
Arkade5 Documentation

Release 1.0.0

Arkitektum AS

des. 08, 2020

Innhold

1	Innhold	3
1.1	Installasjonsveiledning	3
1.2	Bruerveiledning	7
1.3	Systemdokumentasjon	17
1.4	Versjonshistorikk	22

- **Installer Arkade 5**

4. september 2017 lanserte Arkivverket versjon 1.0 av et nytt testverktøy - «Arkade 5».



Arkade vil etter hvert erstatte alle de testverktøy som Arkivverket har for å teste kvalitet av datasett generert fra arkivsystemer. Et viktig poeng er at Arkade kan teste så å si alle aktuelle varianter av arkivuttrekk. I dag støtter Arkade 5 følgende:

- Fagsystemer
- Noark-3
- Noark 5 versjon 3.1, 4.0 og 5.0

Arkade tester arkivuttrekk, produserer testrapporter og «pakker inn» data basert på gjeldende arkivstandarder for metadata. Testing av data baserer seg primært på Arkivverkets standard [ADDML](#) for beskrivelse av datastrukturer.

Arkade er en frittstående applikasjon som fritt kan lastes ned og enkelt installeres av alle. Programmet er svært enkelt i bruk. I prosjektfasen er det hentet innspill, kommentarer og idéer fra både kommunal og statlig sektor. Ved å senke terskelen for bruk av testverktøy håper vi å forenkle prosessene både hos arkivskapere og arkivinstitusjoner. Blant annet vil feil og avvik kunne oppdages mye tidligere, noe som sparer både tid og kostnader for alle involverte. All kildekode og funksjonalitet i Arkade er lagt ut som åpen kildekode, så det vil også være mulig for eksterne å lage spesialtilpasninger.

Arkivverket vektlegger dessuten forvaltning og videreutvikling av programvaren. Det er allerede en løpende prosess for å legge inn mer funksjonalitet basert på registrerte ønsker under utvikling av versjon 1.0. Det vil også etableres brukerforum og tekniske støttefunksjoner, der man tidligere kanskje ikke har vært gode nok utad. Utvikling av Arkade 5 har krevd ett år med utvikling og testing. Prosjektet har vært styrt av Arkivverket, mens programvaren har vært utviklet eksternt av firmaet [Arkitektum AS](#). Samarbeidet med Arkitektum fortsetter, men forvaltning videre vil være Arkivverkets ansvar.

«Arkade vil utgjøre en enorm tidsbesparelse ved testing av uttrekk, både for arkivskaper og depot.» Solveig Heløe Olsen, Rådgiver, Interkommunalt Arkiv, Troms.



Team Arkade, fra Arkitektum AS (A) og Arkivverket (AV): Fra venstre: Jørgen Tellnes (A), Henning Jensen (A), Joachim Fugleberg (AV), Hallstein Søvik (A), Erik Aaberg (AV), Terje Pettersen Dahl (AV), Tor Oskar Ova Johnsen (A), Jørgen Østengen Vik-Strandli (AV).

Foto: Benjamin Dehli (A)

1.1 Installasjonsveiledning

1.1.1 Arkade 5

Skrivebordsapplikasjon for Windows

Installer .NET

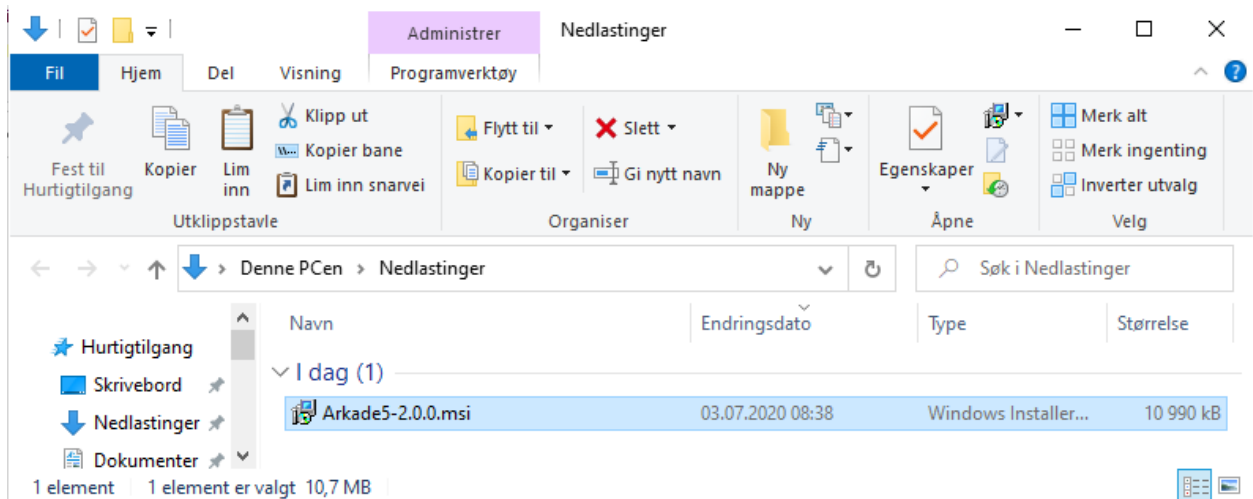
For å kjøre Arkade 5 må .NET Framework (minimum versjon 4.7.2) være installert.

Installer Arkade 5

Last ned siste versjon av Arkade 5 - trykk på knappen:



Start installasjonen ved å dobbeltklikke den nedlastede msi-filen.



