

Sluttrapport: Automatisering av AIP-pakker for Essarch (Arkivspira)

1. Prosjektoppsummering

Dette prosjektet har resultert i utviklingen av en åpen kildekode-applikasjon for automatisert produksjon av informasjonspakker i tråd med DIAS-standard¹. Verktøyet er spesielt optimalisert for Essarch-plattformen, med særlig hensyn til den versjonen som benyttes i KDRS-miljøet per mars 2026. Hovedmålet har vært å effektivisere bevaringsprosessen for digitale arkiv ved å automatisere generering av teknisk dokumentasjon og sikre konsistent validering mot gjeldende krav.

- **Prosjekteier:** IKA Trøndelag
- **Status:** Fullført og klar for operasjonell drift
- **Repository:** <https://github.com/IKATrondelag/dias-package-project>
- **Lisens:** MIT-lisens

2. Bakgrunn og Problemstilling

Prosjektet er forankret i innovasjonsinitiativet **Arkivspira**, et samarbeid mellom IKA Trøndelag (utvikling og test), KDRS (testplattform) og IKA Rogaland (teststøtte). Arbeidet har adressert kritiske flaskehalser i overgangen mellom arkivbegrensning og overføring til langtidslagring.

Følgende seks manuelle flaskehalser ble identifisert og løst:

1. **Gjentatt metadata-utfylling:** Operatører måtte manuelt fylle inn identiske verdier i flere uavhengige systemer og prosessstrinn.
2. **Risiko for inkonsistens:** Manuell overføring av informasjon økte faren for tastefeil og mangelfulle metadata i kritiske felt.
3. **Manglende samlokalisering:** Fysisk eller logisk avstand mellom Essarch-grensesnittet og saksbehandleres arbeidsstasjoner kompliserte dataflyten.
4. **Begrenset skalerbarhet:** Manuelle stoppunkter i pakkeprosessen hindret effektiv håndtering av store volumer og batch-produksjon.
5. **Operatørkompleksitet:** Behovet for å navigere i mange skjermbilder økte opplæringstiden og sårbarheten ved turnover, noe som hindret etablering av en enhetlig "best practice".
6. **Installasjonskompleksitet:** Eksisterende verktøy (ETP og ETA) er basert på eldre teknologi som er utfordrende å installere og vedlikeholde i moderne IT-miljøer.

3. Teknisk Arkitektur og Utviklingsmetodikk

Løsningen er programmert i **Python**, valgt for sitt modne økosystem innen XML-prosessering og filhåndtering. Brukergrensesnittet er bygget med **tkinter** for å sikre en lav brukerterskel. En sentral arkitektonisk beslutning har vært bruken av **YAML-konfigurasjonsfiler**. Dette dekopler institusjonens policy og standardverdier fra selve programkoden, noe som er kritisk for gjenbruk i offentlig sektor. Systemet utfører streng teknisk kontroll gjennom XSD-valideringⁱⁱ og sjekksum-verifiseringⁱⁱⁱ.

KI-integrasjon

Utviklingen har i stor grad nyttiggjort seg KI-verktøy som GitHub Copilot, Claude (Opus/Sonnet) og GPT-Codex. En dedikert instruksjonsfil (copilot-instructions.md) fungerte som en "system prompt" for å styre kodegenereringen mot DIAS-spesifikasjonen og strenge krav til feilhåndtering. Dette sikret at komplekse XML-generatorer ble utviklet med høy presisjon og i tråd med arkivfaglige standarder.

4. Kjernefunksjonalitet

Applikasjonen automatiserer dannelsen av både SIP^{iv} og AIP^v, og reduserer tidsbruken per pakkeoperasjon betydelig.

Funksjon	Beskrivelse
XML-generering	Automatisk produksjon av METS, PREMIS og info.xml basert på GUI og YAML-maler.
Sjekksum-beregning	Generering av SHA-256 sjekksummer for enkeltfiler og ferdige tar-arkiver.
Validering	Maskinell kontroll mot DIAS XSD-skjemaer og interne integritetsregler.
Tar-pakking	Opprettelse av pakkestruktur med sjekksummer i to lag for optimal integritet.
Mappe-håndtering	Automatisk bygging av korrekt kataloghierarki i henhold til DIAS-spesifikasjonen.

5. Testresultater og Distribusjon

Det er gjennomført et omfattende testregime for å verifisere logikk, struktur og skalerbarhet.

Testkategori	Beskrivelse	Antall	Resultat
Enhetstester	Verifisering av metadata-mapping og XML-generatorer.	100	100 Passert
Integrasjonstester	Kontroll av pakkestruktur, sjekksummer og XSD-validering.	82	81 Passert / 1 Hoppet over
Totalt		182	99,5% Suksessrate

Løsningen distribueres som kjørbare filer for Windows (.exe via GitHub Actions), macOS (.app) og Linux (XDG-standard). Dette eliminerer behovet for lokal Python-installasjon hos sluttbruker. Støtte for macOS er inkludert i arkitekturen, men er foreløpig teknisk uprøvd^{vi}.

