



Software jako umělecký materiál

Autor textu: Jan Borner

Anotace: Článek se bude zabývat vznikem a vývojem softwarového umění, přičemž se zaměří na koncepční odlišnosti od umění nových médií. Software sloužil dlouho jako prostředek k tvorbě a reprodukci umění. Postupně se však sám stává předmětem zájmu umělců. Článek se bude snažit především odpovědět na otázky, které tvůrčí postupy jsou stejné jak pro umění nových médií, tak i pro softwarové umění, a které jsou zcela nové. A dále, co vede mediálního umělce k tomu, aby prezentoval svoji tvorbu v podobě počítačového programu, a nikoli jako digitální obraz, zvuk nebo hypertext. Cílem článku je definovat softwarové umění v rámci širší praxe umění nových médií. V článku vycházím především z prací Olgy Goriunové a Floriana Cramera.

Abstract: The article will deal with the origin and development of software art, focusing on conceptual differences from new media art. Software was long time served as a mean of creating and reproducing art. Gradually, however, itself becomes the object of artist's interest. The article will try to answer the particular questions which creative processes are the same as for new media art, as well as for software art, and which are entirely new. And further, what leads media artist to present their work in the form of a computer program, and not as a digital image, sound, or hypertext. This article aims to define software art within the broader practice of new media art. In the article we draw mainly from the works of Olga Goriunová and Florian Cramer.

Klíčová slova: software studies, software art, kultura, umění

Keywords: software studies, software art, culture, arts

Úvod

Způsoby, jakým je nám umění předkládáno, se v poslední době velice změnily. Může za to rozvoj informačních technologií, díky kterým už nemusíme odejít z domu, abychom si například „prošli“ výstavní plochy umělecké galerie. Ať už se jedná o hudební, filmová nebo třeba výtvarná díla, všechna lze digitalizovat pomocí softwaru. Vystává zde ale otázka, co se stane, když se ze softwaru jako prostředku pro tvorbu a prezentaci umění stane umělecký materiál.

Software se stává materiálem

Pojem „software“ odkazuje podle Andree Broeckmanna v obecném smyslu na provozní vrstvy počítačů, které se skládají z kódovaných elektromagnetických signálů a které tvoří rozhraní mezi uživatelským záměrem a technickým aparátem. Software je spojovatel, díky němuž lidé (nebo stroje) mohou působit na digitální stroje. Software je tvořen kódem, t. j. písemným pokynem, který obsahuje přesná pravidla pro procesy běžící na platformě hardware-software. (Broeckmann, 2006, s. 1)

Software byl z tvůrčího hlediska dlouho opomíjeným materiálem a sloužil pouze jako prostředek k výrobě a reprodukci uměleckých děl. Softwarové umění ovšem tuto zavedenou praxi nabouralo a snaží se rebelovat v zaběhnutém světě mediálního umění. Podle Olgy Goriunové můžeme pojem software art chápat jako termín pro soubor postupů, kdy je software brán jako kulturní konstrukt, který vytváří mnoho přístupů zaměřených buď na poetiku kódu, automaticky vytvořené umění, vizuální estetiku rozhraní a jazyka softwaru, programátorský folklór, software jako nástroj sociální a politické kritiky nebo jako umělecké médium. (Goriunová, 2007, s. 1)

Hlavní rozdíl, jenž odlišuje softwarové umění od umění nových médií využívajícího software, je fakt, že softwarové umění odhaluje a dává na odiv kód a pokyny, se kterými pracuje. Softwarový umělec se zaměřuje především na samotný tvůrčí proces a funkční vlastnosti softwaru, přičemž výsledný produkt je druhořadý.

Pojem „softwarové umění“ byl zaveden jako pokus popsat praxi, která je umělecká, non-funkcionalistická, reflexní a spekulativní. S ohledem na estetiku a politiku softwaru se stává počítačové programování součástí vlastní umělecké praxe.

Vznik softwarového umění

Počátky softwarového umění jdou ruku v ruce s vývojem informačních technologií. V padesátých letech dvacátého století, kdy na velkých univerzitách vznikají první programovací jazyky pro sálové počítače, začínají vznikat na této platformě i první hry, viry a jiné formy digitálního folklóru. Současná podoba softwarového umění je přímým důsledkem rozvoje tohoto digitálního folklóru po celém světě. (Goriunová, 2007, s. 1)

Je důležité si uvědomit, že software není v žádném případě vázaný pouze na formální pokyny pro počítač. Příkladem, kdy je algoritmus proveden mentálně, mohou být programovací příručky. Ty obsahují tištěné počítačové kódy, které si může čtenář následně představit krok po kroku ve své mysli.

Takovýto přístup k umění nacházíme již v roce 1970, kdy se v newyorském Židovském muzeu konala výstava s názvem *Software*. Kurátorem této výstavy byl umělecký kritik a teoretik Jack Burnham, jenž

na výstavě představil jako první prototyp hypertextového systému *Xanadu*. Jednalo se výstavu, která poukazovala k analogiím mezi konceptuálním uměním, jehož materiálem jsou pojmy, a softwarovými aplikacemi. (Cramer, 2001, s. 1)

Hodně z toho, čemu dnes říkáme softwarové umění, má svůj původ ve dvou skupinách založených v Londýně. Ta starší se vytvořila okolo Matthewa Fullera, Grahama Harwooda a skupiny I/O/D. Tato skupina vytvořila webový prohlížeč, který místo aby zobrazoval formátovanou stránku, ukazuje její zdrojový kód. Webový prohlížeč je stále možné stáhnout ze stránek skupiny, bohužel však prohlížeč pracuje na starších typech operačních systémů, a tak je již v dnešní době složité jej vyzkoušet.