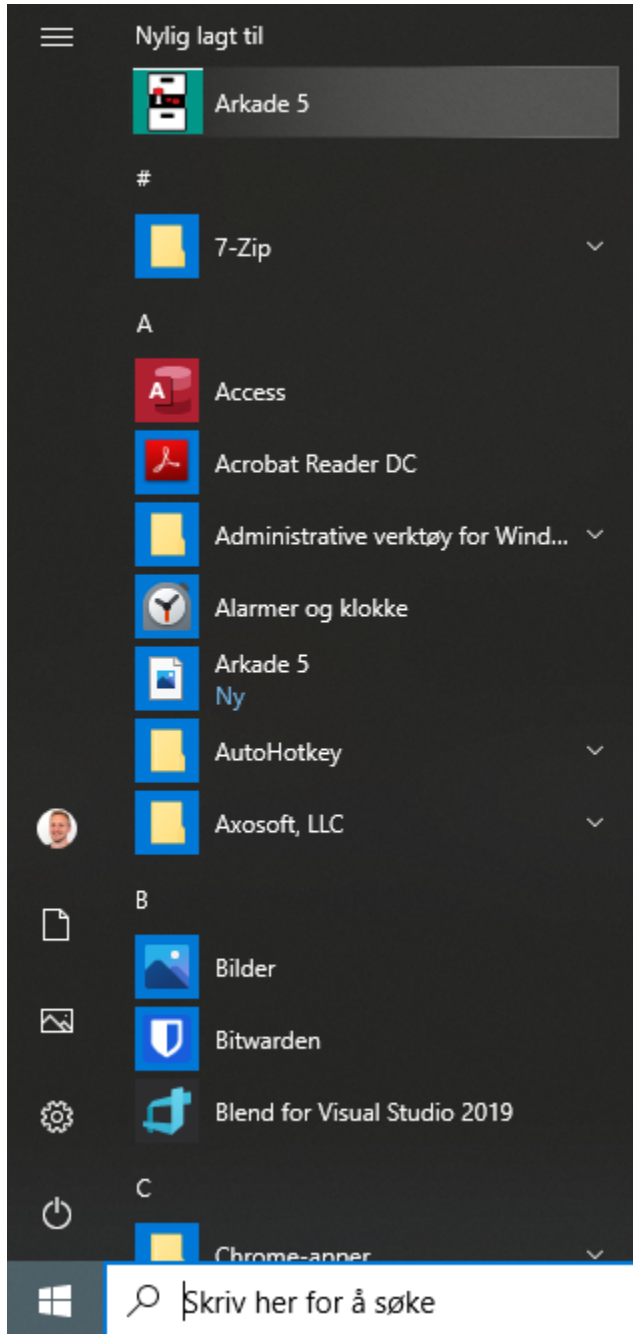
Merk: Windows Smart Screen advarsel Den følgende advarselen vil vises om Windows-maskinen har «Windows Smart Screen» satt på (Windows 10).



- Klikk på «Mer info»
- Klikk «Kjør likevel»

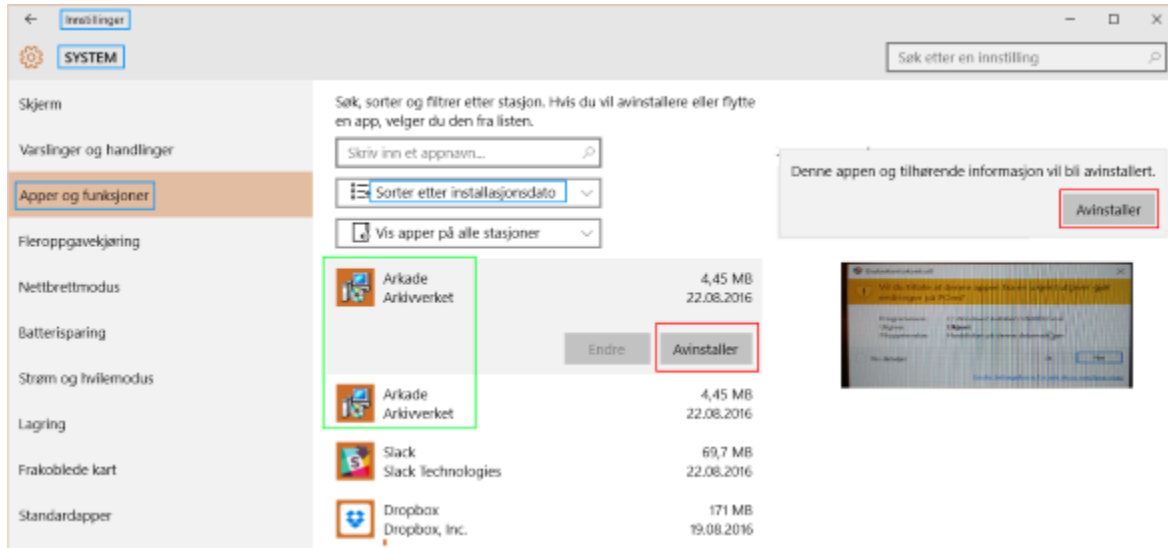
Følg installasjonsveiviseren og aksepter alle de foreslåtte installasjonsvalgene.

Kjør programmet



- Start -> Alle apper
- Finn «Arkade 5» i applikasjons-listen
- Klikk på «Arkade 5» for å kjøre programmet

Avinstallasjon av programmet (Windows 10)



- Klikk Start -> Instillinger -> System -> Apper og funksjoner
- Velg «Sorter etter installasjonsdato»
- Velg «Arkade» i listen over installerte programmer
- Klikk «Avinstaller»
- Klikk «Avinstaller» igjen i nytt vindu
- Klikk «Ja» på Brukerkontroll-advarselen fra Windows

1.1.2 Arkade 5 CLI

Frittstående kommandolinjegrensesnitt for Linux, macOS og Windows

Installer .NET Core

For å kjøre Arkade 5 CLI må .NET Core (minimum versjon 3.0) være installert.

Installer Arkade 5 CLI

Last ned siste versjon av Arkade 5 CLI - trykk på knappen:



Pakk ut den nedlastede zip-filen til der Arkade ønskes kjørt fra.

Det spiller ingen rolle hvor i filsystemet Arkade 5 CLI kjøres fra; plassering for alle inn- og ut-data velges som parametre ved kjøring.

Gjør filen arkade.sh kjørbart:

```
cd Arkade5CLI-<versjon>
chmod +x arkade.sh
```

Ingen ytterligere installasjon er nødvendig.