6. Faglig Diskusjon og Arkivfaglig Kontekst

Overgangen fra statiske pakker til en dynamisk modell representerer et betydelig arkivfaglig fremskritt for kommunal sektor. Ved å støtte Archival Information Collections (AIC) som kan kobles til flere SIP/AIP-pakker, legger verktøyet til rette for en mer fleksibel og nøyaktig representasjon av arkivets utvikling.

Viktigheten av relasjonsfelter: Ved bruk av pakketyperne "Supplement" og "Replacement" sørger applikasjonen for at relasjonsfelter og feltet "Record Status" i PREMIS og METS blir korrekt populert og validert mot AIC. Dette sikrer at koblingen mellom opprinnelige uttrekk og etterfølgende tillegg bevares i depotet.

Integritet vs. Sletting: Å erstatte gamle pakker gjennom sletting svekker arkivets integritet og proveniens. Denne løsningen fremmer derfor bruk av relasjonelle metadata for å bevare kontekst over tid. Det anbefales at institusjoner avklarer støtte for utvidede pakketyper med sin plattformtilbyder før fullskala implementering.

7. Operasjonelle Anbefalinger og Driftsrutiner

For stabil drift av løsningen anbefales følgende daglige rutiner:

- **Logganalyse:** Systematisk kontroll av applikasjonslogger ved bruk av **Correlation IDs** for å spore feil på tvers av prosesser.
- **Diskkontroll:** Overvåking av ledig lagringsplass, spesielt ved produksjon av store pakker (500GB+).
- **Integritetsverifisering:** Regelmessig stikkprøvekontroll av **JSON-kvitteringer** mot de faktiske filene på eksportområdet.
- **Opprydding:** Periodisk sletting av temporære filer i samsvar med lokale sikkerhetskopierings- og sletterutiner.

8. Økonomisk Oversikt

Prosjektet er gjennomført innenfor effektive rammer ved bruk av spesialistkompetanse.

Post	Ressursbruk	Kostnad (NOK)
Utviklingskostnad (totalt)	150 timer à 900,-	135 000,-
Anslåtte årlige driftskostnader	Support, backup og vedlikehold	5 000 – 10 000,-

9. Veien Videre og Prioriteringer

For å sikre løsningens relevans og nasjonal interoperabilitet anbefales følgende prioriteringer:

1. **Høy prioritet:** Implementere automatisert kodesignering av Windows-artefakter i CI/CD-løpet for å fjerne sikkerhetsvarsler ved installasjon^{vii}.
2. **Middels prioritet:** Utvide METS-støtten til å dekke samtlige metadatafelter spesifisert i **Arkade 5** (Digitalarkivets standard), som er et viktig steg mot nasjonal samordning.
3. **Middels prioritet:** Utvikle direkte integrasjon mot **Essarch API** for å eliminere manuelle mellomsteg i overføringen.
4. **Lav prioritet:** Utvikling av et webbasert grensesnitt (Web-UI) for sentralisert styring av pakkeprosesser i større organisasjoner.

Teknisk dokumentasjon for videreutvikling og drift (inkludert HANDOFF.md, CONFIG_GUIDE.md og OPERATIONS.md) er tilgjengelig i prosjektets repository.

10. Referanser og Standarder

- **DIAS: Digital Arkivpakke** Struktur er en felles norsk standard for pakkestruktur for bevaring av digitalt arkivmateriale med utgangspunkt i OAIS-standarden (ISO 14721:2003).
- **KDRS:** Kommunearkivinstitusjonenes Digitale Ressurssenter.
- **METS:** Metadata Encoding and Transmission Standard spesifiserer de overordnede elementene i en [OAIS-basert arkivpakke](#): organisering, struktur og innholdsoversikt.
- **PREMIS:** Preservation Metadata: Implementation Strategies (Vedlikeholdt av Library of Congress) beskriver metadata om de operasjonene en arkivpakke gjennomgår i et arkivdepot. Dette gjør det mulig å dokumentere håndteringshistorikken for materialet i ettertid.
- **Essarch, ETA og ETP:** Programvare for langtidsbevaring fra ESSolutions <https://github.com/ESSolutions>

Fredrik Johnsen, IKA Trøndelag, 11.03.2026

ⁱ DIAS (Digital Arkivpakkestruktur) er den tekniske standarden for informasjonsforvaltning som applikasjonen er bygget for å etterleve.

ⁱⁱ XSD-validering (XML Schema Definition) sikrer at de genererte metadatafilene følger de strengt definerte strukturelle kravene i arkivskjemaene.

ⁱⁱⁱ SHA-256 sjekksummer brukes for å garantere filintegritet og oppdage utilsiktede endringer i data under transport eller lagring.

^{iv} SIP (Submission Information Package) representerer informasjonspakken slik den sendes fra arkivskaper til depotet.

^v AIP (Archival Information Package) er den endelige pakken som lagres for langtidsbevaring i arkivdepotet

^{vi} **Støtte for macOS:** Selv om applikasjonen kan kompileres som en .app-bundle, er funksjonaliteten per nå uprøvd da testmiljøet kun har hatt tilgang til Windows og Linux.

^{vii} **Signering av artefakter:** Automatisert signering er kritisk for sømløs distribusjon i kontrollerte IT-miljøer.