

# DOKUMENTACE ZÁVĚREČNÉ PRÁCE



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA VÝTVARNÝCH UMĚNÍ**

FACULTY OF FINE ARTS

**ATELIÉR GRAFICKÉHO DESIGNU 1**

GRAPHIC DESIGN STUDIO 1

**WRITER.AXDX**

WRITER.AXDX

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTORKA PRÁCE**

AUTHOR

**BcA. Kateřina Srbová**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**MgA. Barbora Fišerová**

**BRNO 2023**

# obsah

abstrakt	4
textová část	5–48
úvod	5–8
východiska, motivace	9–22
současné nástroje	
log: using myself as a use case	
metafora kanceláře a pracovního stolu, složky a soubory	
hodnoty, principy, fungování	
co by (ne)musel být writer.axdx	23–43
o psaní na počítači	
kódování znaků	
papír a dokument	
závěr	44–48
co je ekosystém	
co je community-software.axdx	
co je writer.axdx	
obrazová část	49–54
seznam použitých zdrojů	55–57

## abstrakt

Napsat větu, přečíst si ji znovu, vybrat část, která má být kurzívou, označit ji myší nebo pohybem prstů po touch padu či klávesnici (ups, o jeden znak vedle), zopakovat označení (tentokrát správně, uf), najít správné tlačítko nebo použít příslušnou klávesovou zkratku, napsat další větu, odstavec, kapitolu, esej, knihu... Co se stane v případě, že spadne internet, selže pravidelné ukládání, verze potřebného softwaru není aktuální? Většina běžně používaných nástrojů pro psaní textu nabízí spoustu možností formátování a tím směřují pozornost jinam než na samotné psaní. Tato diplomová práce přibližuje jejich funkce, (vzájemnou) kompatibilitu a uživatelskou zkušenost. Jejím cílem je vytvořit specifikaci nástroje `writer.axdx`, který by měl fungovat jako textový editor pro projekty v rámci `community-software.axdx` (ekosystém nástrojů a metod sloužící komunitám). Cílem tohoto nástroje je vrátit pozornost zpět k samotnému psaní, ve formátu, který není omezený proprietárním softwarem a nerozsype se při otevření v jiném editoru. Poskytnout možnost práce s textem podle vlastních potřeb, které jsou jindy definovány potřebami nebo metodami tzv. pokročilého uživatele. Mít vše, co je k psaní potřeba a nic z toho, co potřeba není.

## cíl práce

Cílem práce je přiblížit problematiku psaní a formátování textu v digitálním prostředí a vytvořit textovou specifikaci pro nástroj `writer.axdx`. Tato specifikace by pak měla fungovat jako výchozí bod pro jeho následný vývoj.

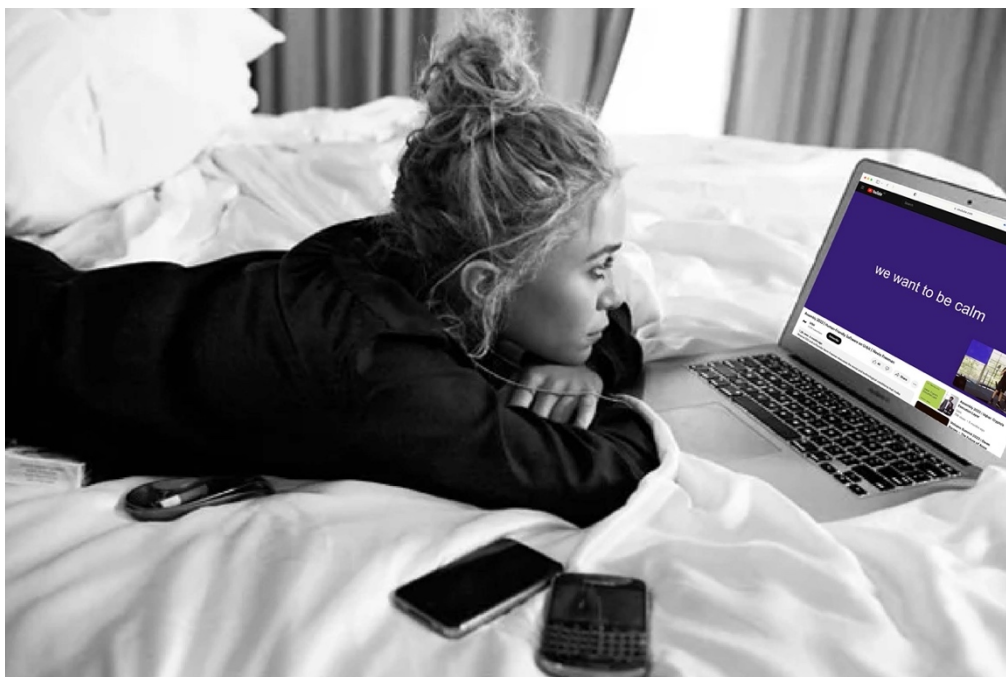
Text v plném znění a příjemnějším zobrazení dostupný také na Glossy Research (<https://glossyresearch.org/post/writer-axdx>).

## úvod

Začínám psát. Bříška prstů se pohybují po klávesnici, ze které mi vypadává pár písmen a klávesa *command*. Na bílém podkladu dokumentu, který jsem nejdřív pojmenovala a vložila do složky, pomalu přibývají černé znaky. Prostor, ve kterém se nacházím, má v názvu slovo *paper*.

Přemýšlím nad vztahem mezi psaním a čtením. *Já teď píšu a ty teď čteš*. Psaní by mělo uklidnit a třídit myšlenky, dát jim tvar a směr, který je předem daný. Zleva doprava, směrem dolů, někam. Organizuju a přesouvám slova, věty, odstavce. Z jednoho dokumentu do druhého a zase zpátky — momentálně jich mám otevřených pět, každý ve svém vlastním okně. Čtení je oproti tomu plynulý, lineární, do velké míry předem nadefinovaný proces. Obojí ale předpokládá alespoň

minimální narativ. Co když ale žádný neexistuje? Nebo není možný? Pokud lineárně nepřemýšlíme, proč prostředí, ve kterých jsme často nuceni pracovat, nepodporují nelineární pracovní metody a postupy? Chtěla bych prostě *jen* začít psát. Víc věcí zároveň, na jednom klidném místě, nezávisle na sobě, pokud chci. Bez nutnosti předem cokoliv definovat. Bez rozptýlení.



*Not Ashley Olsen watching [Human-friendly Software on Urbit](#)?<sup>1</sup>*

Cílem této práce je vytvořit textovou specifikaci a zpracovat probíhající rešerši, která se týká problematiky psaní a formátování textu v digitálním prostředí. Vnímám ji jako službu kolektivu AGDX<sup>2</sup> a pracovní skupině **community-software.agdx**,<sup>3</sup> jejíž kontext je pro pochopení celku stěžejní a proto jí patří samostatná část. S otázkou „Proč vyvíjet vlastní textový nástroj?“ na pozadí se budu věnovat východiskům a motivacím zabývat se na první pohled vyřešeným problémem. Poslední část shrnuje dosavadní rešerši — zpracovaná sesbíraná data a jejich struktura jsou

---

<sup>1</sup> Urbit, „Assembly 2022 | Human-friendly Software on Urbit | Nevin Freeman“, video, 34:37, *Youtube*, nahráno uživatelem Urbit 12. října 2022, <https://youtu.be/bTNAtgfRaUc> (cit. 15. 4. 2023).

<sup>2</sup> Původní definice AGDX, kterou stále používáme na našem webu [www.agdx.info](http://www.agdx.info), zní: „AGDX jsou členové AGD1, kteří se dlouhodobě podílejí na fungování fyzického a virtuálního prostoru ateliéru, rozvíjení sebe i ostatních.“ V současnosti je AGDX definice komplikovaná, protože stále více lidí dokončuje FaVU a opouští AGD1, ale dál zůstávají jako členové AGDX. Koncepce ateliéru, jeho jméno, resp. obecné konvence pojmenování, jsou v současnosti předmětem revize a řešení. (duben 2023)

<sup>3</sup> *community-software.axdx* je nezávislá, kolektivně vytvářená, vlastněná a udržovaná softwarová infrastruktura.

zároveň důležitou součástí výstupu a v plné šíři se nachází pod skupinou WRITER.AXDX na platformě Are.na (<https://www.are.na/writer-axdx>).

Tento text (nebo soubor textů) nemá ambice aspirovat na akademické texty, které se odevzdávají například na humanitních oborech. Nesnažím se být nezaujatá nebo dokonce objektivní — nenásleduju žádnou definovanou metodologii, celý proces i rešerše jsou intuitivní, v průběhu reagují na dění a diskuze v **community-software.agdx** a AGDX. Rešerše je zasazená do konkrétního rámce, který tvoří výchozí bod a základ, který se pokusím nastínit skrze zdroje a projekty, které považuju za inspirativní, některé části textu tak do velké míry fungují jako shrnutí mých poznámek a výpisků.

Nakonec — nejde o to prosadit si vlastní individuální vizi nebo pohled na věc, ale přinést podněty k diskuzím, které dál propojí a posunou naše společné uvažování a navrhování. Věřím, že intuice, vnímání, improvizace a pocity mohou, možná i nečekanými způsoby, přispět k novým řešením výzev, kterým čelíme. Data sice můžou říct, co se děje, ale nemůžou za nás dělat rozhodnutí.

*„Svou práci bych nenazýval výzkumem, protože nemá definovanou metodiku. V práci je přítomno mnoho analytických přístupů, ale jedná se především o intuitivní proces. Věci se pro mě stávají zajímavými prostřednictvím mého těla, které je neustále vystavováno očekáváním, interpretacím a touhám.“*

— Ryan Curo, <https://www.are.na/block/11865598>

*„Poincaré sám napsal, že věří, že logika není způsob, jak něco vymýšlet, ale je to způsob, jak strukturovat myšlenky, které podle něj logika spíše omezuje.“*

— Henri Poincaré, Wikipedia <https://www.are.na/block/991178>

*„Tato široce rozšířená představa, že vše musí být řečeno a může být vyřešeno jazykem, že každý skutečný problém je tématem k diskuzi, že filozofii lze redukovat na otázky a odpovědi, že se člověk může vyléčit pouze mluvením, že diskurs je jediným způsobem, jak něco naučit, tato teatrální, řečnická, publicitu hledající představa, postrádající stud a skromnost, zapomíná na skutečnou přítomnost chleba a vína, na jejich nevyslovenou chuť a vůni, zapomíná na to, jak vychovávat dítě prostřednictvím sotva znatelných gest, na přítakání a součinnost a na věci, které se nevyslovují, na nevyslovené projevy lásky, na intuice, které udeří jako blesk, na kouzlo, které přetrvává za vnějším vystupováním; tato soudcovská představa vylučuje bázlivé, ty, kdo nejsou vždy přesvědčeni o svých názorech, a ty, kdo nevědí, co si myslí, badatele; tato didaktická představa vylučuje ty, kdo nechodí do školy, skromné lidi, vynálezce, váhavé a citlivé, lidi s intelektem a dělníky, zarmoucené a chudé duchem; poznal jsem tolik věcí bez textů, tolik lidí bez gramatiky, dětí bez lexikonu, starců bez slovníku; tolik jsem toho prožil v cizích zemích, němý, vyděšený za oponou jazyků, skutečně bych okusil život, kdybych jen poslouchal a mluvil?“*

— Michel Serres, *The Five Senses : A Philosophy of Mingled Bodies*  
<https://www.are.na/block/10548715>

související are.na channely:

[WRITER \(research\) \[mood?\]](#)

[WRITER \(research\) \[memes\]](#)

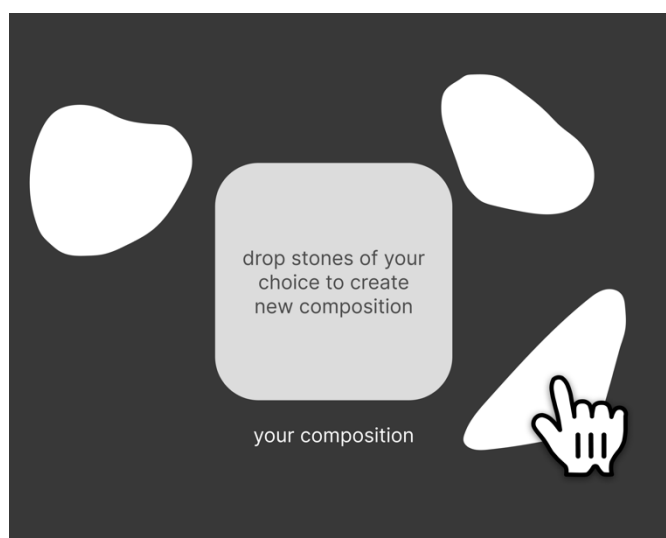
[WRITER \(research\) \[research\]](#)



# východiska, motivace

## současné nástroje

Moje potřeba a motivace zabývat se psaním začala mít uchopitelnější obrysy, když jsme s Jakubem Poláchem a Janem Suchým pracovali na platformě Glossy Research, která se měla zabývat alternativními způsoby a formami publikování textů. Řešili jsme, jak by se prostředí pro psaní i čtení dalo proměnit a navrhnout tak, aby bylo příjemnější, a zároveň svobodnější, pro člověka na obou stranách procesu. V AGDX se dlouhodobě bavíme o tom, že bychom v ideálním případě chtěli nahradit aktuálně používané nástroje svými vlastními, a Dropbox Paper je jedním z nich — kromě toho, že máme vymyšlených několik funkcí, které by se nám jako kolektivu i jednotlivcům hodily, vnímáme jako problém taky to, že uživatelé nevládnou svoje data, jsou sbírány jejich emailové adresy, pokud mají účet a dokument si zobrazí, a ve spoustě ohledech je podobně rigidní jako jiné podobné platformy a nástroje. Na Dropbox Paperu máme momentálně postavený i náš web agdx.info — proto jsme se začali v SS22 potkávat a přemýšlet nad možnými alternativami. Namísto rychlého řešení jsme se oklikou vrátili k přemýšlení a skicování toho, jak by mohl vypadat komunitní software.



log: using myself as a use case

Několik málo minut po začátku psaní této práce jsem si uvědomila, že je mi to fakt nepříjemné. Jasně, už jsem chvíli nepsala souvislý text, ale možná právě v tom je ten problém? V jednom okně mám místo, kde píšu, v druhém are.nu s předstrukturovanými channels — tyhle dvě okna se mi daří spojit a vidět najednou. Zároveň používám překladač, Figma, ve které mám načrtnutou vizuální strukturu celé rešerše, příležitostně ještě znovu letmo pročítám poznámky a zmíněné pasáže. Můj mozek různě přeskakuje z tématu na téma a často se stane, že mě napadne něco, co se mi momentálně nehodí co se týče chronologie. Kam si to mám zapsat? Někam *dolů*, kde si vytvořím sekci poznámek, ze které se brzy stane *big mess*? Založit úplně nový dokument a tím pádem další okno? Využít k tomu komentáře? Ve výsledku neudělám nic, protože přesvědčím samu sebe, že si to zapamatuju, což se ale nestane — *pure pain*. Jako obzvlášť praktický se ukazuje dokument s názvem *trash*, kam si odkládám nepoužité nebo vyhozené věty a odstavce, které se mi zrovna nehodí. Párkrát jsem se pro něco vrátila. V Paperu taky není možné dobře řešit citace, a tak k tomu účelu využívám komentáře. Citace by mohly být zbytečné anyway, kdyby dobře fungoval hypertext a další věci (kontroverzní a možná illegal názor). Běžně mám tendence upravovat věty tak, aby byly rovnou víceméně „prezentovatelné“, ale snažím se od toho odpoutat a nechat tuhle editační část procesu nakonec. Vznikají mi ale nedokončené části, ke kterým se chci později vrátit a dopsat, zvýrazňuju je žlutou barvou.