Kjør Arkade 5 CLI

Se brukerveiledning for Arkade 5 CLI

Avinstaller Arkade 5 CLI

- Slett katalogen Arkade5CLI-<versjon>
- Slett eventuelle gjenværende systemlogger

Mindre enn 1 uke gamle system- og feillogger slettes ikke automatisk etter kjøring. [Les mer ...](#)

1.2 Brukerveiledning

1.2.1 Arkade 5

Skrivebordsapplikasjon for Windows

Arkade 5 brukes ved å lese inn et arkivuttrekk, utføre testing av uttrekket og/eller opprette en arkivpakke av uttrekket. Ved testing genereres det en utfyllende testrapport. Arkade 5 muliggjør også opprettelse/endring av metadata for arkivuttrekk.

Oppstart/avslutning

Ved oppstart av Arkade åpnes innlastingsvinduet. Dersom et tilgjengelig område for midlertidige filer ikke allerede er definert, vil Arkade først be om at dette blir oppgitt (se innstillinger). Arkade avsluttes ved å lukke hovedvinduet. Ved avslutning igangsettes sletting av midlertidige filer.

Innlastingsvinduet

The screenshot shows the 'Arkade 5' application window. The title bar says 'Arkade 5'. The main header is teal with the text 'Arkade 5' and three icons (globe, gear, info) on the right. The main content area is titled 'Last inn arkivuttrekk'. It contains two buttons: 'Velg katalog' and 'Velg SIP/AIP fil'. Below these, it says 'Valgt katalog: C:\N5-arkivuttrekk'. There is a section 'Velg arkivtype' with three radio buttons: 'Fagsystem', 'Noark3', and 'Noark5' (which is selected). At the bottom center is a button 'Last inn uttrekk'. The footer says 'Versjon 2.0.0'.

Innlasting

Velg og last inn arkivuttrekket som skal behandles:

- 1) Klikk på knappen «Velg katalog» dersom uttrekket er en ordinær fil-/mappestruktur. I tilfelle velges katalogen som inneholder arkivbeskrivelse-filen (addml.xml, arkivuttrekk.xml eller NOARKIH.xml). Klikk på knappen «Velg SIP/AIP-fil» dersom uttrekket er en AIP- eller SIP-struktur pakket som en tar-fil.
- 2) Oppgi arkivtype for det valgte uttrekket. Arkade 5 støtter typene «Fagsystem», «Noark 3» og «Noark 5».
- 3) Klikk på knappen «Last inn uttrekk». Det valgte uttrekket vil lastes inn og åpnes i testvinduet.

Testvinduet



Øverst i testvinduet vises:

- Full filsti for det valgte uttrekket
- En unik identifikator (UUID), generert for den gjeldende behandlingen av det valgte uttrekket
- Valgt arkivtype
- Hvilken fil som under testing prosesseres i øyeblikket
- Løpende informasjon om testkjøring*

**For uttrekk av typen Noark 5 vises antall prosesserte XML-elementer. For uttrekk basert på en ADDML-fil vises antall prosesserte filer og antall prosesserte poster.*

Testkjøring

Klikk på knappen «Start testing» for å starte testkjøring på det valgte uttrekket. Testkjøringen vil vare fra noen minutter til mange timer, avhengig av uttrekkets størrelse.

Under testkjøring vil det, i den nedre delen av vinduet, vises meldinger om innlesing, ev. strukturelle feil/mangler ved uttrekket, rapportgenerering og fullført testing.

Testrapport

Etter fullført testing vil en rapport i HTML-format bli generert. Klikk på knappen «Vis rapport» for å åpne den i en nettleser (den vil tilpasse seg gjeldende skjermflate). Ved opprettelse av arkivpakke inkluderes alltid testrapporten. Dersom den (i tillegg) skal tas vare på et annet sted, lagres den ved hjelp av nettleseren. Med rapporten åpen kan dette, i de fleste nettlesere, gjøres ved å taste Ctrl+s.



Testrapport

Testsammendrag

Uuid	d42dae63-e1c8-4f0f-b63e-5d3341fe8266
Arkivskaper	Lewis Carroll, Bob Armstrong
Arkivperiode	1863-03-27 - 1865-12-11
Systemnavn	Test-arkivuttrekk
Systemtype	Sakarkiv (Noark-5)
Arkivtype	Noark5
Tidspunkt for testing	18. oktober 2018
Antall filer som er prosessert	1
Antall avvik funnet	3

Avvik

N5.03 – Validering av XML i henhold til skjema	1
N5.30 – Dokumentfilers sjekksummer	1
N5.32 – Refererte dokumenters eksistens	1

Analyser og kontroller

N5.02 – Validering av sjekksummer

Type: Strukturkontroll

Resultater

Ingen avvik funnet.

N5.03 – Validering av XML i henhold til skjema

Type: Strukturkontroll

Resultater

Lokasjon	Melding
arkivstruktur.xml	Linje 4: Elementet arkiv i navneområdet http://www.arkivverket.no/standards/noark5/arkivstruktur har ugyldig underordnet element systemID i navneområdet http://www.arkivverket.no/standards/noark5/arkivstruktur . Eksempel liste over mulige elementer tittel i navneområdet http://www.arkivverket.no/standards/noark5/arkivstruktur .

Arkivpakkegenerering

Klikk på knappen «Opprett pakke» for å lage en arkivpakke (AIP/SIP) av uttrekket. Dette åpner arkivpakkevinduet der valg for pakken kan gjøres før den opprettes.

Det er mulig å opprette en arkivpakke uten først å utføre testing av det aktuelle arkivuttrekket. En slik pakke vil ikke

inneholde noen testrapport.

Nytt uttrekk / ny kjøring

Ved klikk på knappen «Ny kjøring» avsluttes pågående arkivbehandling og Arkade returnerer til innlastingsvinduet. Dersom det inneværende arkivuttrekket endres, f.eks. som følge av feil/mangler vist i testrapporten, må uttrekket lastes inn på nytt (og ev. tester kjøres på nytt) før knappen «Opprett pakke» oppretter en pakke som inneholder endringene (og knappen «Vis rapport» åpner en gyldig testrapport). Uttrekket lastes inn på nytt ved å klikke «Ny kjøring» (eller ved å starte Arkade på nytt).

NB! Skal det opprettes en arkivpakke som inkluderer resultatene fra inneværende testkjøring, må dette gjøres før «Ny kjøring» klikkes (eller Arkade avsluttes).

