**SIP-creator Beschrijving Software**

1. Doelstelling SIP-creator

De SIP-creator dient een *Plug & Play* software te zijn waarmee kleine archiefinstellingen aan de slag kunnen gaan om archiefdata en metadata op te slaan in een *containerformaat*  (i.e. *exportformaat*).

Plug & play software vereist de creatie van een grafische user interface (GUI), om de gebruiksvriendelijkheid van de software te maximaliseren. Zowel de GUI als de tools die in de SIP-creator verwerkt worden, dienen gebaseerd te zijn op open standaarden en open source software.

Wat de data betreft moet de SIP-creator in staat zijn om bestandsformaten te identificeren, wijzigingen in de data moeten verifieerbaar zijn en de structuur van de data moet duidelijk gemaakt kunnen worden. Daarnaast moeten de bestandsnamen van de opgeladen data in bulk kunnen worden hernoemd, dit ter bevordering van de digitale hygiëne van de bestanden.

Het containerformaat dat de SIP-creator zal genereren zal eveneens conform bestaande standaarden dienen te zijn. De algemene bruikbaarheid van het containerformaat is eveneens wenselijk, zodat dit containerformaat niet op maat is van één soort repository, maar door middel van een exportfunctie op maat van de gekozen repository aangeleverd kan worden.

Naast data dient het containerformaat ook de nodige metadata te bevatten. De technische metadata moet uit het bestandsformaat (indien dit een bekend formaat is) kunnen worden gehaald, bij voorbeeld size, checksum en mimetype. Descriptieve, administratieve en juridische metadata moeten geëxtraheerd kunnen worden uit het registratiesysteem (indien dit gebaseerd is op een standaard als Dublin Core, EAD of MARC), uit een CSV-bestand, of uit een (online) template invulformulier. Het moet eveneens mogelijk zijn om een lijst van metadatarecords toe te voegen aan de container. Vervolgens wordt deze metadata in een XML-formaat van een internationaal aanvaardde standaard (METS, PREMIS) geplaatst, om zodoende samen met de originele data bewaard te worden.

1. Onderzoek naar de beschikbare tools

De SIP-creator zal reeds bestaande open source software gebruiken als bouwstenen voor de creatie van een plug & play software. Hierna volgt een overzicht van de geteste software, hun mogelijkheden en hun tekortkomingen als onderdeel van de SIP-creator software.

* 1. Verpakken van data en metadata
     1. Reeds bestaande *container software*

Een eerste soort software die werd getest, is reeds bestaande SIP-creator software. Er werd nagegaan in welke mate de bestaande SIP-creators voldoen aan de bovengenoemde vereisten.

**Delving SIP Creator**

Delving SIP Creator**[[1]](#footnote-1)** is een software die is bedoeld om (metadata)bestanden op te laden naar het platform CultureHub[[2]](#footnote-2). CultureHub aanvaardt records in meerdere XML-formaten, maakt de records beschikbaar via een OAI-PMH-interface[[3]](#footnote-3) (in de geleverde formaten), en voorziet een zoekinterface om de pakketten te ontsluiten. Om de zoekmogelijkheid te uniformiseren verplicht CultureHub dat de metadata ook wordt aangeleverd in het XML-formaat AFF (Another Fine Format),een eigen XML-formaat dat juist het nodige bevat om te kunnen zoeken in CultureHub. Daarom moeten de metadatavelden gemapt worden naar AFF. CultureHub voorziet mappingen om bestaande XML-standaarden te mappen naar AFF, maar hiervoor is de tussenkomst van technici vereist.

De gemaakte pakketten zijn enkel nuttig voor ingest naar CultureHub, en niet bruikbaar voor andere repositories.

Het programma is geschreven in Java, en is dus beschikbaar voor elk besturingssysteem dat beschikt over Java-ondersteuning. Uitgebracht in Open-Source.

**Europeana SIP Creator**

De Europeana SIP Creator**[[4]](#footnote-4)** gelijkt zeer sterk op Delving SIP Creator. Het biedt dezelfde functionaliteit, maar is bedoeld om pakketten aan te maken om op te laden naar Europeana. De gemaakte pakketten zijn bijgevolg enkel nuttig voor ingest naar Europeana, en niet bruikbaar voor andere repositories.

Eveneens geschreven in Java, en uitgebracht in Open-Source.