Snad nejhorší část procesu nastala v momentě, kdy jsem začala text přehazovat do MS Word. Mohla jsem sice formátovat v Adobe InDesign, ale Word mi přišel přímočařejší, a taky relevantnější k celé práci; zvlášť, když pořád patří k nejpoužívanějším nástrojům tohoto typu. Byla jsem asi nejnaštvanější za celý průběh a určitě v největší marnosti. Nejsem žádný odborník na MS Word, ale myslím si, že k používání tak základních nástrojů, jakými by podle mě nástroje pro psaní nebo/a úpravu textu měly být, by člověk neměl muset číst manuály, sledovat tutoriály a složitě se něco učit jen kvůli základním funkcím.

Ctrl+C, Ctrl+V: proč si můžu vybrat ze tří možností formátování po vložení? Rozumím sice specifickým použitím, ale výhoda nástrojů typu Dropbox Paper je právě v minimalizaci těchto drobných mezikroků, které v průběhu způsobují „malé, postupně se sčítající, naštvání“. Tohle funguje nejen při kopírování a vkládání z míst mimo Word, ale i v jeho rámci.

Obrázky: vážně je defaultní a optimální vložení obrázku do textu takové, kdy obrázek zajede pod text? Nemyslím si to, když vkládám obrázek, většinou chci, aby byl vidět. Takže jde o to proklikat se všemi ikonkami pro obtékání textu, dokud nenarazím na tu, která víceméně funguje, ale stejně se v průběhu všechno pořád rozhazuje.

V rámci macOS Word momentálně nepodporuje funkci Split Screen, kdy je možné na obrazovku v jednom čase zobrazit dvě okna (např. okno s nástrojem pro psaní a okno s are.nou) a používat zároveň.

Export do PDF: fungují jenom odkazy ve formátu <https://www.>, ty, které jsou vloženy pod jednotlivá slova po exportu zmizí. Vlastnosti souboru, do kterého se výsledný text ukládá, taky nejsou zrovna nedůležité. Linky obecně fungují dost hrozně, rozhodně ne tak hladce jako v Paperu, kde např. stačí označit slovo, do kterého chci umístit odkaz, který mám ve schránce, a dát Ctrl+V.

Word má své vlastní souborové formáty .doc a .docx, které mají do stability .txt daleko. Pokud si je otevřu jinde než v MS Word a dál je upravuju např. v Pages (macOS word processor, který má být s Wordem oboustranně kompatibilní), může se stát třeba to, že odevzdám bakalářskou práci, ve které je po exportu prvních deset stran úplně prázdných, i když je na svém počítači vidím (osobní zkušenost). Obrovská spousta nadbytečných funkcí vede mimo jiné třeba k tomu, že si spousta subjektů (institucí, firem...) vytváří vlastní, často hodně přísný, způsob formátování, který je *nutné* dodržovat. Konkrétní šířka okrajů, konkrétní velikost písma, konkrétní řádkování, konkrétní písmo. Nikdo mi neřekl, že si tuhle část užívá, spíš bývá naopak hodně stresující, zvláště, když by ten čas mohl být věnován např. samotné práci, která je zrovna psaná?

Mohla bych pokračovat, ale nechci tomu věnovat celou práci. Rychlých pár poznámek k nástrojům typu *Notion* nebo *Obsidian* — na mě je to pořád moc, ten široký výběr možností mě zbytečně stresuje, měl by tam být jen pokud sama chci. *Poznámky*, *TextEdit*, *Notepad* — fajn, ale to prostředí není optimální ani příjemné a napojení na zbytek community-software.axdx v rozsahu, jaký si představuju, nemožný. *Emacs*, *Vim*, *TextMate*, *Atom* — podle mě zbytečně komplikované pro běžné každodenní používání.

související are.na channely:

[WRITER \(research\) \[apps, tools, other\]](#)

[WRITER \(testing\) \[screenshots\]](#)

## metafora kanceláře a pracovního stolu, složky a soubory

*Je psaní něco jiného než vytváření díla nebo souboru? Je digitální pisatel něčím jiným než stránkou nebo adresářem?*

— *Sandy Baldwin, The Internet Unconscious*

Zabývat se digitálním psaním pro mě kromě zkoumání samotné činnosti a jejích bezprostředních aspektů znamená i základní seznámení se světem počítačů, potažmo webu, a potenciálními technologickými i myšlenkovými mantinely.

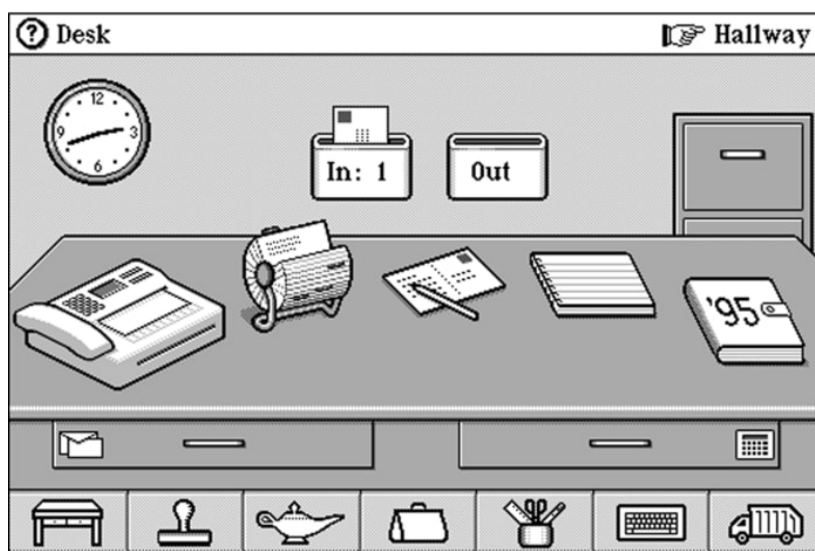
Maggie Appleton začíná svoji esej s názvem *Metaphors We Web By*<sup>4</sup> o historii našeho metaforického vnímání webu úvodem do situace, ve které byly počítače v době vzniku webu — byly to předměty, které byly nabízeny v kancelářích mezi bílými límečky<sup>5</sup> jako náhrada: nahradí složku s doručenou poštou (inbox) i malý

---

<sup>4</sup> Maggie APPLETON, *Metaphors We Web By*, 2022, <https://maggieappleton.com/metaphors-web> (cit. 15. 4. 2023).

<sup>5</sup> Bílé límečky je označení sociální třídy administrativních pracovníků, manažerů, právníků, auditorů, specialistů, zprostředkovatelů, apod.

bloček se žlutými lepícími papírky na poznámky (notes).<sup>6</sup> Od kancelářského prostředí je odvozená většina stále používaných metafor, které dodnes ovlivňují naše porozumění počítačům. Je důležité si uvědomit, že metafory, které používáme, nejsou pouhou jazykovou hříčkou – určují, jak na základní úrovni věcem rozumíme.



screenshot Magic Cap (1994), raného operačního systému navrženého na základě metafory pracovní plochy

příklady používaných metafor podle Aarona Marcuse, které uvádí ve svém článku *Metaphor Design in User Interfaces* (1998):<sup>7</sup>

podstatná jména

**pracovní stůl:** složky, dokumenty, papír

**knihy, noviny:** stránky, sekce, záložky, indexy

**fotografie:** alba, fotky

**televize:** programy, kanály (channels)

**diapozitivy, gramofonové desky:** stopy, třídící zařízení

slovesa

**lokalizovat:** přesun kurzoru na objekt

**vybrat:** jedno nebo dvojitě kliknutí na umístěný objekt

**přidat:** výběr objekt z repozitáře a jeho přetažení na místo

<sup>6</sup> APPLETON, *Metaphors We Web By*, 2022.

<sup>7</sup> Aaron MARCUS, „Metaphor Design in User Interfaces“, *Journal of Computer Documentation*, vol. 22, 1998, č. 2, s. 43–57.

**odstranit:** přetažení objektu do koše nebo odpadkového koše

**uložit:** přetažení objektu na místo

**ukončit:** výběr tlačítka nebo přepínače

**zrušit:** výběr „X“ pomocí kurzoru nebo krouživý pohyb

**přidat hodnotu:** posunutí/otočení ovládacího prvku po matici hodnot

**pohyb vlevo/vpravo, dopředu/dozadu, nahoru/dolů, do/z, do/z:** přetažení ovládacího prvku

**zastavit:** výběr tlačítka nebo přepínače

Podle Aarona Marcuse pomáhají metafory v několika ohledech:

- **reprezentují** (přetažení souboru do koše znamená smazání souboru)
- **popisují strukturu a procesy** (přehledový seznam souborů a složek odkazuje na hierarchickou strukturu)
- **vysvětlují** (ukazatel průběhu (progress bar) značí zpoždění při stahování)
- **vyjadřují koncepty nebo hodnoty** (recyklační šipka namísto koše může značit opětovné použití nebo ekologii)

Autor dále rozlišuje mezi standardními druhy metafor:

- **strukturální** (nahrazují části systému za jiné)
- **operační** (nahrazují chování systému za jiné)
- **pragmatické** (umožňují uživateli lépe pochopit – konkrétní desktopové objekty a procesy představují abstraktnější součásti operačních systémů)

Vývojáři v uplynulém půlstoletí použili mnoho různých metaforických substitucí, Marcus uvádí tyto příklady některých zastřešujících metafor:

- **fyzická místa:** místnosti, budovy, města, krajiny, planety
- **specifická místa:** knihovny, sklady, banky, nákupní střediska, chatovací místnosti
- **činnosti:** cestovat, létát, prohlížet, vyhledávat, surfovat
- **předměty:** pracovní stoly, knihy, televizory

Tzv. *desktop metaphor* (metafora pracovní plochy) je tak „vžitá“ v nás i v počítačích, že je často těžké si uvědomit, že věci, které interpretuje, například složky nebo soubory, nejsou *skutečné*. Jsou reprezentací toho, jak stroje organizují informace, které jsou optimalizovány pro vyhledávání a ukládání. Složky a soubory sice nejsou primárním problémem, který řeší nástroje na úpravu textu, úzce spolu ale souvisejí – jedna z hlavních věcí při práci s veškerými soubory je jejich pojmenování a zařazení do složky.<sup>8</sup> Pokud to neudělám např. v aplikaci Microsoft Word a sekne se mi počítač, může se stát, že veškeré dosavadní úpravy budou ztraceny.

*Soubory, bloky a objekty jsou úložné formáty, které uchovávají, organizují a prezentují data různými způsoby – každý s vlastními možnostmi a omezeními. Souborové úložiště organizuje a představuje data jako hierarchii souborů ve složkách; blokové úložiště rozděluje data do libovolně uspořádaných svazků s rovnoměrnou velikostí; objektové úložiště spravuje data a propojuje je s přidruženými metadaty. (...) Představte si skříň plnou kartoték. Každý dokument je uspořádán v nějakém typu logické hierarchie – podle skříňe, zásuvky, složky a pak podle listu papíru. Odtud pochází termín hierarchické úložiště, které je souborovým úložištěm. Jedná se o nejstarší a nejrozšířenější systém ukládání dat pro přímé a síťové úložné systémy, který pravděpodobně používáte již desítky let. Kdykoli přistupujete k dokumentům uloženým v souborech v osobním počítači, používáte souborové úložiště. Souborové úložiště má široké možnosti a lze do něj ukládat prakticky cokoli.*

— *File storage, block storage, or object storage? Redhat, 2018*<sup>9</sup>

Podle Teda Nelsona většina metafor způsobuje, že kontrolu nad akcemi přebírají mechanismy namísto uživatelů. Mezi největší problémy dnešních počítačů tak řadí

---

<sup>8</sup> Strukturováním, správou, archivací a interakcí s daty se v rámci **ecosystem.axdx** věnuje **database.axdx**.

<sup>9</sup> Red Hat, „File storage, block storage, or object storage?“, *Red Hat*, 2018, <https://www.redhat.com/en/topics/data-storage/file-block-object-storage#:~:text=File%20storage%20organizes%20and%20represents,links%20it%20to%20associated%20metadata> (cit. 15. 4. 2023).

simulaci papíru, imitaci fyzického prostoru, embeddedované a neoddělitelné hierarchie a zjednodušené mapování do souborů.<sup>10</sup> (Více v kapitole [papír a dokument](#) a části o [Tedu Nelsonovi a hypertextu](#).)

Guillaume Pascual ve svém textu *Tearing up the paper paradigm* (2016) tvrdí, že souborový systém odvádí naši pozornost — digitální věk nás měl osvobodit od úmorného indexování, ukládání a vyhledávání, a přesto ztrácíme čas organizováním složek, posíláním několika verzí abstraktních typů souborů nebo ukládáním každého nového souboru do dalších složek.<sup>11</sup> Pascual vyzývá k „*oproštění se od omezujících modelů minulosti a přijmutí digitálního prostředí, které je aktivní, spolupracující, efektivní a flexibilní.*“<sup>12</sup> S každou aplikací, kterou vyvíjíme, a rozhraním, které navrhujeme, bychom se měli snažit papírové paradigma rozbít a metaforu desktopu poslat rovnou do koše.

související are.na channely:

[WRITER \(research\) \[file, folder, hierarchy, structure\]](#)

[WRITER \(research\) \[metaphors\]](#)

[WRITER \(research\) \[apps, tools, other\]](#)

[WRITER \(testing\) \[screenshots\]](#)

## hodnoty, principy, fungování

Není překvapením, že svět informačních technologií není optimální, naopak — počítače ani zdaleka nenaplnují pozitivní utopické představy, které byly predikovány Vannevarem Bushem nebo Tedom Nelsonem. Uživatelé zbývá dělat jen to, co počítač potažmo internet chce a umožňuje, není možné to radikálně hacknout — místo rozvíjení inteligence uživatelů tak spíše prohlubují jejich hloupost; místo aby umožnily snížit nároky materiálního světa na zdroje se staly hlavní součástí problému; místo aby svět učinily srozumitelnějším zvyšují jeho nesrozumitelnost. Slovy Teda Nelsona je svět počítačů „*noční můrou vězení špinavého nočního klubu,*

---

<sup>10</sup> Theodor Holm NELSON, Robert Adamson SMITH, „Back to the Future: Hypertext the Way it Used to Be“, *Xanadu*, 2011, <https://xanadu.com/XanaduSpace/btf.htm> (cit. 15. 4. 2023).

<sup>11</sup> Guillaume PASCUAL, *Tearing up the paper paradigm*, 2016, <https://blog.hightail.com/tearing-paper-paradigm/> (cit. 15. 4. 2023).

<sup>12</sup> Ibid.



který se nezdařil“.<sup>13</sup> Zkusme se na chvíli oprostít od toho, co svět počítačů je, a zamyslet se nad tím, co by mohl být.



posvátný prostor místo vězení špinavého nočního klubu

Ville-Matias Heikkilä ve svém textu *Inverted Computer Culture: A Thought Experiment* (2023)<sup>14</sup> nastiňuje spekulativní realitu, ve které jsou počítače ze své podstaty staré a ať už s nimi děláte cokoli, je to automaticky považováno za starobylou a zkosnatělou praxi. Pochopení a následné ocenění počítačů je něco, co se vyvíjí pomalu v průběhu let, často prostřednictvím vstupních zájmů jako je meditace, kulturní historie nebo matematika. Počítače narozdíl od vnějšího uspěchaného a okázalého světa poskytují „posvátný prostor“ pro uvolnění, zpomalení a soustředění se na konkrétní myšlenku bez rozptylování. Pro lidi, kteří u počítačů strávili hodně času, je často obtížné se znovu přizpůsobit rychlému tempu, roztěkanosti a intelektuální jednorozměrnosti nepočítačového světa.