Arkivpakkevinduet

The screenshot shows the 'Arkade 5' application window. The title bar says 'Arkade 5'. The main header is green with the text 'Arkade 5' and three icons (globe, gear, info). Below the header, the main content area is titled 'Registrer metadata og opprett pakke'. A note says '* = Feltet er påkrevet for å skape en gyldig arkivpakke'. There is a button 'Hent eksterne metadata'. Below this is a section 'Uttreksdato' with a date input field. Then a section 'Merkelapp' with two radio buttons: 'Standard' (selected) and 'Egendefinert'. Under 'Standard' is a text input 'Gjeldende systemnavn, startår og sluttår'. Under 'Egendefinert' is a text input 'Innlastet eller oppgitt verdi'. Below this is a section 'Velg pakketype' with two radio buttons: 'SIP' (selected) and 'AIP'. There is a checkbox 'Inkluder formatinformasjon for dokumentfiler' with a warning icon. At the bottom are two buttons: 'Opprett pakke' and 'Ny kjøring'. The footer says 'Versjon 2.0.0'.

Før arkivpakke opprettes registreres aktuelle metadata. For å skape en gyldig arkivpakke må påkrevde felter (merket *) utfylles. Dersom det aktuelle arkivuttrekket i utgangspunktet var en arkivpakke (AIP/SIP) med allerede definerte metadata, vil Arkade forsøke å lese disse inn og forhåndsutfylle feltene i arkivpakkevinduet. Ved å klikke knappen «Hent eksterne metadata» kan det velges en fil med forhåndsdefinerte metadata, enten en METS-fil (f.eks. {uuid}.xml,

info.xml eller dias-mets.xml fra tidligere produsert pakke) eller metadata på JSON-format (f.eks. generert med Arkade CLI) for automatisk utfylling av feltene. Når pakken opprettes skrives den utfylte informasjonen til filen dias-mets.xml som legges ved i arkivpakken. Metadataene skrives også til filen {uuid}.xml som legges utenfor, på samme nivå som, arkivpakken.

I nedre del av vinduet velges ønsket pakke type, SIP eller AIP.

Dersom «Inkluder formatinformasjon for dokumentfiler» velges vil Arkade lage en csv-fil med forespurt informasjon og lagre denne på følgende lokasjon i tar-filen: {uuid}\administrative_metadata

Når ønskede metadata er oppgitt, klikkes knappen «Opprett pakke». Dette åpner et dialogvindu for valg av pakkens plassering. Ved valgt plassering opprettes arkivpakken.

Arkivpakken vil opprettes som en tar-fil og filnavnet vil være UUID-en som er generert for den gjeldende arkivbehandlingen: {uuid}.tar. Pakken og tilhørende {uuid}.xml-fil plasseres i en katalog Arkadepakke-{uuid}. Når alt er ferdig generert, vises denne katalogen på den valgte plasseringen.

TIPS: Så lenge arkivpakkevinduet ikke forlates, kan metadata endres og «Opprett pakke» klikkes på nytt. Velges samme pakkeplassering, overskrives foregående pakke og {uuid}.xml med oppdaterte metadata. Sørg bare for at filene som skal overskrives ikke er opptatt, f.eks. ved at de er åpnet i andre programmer.

NB! Knappen «Ny kjøring» avslutter gjeldene arkivbehandling.

Innstillinger

Prosesseringsområde

Under kjøring benytter Arkade et filområde til plassering av midlertidige filer fra arkivprosessering, system- og feillogger samt andre systemfiler. Plassering for prosesseringsområdet velges av bruker som en katalog i filsystemet og må være definert før arkiv kan behandles. Plasseringen som velges må være egnet med tanke på størrelse, tilgjengelighet og personvern. Størrelsen må være minst den av alle uttrekk som skal behandles under samme kjøring i tillegg til plass for systemfiler. Ved avslutning av Arkade igangsettes sletting av midlertidige filer og gamle loggfiler. Plasseringen av prosesseringsområdet kan når som helst endres fra innstillingsvinduet. Ved endring av plassering, igangsettes sletting av opprinnelig prosesseringsområde. Ny plassering vil tas i bruk neste gang Arkade startes.

1.2.2 Arkade 5 CLI

Kommandolinjegrensesnitt for Linux, macOS og Windows

Arkade 5 CLI tester og/eller lager en arkivpakke av et arkivuttrekk. Hva slags type oppgave som skal utføres styres ved hjelp av en kommando. Plasseringer for arkivuttrekket som skal behandles, metadata, ferdig arkivpakke og testrapport samt midlertidige filer og systemlogger, kan alle oppgis som parametre, avhengig av hvilken kommando som skal kjøres.

Se [installasjonsveiledning for Arkade 5 CLI](#)

Kjøring

Arkade CLI kjøres med skriptet `arkade.sh` som medfølger installasjonspakken.

For å kjøre skriptet slik som vist i eksemplene (`./arkade.sh`), naviger til installasjonskatalogen:

```
cd Arkade5CLI-<versjon>
```

(Ved å tilgjengeliggjøre skriptet i `PATH` kan det kjøres med `arkade` fra hvor som helst i filsystemet)

Kommandoer

Arkade CLI krever fra og med versjon 2.0 at det oppgis en kommando for å spesifisere hvilken oppgave som skal utføres.

For en oversikt over tilgjengelige kommandoer og de tilhørende parametrene, bruk `help`:

```
./arkade.sh help
```

```
test      Test archive data in accordance with a specified standard. Run this command followed by '--help' for more
          detailed info.

pack      Pack archive data in accordance with a specified standard. Run this command followed by '--help' for more
          detailed info.

process   Process archive data in accordance with specified standard. Run this command followed by '--help' for more
          detailed info.

generate  Generate a specified file. Run this command followed by '--help' for more detailed info.

help      Display more information on a specific command.

version   Display version information.
```

Parametre

Bruk `--help`, etter en kommando for å vise eksempler på bruksmåte og en oversikt over dens parametre:

```
./arkade.sh generate --help
```

```
Arkivverket.Arkade.CLI 0.0.0.0
USAGE:
Generate json file with metadata example:
  arkade generate --metadata-example --output-directory outputDirectory
Generate text file with list of noark5-test:
  arkade generate --noark5-test-list --output-directory outputDirectory
Generate both files:
  arkade generate --noark5-test-list --metadata-example --output-directory outputDirectory

-m, --metadata-example    (Group: file-type) Generate json file with example metadata.
-l, --noark5-test-list    (Group: file-type) Generate text file with list of noark5 tests.
-o, --output-directory    Required. Directory to place Arkade output files.
--help                    Display this help screen.
--version                  Display version information.
```

Alle parametre foruten `--help` og `--version` kan oppgis i kortform og må etterfølges av aktuell verdi f.eks. `--type Noark5` eller `-t Noark5`

Parametrenes rekkefølge er likegyldig.

