

Abstract: “Live coding – an a-speculative paradigm of creative expression?”

Live coding, an emerging, experimental form of improvisational sound art based on the real-time generation and manipulation of digital code, occupies an equivocal position in the field of computer music and electro-acoustic art. Because live coding works on the level of the command line, it requires high-level knowledge of programming languages and of sound theory; the practice could therefore be charged with elitist and exclusive tendencies, and with representing an extreme form of deliberately difficult, inaccessible avant-garde art. But the opposite is true. By stripping their sound-making tools of ornamental user-interfaces, and by removing any intermediaries between themselves and the code, live coding artists perform a return to the ur-form of human-computer-interaction – the command line and input/output terminal. They enact and celebrate this return this doubly, first by hand-writing and hand-manipulating code, and secondly, in accordance with the Live Coding Manifesto, by vowing to always ‘share their screens,’ i.e., to project their laptop screens live on stage. Live coding thus takes a sharp stand against the obscurantism of established computer music, in which audiences were forced to speculate because they never really know what the artists are doing. Here, audiences are forced to witness every single variable change, syntax correction, and palimpsestic alteration. It appears that live coding seeks to demystify, rather than obscure, processes of data generation, data modification, and data flow. Does this mean that live coding is an anti-speculative art form? What if live coding artists incorporate telematic performance practices, in which the code is shown to audiences, but generated elsewhere, and conveyed in real-time over computer networks? How could live coding be mobilized in favour (or against) speculative approaches to creative expression?

Live coding artists, we might say, are gleefully demented purists. Facing the overwhelming complexity of real-time programming, they are happy to incorporate syntax errors, unwieldy code, and undecipherable layers of arrays/functions/variables into their art-making. Using examples and demonstrations, this paper explores possible reasons and the critical potential of this perspective, and asks how we might position the art form in relation to questions regarding the of speculative dimensions of computer-generated sound

Bio: Martin Zeilinger (PhD) is an Austrian media art researcher, practitioner and curator currently living and working in Toronto, and teaching at the University of Toronto and the Ontario College of Art and Design. His work on media art and intellectual property/authorship/agency has appeared in books such as *Sampling Media* (Oxford UP, 2014) and journals such as the *Computer Music Journal*.