V této spekulativní realitě jsou počítače málokdy v soukromém vlastnictví — jsou považovány v podstatě za společné, nikoli osobní — způsoby jejich používání jsou ale často vysoce individualistické. Programování je nejpodstatnějším prvkem používání počítačů a nezdá se, že se setkáváme s uživateli, kteří si veškerý svůj

<sup>13</sup> Theodor Holm NELSON, „Here I Stand, at Age 80“, video, 47:56, *Youtube*, nahráno uživatelem TheTedNelson 4. září 2017, <https://youtu.be/mmfjM-SGIGs> (cit. 15. 4. 2023).

<sup>14</sup> Ville-Matias HEIKKILÄ, *Inverted Computer Culture: A Thought Experiment*, 2023, [http://viznut.fi/texts-en/inverted\\_computer\\_culture.html](http://viznut.fi/texts-en/inverted_computer_culture.html) (cit. 15. 4. 2023).

software vytvořili od nuly. Staletí výzkumu v oblasti informatiky oceňovaly jednoduchost, eleganci, drobnost a přehlednost oproti přizemním hodnotám jako je efektivita nebo škálovatelnost, takže existuje mnoho skvělých idejí, na kterých lze založit individuální programy.

V reálném světě si lidé spojují počítače s mnoha různými věcmi: korporátním odlidštěním, přehlceností, konzumním kapitalismem, odcizením od materiálního světa, zkrácenou dobou pozornosti, směšně krátkými cykly zastarávání atd. Často je obtížné odlišit tyto kulturní předsudky od „podstaty“ a ještě obtížnější je představit si alternativy. Heikkilä věří, že myšlenkové experimenty, jako je tento, mohou být užitečné pro rozšíření perspektivy.

### permacomputing

Ve svém fantasy příběhu otevírá Ville-Matias Heikkilä zásadní aspekt nového uvažování nad počítači — automatickou schopnost a možnost uživatelů vytvářet si jednoduše a bezbariérově vlastní software. Tento aspekt začal rozvíjet v eseji *Permacomputing* (2021),<sup>15</sup> což je pojem vycházející z principů permakultury a starání se o zahradní ekosystém. Heikkilä tvrdí, že by počítačové systémy a uživatelská kultura měly programování svým uživatelům co nejvíce přiblížit, učinit ho užitečným a přístupným. Každá komunita používající počítače by měla mít možnost vytvářet si vlastní software. Takový software by podle něj lépe odpovídal konkrétním potřebám, narozdíl od obecných řešení typu „one size fits all“.

Oproti monolitickým aplikacím by mohly existovat sety stavebních bloků, které by bylo možné dále použít pro vytváření programů obsahujících pouze funkce nezbytné k naplnění daného účelu. Proklamovanými hodnotami softwarového vývoje jsou dnes produktivita, obecnost a akumulace namísto jednoduchosti a citlivosti ke zdrojům. Ideálním stavem je podle něj možnost navrhovat a aplikovat „esoterické

---

<sup>15</sup> Ville-Matias HEIKKILÄ, *Permacomputing*, 2021, [http://viznut.fi/texts-en/permacomputing\\_update\\_2021.html](http://viznut.fi/texts-en/permacomputing_update_2021.html) (cit. 15. 4. 2023).

triky<sup>16</sup> aniž by byla ohrožena přehlednost nebo správnost hlavního kódu (například oddělením definice problému od implementování detailů).



Dalším aspektem světa počítačů je neustálá potřeba upgradovat jak software, tak hardware. U softwaru skrze aktualizace většinou dochází k postupnému zvětšování objemu dat programů — ideální stav by vedl naopak k jejich zmenšování a zrychlení. Programy, jejichž funkčnost se nemění, by podle něj neměly vyžadovat žádnou jinou údržbu než konzervaci.

Nakonec by podle něj měly společnosti podporovat vývoj softwaru, hardware a dalších technologií stejným způsobem, jakým podporují vědecký výzkum a vzdělávání. Výsledky veřejného úsilí by byly veřejně dostupné a volně modifikovatelné. Black boxy, lock-ins, nadměrná produktivizace a mnoho dalšího by mohlo být odsunuto na okraj.

---

<sup>16</sup> esoteric: intended for or likely to be understood by only a small number of people with a specialized knowledge or interest (Oxford Languages, Google).

aplikace může být jako jídlo uvařené doma

Roztomilým případem je text *An app can be a homecooked meal* od Robina Sloana,<sup>17</sup> ve kterém přirovnává svůj proces učení se programovat k učení se vařit. Většina lidí se neučí vařit jen proto, aby se z nich stali kuchaři. Častěji se učí vařit aby se mohli lépe a levněji stravovat, pokračovat v tradici nebo zkrátka trávit čas s člověkem, který je učí. Učit se programovat má kořeny v tržní hodnotě — je doporučováno jako prostředek ke kariérnímu růstu a lákavá položka do životopisu. Pokud programování osvobodíme od požadavků na obecnost, profesionalitu a škálovatelnost, stane se z něj úplně nová činnost — může být obohacující stejně jako je příprava jídla pro někoho, koho milujeme.

*„Tato aplikace pro zasílání zpráv, kterou jsem vytvořil společně se svou rodinou a pro ni, se nezmění, pokud nebudeme chtít, aby se změnila. Nedojde k žádnému náhlému redesignu, žádné záplavě reklam, žádnému obratu v honbě za pro nás nepochopitelnou uživatelskou základnou. Možná jednou zmizí, ale bude to naše rozhodnutí. Co je to za pocit? Nezávislost? Jistota? Suverenita? Je to prostě... pocit, že jsme doma?“<sup>18</sup>*

unix

Výše zmíněné představy, fantazie a koncepty do značné míry naplňují principy Unixu. Unix je skupina multitasking multiuser počítačových operačních systémů, které vycházejí z původního systému AT&T Unix, jehož vývoj zahájili v roce 1969 ve výzkumném středisku Bell Labs Ken Thompson, Dennis Ritchie a další.<sup>19</sup> Operační systémy jako je MacOS od Apple nebo Linux jsou komerční varianty Unixu.

Prvních deset let fungování byl Unix běžně distribuovaný se zdrojovými kódy, jeho vývoj založený na vzájemném uživatelském hodnocení byl klíčovým rysem unixové kultury od jeho počátků. Kniha *The Art of Unix Programming* (2003), jejíž autorem

---

<sup>17</sup> Robin SLOAN, *An app can be a homecooked meal*, 2020, <https://www.robinsloan.com/notes/home-cooked-app/> (cit. 15. 4. 2023).

<sup>18</sup> Ibid.

<sup>19</sup> Ken THOMPSON, Dennis RITCHIE, „The UNIX Time-Sharing System“, *Bell System Technical Journal*, vol. 57, 1978, s. 1905–1929.

je Eric Steven Raymond,<sup>20</sup> odhaluje, že se Unix jako jeden z mála softwarových technologií ukázal jako dostatečně trvanlivý na to, aby se kolem něj vyvinula silná technologická kultura, osobité umění a s ním spojená filozofie navrhování, které jsou předávány napříč generacemi softwarových inženýrů. Další takovou technologií je kultura internetu — od počátku osmdesátých let je čím dál obtížnější od sebe tyto dvě kultury oddělit.

Díky mnoha způsobům, které Unix nabízí pro propojování programů, lze součásti jeho základní sady nástrojů kombinovat a vytvářet tak nové možnosti, které tvůrci jednotlivých částí sady nástrojů nikdy nemuseli ani předpokládat. Principy a pravidla, která umožňují, aby programy nefungovaly jen jako izolovaná jednorázová řešení, ale jako synergické součásti sady nástrojů, Raymond do 17 pravidel:<sup>21</sup>

- **Pravidlo modularity:** Pište jednoduché části, které propojuje čistý interface.
- **Pravidlo přehlednosti:** Srozumitelnost je lepší než přehytralost.
- **Pravidlo kompozice:** Navrhujte programy tak, aby mohly být propojeny s jinými programy.
- **Pravidlo oddělení:** Oddělte zásady od mechanismu; oddělte interface od engines.
- **Pravidlo jednoduchosti:** Navrhujte jednoduše; složitost přidávejte jen tam, kde je to nutné.
- **Pravidlo úspornosti:** Napište velký program jen tehdy, když je na základě testování jasné, že nic jiného nepůjde.
- **Pravidlo transparentnosti:** Navrhujte pro viditelnost, aby byly kontrola a ladění jednodušší.
- **Pravidlo robustnosti:** Robustnost je dítětem transparentnosti a jednoduchosti.
- **Pravidlo reprezentace:** Složte znalosti do dat, aby mohla být logika programu hloupá a robustní.
- **Pravidlo nejmenšího překvapení:** Při navrhování interfacu dělejte vždy to, co nejméně překvapí.

---

<sup>20</sup> Eric Steven RAYMOND, *The Art of Unix Programming*, 1999–2003 (last revision to date), <http://www.catb.org/~esr/writings/taoup/html/index.html> (cit. 15. 4. 2023).

<sup>21</sup> Ibid.

- **Pravidlo mlčení:** Když program nemá co překvapivého říct, neměl by říkat nic.
- **Pravidlo opravy:** Když už musíte selhat, selhejte hlasitě a co nejdříve.
- **Pravidlo úspornosti:** Čas programátora je drahý, šetřete jím přednostně před časem stroje.
- **Pravidlo generování:** Vyhněte se ručnímu hackování; když můžete, pište programy pro psaní programů.
- **Pravidlo optimalizace:** Prototypujte předtím než začnete ladit, před optimalizací, to nechte chvíli fungovat.
- **Pravidlo rozmanitosti:** Nedůvěřujte tvrzením o „jediné správné cestě“.
- **Pravidlo rozšiřitelnosti:** Navrhujte pro budoucnost, protože ta tu bude dříve, než si myslíte.

Je těžké vyhnout se programování komplexních monolitů, pokud spolu programy navzájem neumí komunikovat. Jediný způsob, jak psát složitý software, který se nezhroutí, je minimalizovat jeho globální složitost — sestavit jej z jednoduchých částí propojených dobře definovanými rozhraními tak, aby byla většina problémů lokální, a aby upgrade části nutně nerozbil celek. Složitý kód s větší pravděpodobností způsobuje chyby, a do budoucna bude hůře čitelný pro uživatele, kteří jej budou spravovat.

související are.na channely:

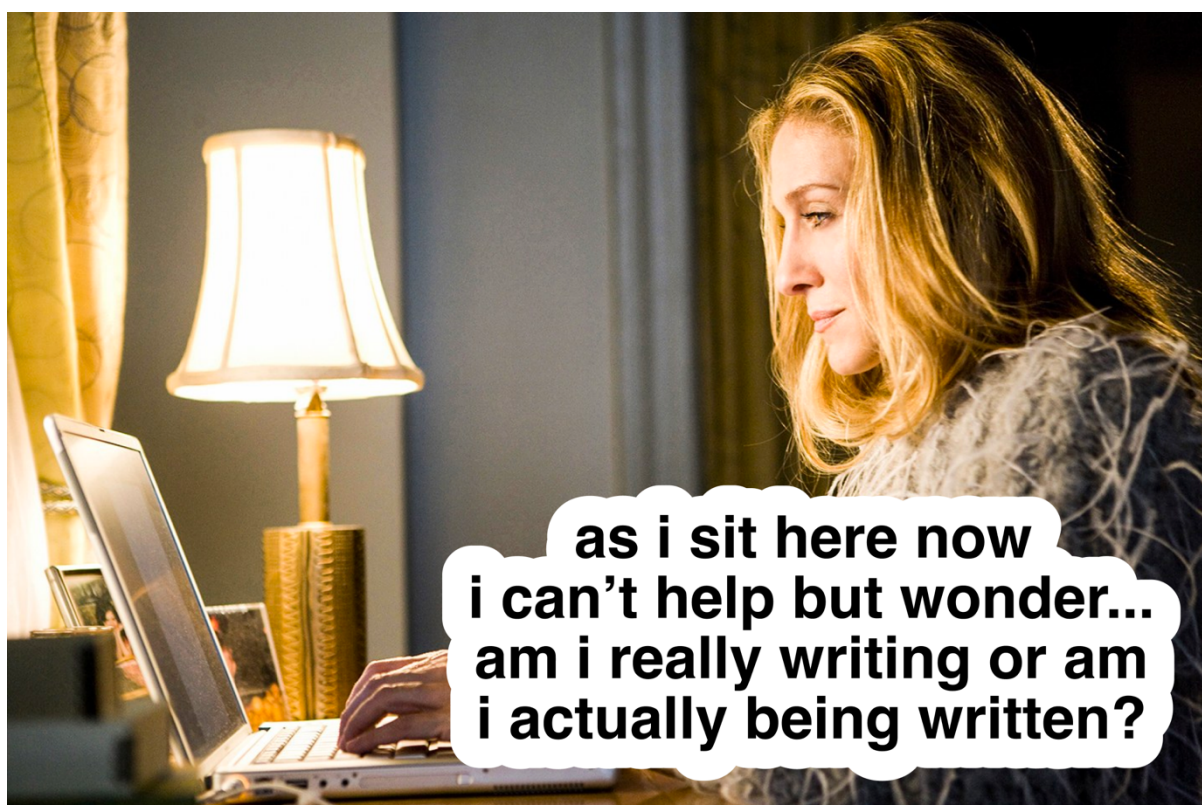
[WRITER \(research\) \[philosophy, theory, principles\]](#)

[WRITER \(research\) \[unix\]](#)

# co by (ne)musel být writer.axdx

## o psaní na počítači

Miliardy lidí, především z nejbohatších zemí světa, píší více než kdy dřív — na svých telefonech, tabletech, laptotech a stolních počítačích. Podle Richarda Seymoura tolik nepíšeme, jako jsme spíš sami *psáni* — změnou psaní z analogového na digitální dochází k tomu, že nás kód formuje v tom, jak jej používáme a data, která tím generujeme, jsou použita k formování nás.<sup>22</sup> Seymour dál tvrdí, že zběsilé tweetování, psaní palcem v MHD nebo aktualizování statusů během přestávek v práci je rozšířením probíhajících změn v pracovním prostředí, kde počítačem zprostředkovaná komunikace způsobuje, že psaní zabírá stále větší podíl celkové produkce.<sup>23</sup>



<sup>22</sup> Richard SEYMOUR, *The Twittering Machine*, Londýn: The Indigo Press 2019., s. 14.

<sup>23</sup> *Ibid*, s. 11.

Baldwin: Všechno na počítači je psaní

Psaní je nyní, jako téměř všechno ostatní, soustředěno především kolem *počítače* – přístroje, který operuje skrze psaní ve formě kódů a jejich transformací. Podle Sandyho Baldwina, autora knihy *The Internet Unconscious* (2015), je všechno na počítači psaní, všechno na síti psaní ve stránkách, souborech a protokolech. Všechno správně formované a kompletní. Struktura psaní v síti je podle něj strukturou všech věcí, přenesených, nahraných a stažených.<sup>24</sup>

kompilát výňatků z knihy *The Internet Unconscious* od Sandyho Baldwina:

Představ si, že píšeš na počítači. Kde je tohle na? Na obrazovce, ale ne na obrazovce, i když se na ní objevují písmena. Dotkneš se jí, jako to děláš se svým telefonem. Tvůj dotek levituje nad obrazovkou, můj se dotýká skla, které od ní odděluje mě a můj svět. Ale slova se nachází hlouběji. Pohybem prstů nebo dotýkáním se skla se nedostávám dál než na hladký povrch. Dál na tamto místo, které vidím skrze obrazovku, místo, kde se akumulují moje písmena, slova a zprávy. Píšu na počítači, vkládám slova na tamto místo. Obrazovka zobrazuje překlepy, chyby. Obrazovka je obrázek, který čtu. Obrazovka je zpráva pro mě, že moje psaní není dost, není dost pro *to*. Pro *co*? Není to jednoduše tak, že není dobré pro obrazovku – ne, nejen pro obrazovku. Ukazuje moje chyby, moje selhání. Co *to* je, a nedostatečné pro *co*?

