Drótos László

Crawlerek és scraperek

Webes tartalmak mentésére szolgáló programok

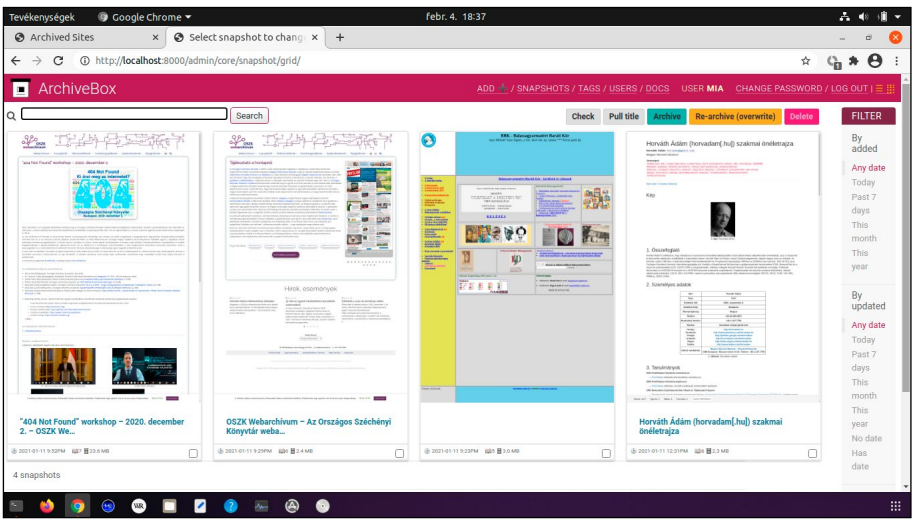
A *Könyv, Könyvtár, Könyvtáros* 2020. decemberi számában[1] Németh Márton szerzőtársammal bemutattunk néhány olyan programot, amelyeket weboldalak vagy teljes webhelyek mentésére fejlesztettek ki, elsősorban magánhasználatra, nem tömeges archiváláshoz. Jelen írás ennek a cikknek a folytatása, és ezúttal a *MIA Wiki*[2], vagyis a Magyar Internet Archívumhoz készülő tudásbázis legfrissebb szócikkeiből válogattam ki néhány érdekes alkalmazást. A wiki 2021 nyarán 136 új bejegyzéssel bővült, s mivel ezek többsége valamilyen program rövid ismertetése, így a *Szoftverek* kategória 2021 szeptemberére elérte a 372 tételt. A paletta igen széles, az egyetlen részfeladatra alkalmas egyszerű szkriptektől a komplett archiváló keretrendszerekig terjed. A két legnépesebb csoportot a *crawler*ek és a *scraper*ek alkotják, vagyis a linkeket követő és a weboldalakat letöltő „aratógépek", illetve a weboldalakról adatokat kigyűjtő „gereblyék", melyek vagy a tartalomszolgáltató szerver programból lekérdezhető felületéhez (API) kapcsolódnak, vagy maguk is egy crawlert futtatnak, és az azzal összeszedett fájlokat elemzik ki.

ArchiveBox[3]

Az *ArchiveBox* egy fiatal szoftvermérnök, *Nick Sweeting* által Python nyelven fejlesztett nyílt forráskódú archiváló szoftver. A kerékpár- és zenemániás Nick jelenleg az Egyesült Államokban és Kanadában működtet egy tanácsadással foglalkozó startupcéget, de gyerekkorát a „Nagy Tűzfal" mögötti Kínában töltötte, így mindennapos élménye volt, ahogy eltűntek vagy elérhetetlenné váltak a számára érdekes weboldalak, vagy hirtelen az egész házban elment az internet, ha

cenzúrázott szavakra próbált rákeresni. Ezért kezdett el először saját maga és barátai számára a webarchiválással kísérletezni.