**Fedora SIP Creator**

Fedora SIP Creator[[5]](#footnote-5) is een webapplicatie om digitale pakketten aan te maken, die in een Fedora-repository kunnen opgeladen worden. Zo'n digitaal pakket bestaat uit een lijst aan bestanden, samen met metadata in meerdere formaten. De bestanden kunnen geselecteerd worden via een bestandskiezer. Het biedt de mogelijkheid om mappingen te maken naar een ander metadata-formaat. Het voortgebrachte pakket is enkel nuttig voor ingest in een Fedora repository.

**SobekCM Editor - U of Florida**

SobekCM Editor**[[6]](#footnote-6)** is een door de University of Florida ontwikkeld programma om op een visuele manier METS-documenten aan te maken, te bekijken, en aan te passen.

Het is enkel beschikbaar voor Windows, en dus niet platformonafhankelijk.

De conclusie is dus dat bestaande SIP-creator software weliswaar vrij beschikbaar is (open source) en in veel gevallen een open standaard (platformonafhankelijk), maar meestal gericht op één bepaalde repository (CultureHub, Europeana). Daarnaast is er ook onvoldoende aandacht voor de data zelf. Bovengenoemde software legt de nadruk op het verpakken van metadata en het leggen van mappingen. De data zelf worden in vele gevallen niet meegeleverd met de metadata. Het volstaat om in de metadata een permanente link op te nemen naar de data.

* + 1. Meer aandacht voor de data: BagIt

BagIt[[7]](#footnote-7) is een standaard die een mappenstructuur afspreekt om zo data en metadata en de bijhorende checksums (zie onder) op te slaan in één “pakket”. Een *bag* bestaat enerzijds uit data, die bestaan uit een reeks mappen en/of bestanden, ook wel *payloads* genoemd, en anderzijds uit *tags*, zijnde metadatabestanden. BagIt zorgt ervoor dat ondubbelzinnig het onderscheid kan maken tussen de data en de metadata, zonder de tussenkomst van de eigenaar te vereisen.

Een BagIt bevat de volgende structuur:

<bagit-directory>

bagit.txt

manifest-md5.txt

bag-info.txt

data/

payload1.jpg

payload2.jpg

Waarbij:

bagit.txt → metadata bevat mbt de versie van BagIt

manifest-md5.txt → de checksums van de payloads bevat

bag-info.txt → de beschrijvende metadata bevat

data → de submap is waarin de data –payloads- wordt bewaard

Een essentieel kenmerk van Bagit is dat wijzigingen in de data (payloads) en de metadata (tags) verifieerbaar zijn. Dit wordt verwezenlijkt aan de hand van zogenaamde controlesommen of *checksums*. Tussen zender en ontvanger van een bagit bestaat een overeenkomst: de zender maakt door middel van een Bagit-software het manifestbestand aan; de ontvanger herberekent de checksums van de payloads, en vergelijkt die met de checksums zoals deze vermeld staan in het bestand *manifest-md5.txt*. Wanneer die niet met elkaar overeenkomen, zijn er wijzigingen binnengeslopen, hetzij bewust (meestal met kwaadwillig opzet), hetzij onbewust (wegens netwerktransport). (zie verder 2.3).

Alle payloads, die vermeld staan in het manifest-bestand, staan verplicht in de datamap. Elke andere locatie wordt verworpen. Optioneel wordt het toegelaten om in het bestand *fetch.txt* een lijst van externe locaties (i.e. via hyperlinks) op te nemen. Die moeten eerst opgehaald worden, vooraleer de validatie kan worden gestart. Die optie is handig wanneer meerdere bagits dezelfde data delen. Bagits met dergelijke gaten worden *holey* genoemd. Het is echter aangewezen om te vermijden links op te nemen naar buiten het exportpakket, om *broken links* te voorkomen.

Naast de data en hun checksums kan een Bagit ook metadata bevatten. Het bestand *bag-info.txt* bevat een vlakke beschrijving van de bag:

Bag-Size: 758.7 KB

Bagging-Date: 2011-01-09

DC-AccessRights: ugent

DC-Creator: Smith, John

DC-DateAccepted: 2011-01-09

DC-Identifier: rug01: 001886277

Daarbij worden de Bag-Size en de Bagging-Date gegenereerd door de BagIt software, de overige beschrijvende metadata kan door de maker van de bag zelf worden toegevoegd, alvorens de bag te maken weliswaar.

Er bestaan verscheidene softwarebibliotheken, die deze specificatie hebben omgezet naar bruikbare code: BagIt gem (Ruby), bagit (Python), pybagit (Python) en BagItPHP (PHP), BagKit (Ruby), maar de bekendste is Bagit Library van de Library of Congress, geschreven in Java.