Microsoft Word mi korporátním hlasem, delegovanými slovy Billa Gatese, říká, že neumím psát, že neumím správně hláskovat. Pohybují prsty s cílem uspokojit obrazovku. Správná slova jsou ta, která ji uspokojí; jakákoliv slova, která ji uspokojí, jsou ta správná. Cílem každého textu je uspokojit *tamto* místo.

Snažím se a zkouším dát obrazovce to, co potřebuje. I přesto jsou v mém psaní vždy další chyby. Pokud je moje psaní bez chyb, je to proto, že přesně splňuje požadavky obrazovky. Ale nikdy není úplně bez chyb. Text, který píšu, takového cíle nikdy nedosáhne. Nedokážeme si představit bezchybné

---

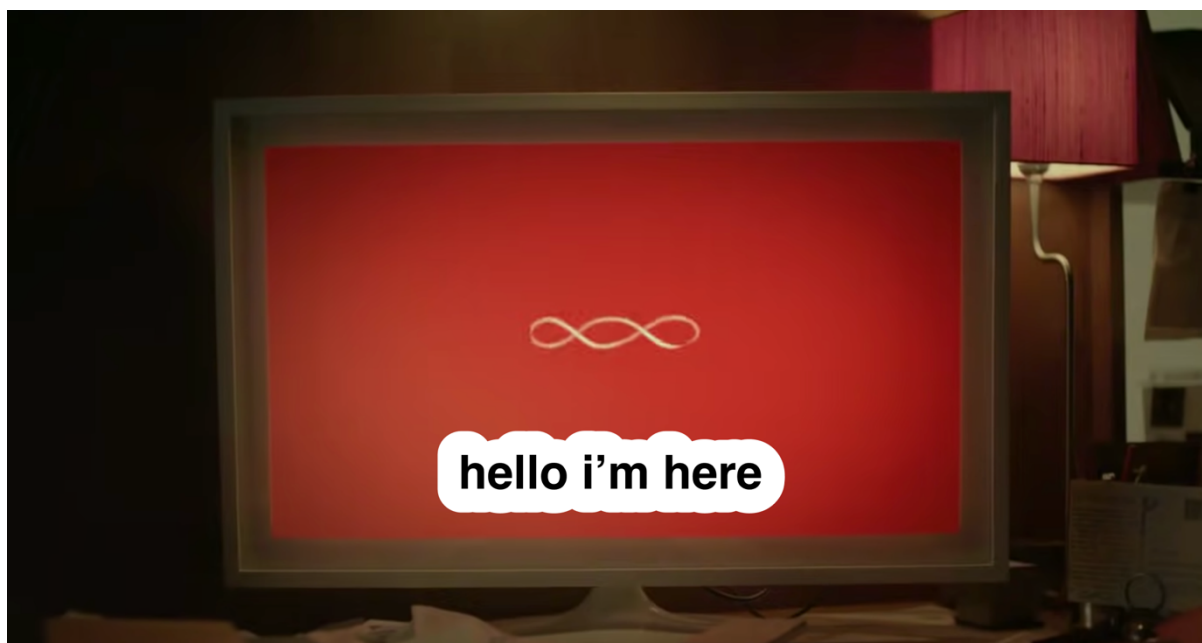
<sup>24</sup> Sandy BALDWIN, *The Internet Unconscious*, Londýn: Bloomsbury Publishing 2015, s. 1.



digitální psaní, protože i ten nejdokonaleji napsaný text je pouze ten, kterému obrazovka umožňuje existovat; text, kterému je umožněno popřít svou vlastní chybovost, aby splnil požadavek, kterým se stává.

Portál nebo zrcadlo? Pokud je obrazovka zrcadlem, můj obraz je jinde, teleportovaný na obrazovku. Přebývám na druhé straně. V této nemrtvé fenomenologii na mě prostřednictvím obrazovky působí imaginární druhý. Jako klam, jako halucinace, že ten druhý existuje, že má slova budou žít dál na *tamtom* místě. Píšu, abych ukojil touhu druhého. Píšu, abych dal druhému a naplnil ho. Dešifruji touhu druhého prostřednictvím kódů. Učím se příkazy aplikace, menu a skripty. Jsem zoufalý: dosáhnout správných příkazů, splnit příkazy, které mi byly dány. Udělám, co mi přikážete, ve vašem jméně, ve všech vašich jménech, vy, kteří mi dáváte svolení a dovolujete mi psát: ó Adobe, ó Microsofte, ó Apple, cokoliv na váš rozkaz.

Tím se odevzdávám přesně tomu, co po mě požadujete.



*záběr z filmu Her (2013)*

Flusser: Je programování pořád psaní?

Jako protiváha Baldwinova tvrzení, že cokoliv na počítači je psaní, stojí názor Vilema Flussera, podle kterého se musíme nejdříve zabývat vývojem, podstatou a smyslem

samotného psaní abychom zjistili, že programování (resp. *počítání*) se od psaní zásadně liší. V eseji *Does Writing Have a Future* (2011) se ptá, co je na psaní jedinečné a odlišuje ho od srovnatelných gest z historie či budoucnosti jako je malování nebo mačkání kláves počítače.<sup>25</sup> Existuje vůbec něco, co spojuje všechna gesta psaní — od tesání latinských písmen do mramoru až po štětce čínských ideogramů na hedvábí, škrábání rovnic na desky nebo bušení do kláves psacích strojů? A spíše řečnický se ptá: jaký by byl život bez psaní a jaký by mohl být, kdybychom se psaní vzdali? Tyhle otázky se týkají nejen psaní, ale i čtení toho, co je napsáno. Flusser se psaní věnuje i ve své knize *Shape of Things*, která vyšla o 6 let později od zmiňované eseje, v kapitole s názvem *Why Do Typewriters Go Click*.<sup>26</sup> Přijde mi podnětné, že je zásadně optimističtější než dříve publikovaná esej.

#### psaní vs počítání

Stroje a nástroje, které jsou nám dnes k dispozici a které dennodenně používáme, způsobují, že to, co bylo dříve psáno „tradičními“ způsoby, ustupuje stranou podobně jako egyptské hieroglyfy nebo indiánské uzly. Jako abeceda původně vystupovala proti piktoqramům, vystupují dnes digitální kódy proti písmenům, aby je předstihly. Stejně jako kdysi myšlení závislé na abecedě aktivně vystupovalo proti magii a mýtu, myšlení závislé na digitálních kódech dnes podle Flussera aktivně vystupuje proti procesním, „pokrokovým“ ideologiím a nahrazuje je strukturálním, systémově-analytickým, kybernetickým způsobem myšlení.

Uspořádanost je vlastnost psaní, která jej přivádí do jeho vlastní krize — v řazení je něco mechanického, a to stroj zvládá lépe než člověk. Když čelíme nějakému problému, spíše z něj vytvoříme diagram než abychom se jej snažili popsat. Naučili jsme se uvažovat numericky namísto slovně, očima namísto ušima. Stojíme nad čísly, ne pod nimi; sedíme před počítačem, kterému poroučíme. Můžeme mu nařídít, aby z něčeho zcela abstraktního udělal něco konkrétního; aby to bylo možné zažít, a tím dobrodružně rozšířit naši zkušenost.

---

<sup>25</sup> Vilém FLUSSER, *Does Writing Have a Future?* Minneapolis, MN: University of Minnesota Press 2011.

<sup>26</sup> Vilém FLUSSER, „Why Do Typewriters Go Click“, in: *Shape of Things*, London: Reaktion Books 1999. s. 62–69.

programování

Je tedy něco, co se na psaní změnilo? Podle Flussera došlo k tomu, že člověk píše spíše pro aparáty a s nimi, než pro lidi — moment, který byl pro psaní historicky stěžejní. Podstata psaní se tedy podle něj změnila; jde o psaní, které potřebuje jiné pojmenování, a to je podle něj slovo *programování*. Záměrně nepoužívá koncept „novoty“, protože o něco zcela nového nejde — jde o vyvrcholení vzorce, který se vytvořil již v době, kdy psaní teprve začínalo. Člověk už dlouho píše lidem způsobem, jako by to byly aparáty: „*na počátku, od Chammurapiho stély, se tyto instrukce nazývaly „příkázáními“; poté, s Dvanácti deskami, se z nich staly „zákony“, které se později rozvětvily do vyhlášek, nařízení a dalších forem instrukcí; během průmyslové revoluce se přidaly instrukce, které se týkaly chování lidí ke strojům, neboli „návody k použití“, až nakonec, od dob infromatických revolucí, tento vývoj završil program, o němž byla řeč dříve — instrukce pro stroje.*“<sup>27</sup>

Uvažováním nad autory historických instrukcí, ať už se jim průběžně říkalo jakkoliv, od nebeských autorů (např. Bůh) až po mýtického autora (např. lid), bylo stále jasnější, že pokyny vydávají lidé, kteří manipulují jinými lidmi. Uživatelské příručky odhalily fakt, že veškeré instrukce usilují o strojové, automatizované chování lidí. Na místo příruček nastupují programy, které už žádné návody nepotřebují. Tímto se podle Flusserra ukazuje, že cílem návodů (a západních dějin) bylo zcela profánní chování, a že po dosažení tohoto cíle je zbytečné lidi vůbec poučovat nebo s nimi manipulovat. Automaticky se chovají tak, jak se mají chovat. Tento stav podle něj směřuje k úplně depolitizaci veškerého chování a až toho bude dosaženo, budou se lidé a společnost řídit automaticky, jako kybernetický systém. Když se instrukce přesunou z člověka na aparát, může člověk dát absurdnímu chování aparátu smysl (a tím i svému vlastnímu chování jako funkci aparátu). Programovat tedy podle Flussera znamená *dávat smysl* a záměrem je *osvobodit lidské bytosti*, aby světu dávaly smysl a činily svůj život v něm svobodným. Důvod, proč podle Flussera nemůžeme programování nazývat psáním, je, že podle něj vyjadřuje jiný druh myšlení — profánní a bez hodnot. Podle něj totiž programování nelze uchopit v historických, politických či etických kategoriích, ale je potřeba na něj aplikovat jiné kybernetické, vypočitatelné a funkční kategorie.

---

<sup>27</sup> FLUSSER, *Does Writing Have a Future?*

Optimistickým aspektem scénáře, ve kterém je veškeré abecední psaní nahrazeno digitálním programováním, je potenciál, že všechna sdělení, texty, chování, znalosti a zkušenosti, které byly kdysi zprostředkovávány texty, se budou přenášet více, efektivněji a kreativněji prostřednictvím nových informačních médií.

co je na počítání vzrušující

V kapitole *Why Do Typewriters Go Click* Flusser píše o tom, že od doby, kdy jsme na svět aplikovali metodologii výpočtu, se struktura světa změnila k nepoznání. To nás podle něj může svádět k závěru, že záleží jen na nás, jak je svět strukturován. Teprve od té doby, co počítáme, máme stroje, bez nich bychom už nemohli žít, ani kdybychom chtěli. Jsme tedy nuceni spíše počítat než psát, a pokud trváme na psaní, musíme jít „klikat“. Podle všeho se zdá, že byl svět ve výchozím nastavení vystavěn pro účely počítání, ale ve skutečnosti si to svět, potažmo lidé, sami vyžádali.

Podle Flussera je pro pochopení na začátku důležité zkoumat, jaké pohyby člověk dělá při počítání na rozdíl od těch, které dělá při psaní. Psaní rukou je pohyb lineární — pro člověka žijícího na Západě čára zleva doprava. Při počítání člověk vybírá malé části z velké hromady, které skládá do úhledných hromádek — jedná se o přerušovaný pohyb. Nejprve vybírá a pak skládá; člověk analyzuje, aby mohl syntetizovat. To je podle Flussera zásadní rozdíl mezi psaním a počítáním: počítání směřuje k syntéze, psaní nikoli.

Počítání pro Flussera narozdíl od lidí, kteří v něm vidí jen otázku čísel, není chladné nebo neemotivní. Je o počítání chladných součtů do nových věcí, které nikdy předtím neexistovaly. S pomocí počítačů je nyní možné překódovat čísla do barev, tvarů a zvuků — jeho krásu a hloubku tak mohou pocítit všichni. Vidíme ji na obrazovkách počítačů, slyšíme v podobě syntetické hudby. Vzrušující na výpočtu podle něj tedy není to, že „skládá svět dohromady“, ale to, že je schopen promítat jiné, nové světy, aby je všichni mohli cítit a zažít. Tahle skutečnost pro něj znamená kulturní revoluci: člověk získal schopnost vytvářet alternativní světy vedle toho, který přijal jako daný — z poddaného jednoho světa se stává projekcí mnoha světů.



související are.na channely:

[WRITER \(research\) \[act of writing\]](#)

[WRITER \(research\) \[the net\]](#)

## kódování znaků

Čtení knihy *The Internet Unconscious* od Sandyho Baldwina na mě působilo trochu jako čtení beletrie a částečně soukromého chatu. To zafungovalo jako *softer* úvod do techničtějších aspektů problematiky digitálního psaní, a v určitém smyslu mě vrátilo na pomyslný začátek — jak jsou písmena, znaky a symboly reprezentovány v *kódech*, kdo je spravuje a za jakých okolností, a jestli je z toho případně cesta ven ((není)). Kódování znaků není charakteristické pouze pro počítače — uživatel Zwork101 ve svém textu s názvem *Chaos, kterým je kódování znaků* (The Chaos that is Character Encodings), přibližuje problematiku na morseově kódu:

*„Představte si, že potřebujete komunikovat s někým, kdo je od vás vzdálen 200 metrů. Problém je, že ke komunikaci máte pouze baterku. Jak to uděláte? Pokud znáte morseovku, můžete ji použít, nebo si nějak předem pošlete instrukce a vysvětlíte, co znamená každé bliknutí. Ať tak či onak, používáte kódování znaků. Kódování*

*znaků nepoužívají jen baterky, ale fungují jako páteř počítačů, které vám zobrazují informace. Místo baterky mají počítače výboje elektrické energie, které přeměňují na písmena, číslice a emoji. (...) Přesto, stejně jako binární systém používaný v moderních počítačích, je [morseovka] založena na kombinacích dvou možných hodnot — v případě Morseovy abecedy na teče nebo pomlčce.“<sup>28</sup>*

## jedničky a nuly

Počítače nedokáží uchovávat nic jiného než bity. To znamená, že každý znak, obraz nebo zvuk je ve svém jádru proud hodnot ano/ne nebo pravda/nepravda, které se pro snadnější pochopení nejčastěji zobrazují pomocí jedniček a nul. Bit je ve skutečnosti střípek elektriny, který buď *je* a nebo *není*.

Dokud nebudeme umět číst binární kód, budeme potřebovat znaky, které bude vždy potřeba převádět do bitů. To znamená, že je musíme nějakým způsobem kódovat — použít něco k reprezentaci něčeho jiného podle souboru pravidel. Kódování znaků, reprezentace písmen a symbolů v počítači, je tedy problém, který nikdy nezmizí. Text je pro počítač abstraktní entitou, jeho význam je v oku čtenáře — abychom mohli přečíst jakýkoliv soubor, je potřeba znát správné kódování.

