# Autonomous Preservation Tools in Minimal Effort Ingest

Asger Askov Blekinge
State and University Library, Denmark
+45 8946 2100
abr@statsbiblioteket.dk

Bolette Ammitzbøll Jurik
State and University Library, Denmark
+45 8946 2322
baj@statsbiblioteket.dk

Thorbjørn Ravn Andersen
State and University Library, Denmark
+45 8946 2317
tra@statsbiblioteket.dk

## ABSTRACT

This poster presents the concept of *Autonomous Preservation Tools*, as developed by the State and University Library, Denmark. The work expands the idea of *Minimal Effort Ingest*, where most preservation actions such as Quality Assurance and enrichment of the digital objects are performed after content is ingested for preservation, rather than before. We present our Newspaper Digitisation Project as a case-study of real-world implementations of Autonomous Preservation Tools.

### Keywords

Long term Preservation, Object Repository, Minimal Effort Ingest, Autonomous Preservation Tools, Tree Iterator, Preservation Actions, Quality Assurance, OAIS

## 1. INTRODUCTION

The State and University Library, Denmark, would like to present an expansion of the Minimal Effort Ingest model [1, 2]. In Minimal Effort Ingest most of the preservation actions are postponed and handled within the repository, when resources are available. We suggest organising these actions using *Autonomous Preservation Tools* – similar to software agents – rather than a static workflow system. This adds flexibility to the repository, as it allows for easy updates, removal or addition of workflow steps. We present the Danish newspaper digitisation project as a case study of Autonomous Preservation Tools. Firstly we introduce the concepts of Minimal Effort Ingest (section 2) and Autonomous Preservation Tools (section 3). We then present the newspaper digitization project (section 4), and how these concepts have been implemented.

## 2. MINIMAL EFFORT INGEST

When ingesting a collection into an object repository for long term preservation it is common to follow the OAIS reference model [3]. In OAIS quality assurance (QA) and enrichments are performed on the submission information package (SIP) before this is ingested into the repository. The Minimal Effort Ingest [1, 2] idea builds on OAIS, but performs the QA and enrichment actions on the archival information package (AIP) inside the repository. The aim is to secure the incoming data quickly, even when resources are sparse.

## 3. AUTONOMOUS PRESERVATION TOOLS

In Minimal Effort Ingest preservation actions should not be orchestrated by a static ingest workflow, but rather be carried out *when resources are available*. From this concept, we developed the idea of Autonomous Preservation Tools.

We use the OAIS term Archive Information Package (AIP) to denote an object stored in a preservation system. We define Preservation Tools as tools that can be used on such an AIP. The implementation of such a tool is very dependent both on the preservation system and the format of the AIP. An AIP could be anything from a simple file to a complex container format or interlinked structure.

An Autonomous Preservation Tool is an extension of a normal preservation tool. Traditionally preservation tools are explicitly invoked on an AIP as part of a static workflow. Autonomous Preservation Tools can discover AIPs to process on their own.

We further assume that AIPs maintain an account of past events. In Digital Preservation such an account can be important for showing data authenticity and provenance, so many repository systems implement this already. From this account the Autonomous Preservation Tool can determine whether it has already processed an AIP or not.

The order in which the Autonomous Preservation Tools process an AIP can be important, such as whether checksums were generated before or after additional metadata was added. Thus, Autonomous Preservation Tools must be able to find the AIPs they have not yet processed, but which have already been processed by specific other Autonomous Preservation Tools.

With Autonomous Preservation Tools the need for a fixed workflow disappears and is replaced with a decentralised implicit workflow. Rather than defining a fixed sequence of steps an AIP must go through, you define the set of events that an AIP must have experienced. Each Autonomous Preservation Tool corresponds to one such event, which it sets when it processes the AIP. The workflow will be a number of Autonomous Preservation Tools each looking for AIPs to process, until each tool has processed every AIP.

This approach brings a great deal of flexibility:

- Removing an Autonomous Preservation Tool is a local operation. No other Autonomous Preservation Tools or workflows will be affected.
- When a new Autonomous Preservation Tool is added, it will automatically start processing old AIPs as well as new. No migration steps are needed.