Do druhé skupiny, jež se spíše zaměřuje na formální poetiku, estetiku programovaného kódu a na individuální subjektivitu vyjádřenou v algoritmech, patří programátoři a umělci Adrian Ward a Alex McLean. Oba vyhráli cenu pro softwarové umění *Transmediale* v Berlíně, kde v roce 2001 bylo softwarové umění jako takové vlastně poprvé definováno právě udílením cen pro tuto kategorii. Ward a McLean se spíše zaměřují na formální poetiku a estetiku programovaného kódu a na individuální subjektivitu vyjádřenou v algoritmech. (Cramer, 2001, s. 1)

Softwarová umělecká díla

Mnoho novomediálních umělců používalo během své práce programování, aniž by se jejich pozornost zaměřila přímo na softwarové umění. Největší rozvoj zažilo softwarové umění mezi léty 2000 a 2005, kdy se software stává sémantickou a ideologickou sadou nástrojů, jež se dá cíleně navrhovat a tvarovat.

Jednou z nejvýznamnějších postav dnešního softwarového umění je Amy Alexander, která často pracuje pod pseudonymy VJ Übergeek a Cue P. Doll. Působí na katedře výtvarných umění na Kalifornské univerzitě v San Diegu. Amy Alexander se věnuje digitálnímu umění a tvorbě internetových domén zabývajících se softwarovým uměním. Mezi její softwarová díla patří například software *Scream* (2005), který na pracovní ploše *Windows* reaguje na lidský křik. Alexander se tím snaží přiblížit software chování lidí, které není vždy strojově klidné. Software reaguje na váš křik a narušuje podle něj rozhraní *Windows*. Tento software si lze snadno stáhnout ze stránek <http://scream.deprogramming.us> nebo si jej můžete prohlédnout na http://www.youtube.com/watch?v=w2Ewc7HSmwo&feature=player_embedded.

Dalším, kdo se aktivně věnuje softwarovému umění, je Denis Roio, známý také jako Jaromil. Od počátku devadesátých let pracuje na vývoji svobodného softwaru a je považovaný za průkopníka v oblasti vytváření sítí, umožňujících přenos audia a videa na internetu. Jeho nejznámější dílo se jmenuje *TimeBasedText* (2005) a jedná se o free software, který zaznamenává činnost psaní textu a přenáší i jeho další informace. Příkazový řádek konzole může nahrávat a přehrávat úhozy na klávesnici s přesností na jednu milisekundu. Tento software poté odkrývá dynamiku písemné komunikace, při níž je možno zachytit váhání při psaní, ale i rozhodnost. Na těchto dvou příkladech je patrné, že softwarový umělec může být původem jak programátor, tak výtvarník.

Dnešní softwarové umění však nemusí splňovat podmínku, že jde o původní software napsaný umělcem, ale funguje v rámci množství dostupných softwarových kódů. (Cramer, 2001, s. 1) Vzhledem k tomu, že se osobní počítače a internet staly populárními, tak se i softwarový kód stal běžně používaným

a dostupným. Současné softwarové umění využívá v mnoha případech postmoderní strategie práce s již existujícím kulturním materiálem, takže se stává součástí postprodukce.

Závěr

Rozdíl mezi softwarovým uměním a uměním nových médií je především v přístupu k samotnému softwaru, jenž je v softwarovém umění chápán už ne jako prostředek k vytvoření uměleckého díla, ale jako materiál díla samotného. Tento posun zájmu od povrchu obrazovky počítače k programovacímu jazyku softwaru dává vzniknout dalšímu důležitému rozdílu, a tím je přesun pozornosti od výsledného produktu k samotnému procesu tvorby.

Použitá literatura:

- AMY, Alexander, 2013. *Amy Alexander* [online]. ©2011–2013 [cit. 22. 1. 2013]. Dostupné z: <http://amy-alexander.com/>.
- ARNS, Inke, 2005. Read_me, run_me, execute_me. Code as Executable Text: Software Art and its Focus on Program Code as Performative Text. In: Rudolf FRIELING – Dieter DANIELS (Hg.), *Medien Kunst Netz 2: Thematische Schwerpunkte*. Wien/New York: Springer. ISBN: 3211238719, s. 177–193 (dt.), s. 197–207 (engl.).
- BROECKMANN, Andreas, 2006. Software Art Aesthetics. Software art. *Mikro in Berlin* [online]. [cit. 22. 1. 2013]. Dostupné z: <http://www.mikro.in-berlin.de/wiki/tiki-index.php?page=Software+Art>.
- CRAMER, Florian – GABRIEL, Ulrike, 2001. Software Art. *Netzliteratur* [online]. [cit. 22. 1. 2013]. Dostupné z: http://www.netzliteratur.net/cramer/software_art_-_transmediale.html.
- GORIUNOVA, Olga, 2007. Software Art. *Digital Artists Handbook* [online]. [cit. 22. 1. 2013]. Dostupné z: <http://www.digitalartistshandbook.org/?q=book/export/html/26>.
- JAROMIL, Denis Roio, 2001. *EduEDA* [online]. [cit. 22. 1. 2013]. Dostupné z: http://www.edueda.net/index.php?title=Jaromil_Denis_Roio.