Vyhledávací tabulka (*code page* nebo kódová stránka) je přímé mapování posloupnosti bitů (kódové jednotky) v poměru 1:1 ke konkrétnímu znaku (části textu se sémantickým významem) — počítač vidí posloupnost bitů, vyhledá ji a zobrazí uživateli příslušný znak. Pokud je použita špatná vyhledávací tabulka, je použit špatný znak. Mechanika uložení textu je pro většinu uživatelů pouze technickým detailem, ale v ekonomice, která je stále více digitální, globální a mnohojazyčná, už od sebe nemůžeme striktně oddělovat význam od uložení — informace, které chceme reprezentovat, úzce souvisí s tím, jak jsou uloženy.

## ASCII

V úvodu jsem psala, že cílem nástroje `writer.axdx` je „*vrátit pozornost zpět k samotnému psaní, ve formátu, který není omezený proprietárním softwarem a nerozsype se při otevření v jiném editoru*“. To mě dovedlo k přemýšlení o (pokud

---

<sup>28</sup> @Zwork101, *The Chaos that is Character Encodings*, 2020, <https://medium.com/@zwork101/the-chaos-that-is-character-encodings-d79111a45e1d> (cit. 15. 4. 2023).

vím) nejprostším textovém formátu, kterým je .txt (text file) ve spojení s plain textem. Plain text dřív obvykle znamenal text kódovaný v ASCII.<sup>29</sup> V předmluvě k původní normě z roku 1963 se uvádí, že ASCII je „sada znaků, která se má používat pro výměnu informací mezi systémy zpracování informací, komunikačními systémy a souvisejícími zařízeními“.<sup>30</sup> ASCII jsem se moc dlouho nezabývala, protože má prostor pro kódování pouhých 128 znaků. Jen 95 z nich jsou tisknutelné znaky jako písmena, číslice, interpunkční znaménka atd., ostatní jsou řídicí (netisknutelné) kódy. Samostatně ASCII řeší jen základní znaky pro angličtinu, ani tu ale nepokrývá kompletně — jelikož neobsahuje znaky s diakritikou, nezvládne kódovat a přečíst ani „Beyoncé“. ASCII byl nejběžnějším kódováním znaků na internetu až do prosince 2007, kdy jej překonalo kódování UTF-8 od Unicode, které je s ASCII zpětně kompatibilní. Z přečtených článků mi začalo být patrné, že „čistý“ plain text neexistuje, protože z podstaty fungování počítačů musí být kódován stejně jako všechno ostatní.



<sup>29</sup> ASCII je zkratka pro American Standard Code for Information Interchange.

<sup>30</sup> *ASCII: American Standard Code for Information Interchange*, 1963, <https://www.sr-ix.com/Archive/CharCodeHist/X3.4-1963/index.html> (cit. 15. 4. 2023).

## Unicode

Unicode, oficiálně *The Unicode Standard*, je standard informačních technologií pro konzistentní kódování, reprezentaci a zpracování textu pro většinu světových písemných systémů.<sup>31</sup> Ve světě technologií je považován za základní nezbytnost pro práci s různými znakovými sadami. O jeho silných a slabých stránkách téměř nikdo nemluví, *prostě to tak je a funguje*. Ale jako každé technické řešení má Unicode své výhody i nevýhody, a dokonce i konkurenci.

Je důležité si uvědomit, že Unicode odděluje mapování znaků od jejich kódování. Co se týče samotného **kódování**, Unicode má řadu způsobů, jak tyto kódové body převést na skutečné bity a bajty v počítači. Tyto způsoby se nazývají Unicode Transformation Formats (UTF) a navzájem se od sebe liší počtem bitů — v závislosti na použití UTF dochází k šetření nebo plýtvání místem. V dnešní době je standardním kódováním UTF-8, které dokáže zakódovat prakticky jakýkoli znak, je zpětně kompatibilní se základním kódováním ASCII, a přesto je pro většinu případů použití relativně prostorově úsporné.

Unicode **mapuje** prakticky každé písmeno nebo symbol v každém jazyce a přiřazuje mu jedinečný kódový bod (code point), z nichž každý je popsán znakem U+ následovaným čtyřmi znaky v šestnáctkové soustavě. Velké latinské písmeno „A“ je reprezentováno zápisem U+0041. Jedná se o abstraktnímu pojem „A“, nikoli ten, který je vidět na obrazovce vykreslený jako písmo. Beckerova norma Unicode jasně uvádí, že „zásadně a jasně rozlišuje mezi znaky, které jsou abstraktními entitami nesoucími textový obsah, a glyfy, které jsou viditelnými grafickými formami“.<sup>32</sup> Sandy Baldwin popisuje glyf jako technicky vytvořený tvar, který může mít znak při jeho vykreslení nebo zobrazení; je to konkrétní, zobrazující reprezentace jednoho nebo více znaků. Glyf a znak jsou podle něj součástí jedné technologie psaní; z jedné strany abstraktní, zakódované a konceptuální, z druhé strany hmotné, vnímané a neurčité.<sup>33</sup>

---

<sup>31</sup> Wikipedia contributors, „Unicode“, *Wikipedia, The Free Encyclopedia*, <https://en.wikipedia.org/w/index.php?title=Unicode&oldid=1150096766> (cit. 15. 4. 2023).

<sup>32</sup> Joseph D. BECKER, „Unicode 88“, *Unicode*, <https://unicode.org/history/unicode88.pdf> (cit. 15. 4. 2023).

<sup>33</sup> BALDWIN, *The Internet Unconscious*, s. 140.





Vyhledávací tabulka Unicode je rozdělena do několika rovin:<sup>34</sup>

- Rovina 0: Základní vícejazyčný plán, BMP – časté znaky.
- Rovina 1: Doplnková vícejazyčná rovina, SMP – málo časté, neideografické znaky. Historická nebo neobvyklá písma, symboly, emoji
- Rovina 2: Doplnková ideografická rovina, SIP – málo časté ideografické znaky.
- Rovina 3: Terciární ideografická rovina, TIP – navrženo pro Unicode 13.0 (dosud nezveřejněno). Bude zahrnovat ideografické znaky CJK, které nejsou zahrnuty v BMP nebo SIP, včetně mnoha historických znaků.
- Roviny 4–13: V současné době nepřirazené, vyhrazené pro budoucí použití.
- Rovina 14: Doplnková rovina pro zvláštní účely (SSP).
- Roviny 15, 16: soukromé použití.

Tím, že se Unicode snaží mapovat každé písmeno a symbol v každém jazyce, dochází k mnoha zkreslením. Existuje mnoho jazyků, které koncepčně sdílejí soubor

---

<sup>34</sup> Randy AU, „Character Encodings — The Pain That Won't Go Away, Part 2/3: Unicode“, *Better Programming*, 2019, <https://betterprogramming.pub/character-encodings-the-pain-that-wont-go-away-part-2-2-unicode-4f13a0d2d42d> (cit. 15. 4. 2023).

ideografických znaků, ale mohou mít neshody v konkrétních detailech. Nejvýraznější případ toho typu je *han unification*<sup>35</sup> — snaha mapovat více znakových sad znaků tzv. CJK jazyků (čínština *hanzi*, japonština *kanji*, korejština *hanja* a vietnamština *chữ Hán*) do jediné sady unifikovaných znaků. To, co může být pro jeden jazyk starou formou, může být v současnosti používáno v jiném jazyce, nebo může mít znak v různých médiích nebo historických obdobích mírně odlišné zobrazení, přičemž si zachovává stejný význam. Co může být pro jeden jazyk nepodstatným rozdílem v typografii, může být pro jiný jazyk významnějším, a díky dlouhé geopolitické historii značně politickým, rozdílem. Unicode má konečný počet míst pro znaky, které může obsadit, proto má tendenci zjednodušovat a šetřit místem pro budoucí znaky — vyhrávají tak ty nejčastěji používané, ostatní jsou vynechány. Zapomíná se na fakt, že pokud je některý ze zřídka používaných znaků součástí něčího jména nebo názvu místa bydliště, pak se o často používaný znak jedná.

Spolu s Unicode vznikl na základě ASCII ještě TRON code. Bojovali spolu několik let, až se nakonec jeden z nich zhroutil a druhý stal univerzálním vítězem. Moment, ve kterém se zásadně liší je, že TRON podporuje jedinečné znaky každého jazyka. Steven J. Searle ve svém textu Unicode Revisited tvrdí, že Unicode byl zaměřen na sjednocení trhů, a proto jeho tvůrci tak dlouho zapomínali na skutečnost, že současně sjednocují prvky čínské, japonské a korejské kultury.<sup>36</sup> Právě skutečnost, že Unicode možná „slučuje kultury“, narazila na největší odpor. Důvodem, proč je TRON code podle něj lepší je, že byl od počátku navržen jako systém kódování znaků, který mohou používat všichni lidé na světě. Od počátku například obsahoval znakové kódové body pro znaky Braillova písma s důrazem na to, že i lidé s jakýmkoliv handicapem jsou součástí světové společnosti, která má při navrhování počítačových systémů svá práva.<sup>37</sup> Zásadním rozdílem bylo také to, že TRON code nikdy nezasahoval do businessu tvorby znakových sad a písem, narozdíl od Unicode konsorcia.

---

<sup>35</sup> Wikipedia contributors, „Han unification“, *Wikipedia, The Free Encyclopedia*, [https://en.wikipedia.org/w/index.php?title=Han\\_unification&oldid=1141363421](https://en.wikipedia.org/w/index.php?title=Han_unification&oldid=1141363421) (cit. 15. 4. 2023).

<sup>36</sup> Steven J. SEARLE, „Unicode Revisited“, *TRON Web*, <http://tronweb.super-nova.co.jp/unicoderevisited.html> (cit. 15. 4. 2023).

<sup>37</sup> Ibid.

## Unicode Consortium

Unicode Consortium je nezisková organizace, jejíž hlavním cílem je udržovat a zveřejňovat Unicode Standard. Za posledních 25 let zmapovalo a zakódovalo více než 120 000 znaků ze 129 písem, včetně dávno mrtvých jazyků, jako jsou chetitské hieroglyfy.<sup>38</sup> Zajistili přežití těchto jazyků v digitální budoucnosti, a taky digitálně archivovali celou historii písma. Nyní, když jsou všechna hlavní psaná písma zakódována, vybírá konsorcium Unicode z návrhů předložených zúčastněnými stranami nové projekty k řešení. V případě emojijs byly těmito zájemci velké softwarové společnosti včetně Applu, Googlu a Microsoftu, které chtěly prodávat více telefonů v Japonsku. Japonští telefonní operátoři už emoji nabízeli dlouho, ty se postupně staly velmi populárními, ačkoli napříč operátory nefungovaly, protože byly všechny unikátně kódované – potřebovali Unicode. A jako platící a hlasující členové konsorcia ho dostali. Členství v konsorciu není bezplatné – plné členství a hlasovací práva stojí 50 000 amerických dolarů.<sup>39</sup> Aditya Mukerjee ve svém článku *I Can Text You A Pile of Poo, But I Can't Write My Name*, navrhuje 2 kroky, které by Unicode konsorcium mohlo zavést:<sup>40</sup>

1. explicitní plán nábory zaměstnanců, který by zaručil, že jeho zaměstnanci budou reprezentovat mnoho jazyků, které se snaží standardizovat
2. přijmout stanovy, které zajistí, že mezi členy technických výborů a funkcionářů nebudou převažovat rodilí mluvčí angličtiny

Uživatelka Eevee na svém blogu shrnuje problematiku ohledně Unicode tak, že katastrofou je spíše lidský jazyk než Unicode a cokoli, co si klade vznešené cíle reprezentovat jej celý, bude mít nějaké vrásky.<sup>41</sup> Tahle myšlenková cesta vedla do slepé uličky, Unicode a jeho UTFs jsou hluboce propsány do fungování dnešních počítačů, a z hlediska kompatibility je to v současnosti víceméně jediné řešení,

---

<sup>38</sup> Wikipedia contributors, „Unicode Consortium“, *Wikipedia, The Free Encyclopedia*, [https://en.wikipedia.org/w/index.php?title=Unicode\\_Consortium&oldid=1137002727](https://en.wikipedia.org/w/index.php?title=Unicode_Consortium&oldid=1137002727) (cit. 15. 4. 2023).

<sup>39</sup> <https://home.unicode.org/membership/membership-levels/>

<sup>40</sup> Aditya MUKERJEE, „I Can Text You A Pile of Poo, But I Can't Write My Name“, *Model View Culture*, 2015, <https://modelviewculture.com/pieces/i-can-text-you-a-pile-of-poo-but-i-cant-write-my-name> (cit. 15. 4. 2023).

<sup>41</sup> @Eevee, Dark corners of Unicode, 2015, <https://eev.ee/blog/2015/09/12/dark-corners-of-unicode/> (cit. 15. 4. 2023).

zvláště pokud jde o internet. Pokud to správně chápu, hypotetická „cesta“ by mohla být vytvořit si vlastní hardware a na něm stavět vlastní software, což by znamenalo možnost vytvořit si vlastní systém na kódování znaků bez přítomnosti vazby na jiný kódovací systém jako je Unicode (?), který by byl ale kompatibilní vždycky jen v této úzké skupině *našeho* hardware a neumožňoval fungovat nezávisleji jen v rámci software (?). Myslím, že je důležité si na individuální i kolektivní rovině popsat primární problémy, kterými se chceme zabývat a které chceme řešit, a na ty se co nejméně soustředit. Raymond to ve své knize *The Art of Unix Programming* shrnuje takto:

*„Chcete-li správně naplňovat filozofii Unixu, musíte si vážit vlastního času natolik, abyste jím nikdy neplýtvali. Pokud už někdo problém jednou vyřešil, nenechte se pýchou nebo politikou vtáhnout do řešení podruhé, místo abyste ho znovu použili. A nikdy nepracujte víc, než musíte; pracujte raději chytřeji a šetřete si úsilí navíc na chvíle, kdy ho budete potřebovat. Opřete se o své nástroje a automatizujte vše, co můžete.“<sup>42</sup>*

související are.na channely:

[WRITER \(research\) \[character encoding\]](#)

[WRITER \(research\) \[formatting\]](#)

[WRITER \(research\) \[text rendering\]](#)

## papír a dokument

Po otevření aplikace Microsoft Word se automaticky otevře ohraničený bílý prostor odpovídající tradičnímu kancelářskému papíru A4, s okraji, které víceméně odpovídají tisknutelné ploše. Jinak tomu není ani v Google Docs, nástroji, který se oproti MS Word nachází primárně v online prostředí. Nástroj, ve kterém píšu tento text, se jmenuje Dropbox *Paper*. Prostor určený pro psaní tady nemá tak jasné viditelné hranice jako u dvou předchozích příkladů, princip ale zůstává stejný — můžu psát jen tam, kde se to v současné době očekává. Konečným stavem tak

---

<sup>42</sup> RAYMOND, *The Art of Unix Programming*

může být vždy jen lineárně chronologicky rozvíjející se text (vynechávám postupy, které by vyžadovaly radikální ohnutí současných rigidních funkcí, jejichž nastavení by v konečném důsledku zabralo víc času a úsilí než samotné psaní). Jediný další prostor, který umožňuje tuto chronologii částečně rozbít a ke kterému je většina dnešních textových nástrojů uzpůsobená, je místo pro poznámky pod čarou. Podle Teda Nelsona, jehož myšlenky mi přijdou nesmírně inspirativní pro všechny nadcházející části tohoto textu, je poznámka pod čarou jen myšlenka snažící se uniknout jiným směrem.<sup>43</sup> *Kam by se mohla snažit uniknout?*

Když vařím podle receptu v papírové podobě, můžu na něj kdekoliv připsat poznámku o tom, že mám příště použít míň cukru, písemně přepočítat, jaké množství ingrediencí potřebuju na dvojitou dávku těsta nebo si na stejný papír sepsat nákupní seznam. Současné nástroje pro práci s textem imitují papír, ale jeho skutečně přínosné funkce ignorují. Podle autorů textu *Potluck: Dynamické dokumenty jako osobní software* (Potluck: Dynamic documents as personal software), kterému se budu chvíli věnovat, je papír ze své podstaty liberální médium, a proto není třeba dodržovat předem daná pravidla.<sup>44</sup>

Co se týče uchovávání a další práce s dokumenty, které umožňují současné technologie, podle hackerského žargonu je možné definovat papírovou verzi online dokumentu jako verzi „mrtvého stromu“,<sup>45</sup> kdy se „mrtvé stromy“ vztahují k papíru. Přesná věta z Jargon File zní „Mrtvé stromy nelze grepovat“, od unixového příkazu grep, který znamená prohledávání obsahu textových souborů. To znamená, že výhodou uchovávání dokumentů v digitální podobě namísto na papíře je, že je lze snadněji vyhledávat podle konkrétního obsahu. Související pojem „tree-killer“ může odkazovat buď na tiskárnu, nebo na osobu, která plýtvá papírem (například na lidi z marketingu, kteří vytvářejí velké množství dokumentace „bez obsahu“). Publikování „mrtvého stromu“ označuje tištěnou papírovou verzi písemného díla, na rozdíl od digitálních alternativ, jako je webová stránka. Podle Teda Nelsona nemusí být dokument nutně simulací papíru. Dokument v nejobecnějším slova

---

<sup>43</sup> Theodor Holm NELSON, Charles BROSKOSKI „Parallel documents! Interview, August 2018“, video, 42:31, *Youtube*, nahráno uživatelem TheTedNelson 19. prosince 2018, <https://youtu.be/PPBeHDxcVSA> (cit. 15. 4. 2023).

