KANSALLISARKISTO

THE CREATIVE ARCHIVES´ AND USERS´ NETWORK

Report on File Formats for Hand-written Text Recognition (HTR) Material

CO:OP

Community as Opportunity

The Creative Archives´ and Users´ Network

Author: Sami Nousiainen, National Archives of Finland

December 16, 2016

Co-funded by the Creative Europe Programme of the European Union

This work is licensed under Creative Commons CC BY 4.0.

Abbreviations and terms

Abbreviation or term	Explanation
ADDML	Archival Data Description Markup Language
AHAA	A metadata project of the National Archives of Finland
AIP	Archival Information Package
ALLÄRS	General Swedish Thesaurus
ALTO	Analyzed Layout and Text Object
binarize	To convert the document image into bi-level form as an attempt to separate the text from the background.
CHANGE	Change Vocabulary Specification
CSS	Cascading Style Sheets
DC	Dublin Core
deskew	To correct the angle of the text.
dewarp	To correct the distortion due to page curl and perspective that is present in document page images captured using digital cameras.
DIP	Dissemination Information Package
Dublin Core Schema	Set of terms to describe web resources or physical resources.
EAC-CPF	Encoded Archival Context – Corporate bodies, Persons and Families
EAD	Encoded Archival Description
EDM	Europeana Data Model
ESE	Europeana Semantic Elements
faceted search	A way of accessing information enabling the usage of several filters (e.g. filter for organization, format, year).
FINNA	Search interface for Finnish archives, libraries and museums
FINTO	Finnish Thesaurus and Ontology service
GTS	Ground Truth Storage
HTML	HyperText Markup Language
HTR	Handwritten Text Recognition
IE	Information Extraction
indexing	Can be used in two meanings in this document: 1) Providing a more detailed structural description of an archival unit by specifying page numbers and corresponding topic. 2) Automatic processing of documents resulting in the creation of an index within indexing/search software enabling (e.g. full-text) searches to be carried out. For example, the inverted index lists for each term the documents in which the term appears.
ISAAR (CPF)	International Standard Archival Authority Record For Corporate Bodies, Persons and Families
ISAD(G)	General International Standard Archival Description
ISNI	International Standard Name Identifier
JPEG	Joint Photographic Experts Group
JSON	JavaScript Object Notation
KAM	Libraries, archives and museums (acronym in Finnish)

CC BY

KDK	National digital library (acronym in Finnish)
KOKO	Collection of Finnish core ontologies (including e.g. YSO)
LAM	Libraries, Archives and Museums
MARC	MAchine-Readable Cataloging
METS	Metadata Encoding and Transmission Standard
MIX	Metadata for Images in XML Schema
MODS	Metadata Object Description Schema
NDL	National Digital Library
NER	Named Entity Recognition
NLP	Natural Language Processing
NLP	Negative Log Probability
OAI-PMH	Open Archives Initiative Protocol for Metadata Harvesting
OAIS	Open Archival Information System
OCR	Optical Character Recognition
ORCID	Open Researcher and Contributor ID
OWL	Web Ontology Language
PAGE	Page Analysis and Ground-truth Elements
PAS	Long-term preservation (acronym in Finnish)
PDF	Portable Document Format
POS	Part-Of-Speech
PREMIS	PREservation Metadata: Implementation Strategies
RDA	Resource Description and Access
RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
SAPO	Finnish Spatio-Temporal Ontology (acronym in Finnish)
SFTP	SSH File Transfer Protocol
SIP	Submission Information Package
SKOS	Simple Knowledge Organization System
SPARQL	SPARQL Protocol and RDF Query Language
SUO	Finnish Geo-ontology
TEI	Text Encoding Initiative
textMD	Technical Metadata for Text
TIFF	Tagged Image File Format
TISC	Open Time and Space Core Vocabulary Specification
Turtle	Terse RDF Triple Language
URI	Uniform Resource Identifier
VIAF	Virtual International Authority File
XML	Extensible Markup Language
YSA	General Finnish Thesaurus (acronym in Finnish)
YSO	General Finnish upper ontology (acronym in Finnish)

CC BY

Table of Contents

CC BY

CC BY

1 Introduction and background

The **primary purpose** of this document is to review and analyze the available **file formats** for the storage of automatically recognized text or manually input text (transcription). The automatic recognition can be either OCR-based (i.e. recognition of printed text) or HTR-based (i.e. recognition of hand-written text).