Az ArchiveBox legelső verzióját 2017-ben – talán nem véletlenül – a függetlenség napján, július 4-én tette fel a GitHub kódmegosztó oldalra.[4] A cikk írásakor még mindig csak a 0.6.2-es változatnál tart, de már most is jónéhány hasznos funkciót tartalmaz ez a rendszer, amely a *wget* nevű linuxos crawlert, a *Google Chrome headless* módban futtatható motorját és a *youtube-dl* videóletöltőt integrálja. Sokféle mentési lehetőséget választhatunk, akár egyszerre többet is (szabványos WARC-formátum, fájlrendszer, egyetlen HTML-állomány, JSON-szöveg, PNG-oldalkép, PDF-dokumentum), és külön is letölthetők a megadott weboldal egyes elemei (title, favicon, response header, médiafájlok), illetve az oldal automatikusan generált technikai metaadatai, sőt az Internet Archive szerverére is elmenthetünk egy másolatot. Fontos korlátozás, hogy a linkeket legfeljebb egy szintig követi a robot, vagyis teljes webhelyek aratására már nem használható. A letöltendő URL-címek *(seed)* megadhatók egy webes űrlapon vagy egy szövegfájlban, de könyvjelzőkből vagy bookmark-szolgáltatásokból (pl. Pocket, Pinboard), böngészési előzményekből és RSS feedekből is ki tudja gyűjteni őket az ArchiveBox. Továbbá képes olyan szoftverrepozitóriumok egyes részeinek klónozására is, mint a GitHub, a Bitbucket vagy a GitLab.



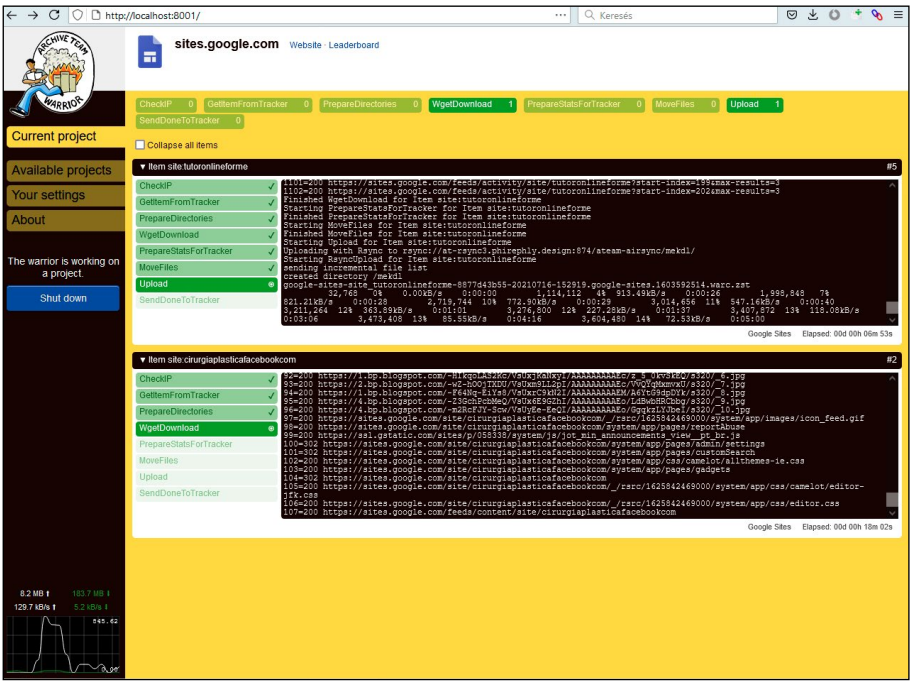
Lementett weboldalak az ArchiveBox adminisztrátori felületén

A távlati fejlesztési tervek között szerepel a *PyWb* programmal való archiválás és megjelenítés beépítése, valamint a felhasználó által definiálható szkriptek futtatása a böngészőben, ami az emberi aktivitást igénylő, webkettes platformokról való automatikus mentésekhez lenne nagyon hasznos. Az ArchiveBox összes

funkciója csak parancsmódban érhető el, de létezik hozzá egy webes adminisztrátori felület is, amivel paraméterezhetők és elindíthatók egyszerűbb feladatok, valamint visszanézhetők a korábbi mentések. Érdemes még megemlíteni, hogy az ArchiveBox wikijében van egy kiváló linkgyűjtemény[5] a különféle archiváló szervezetekről, projektekről, szoftverekről, illetve a témával foglalkozó cikkekről, blogokról és fórumokról.

Warrior[6]

A *Warrior* egy civilekből és szakemberekből álló laza szerveződés, az *Archive Team*[7] számára összeállított virtuális gép, melynek segítségével önkéntesek is be tudnak kapcsolódni az archiváló projektekbe a saját számítógépükkel. A csoport tagjai elsősorban az egyéni szintű archiváláshoz nyújtanak segítséget információkkal és technológiákkal, továbbá figyelik a veszélyeztetett, bezárásra készülő webhelyeket vagy online platformokat, és ilyenkor mentési akciókat szerveznek. A mozgalmat az osztrák közszolgálati média által a digitális archiválás frontemberének nevezett különc, *Jason Scott* indította el 2009-ben, aki egyéb tevékenységei (technológiatörténész, filmkészítő, színész és előadó) mellett a régi szövegfájlokból és ASCII-karakteres rajzokból álló *http://textfiles.com/* honlap létrehozója, és annak a *Sockington* nevű macskának az egyik gazdája, „akinek" több mint egymillió