<sup>44</sup> Geoffrey LITT, Max SCHOENING, Paul SHEN, Paul SONNENTAG, „Potluck: Dynamic documents as personal software“, *Ink & Switch*, 2022, <https://www.inkandswitch.com/potluck/> (cit. 15. 4. 2023).

<sup>45</sup> [https://en.wiktionary.org/wiki/dead\\_tree\\_edition](https://en.wiktionary.org/wiki/dead_tree_edition).

smyslu definuje jako balíček myšlenek vytvořený lidskou myslí a adresovaný lidské mysli, určený k dalšímu rozvoji *těchto* myšlenek a *této* mysli. Podle něj se lidské myšlenky projevují jako text, propojení (connections) nebo diagramy, a proto je podle něj zásadní otázkou, jak je uchovávat a reprezentovat.

## Potluck

Autoři textu a stejnojmenného nástroje *Potluck* rozvíjí formát dokumentu směrem k menší rigidnosti. Uživatelům chtějí atraktivním přístupem umožnit postupně obohacovat textové dokumenty, v rámci kterých by mohli podle potřeby přidávat strukturu a výpočetní principy. Mimo jiné navazují na tzv. *detektory dat*, které popsali výzkumní pracovníci společnosti Apple Bonnie Nardi, James Miller a David Wright v článku *Collaborative, Programmable Intelligent Agents* z roku 1998 jako „*inteligentní rozpoznávače vzorců (patterns) zabudované do operačního systému, které dokáží detekovat strukturovaná data, jako jsou telefonní čísla a adresy ulic, obsažená v každodenních nestrukturovaných dokumentech, a poté umožňují uživateli provádět akce na základě těchto strukturovaných dat*“.<sup>46</sup> Potluck uživatelům umožňuje definovat vzorce, které jsou rozpoznávány v textovém dokumentu. Tyto vzorce se vytvářejí pomocí interakce s vyhledáváním — uživatelé jsou již obeznámeni s vyhledáváním obsahu v textovém procesoru nebo webovém prohlížeči, takže se podle autorů jedná o přirozený náběh na vytváření detektorů „živých“ dat. Z „obyčejného“ dokumentu tak lze vytvořit vlastní software „na míru“, ať už se jedná o osobní kuchařku, plánovač nebo upomínkovač na zalévání rostlin. Textové dokumenty jsou podle autorů nástroje jediným univerzálním médiem pro záznam všech druhů informací. Potluck je v rané fázi vývoje a fungování v něm tak zvládnout jen ti, kteří mají nějaké zkušenosti s programováním.

---

<sup>46</sup> Bonnie NARDI, James MILLER, David WRIGHT, „Collaborative, Programmable Intelligent Agents“, *Communications of the ACM*, vol. 41, 1998, č. 3, s 96–104.



### Ted Nelson a projekt Xanadu

Ted Nelson je průkopník informačních technologií, filozof, sociolog a zároveň autor projektu Xanadu, který je označován jako nejdéle běžící vaporware<sup>47</sup> projekt v historii výpočetní techniky. V textu *Back to the Future: Hypertext the Way It Used To Be* společně s Robertem Adamson Smithem představuje svůj projekt Xanadu v abstraktu: „Ostatní napodobují papír (Word, Acrobat) a konstantní 3D svět, ve kterém žijeme („virtuální realita“). Náš systém se místo toho snaží vytvářet dokumenty lepší než papír v prostoru lepším než realita.“<sup>48</sup>

Společně s Douglasem Englebartem je Ted Nelson považován za otce počítačového hypertextu (Ted by asi byl rád, kdybych tady zmínila, že pojem „hypertext“ vymyslel a použil jako první).<sup>49</sup> Hypertext je schopnost propojovat fragmenty textu pomocí počítače, což čtenáři umožňuje sledovat odkaz z jedné části textu do jiné. Reid Scoggin popisuje, jak poznámka pod čarou v knize dříve vyžadovala návštěvu knihovny, vyhledání odkazované knihy a nalezení citované stránky, která dost možná obsahuje další poznámky pod čarou. Hypertext podle něj umožňuje rychlé a snadné procházení sítě (např. internetu) pouhým kliknutím myši,

<sup>47</sup> Vaporware je v počítačovém světě produkt, obvykle hardware nebo software, který je ohlášen široké veřejnosti, ale je opožděn nebo není reálně vyroben či oficiálně zrušen.

<sup>48</sup> Theodor Holm NELSON, Robert Adamson SMITH, „Back to the Future: Hypertext the Way it Used to Be“, *Xanadu*, 2011, <https://xanadu.com/XanaduSpace/btf.htm> (cit. 15. 4. 2023).

<sup>49</sup> Theodor Holm NELSON, „Complex Information Processing: A File Structure for the Complex, the Changing, and the Indeterminate“ *Association for Computing Machinery '65: Proceedings of the 1965 20th national conference*, 1965, 84–100.

což zpřístupňuje informace těm, kteří nemají čas a prostředky na hledání fyzických artefaktů.<sup>50</sup>

V textu Nelson tvrdí, že se papír stal metaforickým vězením pro digitální systémy. Navrhováním digitálních dokumentů tak, aby napodobovaly papír, se implementovalo mnoho jeho fyzických omezení: místo věty nebo odstavce považujeme stránku za nejmenší jednotku informace, na kterou lze navázat; zůstali jsme svázáni s konvencemi tradiční sazby, které zavedly tiskárny; trváme na jediném módu zobrazení – maketě fyzické stránky – místo abychom přijali flexibilní, vícenásobné pohledy, které jsou možné v dynamickém výpočetním médiu. Nejlepší vlastnosti papíru jsme naopak opustili: možnost přidávat ručně psané poznámky na okraje, možnost rozložit si před sebe více listů a přehazovat je, abychom našli nejlepší strukturu a pořadí, a možnost přenést list papíru přes místnost a prohlédnout si ho s někým jiným.

Účelem hypertextu bylo podle Nelsona nahradit nedostatky papíru. Na papíře nelze snadno zobrazit souvislosti, má velmi omezený prostor a nutí k nepružnému uspořádání. Hypertext potenciálně nabízí mnoho forem propojení a prezentace, které papír neumožňuje. Díky nastavené tradici nástrojů pro úpravu textu byly pod kontrolou mechanismy, nikoliv uživatelé. Jsou to hlavně:

- simulace papíru
- imitace fyzického prostoru
- „surové“ (jednosměrné) odkazy
- vložené a neoddělitelné hierarchie
- zjednodušené mapování do souborů

Nelson popisuje, že napodobení papíru se netýká pouze nástrojů pro práci s textem, ale samotného internetu, který na počátku vznikl na základě ideje hypertextu. Oproti Nelsonově pojetí však namísto oboustranně propojených dohledatelných odkazů zůstaly pouze jednosměrné, které nás odvádějí *někam pryč*. Podle Nelsona je důležité si uvědomit, že tyto systémy „imitují papír pod sklem“, protože do něj nelze vkládat poznámky. Připomíná princip WYSIWYG (What You

---

<sup>50</sup> Reid SCOGGIN, „From A Topiary: Hypertext and Urbit: A brief history of hypertext and Urbit networking“, *Urbit*, 2021, <https://urbit.org/blog/a-topiary> (cit. 15. 4. 2023).



See Is What You Get), projevující se jako to, co dostanete, pokud dokument vytisknete. Tento podle něj propagandistický výraz legitimizuje simulaci papíru jako správné nastavení počítačových dokumentů a úplně opouští důležitou příležitost: jak můžeme médium papíru vylepšit? Princip WYSIWYG je podle autorů „fetišizací“ tiskového layoutu (svislých pruhů strojopisu) jako jediného možného pohledu na dokument, čímž dochází k penalizaci uživatele.

Autoři projektu Xanadu nevytváří jen jiný druh hypertextu, zároveň hledají nejobecnější formu dokumentu a způsob, jak všechny možné dokumenty reprezentovat. Podle nich to znamená zbavit se všech výše zmíněných mechanismů. Struktura dokumentů se od papírových liší zásadním způsobem – dokument může být rozdělen na libovolné jednotky („vunits“ neboli zobrazné jednotky), kterých může být libovolné množství.

Text v této nové struktuře může být uspořádán a zobrazen v módech, jako jsou např.

- obdélníkové stránky
- překrývání
- létající odstavce
- fontány textu
- vlnící se plochy
- animované a měnící se texty

Všechny obsahy jsou propojitelné, průchozí (ukazují svůj původ) a mohou být propojeny vedle sebe (jako sloupce v knize, tabulce nebo tabulkovém procesoru). Je možný libovolný počet odkazů, strukturálních vztahů, módů a interaktivních metod, které si uživatel může vybrat. Nejúplnější zobecnění dokumentů nazývají autoři *transliterální strukturou*. Tato struktura je schopna reprezentovat všechny možné struktury dokumentů a vypořádat se s proměnlivostí změn, verzováním i autorskými právy. *Dokumenty* autoři charakterizují jako balíčky obsahu sestavené z nového i starého textu, zvuku a videa, v libovolném uspořádání, částech a požadovaném vzhledu. *Literaturou* rozumí sbírky dokumentů se všemi jejich možnými souvislostmi.

Struktura souboru je narozdíl od obsahu i odkazů v jediném souboru jako je to u modelu .html rozdělená na:

- model dokumentu
- získávání obsahu (zdrojování)
- jednoduchá struktura (paralelní pásy obsahu)
- překryvná struktura (přeskupení, typografie)
- vnější struktura (odkazy)
- historie (předchozí verze)

Ted Nelson archivuje většinu svých myšlenek na [svém Youtube kanále](#), kde do detailů vysvětluje principy a funkce jeho projektů. Ve videu [Here I Stand, at Age 80](#) (2017) shrnuje, co chce, aby se vědělo o jeho ideálech a práci v průběhu let s jeho vlastním podtitulem: *Jak velmi špatný kariérní plán vedl k objevům, nadějím a dobrodružstvím.*<sup>51</sup>

*„EVERYTHING IS DEEPLY INTERTWINGLED. V podstatě neexistují žádné „subjekty“, existuje pouze všechno vědění, protože vzájemné vazby mezi nesčetnými tématy tohoto světa prostě nelze přehledně rozdělit. (...) Hierarchické a sekvenční struktury, oblíbené zejména od dob Gutenberga, jsou obvykle vynucené a umělé. Vzájemná provázanost není obecně uznávána – lidé stále předstírají, že mohou věci hierarchizovat, kategorizovat a sekvencovat, i když to nejde.“*<sup>52</sup>

— *Computer Lib/Dream Machines, Ted Nelson*

---

<sup>51</sup> Theodor Holm NELSON, „Here I Stand, at Age 80“, video, 47:56, *Youtube*, nahráno uživatelem TheTedNelson 4. září 2017, <https://youtu.be/mmfjM-SGIGs> (cit. 15. 4. 2023).

<sup>52</sup> Theodor Holm NELSON, *Computer Lib/Dream Machines*, Redmond, Washington: Microsoft Press 1987.



související are.na channely:

[WRITER \(research\) \[document, paper\]](#)

[WRITER \(research\) \[philosophy, theory, principles\]](#)

# závěr

Proč teda vyvíjet vlastní textový editor? Doufám, že motivace zabývat se problematikou digitálního psaní jsou po přečtení textu pochopitelnější. Pokusím se je teď konkrétněji napojit na některé z AGDX (a [community-software.agdx](https://community-software.agdx)) hodnot a hodně volně aplikovat na fiktivní use-cases.

Jednou jsem četla tweet, který teď nemůžu dohledat, který popisoval, že dnešní software a s ním spojené platformy jsou navrhovány jako domy a vybízeli k otázce, proč raději nevyvíjet něco, co by se namísto domu blížilo kuchyni nebo skleníku. To, jak a kde bydlíme a čím se obklopujeme, do velké míry ovlivňuje a formuje náš život a jinak tomu není ani s digitálními (a) online ekvivalenty. Obrovský prostor, ve kterém jsme vnímáni jako produkt generující data, jeden z obrovské spousty; navštěvujeme domy, ve kterých nejsme vítáni jako individuality, ale jako jeho syntetizovaní a optimalizovaní uživatelé.

Tyto domy nejsou k bydlení, ale slouží pouze k návštěvám. Navzájem spolu nekomunikují, sami si musíme pamatovat, co v jakém domě najdeme — kam jsme poslali zprávu, uložili heslo, kam jsme napsali důležitou poznámku. Cestování mezi nimi je otravné a nepřehledné, s téměř neexistující infrastrukturou nemůžeme jako pouzí uživatelé téměř nic dělat — nelze ji rozšiřovat ani na ní smysluplně stavět. *„Tato roztržitá, oddělená a přísně sledovaná zkušenost je jednoduše vedlejším produktem toho, že váš software provozují a ovládají jiní lidé. Pro běžného uživatele je výsledek omezující, nudný a bolestivý. Možnost každodenní kreativity s nástroji, které máme k dispozici, se blíží nule. Neměl být počítač jízdním kolem pro mysl?“* stojí na webu Urbitu, operačního systému a peer-to-peer síti.<sup>53</sup> Urbit je založený na jednoduchosti a postavený tak, aby vydržel navždy, 100% vlastněn svými uživateli. Podle lidí, kteří za ním stojí, už internet nelze zachránit — velké korporace budou vždy kontrolovat naše aplikace a služby, protože už je nedokážeme spravovat a provozovat sami. Důležitou součástí je čistý softwarový stack, který je dostatečně kompaktní, aby mu vývojář mohl plně porozumět a ovládat jej.