**Library of Congress – TransferTools – BagIt[[8]](#footnote-8)**

Console-applicatie[[9]](#footnote-9) waarmee bagits kunnen worden gemaakt, gevalideerd, en aangepast. Deze softwarebibliotheek werd geschreven in Java, waardoor het platformonafhankelijk is. De console-applicatie fungeert als een wrapper rondom deze bibliotheek.

Doordat het een console-applicatie is, en geen grafische user interface heeft, voldoet deze tool niet aan de vooropgestelde eisen van de SIP-creator.

**Library of Congress – TransferTools - Bagger**[[10]](#footnote-10)

Net als Bagit ontwikkeld door de Library of Congress, maar in tegenstelling tot bovengenoemde, werd bij Bagger een GUI rondom de software bibliotheek van Bagit geschreven. Interessant is de toevoeging van de BagView, een grafisch paneel waarmee de metadata, opgeslagen in het bestand bag-info.txt, kan worden aangepast. De velden van de bag-info.txt kunnen eveneens onderworpen worden aan regels zoals vastgelegd in een profiel. Handig dus voor organisaties die op een vaste manier hun metadata beschrijven.

**UGent – BagKit[[11]](#footnote-11)**

GUI rondom de hierboven vermelde software bibliotheek BagIt. Bevat dezelfde functionaliteit als hierboven vermelde applicatie, en is eenvoudig in gebruik. BagKit echter maakt onrechtstreeks gebruik van de bagit-library door de bagit-tool van LOC op de achtergrond op te roepen en de output ervan te parsen. Bagger daarentegen maakt rechtstreeks gebruik van de java bibliotheek bagit. Nog een extra troef van Bagger ten opzichte van BagKit is dat het aanpassen van metadata reeds zit ingebed in de Bagger.

BagIt is ontwikkeld met het oog op disk-based storage en het transporteren over een netwerk van digitale content. Van alle bovengenoemde geteste tools, komt de Bagger tool van LoC het beste naar voren. Maar net zoals de bestaande SIP creators de data stiefmoederlijk behandelden, wordt bij een BagIt tool de metadata zeer vlak behandeld. De bag-info.txt kan enkel een vlakke beschrijving bevatten, geen gestructureerde. Dit kan worden opgelost door de BagIt te combineren met METS.

* + 1. METS als kers op de *bag*

METS[[12]](#footnote-12) (Metadata Encoding and Transmission Standard) is een open metadatastandaard om descriptieve, administratieve en structurele metadata van digitale objecten uit een collectie te beschrijven. De voertaal daarbij is XML.

METS combineert de digitale wereld met de wereld van de metadata, doordat een METS document zowel data als metadata kan bevatten, en wel op twee manieren:

* Door het kopiëren van de data in het METS document (mits speciale encodering)
* Door een verwijzing op te nemen naar de bestanden, vaak door middel van hyperlinks

Met het oog op het creëren van containerbestanden is vooral deze eerste methode interessant. Hyperlinks (de tweede methode) kunnen verwijzen naar data die in de toekomt niet meer op de aangeduide plaats aanwezig zou kunnen zijn. De eerste methode, het opnemen van data en metadata in het METS document, maakt dat METS in theorie bruikbaar is als containerbestand, ware het niet dat bestanden op deze manier zeer groot en onhandelbaar kunnen worden. Een bijkomend bezwaar is dat enkel de bestanden daarmee worden opgenomen, en niet de mappenstructuur, waardoor enige structuur verdwijnt. Dit valt weliswaar op te lossen door het gebruik van een structuurmap (structMap).

Een grote troef van METS is dat deze standaard zich leent om de complexiteit van digitale data te structureren. Wanneer men bijvoorbeeld de pagina's van een dichtbundel scant, en men deze wil beschrijven in een document, moet men de volgende zaken bekijken:

* technische metadata van elk bestand (locatie, formaat, checksum[[13]](#footnote-13), type)
* bibliografische metadata (bestaat vaak in meerdere formaten. Uit het document moet ondubbelzinnig blijken dat de metadata handelt over ofwel het geheel van de beschreven bestanden, ofwel een deel ervan.)
* administratieve metadata (al dan niet gekoppeld aan het geheel van de bestanden, of een deel ervan. Bvb auteursrechten)

Er moeten verbanden kunnen worden gelegd tussen de verschillende onderdelen. De computer moet kunnen interpreteren dat een bibliografische beschrijving in het document van toepassing is op pagina 1 en 2 van de dichtbundel, en niet op de rest.