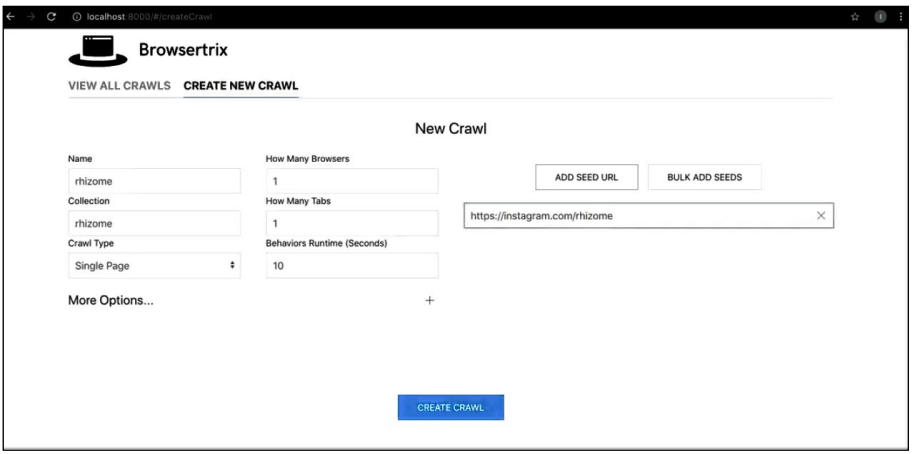
Két futó munkafolyamat az Archive Team Warrior nevű virtuális gépen

követője van a Twitteren. Az ArchiveTeam magyar honlapja[8] 2021 elején indult, és saját webarchívumot is szolgáltat *Régi Szép Idők* (RSZI) néven. A Warrior Windows, macOS és Linux alatt egyaránt futtatható VirtualBox vagy VMware Player virtualizációs környezetben, de Docker-formában is telepíthető. A *wget* crawlert használja, az egyes jobokat pedig a központi szerveren futó *Tracker-rendszer* vezérli. A felhasználó kiválaszthatja, hogy hány ilyen munkafolyamatot akar futtatni párhuzamosan, illetve hogy az Archive Team aktuális projektjei közül melyekben szeretne részt venni. Ezután a Warrior a Trackertől kapott paraméterek alapján elindítja a jobokat, majd ha valamelyik webhely letöltése befejeződött, akkor a WARC-fájlokat úgynevezett *rsync-kapcsolaton* keresztül feltölti előbb az Archive Team tárhelyére, onnan pedig bekerülnek az Internet Archive gyűjteményébe is. A Warrior paraméterezése és működésének követése böngészőből történik, egy nagyon egyszerű, logikus felületen a *localhost:8001* címet megnyitva. Regisztrációt nem igényel, csak egy felhasználói nevet kell választanunk, amely az „önkéntes bedolgozók" aktuális listáján fog megjelenni a Tracker weboldalán.

Browsertrix[9]

A *Browsertrix* egy böngészőalapú archiváló szoftver, mellyel a weblapok HTML-kódját elemző „buta" robotoknál jobb minőségben – igaz, jóval lassabban – lehet lementeni a mai bonyolult, gyakran szkriptekkel generált weboldalakat. A Browsertrix mögött az a San Franciscóban élő *Ilya Kramer* áll, aki a már említett tavalyi 3K-cikkben ismertetett *Webrecorder*t és *ReplayWeb.page*-t is írta, több más nagyszerű archiváló és megjelenítő eszközzel együtt. Korábban az Internet Archive-nál dolgozott, majd átment a *Rhizome* nevű, digitálisművészet- és digitálismédia-megőrzéssel foglalkozó nonprofit szervezethez. A Rhizome működteti a Webrecorder technológiáját használó *conifer.rhizome.org* archiváló szolgáltatást és az *oldweb.today* oldalt, ahol emulált környezetben futó régi böngészőkkel nézhetők meg a régi webhelyek. A Browsertrix a nevét a „browser" és a „Heritrix" szavak összevonásából kapta (utóbbi az Internet Archive crawlere). Fejlesztése a 2010-es évek közepére nyúlik vissza, és három fázisra tagolható. A legelső változat a weboldalak bárki által annotálhatóvá tételével foglalkozó *Hypothes.is* projekt Annotator szoftvere számára készült 2015-ben, de mivel már az is nyílt forráskódú volt, így más rendszerekbe is be lehetett építeni.[10] A *Selenium* nevű böngészőautomatizáló eszközt használta a *Chrome* vagy a *Firefox* vezérlésére is, és a betöltött weboldalakat elküldte a *webrecorder.io* vagy a *Save Page Now* szolgáltatásnak. (Előbbi a Conifer elődje, utóbbi pedig az Internet Archive *archive-on-demand* funkciója.) A technikai metaadatok – beleértve az eredeti és az archív URL-címet is – pedig egy JSON-naplófájlba kerültek. Az IIPC (International Internet Preservation Consortium) 2019-es konferenciáján Ilya Kramer bemutatott egy továbbfejlesztett változatot, amely egy Docker-konténerben futtatható önálló archiváló eszköztár.[11] Ehhez