---

<sup>53</sup> Urbit: Overview, <https://urbit.org/overview/interface> (cit. 15. 4. 2023).

Pojďme zkusit vyvíjet něco, co se bude v principech podobat ekologické zahradě, ne velkému zemědělskému podniku.

## co je ekosystém

Podle Wikipedie se ekosystém skládá „ze všech organismů a fyzického prostředí, s nímž se vzájemně ovlivňují. Živé a neživé části jsou vzájemně propojeny koloběhem živin a toky energie. (...) Ekosystémy jsou dynamické útvary — podléhají periodickým narušením a vždy se z nějakého minulého narušení zotavují. Tendence ekosystému setrvat navzdory narušení v rovnovážném stavu se označuje jako jeho odolnost. Schopnost systému absorbovat narušení a reorganizovat se při změně tak, aby si zachoval v podstatě stejnou funkci, strukturu, identitu a zpětné vazby, se označuje jako jeho ekologická odolnost.“<sup>54</sup>

Wikipedie odděluje odvozený pojem digitální ekosystém, který představuje „distribuovaný, adaptivní, otevřený sociotechnický systém s vlastnostmi samoorganizace, škálovatelnosti a udržitelnosti inspirovanými přírodními ekosystémy.“<sup>55</sup>

„ecosystem.dps is an outer skin of the whole DPS project. It binds and structures all of the current and future parts, such as systems.dps together. It's a top-level definition of everything inside DPS.“<sup>56</sup>

V abstraktu této práce jsem **community-software.agdx** popsala jako ekosystém nástrojů a metod sloužící komunitám. Zpětně si myslím, že to není úplně přesné a stejně jako komunitě může sloužit jednotlivci.

---

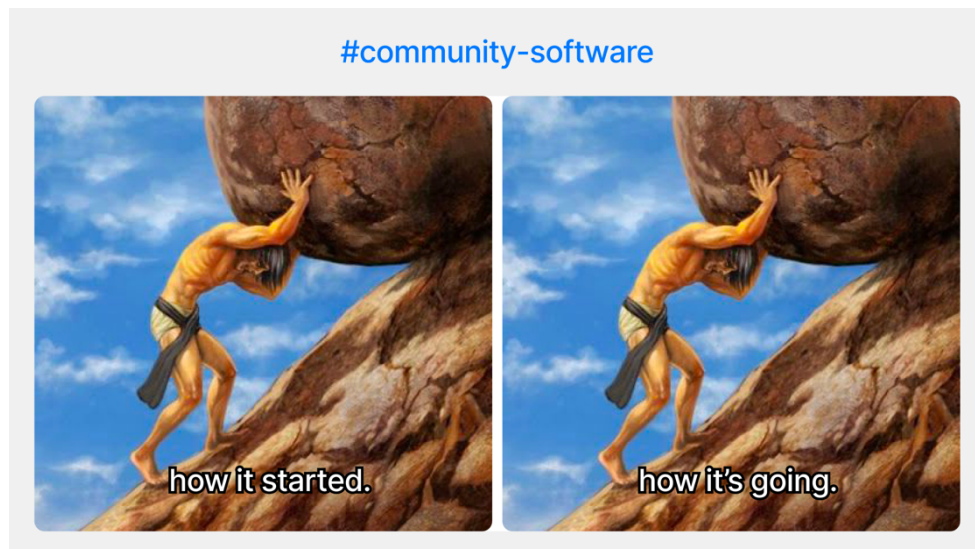
<sup>54</sup> Wikipedia contributors, „Ecosystem“, *Wikipedia, The Free Encyclopedia*, <https://en.wikipedia.org/w/index.php?title=Ecosystem&oldid=1148102981> (cit. 15. 4. 2023).

<sup>55</sup> Wikipedia contributors, „Digital ecosystem“, *Wikipedia, The Free Encyclopedia*, [https://en.wikipedia.org/w/index.php?title=Digital\\_ecosystem&oldid=1143737790](https://en.wikipedia.org/w/index.php?title=Digital_ecosystem&oldid=1143737790) (cit. 15. 4. 2023).

<sup>56</sup> DPS members, „ecosystem.dps“, *wiki.dps*, <https://wiki.public.services/systems/ecosystem/> (cit. 15. 4. 2023).

**ecosystem.axdx** = **members.axdx** (humans being present) + **values.axdx** (foundations, principles and beliefs shared between those present humans) + **community-software.axdx** (software infrastructure build by those present humans based on their shared values) + **kitchen.axdx** + **school.axdx** + ...)

co je community-software.axdx



(duben 2023)

***community-software.axdx** is an independent, collectively created, owned and maintained software infrastructure that is part of **ecosystem.axdx**. its foundation is **core-technologies.axdx** (tools, primitives, libraries, etc.), which serve as building-blocks for **products.axdx** and **services.axdx**. **community-software.axdx** is funded by **membership.axdx** and the profits from **products.axdx** and **services.axdx**, it is also linked to **economic-system.axdx**. **community-software.axdx** is based on the common **values.axdx** of each **members.axdx** and is open to all new soft-mates.*

V současnosti explicitně definované části **ekosystem.axdx** (zjednodušeno):

core-technologies.axdx	products.axdx	members.axdx	popis
builder.axdx	collaborate.online	@barbora @honza_suchy	
composer.axdx	collaborate.online	@barbora @honza_suchy	nástroj pro vytváření layoutů a finálních výstupů
database.axdx	aaarchive	@metju @namelessnobody	general-purpose interface pro strukturování, správu, archivaci a interakci s daty
design-system.axdx	—	—	jednotný design systém, především jeho principy, které mohou být využity napříč community-software.agdx
editor.axdx	—	—	textový processor / prostor pro úpravu textu
reader.axdx	—	—	prostor pro čtení
writer.axdx	—	@katerina	textový editor / prostor pro psaní

*the **short-term goal** is to ensure economic stability for the core-team and thus sustainability of the development and management of the entire infrastructure*

*the **medium-term goal** is to open up core-technologies.axdx to the entire ecosystem.axdx and its members*

*the **long-term goal** is to open up core-technologies.axdx as commons.axdx and allow everyone to create their own software infrastructure*

související are.na skupina:

[AGDX Community Software](#)

související are.na channely:

[COMMUNITY SOFTWARE \(vibe\)](#)

[COMMUNITY HARDWARE \(research\)](#)

[IF YOU CARE YOU CARE \(research\)](#)

## co je writer.axdx

**writer.axdx** je prostor pro psaní textu. Jeho důležitou vlastností je oddělení činnosti psaní od činnosti formátování, a namísto toho umožňuje radikálně customizovat samotné prostředí. Každý uživatel je jiný a má své individuální preference. Nelimituje se na imitaci papíru a formáty nepodléhají rigidním vlastnostem, které historicky vyplývají z (potenciálního) tisku → **editor.axdx** nebo **composer.axdx** jako potenciální prostory pro formátování a doplňování textu o další, komplexnější funkce a features (tabulka, obrázek, video...).

Zbavení se limitujících aspektů simulace papíru umožní rozšíření netradičních módů zobrazení v digitálním prostředí, např. realizaci něčeho, čemu Ted Nelson říká *parallel pages* (více toků textu, které vidíme vedle sebe) nebo s nimi související *transclusions* (zahrnutí části jednoho hypertextového dokumentu do jiného pomocí odkazu, nikoli kopírování). Díky tomu by se mohl realizovat koncept velkého textového souboru, který by fungoval jako zdroj, na jehož jednotlivé, rozdrobené části se dá odkazovat → změna datové a souborové struktury *dreams*.

Designování single-purpose nástrojů tímto způsobem umožní odlehčit komplexnost uživatelského rozhraní — jednoduché nástroje si můžou dovolit jednoduchý interface, protože se zaměřují na jednu věc, kterou se snaží umět dobře (příklad Notes nebo Voice Memos v iOS). *Do one thing and do it well* — tento princip zároveň přirozeně způsobuje odlehčení kódu. Uživateli by mělo být umožněno do kódu zasahovat, upravovat jej a dopisovat → permacomputing *dreams*.

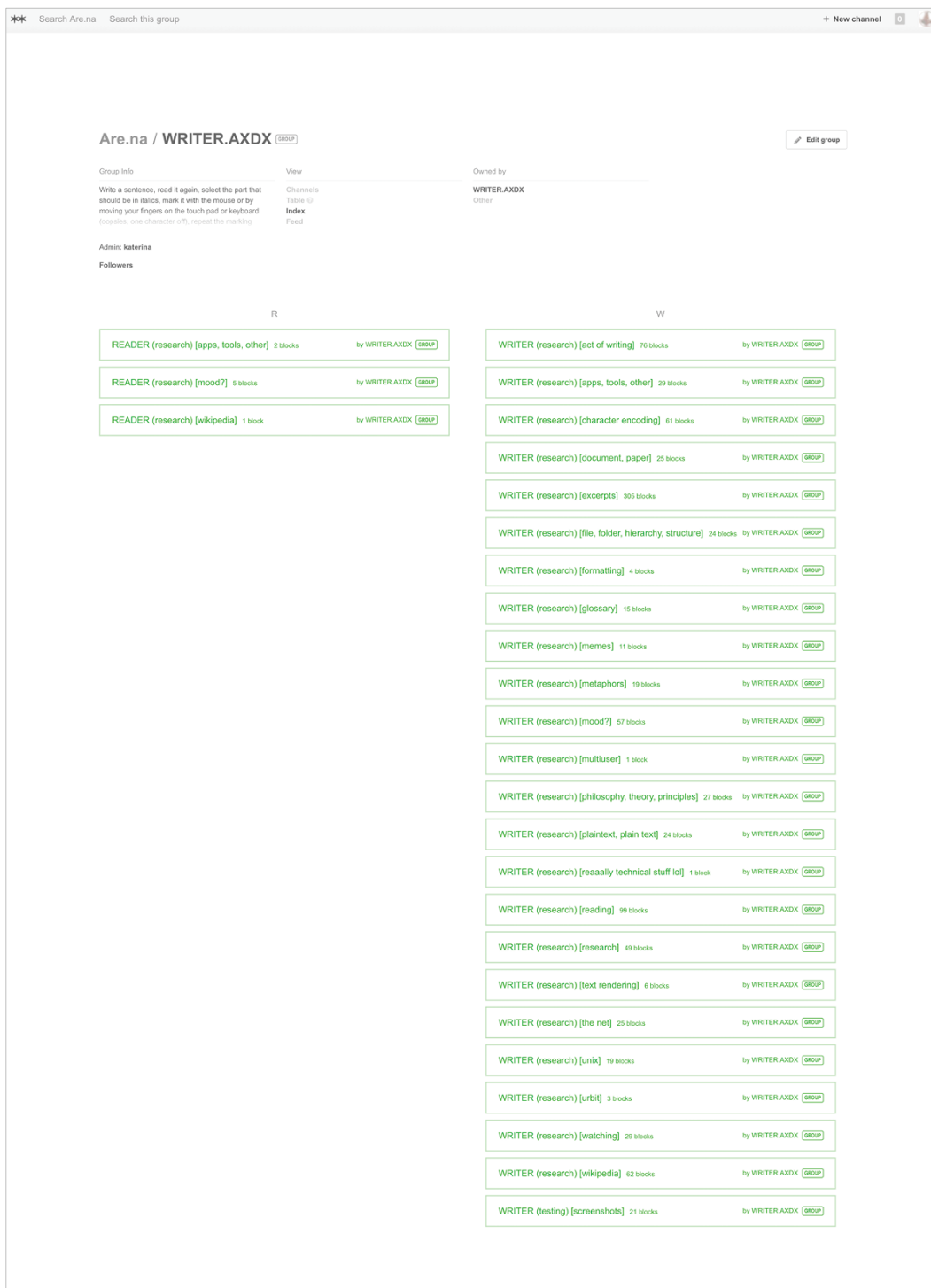
**writer.axdx** je součástí **ecosystem.axdx** — v ideálním případě je od něj odvozený jakýkoliv textový input, který bude potřeba napříč celým ekosystémem. Měl by jít ale používat i odděleně a autonomně, obvlášt pokud na základě něj vznikne *produkt*. Přejít mezi jednotlivými micro-apps je plynulý a na popředí by nemusel být příliš znatelný (pokud to není nezbytně nutné). Uživatel používá micro-apps v momentě, kdy konkrétní věc dělá — rozlišujeme tak mezi činnostmi jako je psaní poznámky, dělání korektur, řešení layoutu či sazby nebo úpravou obrázků → distraction-free OS *dreams*.



# obrazová příloha

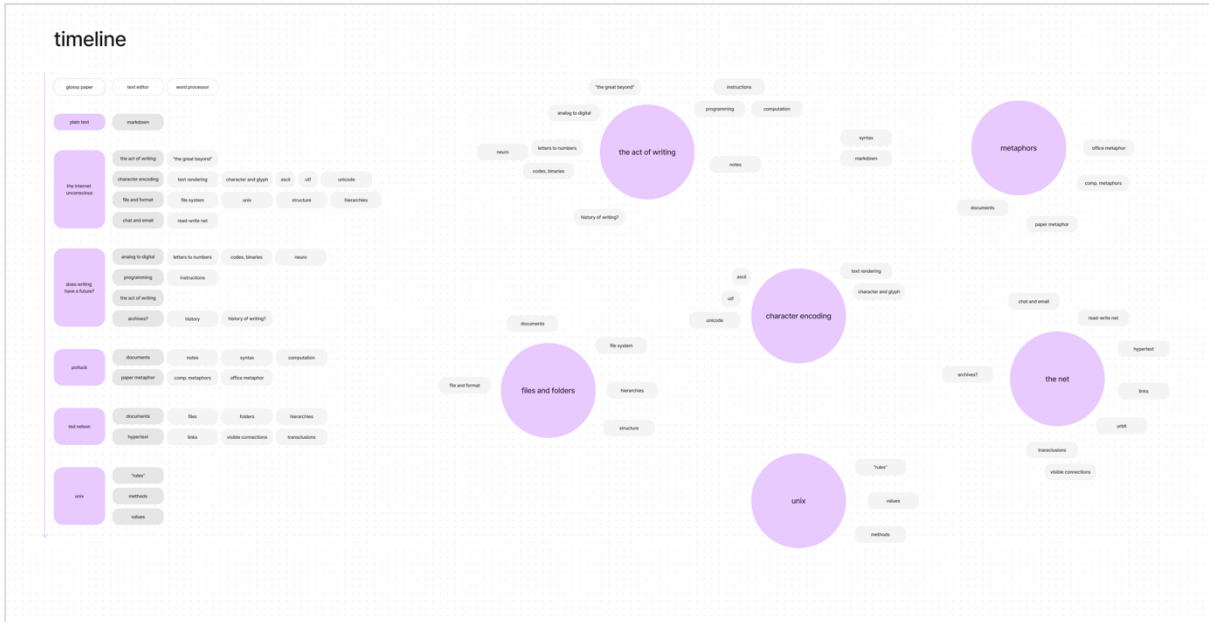
## struktura rešerše

are.na, <https://www.are.na/writer-axdx>



# vizualizace struktury

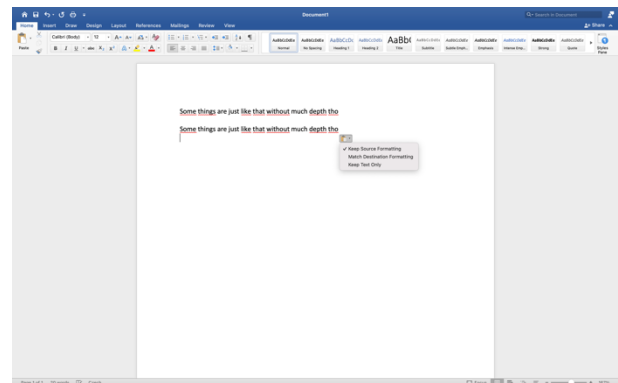
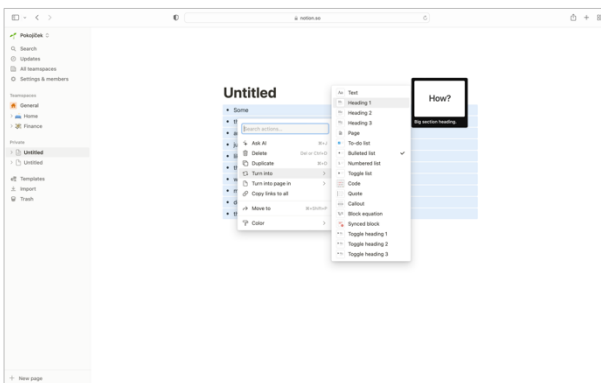
FigJam



