Implementing Automatic Digital Preservation for a Mass Digitization Workflow

Henrike Berthold
Saxon State and University Library
Dresden, Germany
+49 351 4677240
henrike.berthold@slub-dresden.de

Andreas Romeyke
Saxon State and University Library
Dresden, Germany
+49 351 4677216
andreas.romeyke@slub-dresden.de

Jörg Sachse
Saxon State and University Library
Dresden, Germany
+49 351 4677216
joerg.sachse@slub-dresden.de

Stefan Fritzsche
Technische Universität Dresden, Germany
+49 351 46333212
stefan.fritzsche@tu-dresden.de

Sabine Krug
Saxon State and University Library
Dresden, Germany
+49 351 4677232
sabine.krug@slub-dresden.de

ABSTRACT

The Saxon State and University Library Dresden (SLUB) has built up its digital preservation system SLUBArchiv from 2012 to 2014. In January 2015, we launched the preservation workflow for digitized documents. This workflow extends the in-house mass digitization workflow, which is based on the software Kitodo.Production. In this paper, we describe the three major challenges we faced while extending our mass digitization workflow with an automatic preparation and ingest into our digital long-term preservation system and the solutions we found. These challenges have been

(1) validating and checking not only the target file format of the scanning process but also the constraints to it,

(2) handling updates of digital documents that have already been submitted to the digital long-term preservation system, and

(3) checking the integrity of the archived data as a whole in a comprehensive but affordable fashion.

Keywords

Digital preservation, Preservation strategies and workflows, Case studies and best practice, file formats, updates of archival information packages, bit-stream preservation, Saxon State and University Library Dresden, SLUB, SLUBArchiv

1. INTRODUCTION

SLUB has been digitizing its documents since 2007. The Dresden Digitization Center at SLUB is one of Germany's leading centers of mass digitization in the public sector. It produces 2 to 3 million scans a year. In addition, service providers digitize collections of other institutions as part of a digitization program of the federal state of Saxony in Germany. The software Kitodo.Production manages the digitization workflow i.e. the scanning process, the enrichment with structural and descriptive metadata and the export to the catalogue and to the digital collections presentation.

To preserve the resulting digital documents, SLUB has built up the digital preservation system SLUBArchiv. SLUBArchiv is based on the extendable preservation software Rosetta by ExLibris Corp. and complemented by a submission application for pre-ingest processing, an access application that prepares the preserved master data for reuse, and a storage layer that ensures the existence of three redundant copies of the data in the permanent storage and a backup of data in the processing and operational storage. Rosetta itself has been customized to SLUB's needs e.g. by plugins that have been developed in-house.

To complete SLUB's new digitization workflow (see Figure 1), an automatic pre-validation of the produced images has been added to check early if the requirements to the scanned files are met.

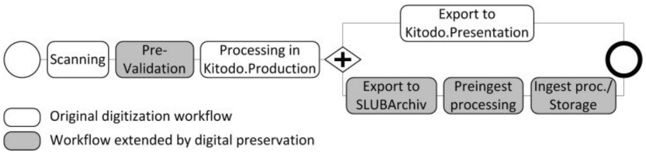
Scanning
Pre-Validation
Processing in Kitodo.Production
Export to Kitodo.Presentation
Export to SLUBArchiv
Preingest processing
Ingest proc./ Storage
Original digitization workflow
Workflow extended by digital preservation

Figure 1: Digitization workflow

In January 2015, the digitization workflow with digital preservation went live. In June 2015, SLUBArchiv has received the Data Seal of Approval [4].

During the development of the digital preservation for the digitization workflow, we faced a number of challenges. In this paper in section 3, we describe the three major challenges. The solutions are outlined in section 4. In section 2, we describe the mass digitization workflow in more detail, which is the setting for our challenges and solutions. In the last section, we summarize the future development goals.

2. MASS DIGITIZATION WORKFLOW

In SLUB's Digitization Center, a record is created in Kitodo.Production for each print document to be scanned. This record represents the digital document that corresponds to the print document. The document is then scanned on the appropriate scan device or, depending on the contents, possibly also on different scan devices (e.g. an enclosed map is scanned on a device that can handle large formats). All scans of a document are stored in the directory assigned to the digital document in Kitodo. When the scanning step is finished, checksums for all files are calculated and the processing starts. Descriptive metadata of the original print document are taken from the local or a remote library catalogue. Further descriptive and structural metadata are added. Finally, when the processing step is completed, the presentation data (in JPEG format) are exported to Kitodo.Presentation and the preservation data are exported to a transfer directory. The master data consist of one or more METS/MODS metadata files, zero or multiple ALTO XML files, and one or more master scans in the TIFF format. The steps of the workflow are visualized in Figure 1.