már készült egy egyszerű, grafikus felület is, de bonyolultabb aratásokat csak parancsmódban vagy YAML-formátumú fájlokon keresztül lehet konfigurálni. A headless-módban futtatott Chrome-hoz külön definiálhatók „viselkedések" *(behaviors)*, ezekkel vezérelve például a Twitter- vagy a Facebook-oldalak letöltését, a Webrecorder *autopilot* funkciójához hasonlóan. Opcionálisan oldalképek is készíthetők vele, a weboldalak szövege pedig *Solr*ral indexelhető. A bejelentkezést igénylő webhelyekhez külön böngészőprofilok definiálhatók, így elkerülhető (bár nem százszázalékos biztonsággal), hogy a bejelentkezési adatok bekerüljenek a WARC-fájlba. A mentéseket PyWb-vel lehet visszanézni. Két évvel később a projekt ismét új irányt vett: egyetlen integrált rendszer helyett modulokra szedte szét a fejlesztője. 2021 őszén még csak a crawler-komponens volt letölthető (szintén Dockerben), amihez még nem készült el a grafikus felület.[12] A Browsertrix Crawler a *Puppeteer* nevű eszköz segítségével vezérli a böngészőt, a mentést pedig a PyWb végzi *capturing*-üzemmódban. Van egy *screencasting*-opciója is, amivel a böngészőben figyelhetjük, ahogy a robot letölti a weboldalakat. A kiinduló URL-lista fájlban is megadható, és seedenként külön konfigurációs szabályok határozhatók meg. Több szálon futtatható és ehhez is vannak *behavior*-szkriptek (pl. autoscroll, video autoplay, valamint webhelyspecifikus viselkedések). A WARC- mellett WACZ-formátumba is tud menteni, ami a ReplayWeb.page számára szükséges indexeket és technikai adatokat tartalmazza, de természetesen PyWb-vel is visszanézhető az archivált tartalom.

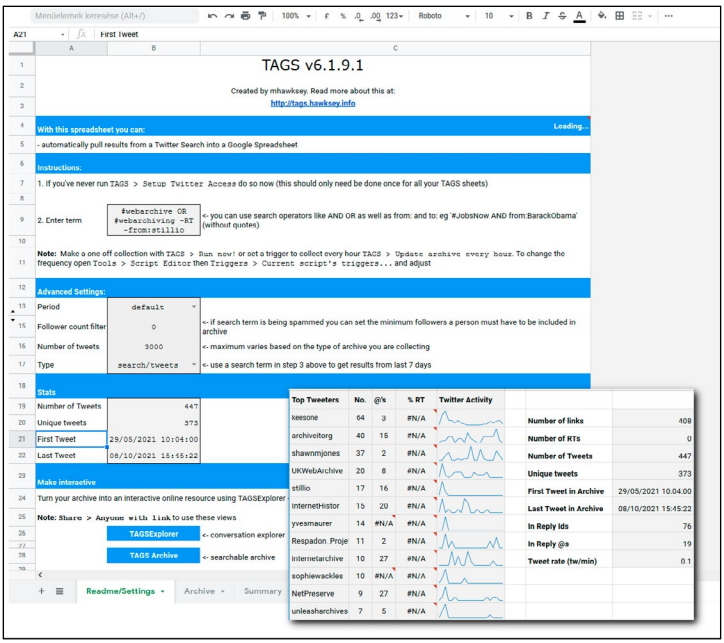


Instagram-oldal mentésének beállítása a Browsertrix programban

TAGS[13]