METS lost deze complexiteit op door het probleem te ontbinden in zijn onderdelen, en de linken later te leggen. Een typisch METS-document bestaat uit de volgende “losse” onderdelen:

* Één of meerdere dmdSec (Descriptieve Metadata Section): bevat XML-data met beschrijvende metadata, of verwijst naar een externe locatie. Elke dndSec heeft een identifier.
* Één of meerdere amdSec (Administrative Metadata Section): bevat XML-data met administratieve metadata, of verwijst naar een externe locatie. Elke amdSec heeft een identifier.
* Één fileSec (File Section): bevat één of meerdere fileGrp en file. Elke file heeft een identifier.

Omdat die onderdelen op zich staan, kunnen we meerdere verbanden tussen hen leggen, louter door naar hun identifier te verwijzen. METS gebruikt hiervoor een structMap. Zo kunnen we in structMap 1 van elke ingescande pagina zeggen, welke gedichten er afgebeeld worden, door een file met een dmdSec te koppelen. De structuur van een structMap is vrij.

Daarnaast kunnen checksums van de data worden bijgehouden, maar deze optie is niet verplicht. Verder kan van een METS-document enkel worden gecontroleerd of het een geldige XML is, en of het voldoet aan het schema. Controleren of de bestanden wel degelijk bestaan en of de checksums kloppen, valt daar buiten.

Er bestaan verschillende softwarebibliotheken[[14]](#footnote-14), voornamelijk geschreven in Java: METS Java Toolkit[[15]](#footnote-15), METS API 1.2[[16]](#footnote-16), jpstroop / mets-api[[17]](#footnote-17). 7train METS Generation Tool[[18]](#footnote-18) is een commandline tool om METS-documenten te genereren. Voor meer informatie omtrent de GUI's voor METS, zie website van Library Of Congress[[19]](#footnote-19).

* 1. Batch Rename tools

Naast het opnemen van data en metadata, dient deze SIP-creator software ook nog andere functionaliteiten aan te bieden.

Een eerste daarvan is de mogelijkheid om in bulk bestanden te hernoemen. Zo bekomt men eenvormig benoemde archiefbestanden, wat de digitale hygiëne[[20]](#footnote-20) van dergelijk archiefbestanden aanzienlijk kan vergroten. Voor het hernoemen van bestanden is een “*find & replace*” functionaliteit met een optionele geavanceerde modus het meest gebruiksvriendelijke.

Onderstaand een overzicht van de geteste Batch Rename tools, en hun plus- en minpunten met het oog op de ontwikkeling van een plug & play SIP Creator.

**Batchfilerenametool**[[21]](#footnote-21)

Grafische tool, gemaakt door ontwikkelaars van Google, om een reeks bestanden in een map te hernoemen. De tool is geschreven in Java, en dus beschikbaar voor elk systeem dat Java ondersteunt. Deze tool heeft een gebruiksvriendelijke basisset aan functionaliteiten (prefix en suffix toevoegen, nummering –met optionele voorloopnullen- toevoegen, nieuwe bestandsnaam toekennen). Nadelen zijn dat de plaats van de nummering is vastgelegd (achteraan de bestandsnaam, voor de extensie), en dat er gevaar is voor overschrijving zonder een optie om dergelijke gevallen ongedaan te maken.

**Bulk Rename Utility**[[22]](#footnote-22)

Grafische tool, enkel voor Windows beschikbaar. De tool is zeer populair op het internet door de grote waaier aan opties bij het hernoemen van bestanden. Bestanden kunnen bijvoorbeeld hernoemd worden op basis van reguliere expressies. Maar juist dit groot aantal mogelijkheden kan de modale gebruiker te veel overdonderen. Alle mogelijke opties zijn zichtbaar in het hoofdscherm, waardoor de interface onoverzichtelijk wordt.

**File Renamer Basic**[[23]](#footnote-23)

Grafische tool, enkel voor Windows beschikbaar, en niet gratis ter beschikking. De selectie van bronbestanden gebeurt op basis van reguliere expressies, net zoals de hernoeming. Technische metadata van het bestand kan ook worden opgenomen in de nieuwe bestandsnaam.

