerissen.name

Code, Design, Subjekt

Kritischer Designdiskurs und Critical Software Studies in bildungstheoretischer Perspektive

Otto-von-Guericke-Universität Magdeburg, 3.7.2015