A *TAGS,* ami a Twitter Archiving Google Sheet rövidítése és persze a Twitteren használt *hashtag*ekre utal, egy régi és sajnos már nem frissülő esz-

köz, de két szempontból is érdemes bemutatni. Egyrészt, mert egy egyszerű példa arra, hogy hogyan lehet adatokat „összegereblyézni" a webről, másrészt a története jól illusztrálja, hogy az archiváló technológiák milyen gyorsan elavulnak az internetes szolgáltatások gyakori változásai miatt. *Martin Hawksey*, aki most a skóciai *Edinburgh Futures Institute* tanulástervezési és oktatástechnológiai vezetője 2010-ben hobbiprojektként kezdett el foglalkozni azzal, hogy hogyan lehetne a közösségi médiából gyűjtött adatokat felhasználni az oktatásban. Ekkor írta meg a *TAGS scraper*t, majd egy évvel később a *TAGSExplorer* modult, amivel elemezhetők, vizualizálhatók és kereshetők a begyűjtött információk. A scraperek más megközelítést alkalmaznak, mint az előzőekben bemutatott crawlerek, és jellemzően más célokra is használjuk őket. Ahelyett, hogy az eredeti weboldalakról próbálnánk meg minél hűségesebb másolatokat készíteni, azok lényegi tartalmát és/vagy metaadatait gyűjtjük be, például tudományos vagy piackutatási elemzésekhez. Ez technikailag egyszerűbb feladat, mint komplex webhelyek lementése és megtekinthető állapotban tartása, viszont elvész az eredeti külalak és kontextus. A TAGS egy sablon és egy szkript

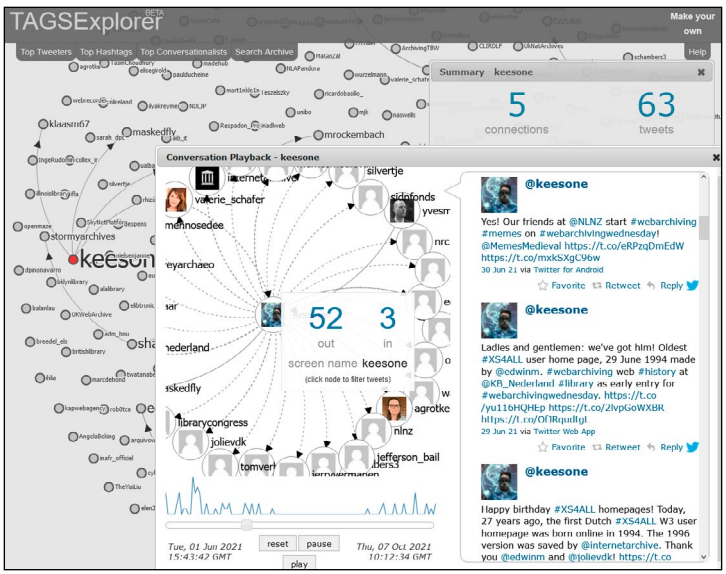


A TAGS paraméterezése a webarchiválással kapcsolatos tweetek gyűjtéséhez és az összesített adatokat tartalmazó munkalap egy részlete

a Google Dokumentumok táblázatkezelőjéhez, amiről egy másolatot készíthetünk a saját tárterületünkre, majd engedélyeznünk kell, hogy hozzáférjen a Google- és a Twitter-fiókunkhoz. (Ez a biztonsági szigorítások miatt manapság

egy elég körülményes műveletsor.) A táblázat *Readme/Settings* munkalapján tudjuk paraméterezni a szkriptet. Az archiválandó tweetek szűrhetők hasthtag és felhasználó szerinti keresőkérdéssel, illetve a kedvencek listája alapján, továbbá megadható a maximális számuk és a frissítés gyakorisága is. A scraper a TAGS menü *Run Now!* menüpontjára való kattintás után a Twitter API-ján keresztül elkezdi letölteni a tweetek szövegét és tizenhétféle metaadatát (pl. felhasználói név, profilkép-URL, időpont, földrajzi hely, nyelv), melyek az *Archive* munkalapon jelennek meg. Visszamenőleg legfeljebb az utolsó hét nap üzeneteit tudjuk leszedni a nyilvános API korlátai miatt. A menüből két további fület *(Summary* és *Dashboard)* is hozzáadhatunk a táblázathoz, ahol összesítéseket és grafikonokat látunk a begyűjtött adatokról.