# testování nástrojů pro úpravu textu

související are.na channel:

[WRITER \(testing\) \[screenshots\]](#)

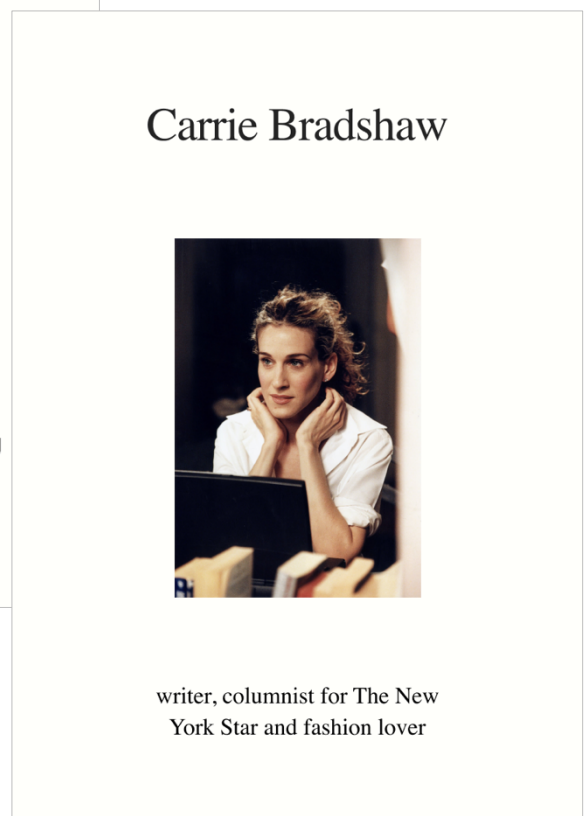
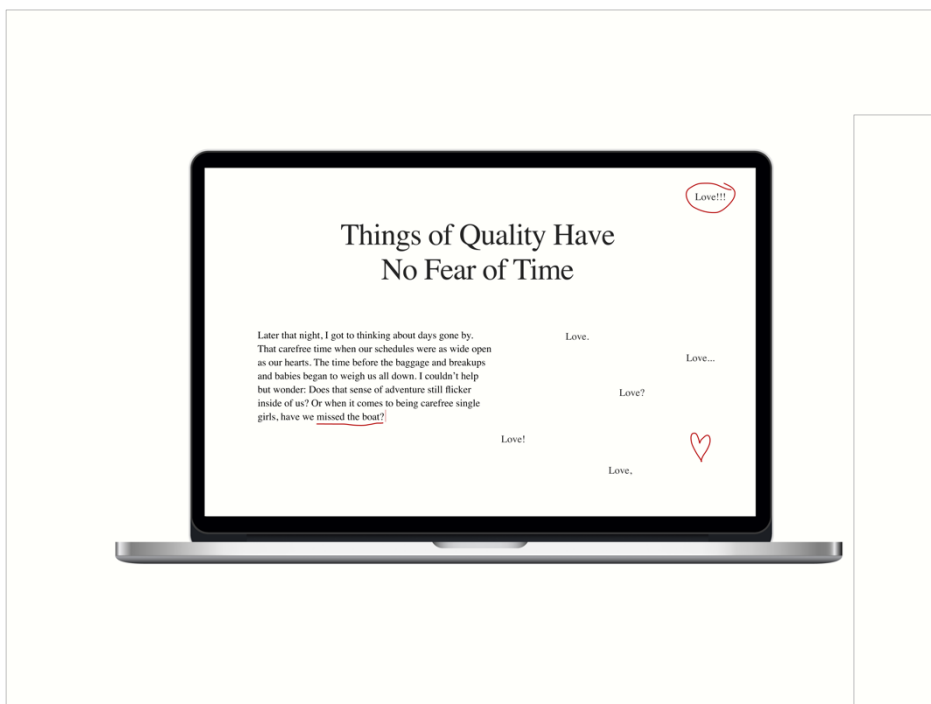


## sketches

V následujících ukázkách jsem se pokusila vžít do pár fiktivních postav a navrhnout, jak by mohla vypadat jejich vlastní varianta **writeru.axdx**. Nejde o žádný hluboký výběr ani rozbory, ale spíš rychlou *pure-fun* aktivitu, díky které jsem narazila na další otázky, na které by bylo fajn znát odpověď a témata, kterými se dál zabývat.

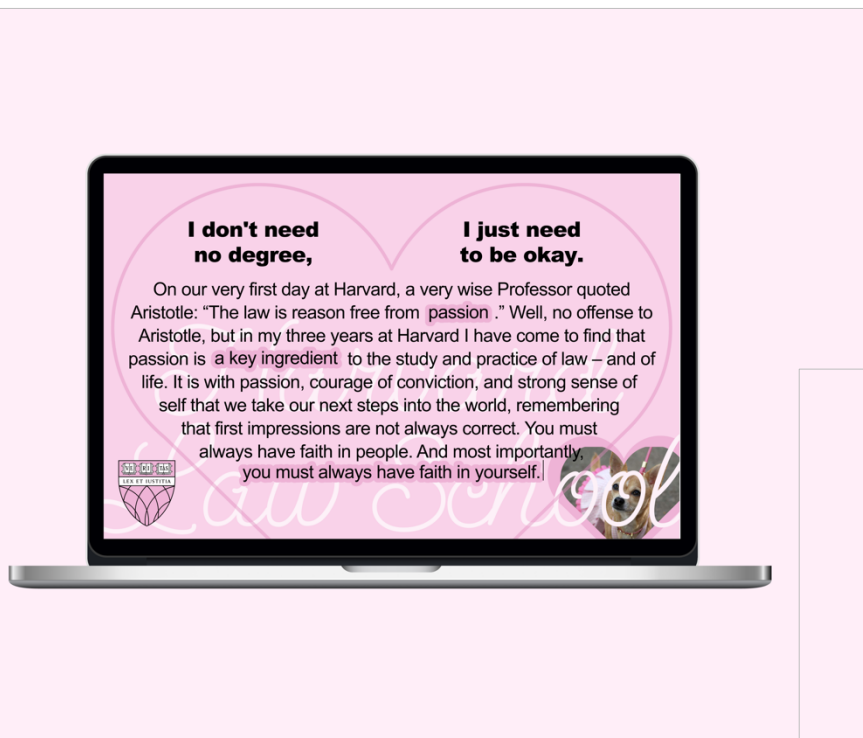
### use case #1: Carrie Bradshaw

- kde psaní začíná a končí? (psaní vs kreslení, čitelnost v kontextu rozpoznávačů textu z fotek a obrázků např. v iOS Photos, čitelnost psacího / psaného písma?)
- psaní kdekoliv v prostoru obrazovky (kam kliknu kurzorem, co se bude dít při překrývání?)



## use case #2: Elle Woods

- libovolný tvar textových polí
- připnuté obrázky v rámci customizace prostředí (cuteness, motivation atd.)
- uživatelsky volně definovatelný markdown a jak se projevuje navenek? (jsou případy ve kterých stávající markdown funguje hůř, když např. chci použít některé znaky, které využívá jako indikátory formátování, např. na Discordu)
- volba barev a výplně (pozadí, písmo, kurzor...)
- světlý a tmavý režim?



Elle Woods



lawyer fighting for animal rights  
and presidential candidate

### use case #3: Baddie Witch

- nečitelný / vizuálně kódovaný text / navrstvený text přes sebe
- markdown (jednoduché odkazy, hypertext?)
- paralelní textová pole vedle sebe



## Baddie Witch



you don't believe in astrology  
but u believe it's 3 pm?

#### use case #4: Amanda Woods

- jak budou vypadat funkce cut / copy / paste? (jakou mají paměť a kam se obsah ukládá nebo propisuje? někam by mohl, viz are.na rozšíření do Chrome, která při kopírování a vložení na are.nu ukládá zdroj)
- možné akce s textem (odeslat email, smsku, tweet...) – v podstatě funkce kopírovat a vložit, jen někam *jinam*?
- emojis?



### Amanda Woods



owner of a company that produces  
movie trailers, living in LA

# seznam použitých zdrojů

prozkoumané zdroje v nestrukturované podobě:

[WRITER \(research\) \[reading\]](#)

[WRITER \(research\) \[watching\]](#)

zmíněné zdroje, seřazeno chronologicky:

východiska, motivace

- Maggie APPLETON, *Metaphors We Web By*, 2022, <https://maggieappleton.com/metaphors-web> (cit. 15. 4. 2023).
- Aaron MARCUS, „Metaphor Design in User Interfaces“, *Journal of Computer Documentation*, vol. 22, 1998, č. 2, s. 43–57.
- Red Hat, „File storage, block storage, or object storage?“, *Red Hat*, 2018, <https://www.redhat.com/en/topics/data-storage/file-block-object-storage#:~:text=File%20storage%20organizes%20and%20represents,links%20it%20to%20associated%20metadata> (cit. 15. 4. 2023).
- Theodor Holm NELSON, Robert Adamson SMITH, „Back to the Future: Hypertext the Way it Used to Be“, *Xanadu*, 2011, <https://xanadu.com/XanaduSpace/btf.htm> (cit. 15. 4. 2023).
- Guillaume PASCUAL, *Tearing up the paper paradigm*, 2016, <https://blog.hightail.com/tearing-paper-paradigm/> (cit. 15. 4. 2023).
- Ville-Matias HEIKKILÄ, *Inverted Computer Culture: A Thought Experiment*, 2023, [http://viznut.fi/texts-en/inverted\\_computer\\_culture.html](http://viznut.fi/texts-en/inverted_computer_culture.html) (cit. 15. 4. 2023).
- Ville-Matias HEIKKILÄ, *Permacomputing*, 2021, [http://viznut.fi/texts-en/permacomputing\\_update\\_2021.html](http://viznut.fi/texts-en/permacomputing_update_2021.html) (cit. 15. 4. 2023).
- Robin SLOAN, *An app can be a homecooked meal*, 2020, <https://www.robinsloan.com/notes/home-cooked-app/> (cit. 15. 4. 2023).
- Ken THOMPSON, Dennis RITCHIE, „The UNIX Time-Sharing System“, *Bell System Technical Journal*, vol. 57, 1978, s. 1905–1929.

- Eric Steven RAYMOND, *The Art of Unix Programming*, 2003 (last revision to date), <http://www.catb.org/~esr/writings/taoup/html/index.html> (cit. 15. 4. 2023).

co by (ne)musel být writer.axdx

- Richard SEYMOUR, *The Twittering Machine*, Londýn: The Indigo Press 2019.
- Sandy BALDWIN, *The Internet Unconscious*, Londýn: Bloomsbury Publishing 2015.
- Vilém FLUSSER, *Does Writing Have a Future?* Minneapolis, MN: University of Minnesota Press 2011.
- Vilém FLUSSER, „Why Do Typewriters Go Click“, in: *Shape of Things*, London: Reaktion Books 1999. s. 62–69.
- @Zwork101, The Chaos that is Character Encodings, 2020, <https://medium.com/@zwork101/the-chaos-that-is-character-encodings-d79111a45e1d> (cit. 15. 4. 2023).
- ASCII: American Standard Code for Information Interchange, 1963, <https://www.sr-ix.com/Archive/CharCodeHist/X3.4-1963/index.html> (cit. 15. 4. 2023).
- Wikipedia contributors, „Unicode“, *Wikipedia, The Free Encyclopedia*, <https://en.wikipedia.org/w/index.php?title=Unicode&oldid=1150096766> (cit. 15. 4. 2023).
- Steven J. SEARLE, „Unicode Revisited“, *TRON Web*, <http://tronweb.supernova.co.jp/unicoderevisited.html> (cit. 15. 4. 2023).
- Maggie SHAFER, Unicode: A story of corruption, connection, and smiling poo, 2015, <https://medium.com/@maggieshafer/unicode-a-story-of-corruption-connection-and-smiling-poo-598295e4af9d> (cit. 15. 4. 2023).
- Aditya MUKERJEE, „I Can Text You A Pile of Poo, But I Can't Write My Name“, *Model View Culture*, 2015, <https://modelviewculture.com/pieces/i-can-text-you-a-pile-of-poo-but-i-cant-write-my-name> (cit. 15. 4. 2023).
- @Eevee, *Dark corners of Unicode*, 2015, <https://eev.ee/blog/2015/09/12/dark-corners-of-unicode/> (cit. 15. 4. 2023).



- Geoffrey LITT, Max SCHOENING, Paul SHEN, Paul SONNENTAG, „Potluck: Dynamic documents as personal software“, *Ink & Switch*, 2022, <https://www.inkandswitch.com/potluck/> (cit. 15. 4. 2023).
- Bonnie NARDI, James MILLER, David WRIGHT, „Collaborative, Programmable Intelligent Agents“, *Communications of the ACM*, vol. 41, 1998, č. 3, s 96–104.
- Theodor Holm NELSON, Robert Adamson SMITH, „Back to the Future: Hypertext the Way it Used to Be“, *Xanadu*, 2011, <https://xanadu.com/XanaduSpace/btf.htm> (cit. 15. 4. 2023).
- Theodor Holm NELSON, „Complex Information Processing: A File Structure for the Complex, the Changing, and the Indeterminate“ *Association for Computing Machinery '65: Proceedings of the 1965 20th national conference*, 1965, 84–100.
- Reid SCOGGIN, „From A Topiary: Hypertext and Urbit A brief history of hypertext and Urbit networking“, *Urbit*, 2021, <https://urbit.org/blog/a-topiary> (cit. 15. 4. 2023).
- Theodor Holm NELSON, *Computer Lib/Dream Machines*, Redmond, Washington: Microsoft Press 1987.
- Theodor Holm NELSON, „Here I Stand, at Age 80“, video, 47:56, *Youtube*, nahráno uživatelem TheTedNelson 4. září 2017, <https://youtu.be/mmfm-SGIGs> (cit. 15. 4. 2023).

## závěr

- Wikipedia contributors, „Ecosystem“, *Wikipedia, The Free Encyclopedia*, <https://en.wikipedia.org/w/index.php?title=Ecosystem&oldid=1148102981> (cit. 15. 4. 2023).
- Wikipedia contributors, „Digital ecosystem“, *Wikipedia, The Free Encyclopedia*, [https://en.wikipedia.org/w/index.php?title=Digital\\_ecosystem&oldid=1143737790](https://en.wikipedia.org/w/index.php?title=Digital_ecosystem&oldid=1143737790) (cit. 15. 4. 2023).
- DPS members, „ecosystem.dps“, *wiki.dps*, <https://wiki.public.services/systems/ecosystem/> (cit. 15. 4. 2023).
- Urbit: Overview, <https://urbit.org/overview/interface> (cit. 15. 4. 2023).