NB! Alle kataloger som oppgis som må eksistere på forhånd.

Prosesseringsområde (CLI)

Under kjøring benytter Arkade et filområde til plassering av midlertidige filer fra arkivprosessering, system- og feil-logger samt andre systemfiler. Plassering for prosesseringsområdet velges av bruker som en katalog i filsystemet og må oppgis med parameteret `--processing-area` eller `-p`. Plasseringen som velges må være egnet med tanke på

størrelse, lese-/skrivehastighet og personvern. Størrelsen må være minst den av uttrekket som skal behandles, i tillegg til noen MB for systemfiler. Etter en arkivprosessering vil Arkade igangsette sletting av midlertidige filer og gamle loggfiler. Av tekniske hensyn fjernes ikke automatisk mindre enn 1 uke gamle loggfiler. Vær oppmerksom på at loggfiler kan inneholde personopplysninger. Det anbefales å bruke den samme plasseringen for prosesseringsområdet for hver kjøring. Slik kan Arkade automatisk fjerne loggfiler som er mer enn 1 uke gamle. Sørg ellers for manuelt å fjerne loggfilene.

Eksempel på bruk

Plasseringer og navn på kataloger for inn- og utdata er valgfrie. I eksemplene som følger brukes katalogene tmp/ og output/ i hjemmekatalogen ~/. Opprett de samme katalogene dersom du vil bruke eksempelkommandoene i denne veiledningen slik som de står.

```
mkdir ~/tmp
mkdir ~/output
```

process - Prosessering av arkivuttrekk

Arkivuttrekket som skal prosesseres kan foreligge som en ordinær mappestruktur eller en AIP/SIP-struktur pakket som en tar-fil. Parameterverdien for --archive/-a settes til hhv. katalogen som inneholder arkivbeskrivelse-filen (addml.xml, arkivuttrekk.xml eller NOARKIH.xml) eller til tar-filen.

Under vises et eksempel på hvordan kommandoen **process** brukes. Det leses inn et arkivuttrekk fra katalogen ~/N5-arkivuttrekk/. Arkivuttrekkets type er oppgitt til noark5. *Metadata* for uttrekket leses fra filen ~/metadata.json. Katalog for plassering av prosesseringsområdet er satt til ~/tmp/ og katalog for resulterende data er satt til ~/output/.

```
./arkade.sh process -a ~/N5-arkivuttrekk/ -t noark5 -m ~/metadata.json -p ~/tmp/ -o ~/
↳output/
```

Obligatoriske parametre ved arkivprosessering: --archive--type--metadata-file--processing-area--output-di

Valgbare parametre ved arkivprosessering:

```
--information-package-type/-i - Standardverdi 'SIP'. Kan overstyres til 'AIP'.
--noark5-test-list/-n - Oppgi sti til en fil som inneholder en liste med tester skal kjøres.
--document-file-format-check/-f - Arkade utfører formatsjekk av dokumentfiler og lagrer resultatet i en
csv-fil i tar-pakken.
```

test - Testing av arkivuttrekk

Test om arkivuttrekk er i henhold til en spesifisert standard. Påkrevde parametre er --archive, --type, --processing-area og --output-directory. Dersom det kun skal kjøres et utvalg av tester må en tekstfil som inneholder en liste over disse oppgis med parameteret --noark5-test-list. Arkade CLI kan *lage en fil med et eksempel på en liste over tester*.

```
./arkade.sh test -a ~/N5-arkivuttrekk/ -t noark5 -p ~/tmp/ -o ~/output/ -l ~/n5-
↳testlist.txt
```

pack - Pakking av arkivuttrekk

Lag en arkivpakke. På krevde parametre er `--archive`, `--type`, `--metadata-file` `--processing-area` og `--output-directory`. Standard pakketype er SIP, dette kan endres ved å oppgi 'AIP' til parameteret `--information-package-type`. Det kan også utføres en formatsjekk av filene ved å oppgi parameteret `--document-file-format-check`.

```
./arkade.sh pack -a ~/N5-arkivuttrekk/ -t noark5 -m ~/tmp/metadata.json -p ~/tmp/ -o ~  
↪ /output/ -f
```

generate - Lag en eksempelfil

Kommandoen under lager både en metadata-fil og en testliste-fil i katalogen som er gitt til parameteret `--output-directory`. Filene lagres med standardnavn `arkade-ip-metadata.json` og `noark5-testlist.txt`.