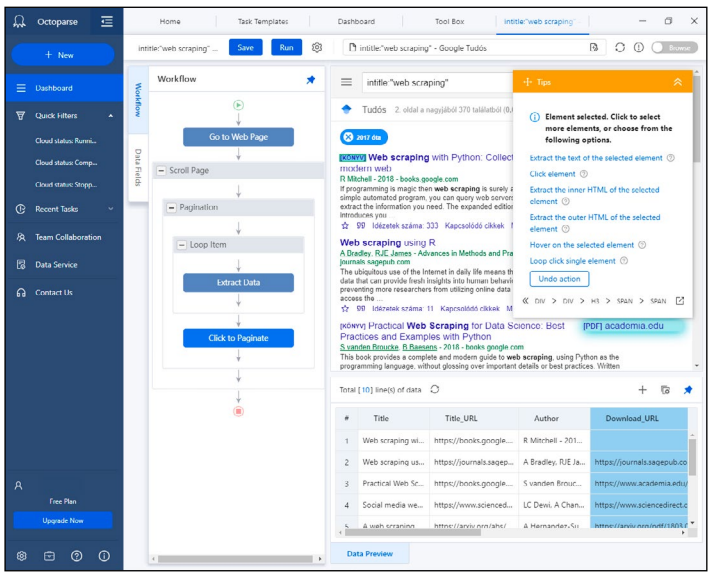
A TAGSExplorer linkre kattintva pedig egy külön böngészőablakban megnézhető az emberek közötti kapcsolati háló, és visszajátszható egy kiválasztott felhasználó tevékenysége harminc másodpercbe sűrítve. A Google és a Twitter szigorításai miatt Martinnak többször is módosítania kellett a szkriptet, míg végül 2018-ban – úgy tűnik – végleg abbahagyta a karbantartását, így ez az egyszerű és sokak által használt eszköz lassan működésképtelen lesz. A TAGSExplorer és a TAGS Archive-linkek például már nem jók az eredeti sablonban, és az egyik fórumtag által 2020 szeptemberében javasolt módosítás[14] is csak az előbbit javítja meg, e cikk írásakor az archívum keresője már nem volt elérhető.



A Holland Királyi Könyvtár digitális gyűjteményében dolgozó, magyar származású történész, Kees Teszelszky webarchiválási témájú tweetjeinek visszajátszása a TAGSExplorerrel

Octoparse[15]

Az *Octoparse* az egyik legjobb webscraper-rendszer, ami Windows vagy macOS rendszerű asztali gépen futtatható változatban és előfizetéses felhőszolgáltatásként is elérhető. A kaliforniai központú *Octopus Data Inc.* céget 2015 szeptemberében hozta létre a kínai *Shenzhen World Information Technology* és a következő év márciusában mutatták be az eredetileg *Octopus Collector* nevű szoftverük Octoparse-ra átkeresztelt, angol nyelvű változatát. (A szóösszetétel feltehetően arra utal, ahogy a program polipszerű karjaival kinyúl az internetre, majd kielemzi a begyűjtött weboldalakat.) Sok más hasonló célú versenytársától eltérően az Octoparse használata nagyon könnyen megtanulható a rámutatással és kattintással működő konfiguráló felületnek, az intelligens elemző algoritmusnak és a helyzetérzékeny súgónak köszönhetően. Miután megadtuk a kiinduló weboldal URL-címét, a program betölti azt és felismeri a főbb HTML-elemeket, majd táblázatos formába rendezi a bennük talált tartalmakat, melyek lehetnek adatok, szövegek és linkek. A táblázat oszlopait átnevezhetjük, átrendezhetjük vagy törölhetjük, illetve felvehetünk továbbiakat is úgy, hogy az egérrel kijelölünk elemeket a weblapon.

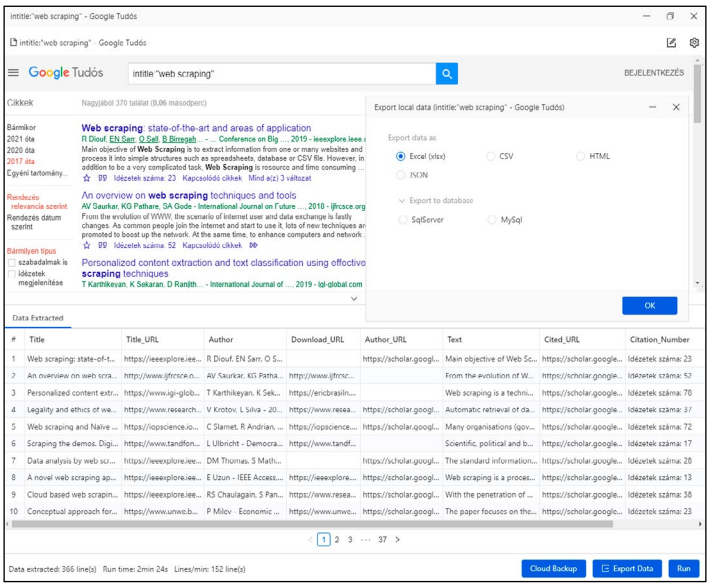


Web scraping témájú publikációk adatainak kigyűjtése a Google Tudós találati listájából az Octoparse oldalelemző funkciójának segítségével