The transfer to the SLUBArchiv happens asynchronously. The submission application (i.e. our pre-ingest software) scans the transfer directory and processes the data of each newly arrived digital document. It checks completeness and integrity, transforms metadata from METS/MODS to METS/DC and converts the data to a SIP that is accepted by the software Rosetta. It then uses a Rosetta web service to initiate the ingest processing. During SIP processing, completeness and integrity is checked again. A plugin performs a virus check. The data format of each file is identified, validated and technical metadata are extracted. If SIP processing is successful, an AIP is built and stored in the permanent storage. The storage layer creates two more copies and manages them. The storage media used are hard disks in disk-based storage systems and LTO tapes in tape libraries. The archiving workflow is shown in Figure 2. It is fully automated.

Although the process seems to be simple, we faced a number of challenges. The most important challenges are described in detail in the next section.

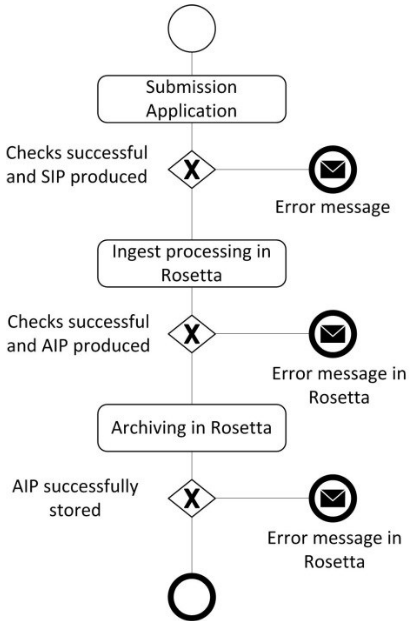

Figure 2. Archiving workflow

3. CHALLENGES

3.1 File Format Specification with Constraints

The specification of the data format TIFF 6.0 [1] specifies baseline TIFF and a number of extensions. During the ingest processing, the data format of each file is identified and the file is validated (i.e. it is checked whether the file is correct regarding to the file format specification). However, we have additional requirements regarding the digitized data. They have to be compliant with the guidelines specified by the Deutsche Forschungsgemeinschaft [2]. These guidelines specify values for technical parameters such as resolution or color depth. In addition, we have requirements that are important for digital long-term preservation [3]. One example is compression, which is allowed in TIFF 6.0 and encoded in tag 259 (“Compression”). So, to ensure robustness, we only accept uncompressed TIFF. Another example is the multipage feature, which allows for embedding multiple images in a single file (tag 297, “PageNumber”). To ensure that the metadata correspond to the image, we only allow one-page TIFF files.

3.2 Updates of Archival Information Packages

In contrast to the assumption that archived digital documents do not change, we have to deal with updates of digital documents. Reasons are manifold; some of them are:

- metadata are extended by a specific collection, e.g. due to an indexing project,
- a new representation is added, e.g. geo-referencing metadata,
- an image master has to be added or replaced, e.g. due to an error in the initial scanning process.

If a digital documents need to be changed, it is re-activated in Kitodo.Production. If the master data need to be changed, they are retrieved from Rosetta and reloaded into Kitodo. The digital document is then updated and exported again to Kitodo.Presentation and the SLUBArchiv.

3.3 Checking the Bit Stream of a Large Archive

SLUBArchiv currently preserves digital documents with a data volume of approx. 40TB. By the end of this year, the data volume will be approx. 100TB. We manage three copies of the data. Therefore, the total data volume is currently 120 TB, in 12/2016 it will be approx. 300TB. The storage layer of SLUBArchiv uses hierarchical storage management, in which large files are stored on tapes. An integrity check of all digital documents (and their three copies) is not feasible due to the time that is required to read all data from tape storage and check them. The estimated time effort is in the range of weeks. Since the complete restore cannot be done at once (because the disk storage cannot be held available for this amount of data), it would have to be organized in an incremental process. Such a process would stress the tapes. Therefore, we need a method to get reliable results without checking all data in the archive.

4. SOLUTIONS

4.1 File Format Specification with Constraints

Based on the code library libtiff [5], we have implemented the open-source tool checkit-tiff [6], which takes a human-readable configuration file and checks a tiff file against the specified configuration. It contains one rule per line (see example below). Each rule line has three entries:

- the ID number of a tiff tag,
- a string that specifies if the tag must be encoded in the file (“mandatory” or “optional”), and
- the feasible values as a single value “only(*value*)”, a range “range(*from_value, to_value*)”, a regular expression and some more options (see [6] for a complete list)

```
# Compression is not allowed
259; mandatory; only(1)
# XResolution
282; mandatory; range(300, 1200)
# YResolution
283; mandatory; range(300, 1200)
#   Make   i.e.   name   of   the   scanner
manufacturer
271; optional; regex("^[[:print:]]*$")
#PageNumber
297; optional; only(0,1)
```

Currently, we automatically check the files after the scanning step. If one or more files are not correct regarding the requirements, the digitization center or the external digitization service provider re-scans the files, replaces them in the directory of the digital document and the check is run again. This is to make sure that Kitodo processing only starts if all files are correct. Currently, the error rate is about 6%.