- When an Autonomous Preservation Tool is changed, it can be marked as a new Autonomous Preservation Tool, and thus start processing all previously processed AIPs. Alternatively, the tool can be configured to continue where it left off, and thus only process new AIPs.

## 4. CASE STUDY

The Danish newspaper digitisation project [1] is an in-production example of Minimal Effort Ingest using Autonomous Preservation Tools. In this project we receive scanned newspaper pages in batches of about 25,000 pages along with MIX[2], MODS[3] and ALTO[4] metadata. We receive two batches a day and a total of about 30 million newspaper pages throughout the duration of the project. All ingest, validation and enrichment preservation actions are performed with the Autonomous Components described in section 4.2.

### 4.1 Repository

Each new batch of scanned newspaper pages must be ingested in our repository system, undergo a large number of quality checks and have access copies generated.

In keeping with the Minimal Effort Ingest model, we first ingest the batch of pages, and then perform the quality checks. Metadata is stored in DOMS, our Fedora Commons[5] 3.x based repository, whereas the data files are stored in our BitRepository[6]. We use Solr[7] to index the content of DOMS for our discovery platforms.

We store an additional object in DOMS, the batch object, which represents the batch of scanned pages, rather than any single page. In this object, we store PREMIS[8] Events detailing which actions and transformations have been performed on the batch. This information is also indexed by Solr.

### 4.2 Autonomous Components

We implemented the Autonomous Preservation Tool model in what we call Autonomous Components. Each component corresponds to a single action, such as “Ingest batch into repository” or “Schema-validate all XML files in batch”.

All autonomous components are Linux executables and have the following characteristics:

- Can query Solr for batch objects having specific combinations of PREMIS Events.

- Registers a component-specific PREMIS Event on the batch object after execution.

The current location of a batch in the workflow is determined by the set of PREMIS events present on the batch object - in other words which components have processed the batch so far. Each component knows which PREMIS events must be present or absent on a given batch for it to be ready to be processed by the component.

We have created *Tree Iterators* as a framework for autonomous components to handle batches in a storage-agnostic way. Tree iterators allow you to iterate through complex directory structures, whether in the repository or on disk, in a uniform way. With this framework, the autonomous components are able to work identically on batches not yet ingested, and batches inside the repository. This gives us great flexibility when testing, and allows us to easily rearrange which components should be run before ingest, and which should be run after.

## 5. CONCLUSIONS

We have shown that Autonomous Preservation Tools can be considered a viable alternative to a static workflow. We believe that Autonomous Preservation Tools should become a standard part of the digital preservationist’s toolbox, especially when using Minimal Effort Ingest.

## 6. FURTHER WORK

The State and University Library is currently in the process of replacing our Fedora based metadata repository. This will require a number of components to be reimplemented but we remain dedicated to the Minimal Effort Ingest and Autonomous Preservation Tools concepts.

During 2016 we will begin a project of receiving Danish born-digital newspapers. The principles described here will be carried further in this project.

## 7. ACKNOWLEDGEMENTS

We would like to thank our colleagues Kåre Fieldler Christiansen, Jette G. Junge and Gitte Bomholt Christensen for their help in developing both these concepts and this poster.

## 8. REFERENCES

[1] B. A. Jurik, A. A. Blekinge, and K. F. Christiansen. Minimal effort ingest. In Lee et al. [2].

[2] C. Lee, E. Zierau, K. Woods, H. Tibbo, M. Pennock, Y. Maeda, N. McGovern, L. Konstantelos, and J. Crabtree, editors. *Proceedings of the 12th International Conference on Digital Preservation.* School of Information and Library Science, University of North Carolina at Chapel Hill, 2015.

[3] Space Data and Information Transfer Systems. *ISO 14721:2012 Open Archival Information System (OAIS) - Reference Model.* The International Organization of Standardization, 2012.

[1]http://en.statsbiblioteket.dk/national-library-division/newspaper-digitisation/newspaper-digitization
[2]https://www.loc.gov/standards/mix/
[3]https://www.loc.gov/standards/mods/
[4]https://www.loc.gov/standards/alto/
[5]http://fedorarepository.org/
[6]http://bitrepository.org/
[7]http://lucene.apache.org/solr/
[8]http://www.loc.gov/standards/premis/