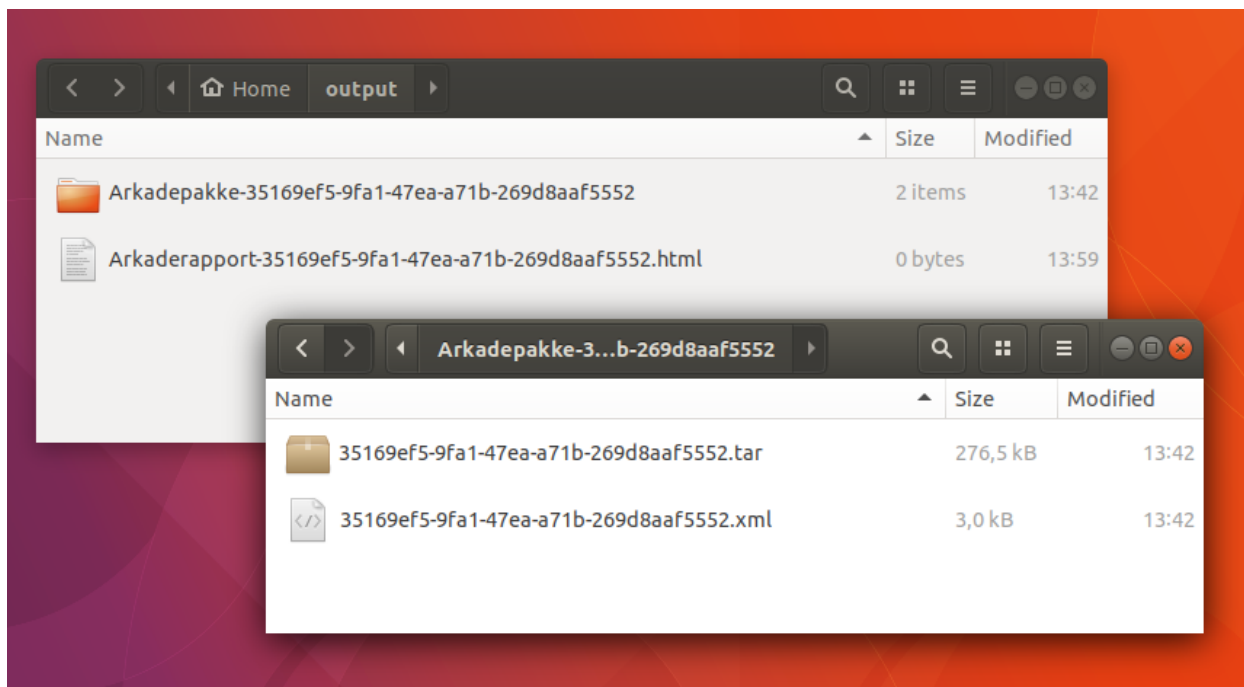
```
./arkade.sh generate -o ~/output/ -m -n
```

Obligatoriske parametre ved filgenerering:

`--metadata-example/-m` *eller* `--noark5-test-list/-n`
`--output-directory`

Resulterende data

process og **pack** kommandoene produserer en arkivpakke som en tar-fil, med tilhørende {uuid}.xml på METS-format, samlet i en katalog. Ved siden av katalogen plasseres en testrapport på HTML-format.



For hver prosessering genereres en unik UUID som bl.a. brukes i fil- og katalognavn for resultatene.

1.3 Systemdokumentasjon

1.3.1 Source code

The source code is located at the GitHub-repository: <https://github.com/arkivverket/arkade5/>

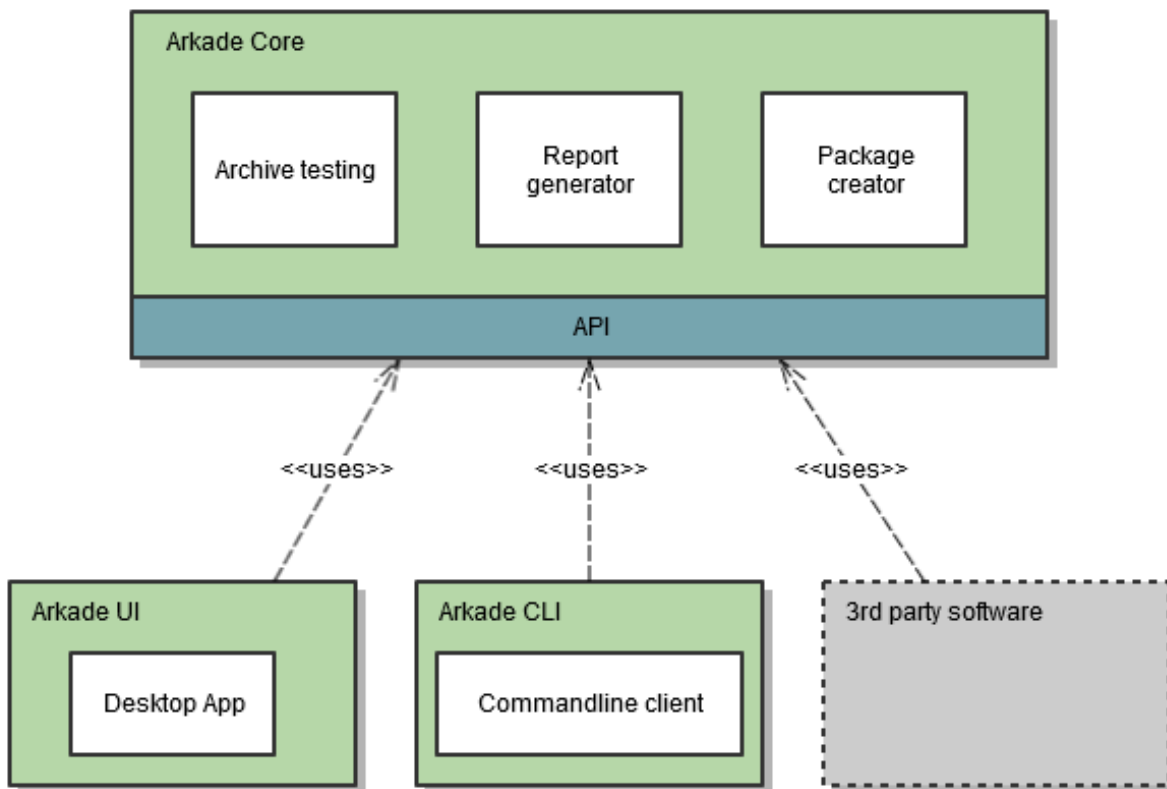
Arkade is developed with .Net and C#. The solution-file (.sln) is compatible with Visual Studio 2015 and above.

Overview

Arkade provides mainly three different functions:

- Archive testing
- Report generator
- Package creator

These functions are exposed in the API and the graphical user interface project is also using the API-class to interact with the core functions.



Below is a brief description of each project in the solution.

Arkivverket.Arkade

This is the core library with functions for testing archive extractions, generating reports and creating SIP/AIP-packages.