4.2 Updates of Archival Information Packages

We use the workflow software Kitodo.Production to produce and also to update digital documents. Hence, a specific document can be transferred multiple times to the SLUBArchiv – the first time is an ingest, all transfers after that are updates.

The transfer is done asynchronously. After the processing in Kitodo.Production, the files that belong to a document and that need to be preserved (i.e. master scan files, one or more METS/MODS files, ALTO XML files) are copied to a folder in the transfer directory.

The software Rosetta supports updates. It offers specific update functions through its ingest web service API. Rosetta can manage multiple versions of an AIP and creates a new version each time an AIP is updated. Older versions of digital objects remain stored and accessible for staff users.

The submission application (which takes the data that belong to a digital document and prepares a SIP) has to distinguish an initial ingest from an update. It uses the identifier of the digital document to check whether a digital document is currently processed by Rosetta or already archived. If the check is successful, it uses the update function, otherwise it uses the "normal" ingest function.

The Rosetta web service API provides functions to add a file, delete a file and replace a file. Using the checksums of the files, the submission application derives which file has been added, deleted or updated and applies the corresponding web service functions.

Currently, we are re-implementing the transfer from Kitodo.Production to the SLUBArchiv. We will make sure that all versions of a digital document that are copied to the transfer directory are archived. Since Kitodo.Production has no built-in versioning, we use the time of the export from Kitodo as our ordering criterion.

4.3 Checking the Bit Stream of a Large Archive

Each AIP is stored in three physical copies in two different locations, both of which are equipped with disk and tape storage systems.

Each AIP is stored in two storage pools - a primary and a secondary storage pool - of a clustered file system (IBM General Parallel File System, GPFS). The two storage pools are located at different locations. In these storage pools, large files are migrated to tape with IBM's Hierarchical Storage Management (HSM), which is an extension of the IBM Tivoli Storage Manager (TSM) software. The third copy of an AIP is a backup copy. TSM is used as backup software. Backup copies of new data in the GPFS-based permanent storage are made regularly (currently three times in 24 hours) and written to tape every day. All tape pools (i.e. HSM and backup tape pools) are protected by Logical Block Protection (LBP, a CRC checksum technology).

The integrity of archival copies is checked using two different methods.

4.3.1 Sample Method

Integrity of archival copies is checked yearly for a 1%-sample of all files. The sample of AIPs is produced automatically. Using Rosetta, a list of all files that belong to the selected AIPs is produced. The integrity of all three copies of these files is then checked automatically in the storage system. If an error is detected, corrupt files are replaced by correct copies and AIPs that are produced or updated on the same day are checked as well. Due to the applied storage policy, AIPs that are ingested on the same day are located in the same storage area. Depending on the results, the check is extended to a longer time period in which AIPs are stored in permanent storage. We have executed this check once. No integrity failures were found.

4.3.2 Pattern Method

The directory structure of the permanent storage can be controlled in Rosetta using a storage plugin. We have implemented a storage plugin that stores all files of an AIP in a single directory. These AIP directories are structured according to the year, month and day of the ingest. A file with a specified fixed bit pattern is stored daily in the directory of that specific day in the storage system. All these pattern-files are checked quarterly. Due to the specified bit pattern, single and multiple bit failures can be detected. If an error is identified, the data that are produced the same day are checked. Depending on the results, the check is extended to a longer time period in which AIPs are stored in permanent storage. We have executed this check once. No integrity failures were found.

5. CURRENT CHALLENGES

The SLUB Archive is developing towards new media types (digital video, audio, photographs and pdf documents), unified pre-ingest processing, and automation of processes (e.g. to perform tests of new software versions). Additionally, we currently conduct a pilot project of a digital preservation service for another Saxon institution.

6. REFERENCES

[1] Adobe Developers Association. 1992. *TIFF revision 6.0*. http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf

[2] Deutsche Forschungsgemeinschaft. 2013. *DFG Practical Guidelines on Digitisation*. http://www.dfg.de/formulare/12_151/12_151_en.pdf

[3] Rog, J. 2008. *Evaluating File Formats for Long-term Preservation*. https://www.kb.nl/sites/default/files/docs/KB_file_format_evaluation_method_27022008.pdf

[4] Saxon State and University Library Dresden. 2015. *Assessment of SLUBArchiv for the Data Seal of Approval*. https://assessment.datasealofapproval.org/assessment_178/seal/pdf/

[5] http://libtiff.maptools.org

[6] https://github.com/SLUB-digitalpreservation/checkit_tiff