**Advanced Renamer**[[24]](#footnote-24)

Grafische tool, enkel voor Windows beschikbaar. De tool is in gebruik een stuk eenvoudiger dan de Bulk Rename Utility. Zo worden geen reguliere expressies gebruikt, maar wel een eenvoudigere syntax, vergelijkbaar met die van RenameWand (zie verder). Technische metadata van het bestand kan ook worden opgenomen in de nieuwe bestandsnaam.

**PFrank**[[25]](#footnote-25)

Grafische tool, enkel voor Windows beschikbaar. De tool is even complex als Bulk Rename Utility, en bevat grosso modo dezelfde functionaliteit, maar heeft een overzichtelijkere interface. Zo worden de meest gebruikte functies in het hoofdscherm gezet, terwijl de geavanceerde opties terug te vinden zijn in het menu.

**RenameWand**[[26]](#footnote-26)

Console applicatie[[27]](#footnote-27), geschreven in Java, en dus geschikt voor alle platformen. De applicatie hanteert een eigen syntax om bestanden te hernoemen, die veel eenvoudiger is dan de syntax van een reguliere expressie. De applicatie selecteert bestanden in een opgegeven map op basis van een *source pattern*, en hernoemt ze op basis van een *target pattern*.

De geteste software bleek in de meeste gevallen te geavanceerd te zijn voor de basismogelijkheden die aangeboden worden binnen de SIP Creator. Daarom werd een op maat gemaakte Rename tool geschreven, weliswaar gebaseerd op de GUI en de functionaliteiten van bovengenoemde.

Van Batchfilerenametool werd de eenvoud in functies overgenomen: prefix, suffix en hernummeren (met voorloopnullen). Bulk Rename Utility heeft dezelfde functionaliteiten, maar eveneens de optie tot een geavanceerde modus (via reguliere expressie). RenameWand biedt de optie om te gaan met overschrijvingen (abort, undo all, ignore). Van PFrank werd de gedachtengang overgenomen om de meest gebruikte opties in het zicht te plaatsen, en geavanceerde te verbergen achter menu’s.

* 1. Wijzigingen in de data verifieerbaar maken

Net zoals fysieke archiefstukken schade kunnen oplopen tijdens transport of bewaring, geldt dat ook voor digitale archiefstukken. Daarom is het noodzakelijk om mechanismen in te bouwen die dergelijke digitale schade kunnen opmerken.

Er bestaan verschillende algoritmes om deze wijzigingen in data op te merken. Deze algoritmes aanvaarden als input binaire data, en geven een 'code' terug. Ze zijn zo gemaakt dat de wijziging van één bit in de data een andere code oplevert. Zo'n code wordt aangemaakt, en meegeleverd met de data bij een digitale overdracht (CDROM, USB, Internet ..). De ontvanger van de data produceert met hetzelfde algoritme de code opnieuw, en vergelijkt die met de meegeleverde code. Deze codes worden daarom ook *checksums* genoemd.

Voorbeelden van deze algoritmes zijn MD5 en SHA-1.

Deze functionaliteit vormt een belangrijk onderdeel van de BagIt standaard (zie boven).[[28]](#footnote-28)

Het is aangeraden om bij elke verplaatsing van de data een dergelijke checksum te berekenen, en ook na archivering regelmatig een checksum uit te voeren op de gearchiveerde bestanden. Indien een bestand schade heeft opgelopen, kan het dan hersteld worden aan de hand van een back up kopie (checksums op zich kunnen niet gebruikt worden om data te herstellen naar zijn originele vorm).

* 1. Grafische User Interface (GUI)

Een laatste belangrijk aspect van de SIP creator is de grafische user interface, een belangrijke voorwaarde om een plug & play software te kunnen aanbieden.

Na het testen van bovengenoemde tools en software, werd Bagger van Library of Congress gekozen als vertrekpunt voor de globale grafische interface. Deze omvat de BagIt van Library of Congress, die alle opties van de standaard ondersteunt. Daarnaast werd BagIt geschreven in Java, en is dus bruikbaar voor elk systeem dat Java ondersteunt.

1. Conclusie

Uit bovengenoemde standaarden, software en tools werden de bouwstenen voor de SIP-creator gehaald. Elke standaard, software en tool bevat interessante ideeën, voordelen en nadelen.

Als basis van het containerformaat werd ervoor gekozen om de BagIt standaard te combineren met het gebruik van METS. Met andere woorden: een bagit met een METS-document er in. Bagit kan beter om met het opslaan van de data en het bijhouden van checksums, terwijl METS beter is in het beschrijven van de metadata en het leggen van de links tussen de metadata en de data.