The most notable classes in the core project are the test engines, package creator and report generator:

- Arkivverket.Arkade.Core.Noark5.Noark5TestEngine
- Arkivverket.Arkade.Core.Addml.AddmlDatasetTestEngine
- Arkivverket.Arkade.Core.InformationPackageCreator
- Arkivverket.Arkade.Report.HtmlReportGenerator

A short description of the packages in the core project:

Core - Domain classes

ExternalModels - Classes generated from xml schemas

Identify - Identification classes for reading and identifying an archive extraction

Logging - Classes related to logging of events during testing

Metadata - Contains classes related to creating metadata files for archive extractions

Report - Classes for generating test reports

Resource - Various resource files, language files, images etc.

Tests - Contains test classes for testing archive extractions

Util - General utilities

Arkivverket.Arkade.UI

This project provides the graphical user interface of the Arkade 5 software. It is based on WPF, Windows Presentation Foundation. Together with WPF, the application uses the [Prism](#) library for creating a loosely coupled, maintainable and testable XAML application.

[Autofac](#) is used as a dependency framework. Bootstrapping of the applications happens in **Bootstrapper.cs**. It is based on the bootstrapper provided by Prism and it loads the Autofac-module provided by the Arkade core library.

The design and layout is based on Google's [Material](#) Design. This has been implemented with the help of the [MaterialDesignThemes-library](#). Note that the user interface is only inspired by the material design, not necessary strictly following it in every situation.

Arkivverket.Arkade.Test

This project contains the unit tests and other tests classes for the project. Unit tests are created with xUnit.

Setup

This is the setup project for creating installation binaries. You need the [Wix-toolset](#) to be able to use the Setup-project.

Signing the installation file

In order to sign the msi file, you need the **signtool.exe** on your computer. This can be installed together with the Visual Studio. The ClickOnce Publishing package contains this tools.

Signing of the installation file is done by the continuous integration server. Signing is performed with a certificate provided by Arkivverket. The following command is run to sign the installation file:

```
"C:\Program Files (x86)\Windows Kits\8.1\bin\x64\signtool.exe" sign /f PATH_TO_
↪CERTIFICATE_FILE.pfx /p CERTIFICATE_PASSWORD src\Setup\bin\Release\Setup.msi
```

Sample.ConsoleApp

This is a sample application, which demonstrates the use of the Arkade API.

Porting to other platforms

For now the application is only developed for use on the Microsoft Windows platform. However, with the new **.Net Core** project from Microsoft, a cross platform application should be possible when the framework and tools has matured. The .Net Core platform allows running .net applications on linux and mac in addition to windows.

The Arkade project has few external dependencies and it should be possible to either update them to newer versions when they are compatible with .net core or replace them with other compatible libraries. Currently there are only three external libraries in use:

- **SharpZipLib**
- **Autofac**
- **Serilog**

In addition there are some of the .net packages that has been restructured, deprecated or removed that need to be fixed before the application is fully cross platform compatible.

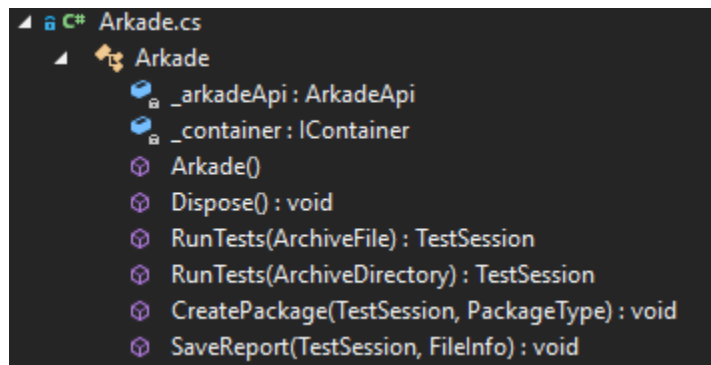
The graphical user interface can be a challenge to port, currently Microsoft has no plans for porting the Windows Presentation Framework to other platforms. This means that for creating a linux desktop app, you might have to recreate the user interface with another graphics library. Preferably a library that works on both linux and mac.

Some useful links regarding porting to .net core: * <https://blogs.msdn.microsoft.com/dotnet/2016/02/10/porting-to-net-core/> * <https://marketplace.visualstudio.com/items?itemName=ConnieYau.NETPortabilityAnalyzer>

1.3.2 Arkade API

The Arkade project provides API-classes for simplified use of the core functionality. There are two API-classes included: **Arkade.cs** and **ArkadeApi.cs**. They are located inside the namespace **Arkivverket.Arkade.Core**. Both classes provides the same functionality, the difference is that **Autofac** is used for dependency injection in the **Arkade** class. The **ArkadeApi** class must be instantiated manually. There is an Autofac module that can be used, **Arkivverket.Arkade.Util.ArkadeAutofacModule**, if the client software already is using Autofac for dependency injection.

This is the signature of the Arkade API class:



```
Arkade.cs
Arkade
  _arkadeApi : ArkadeApi
  _container : IContainer
  Arkade()
  Dispose() : void
  RunTests(ArchiveFile) : TestSession
  RunTests(ArchiveDirectory) : TestSession
  CreatePackage(TestSession, PackageType) : void
  SaveReport(TestSession, FileInfo) : void
```

There are two **RunTests** methods that runs for a given archive, either from a directory structure or a SIP/AIP package file (.tar). After the tests are run, the api returns a **TestSession**. The **TestSession** class contains all necessary information for creating a package with tests results or generating a report.

A simple test run may look like this:

```
var arkade = new Arkade();
var testSession = arkade.RunTests(ArchiveFile.Read("c:\\tmp\\ExampleArchive.tar",
↳ ArchiveType.Noark5));
arkade.SaveReport(testSession, new FileInfo("c:\\tmp\\TestReport.html"));
arkade.CreatePackage(testSession, PackageType.SubmissionInformationPackage);
```

The **TestSession** class contains various information about the testing that has been done. The **TestSuite** property contains a list of all tests that has been run and their results.

See the sample project (Sample.ConsoleApp) in the source code, for a complete example that runs testing on a Noark5 archive.

1.3.3 ADDML

Arkade is built to support ADDML version 8.2.