Ha van lapozási vagy további tartalombetöltési lehetőség az oldalon, akkor azt is megtaníthatjuk a scrapernek – amennyiben nem ismeri fel automatikusan. A főbb közösségi, üzleti, utazási és egyéb szolgáltatásokhoz előre elkészí-

tett sablonok állnak rendelkezésre, így azokkal még könnyebb ez a munkafázis. A *Run* gombra kattintva a beépített böngészőben elindul a weboldalak betöltése és azokból az adatok begyűjtése, melyek azután XLS-, CSV-, JSON-, HTML-fájlokba vagy SQL-adatbázisokba exportálhatók.

Az egyes *task*ok beállításánál több hasznos lehetőség is van: kiválasztható, hogy a program milyen böngészőként azonosítsa magát, blokkolja-e a reklámokat, letöltse-e a képeket, leálljon-e, ha úgy tűnik, hogy végtelen ciklusba került, cserélgesse-e az IP-címeket a kitiltás elkerülése érdekében stb. Az aratások természetesen ütemezhetők is, és amennyiben a saját gépünk helyett a felhőben futtatjuk őket, akkor kérhetünk e-mail-értesítést, amikor véget értek, és ilyenkor API-n keresztül is letölthetők az adatok. Az ingyenes változatnál tíz scraper definiálható, legfeljebb két feladat futhat egy időben és csak a saját gépünkön, maximum tízezer adatrekord exportálható ki egyszerre, viszont a bejárható oldalak száma nem korlátozott.

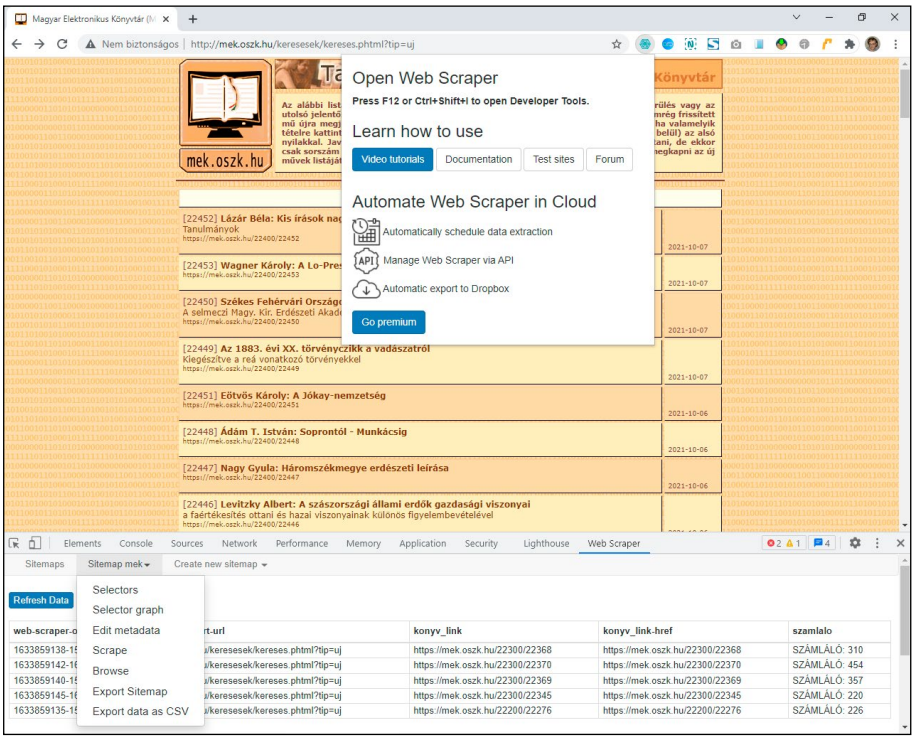


Az Octoparse által begyűjtött adatok exportálása Excel-táblázatba

Web Scraper[16]