De Bagger van Library of Congress vormt de basis voor de SIP Creator, maar is méér dan een vertaling ervan. Het is eerder een remake, want het werd grotendeels herschreven. Door gebruik te maken van de Bagit standaard, is het voor de gebruiker mogelijk om wijzigingen in de data te verifiëren, en door gebruik te maken specifiek van de Bagger van LoC, wordt gekozen voor een open standaard, die algemene bruikbaarheid van de containerformaten garandeert.

SIP Creator vormt een uitbreiding op de functionaliteiten van Bagger, doordat SIP Creator ondersteuning biedt voor METS en PREMIS. Hierdoor is het mogelijk om bestandsformaten te identificeren en structuur in de data duidelijk te maken. Daarnaast zit er in de SIP Creator een renamer vervat, die het mogelijk maakt om bestanden in bulk te hernoemen alvorens ze te verpakken tot een bag.

Ook voorziet de SIP Creator een exportmogelijkheid zodat de containerformaten kunnen worden opgeladen naar een repository.

Om het gebruiksgemak zo optimaal mogelijk te maken, worden er voor de gebruiker meerdere mogelijkheden voorzien om metadata toe te voegen aan een dataset:

* Door middel van een XML, die gegenereerd werd uit het registratiesysteem van de archiefinstelling
* Door middel van een CSV, waarvan het AMVB een template voorziet
* Door middel van een invulformulier met DC-velden

Deze metadata wordt vervolgens door de SIP Creator verwerkt tot een METS-bestand, en als tag toegevoegd aan de bag. Zodoende worden data en metadata in één zelfde map bewaard.

1. https://github.com/delving/sip-creator [↑](#footnote-ref-1)
2. http://www.delving.eu/the-delving-platform [↑](#footnote-ref-2)
3. http://www.openarchives.org/pmh/ [↑](#footnote-ref-3)
4. http://europeanalabs.eu/sip-creator/ [↑](#footnote-ref-4)
5. http://www.fedora-commons.org/download/2.2/services/sipcreator/doc/index.html [↑](#footnote-ref-5)
6. http://ufdc.ufl.edu/?m=hesoftware\_mets [↑](#footnote-ref-6)
7. http://www.digitalpreservation.gov/documents/bagitspec.pdf [↑](#footnote-ref-7)
8. http://loc-xferutils.sourceforge.net/ [↑](#footnote-ref-8)
9. Command line applicatie; geen grafische user interface. [↑](#footnote-ref-9)
10. http://sourceforge.net/projects/loc-xferutils/files/loc-bagger/ [↑](#footnote-ref-10)
11. https://github.com/nics/BagKit [↑](#footnote-ref-11)
12. http://www.loc.gov/standards/mets/

    <http://www.loc.gov/standards/mets/METSPrimerRevised.pdf>

    <http://www.projectcest.be/index.php/METS>

    https://biblio.ugent.be/publication/480734 [↑](#footnote-ref-12)
13. Zie verder 2.3 [↑](#footnote-ref-13)
14. http://www.loc.gov/standards/mets/mets-tools.html [↑](#footnote-ref-14)
15. http://hul.harvard.edu/mets/ [↑](#footnote-ref-15)
16. http://sourceforge.net/projects/mets-api/ [↑](#footnote-ref-16)
17. https://github.com/jpstroop/mets-api [↑](#footnote-ref-17)
18. http://sourceforge.net/projects/seventrain [↑](#footnote-ref-18)
19. http://www.loc.gov/standards/mets/mets-tools.html [↑](#footnote-ref-19)
20. Zie publicaties eDavid

    http://www.edavid.be/publicaties.php

    Bijvoorbeeld: http://www.edavid.be/davidproject/teksten/Richtlijn3.pdf [↑](#footnote-ref-20)
21. http://code.google.com/p/batchfilerenametool/ [↑](#footnote-ref-21)
22. http://www.bulkrenameutility.co.uk/Main\_Intro.php [↑](#footnote-ref-22)
23. http://download.cnet.com/File-Renamer-Basic/3000-2248\_4-11753579.html [↑](#footnote-ref-23)
24. http://www.advancedrenamer.com [↑](#footnote-ref-24)
25. http://www3.telus.net/pfrank/index.html [↑](#footnote-ref-25)
26. http://renamewand.sourceforge.net/ [↑](#footnote-ref-26)
27. Command line applicatie; geen grafische user interface. [↑](#footnote-ref-27)
28. Zie 2.1.2 [↑](#footnote-ref-28)