List of supported ADDML processes:

- A.01 Analyse_CountRecords - Antall poster
- A.02 Analyse_CountChars - Antall tegn
- A.03 Analyse_FindExtremeRecords - Lengste og kortest post
- A.04 Analyse_CountRecordDefinitionOccurences - Antall poster for posttype
- A.05 Analyse_AllFrequenceList - Koder brukt i datasettet
- A.06 Analyse_CrossTable
- A.07 Analyse_CountNULL - Antall null-verdi for felt
- A.08 Analyse_FindExtremeValues - Lengste og korteste verdi for felt
- A.09 Analyse_FindMinMaxValue - Laveste og høyeste verdi for felt
- A.10 Analyse_FrequenceList - Antall forekomster av verdier for felt
- A.11 Control_AllFixedLength - Kontroll av oppgitt lengde for alle posttyper
- A.12 Control_NumberOfRecords - Antall og kontroll av antall poster
- A.13 Control_FixedLength - Kontroll av oppgitt lengde for posttype
- A.14 Control_NotUsedRecordDef - Kontroll om posttype benyttes
- A.15 Control_Key - Kontroll av unik nøkkel
- A.16 Control_ForeignKey - Kontroll av fremmednøkkel
- A.17 Control_MinLength - Kontroll av oppgitt minste lengde
- A.18 Control_MaxLength - Kontroll av oppgitt største lengde
- A.19 Control_DataFormat - Kontroll av oppgitt dataformat for felt
- A.20 Control_NotNull - Kontroll av null-verdier i felt
- A.21 Control_Uniqueness - Kontroll av unike verdier i felt

- A.22 Control_Codes - Koder som benyttes i felt
- A.32 Control_Birthno - Kontroll av fødselsnummer i felt
- A.33 Control_Organisationno - Kontroll av organisasjonsnummer i felt
- A.34 Control_Accountno - Kontroll av kontonummer i felt
- A.35 Control_Date_Value - Kontroll av dato i felt
- A.36 Control_Boolean_Value - Kontroll av boolsk verdi i felt

1.3.4 NOARK 5

Arkade supports the NOARK5 standard.

List of implemented Noark5 Tests:

- N5.02 - Kontroll av sjekk-summene for filene arkivuttrekk.xml og addml.xsd
- N5.03 - Kontroll av sjekk-summene for XML-filene og XML-skjemaene i avleveringspakken
- N5.04 - Antall arkiver i arkivstrukturen
- N5.05 - Antall arkivdeler i arkivstrukturen
- N5.06 - Arkivdelen[e]s status i arkivstrukturen
- N5.07 - Antall klassifikasjonssystemer i arkivstrukturen
- N5.08 - Antall klasser i arkivstrukturen
- N5.09 - Antall klasser uten underklasser eller mapper i det primære klassifikasjonssystemet i arkivstrukturen
- N5.10 - Antall mapper i arkivstrukturen
- N5.11 - Antall mapper for hvert år i arkivstrukturen
- N5.12 - Kontroll på at mappene bare er knyttet til klasser uten underklasser i arkivstrukturen
- N5.13 - Antall mapper som er klassifisert med hver enkelt klasse i arkivstrukturen
- N5.14 - Antall mapper uten undermapper eller registreringer i arkivstrukturen
- N5.15 - Saksmappenens status i arkivstrukturen
- N5.16 - Antall registreringer i arkivstrukturen
- N5.17 - Antall forskjellige journalposttyper i arkivstrukturen
- N5.18 - Antall registreringer for hvert år i arkivstrukturen
- N5.19 - Kontroll på at registreringer bare er knyttet til klasser uten underklasser i arkivstrukturen
- N5.20 - Antall registreringer som er klassifisert med hver enkelt klasse i arkivstrukturen
- N5.21 - Antall registreringer uten dokumentbeskrivelse i arkivstrukturen
- N5.22 - Journalpostenes status i arkivstrukturen
- N5.23 - Antall dokumentbeskrivelser i arkivstrukturen
- N5.24 - Antall dokumentbeskrivelser uten dokumentobjekt i arkivstrukturen
- N5.25 - Dokumentbeskrivelsenens status i arkivstrukturen
- N5.26 - Antall dokumentobjekter i arkivstrukturen
- N5.27 - Start- og sluttdato for dokumentene i arkivstrukturen

- N5.28 - Antall dokumentfiler i arkivuttrekket
- N5.29 - Antall dokumenter i arkivuttrekket fordelt på dokumentformat
- N5.30 - Kontroll av sjekk-summen for hver dokumentfil i arkivuttrekket
- N5.32 - Kontroll på om dokumentobjektene i arkivstrukturen refererer til eksisterende
- N5.33 - Kontroll på at det ikke finnes dokumentfiler i arkivuttrekket som mangler referanse fra dokumentobjektet
- N5.34 - Antall dokumentfiler som blir referert til av flere enn ett dokumentobjekt
- N5.35 - Antall saksparter i arkivstrukturen
- N5.36 - Antall merknader i arkivstrukturen
- N5.37 - Antall kryssreferanser i arkivstrukturen
- N5.38 - Antall presedenser i arkivstrukturen
- N5.39 - Antall korrespondanseparter i arkivstrukturen
- N5.40 - Antall avskrivninger i arkivstrukturen
- N5.41 - Antall dokumentflyter i arkivstrukturen
- N5.42 - Eventuelt - antall skjerminger i arkivstrukturen
- N5.43 - Eventuelt - antall graderinger i arkivstrukturen
- N5.44 - Eventuelt - antall kassasjonsvedtak i arkivstrukturen
- N5.45 - Eventuelt - antall utførte kassasjoner i arkivstrukturen
- N5.46 - Eventuelt - antall konverterte dokumenter i arkivstrukturen
- N5.47 - Kontroll av system-identifikasjonene i arkivstrukturen
- N5.48 - Kontroll av referansene til arkivdel i arkivstrukturen
- N5.51 - Kontroll av referansene til sekundær klassifikasjon i arkivstrukturen
- N5.59 - Antall journalposter i arkivuttrekket
- N5.60 - Start- og sluttdato i arkivuttrekket
- N5.61 - Antall endringer i endringsloggen
- N5.62 - Kontroll av referansene i endringsloggen

1.4 Versjonshistorikk

Se full versjonshistorikk for Arkade 5 på [GitHub](#)