A szimplán csak *Web Scraper*nek elnevezett ingyenes böngészőkiegészítőt, és mellé a fizetős, felhőalapú, webes adatgyűjtő szolgáltatást egy 2017-ben alapított cég, a Lettországban bejegyzett *Web Graph* fejlesztette ki. Előbbinek az alapötlete nagyon egyszerű: magát a Chrome vagy Firefox böngészőt használjuk scraperként, így a JavaScripttel generált, Ajax-kéréseket küldő vagy bejelentkezést igénylő weboldalakkal sincs gond. A bővítmény telepítése után az F12 gomb

megnyomásával elérhető fejlesztői eszközök között megjelenik egy további fül három fő menüponttal. Új feladatot a *Create new sitemap* menüben hozhatunk létre, és ugyanitt arra is lehetőségünk van, hogy egyszerűen bemásoljunk egy kész utasítássort JSON-formátumban. (Ilyeneket a honlapon levő oktatóvideók alatt és a fórumban találhatunk.) Új sitemap esetén adni kell neki egy nevet, valamint beírni egy vagy több URL-címet, ahonnan majd a scraper elindul. A már létrehozott feladatok a *Sitemap* alatt listázhatók ki és szerkeszthetők vagy törölhetők. Szerkesztés módban a Sitemap menü *Selectors* alpontja nyílik meg, itt tudjuk megadni a *Select* gomb megnyomásával, majd az egérrel való rámutatással, hogy a weboldal mely elemét válassza ki a program, és hogy az milyen típusú. A legegyszerűbb a *text* típus, melynél csak az adott oldalelem szövege (pl. egy név vagy e-mail-cím) kerül begyűjtésre. Ha több ilyen elem is van az oldalon, akkor a shift billentyű lenyomása mellett érdemes még néhányat kiválasztani, hogy megtanítsuk a Web Scrapert ezek felismerésére, és ilyenkor a *Multiple* opciót is ki kell pipálni a selector űrlapján. A *text*en kívül további tizenkét elemtípus közül választhatunk, amikkel különböző műveleteket végeztethetünk el, például linkek követése, oldal görgetése, lapozás. Ezek a selectorok



Megtekintésszámok kigyűjtése a Magyar Elektronikus Könyvtárból az újdonságok listája alapján Web Scraperrel

szülő-gyerek kapcsolatban lehetnek egymással, így egy munkafolyamattá fűzhetők össze, mely gráfformában is megnézhető. Az adatgyűjtés a Sitemap menü *Scrape* alpontjával indítható, az eredmény pedig a *Browse* menüpont alatt jelenik meg táblázatként, ami azután CSV-fájlba exportálható.

A program működési logikájának megtanulása időigényes, de szerencsére jó dokumentáció, videós oktatóanyag, blog, fórum és többféle tesztoldal is van hozzá, így aki ezekbe beleássa magát, az egészen bonyolult feladatokat is meg tud oldani a böngészőjével. A cég felhőjében futó, *Web Scraper Cloud* elnevezésű szolgáltatás többféle csomagban is előfizethető. A legdrágábbnál gyakorlatilag korlátlan a robottal bejárható oldalak száma, ötnél több párhuzamos job is futtatható, megadható külső proxy, az aratások ütemezhetők, az eredmények hatvan napig maradnak a felhőtárhelyen, és a CSV- mellett van XLSX-, JSON-, Dropbox- és Google Sheets-export, valamint API-n keresztüli elérés is.

Jegyzetek

1. Drótos László – Németh Márton: *Egyedi mentésekre szolgáló webarchiváló szoftverek.* = Könyv, Könyvtár, Könyvtáros, 29. évf. 2020. 12. sz. 3–11. p. https://epa.oszk.hu/01300/01367/00334/pdf/EPA01367_3K_2020_12_003-011.pdf (2021.09.20.)
2. https://webarchivum.oszk.hu/mediawiki/ (2021.09.20.)
3. https://archivebox.io/ (2021.09.20.)
4. https://github.com/ArchiveBox/ArchiveBox (2021.09.20.)
5. https://github.com/ArchiveBox/ArchiveBox/wiki/Web-Archiving-Community (2021.09.20.)
6. https://wiki.archiveteam.org/index.php/ArchiveTeam_Warrior (2021.09.20.)
7. https://en.wikipedia.org/wiki/Archive_Team (2021.09.20.)
8. http://archiveteam.hu (2021.09.20.)
9. https://github.com/ikreymer/browsertrix (2021.09.20.)
10. https://github.com/ikreymer/browsertrix (2021.09.20.)
11. https://github.com/webrecorder/browsertrix (2021.09.20.)
12. https://github.com/webrecorder/browsertrix-crawler (2021.09.20.)
13. https://tags.hawksey.info (2021.09.20.)
14. https://tags.hawksey.info/forums/topic/explorer-archive-hyperlinks-broken-in-6-1-9-1/ (2021.09.20.)
15. https://www.octoparse.com (2021.09.20.)
16. https://webscraper.io (2021.09.20.)