The existing file formats are described from the point of view of their structure and special characteristics and links to schema files or more detailed descriptions of the formats are given. Also, an attempt is made to list some of the projects, organizations and pieces of software using the formats. Finally, a summary and comparison of the reviewed file formats is provided.

Another purpose of this document is to analyze the **applicability of the file formats** in the environment of the **National Archives of Finland**. This requires state-of-the-art analysis identifying current systems related to e.g. long-term preservation of documents, metadata handling and information search as well as describing the foreseen changes in the environment in the near future. In addition to that, requirements concerning the types of usage potentially enabled by the existence of OCR:ed / HTR:ed document text are listed. Finally, the potential implications of fulfilling the listed requirements on processes, other systems and processing are analyzed.

2 Analysis of file formats

2.1 General

This section provides the description and analysis of file formats suitable for the storage of automatically or manually recognized text. The file formats studies are divided into two groups: 1) file formats specifically designed for the storage of OCR / HTR results and 2) file formats that could be used also for the storage of OCR / HTR results (in addition to other purposes).

From the point of view of text recognition, OCR and HTR differ from each other with respect to the level of difficulty: HTR results (i.e. the recognized text) can be expected to be less accurate than OCR results in general. This also emphasizes the fact that for HTR it is even more important to be able to express the level of **confidence** concerning the recognition results in the HTR results file. This is important from the point of view of further utilization of the recognition results e.g. for metadata generation or search indexing. Another difference between OCR and HTR concerning the results file is the fact that the recognized **text style** (e.g. font) is an easily understandable concept in case of OCR, but not directly applicable in case of HTR; the question is, whether there is an analogous concept in case of HTR.

The OCR / HTR formats reviewed in this section will be analyzed from several points of view. The objective is to make the analysis comparable between the file formats. For example, the following issues will be addressed, whenever applicable and whenever information can be found. However, note that this list does not include all the characteristics of the file formats and the reader is encouraged to check the original format specification and/or schema file for more details.

CC BY

- **General description**: General description of the file format and special characteristics of it if any. Encompassing (when applicable and suitable information available) e.g.:
    - **Format name**: The name of the file format analyzed.
    - **Links**: Links to e.g. the format specification and/or schema file (in case of XML) and other resources and documents related to the format.
    - **Organizations**: Main organizations that have specified / are specifying the format or are maintaining it or are using / supporting the format.
    - **Projects**: Projects that are using the file format or that have specified / are specifying the format.
    - **Pieces of software**: Pieces of software using the format. This could mean pieces of software that can read or export the format. E.g. OCR software using the format as (one) output format.
    - **When**: When has the format been introduced and when was it updated. The purpose of this criterion is to shed light on the maturity and longevity of the format.
- **Tags / attributes / properties**: The purpose of this part is to indicate how some of the main issues of the OCR / HTR results are expressed in the file formats analyzed. In XML formats, the terminology used is such that there are **tags** and **attributes**. An example about a tag is the following line in XML containing the tag named `text`: `<text>This is just text.</text>`. An example about and attribute is the `fontsize` in the following line in XML: `<text fontsize=”10”>This is just text.</text>`. Depending on the format studied, other terminology might also be used. The expression of the following issues will be studied:
    - **Image file**: Reference to the image file from which the text was recognized.
    - **Processing**: Description of the processing steps for OCR / HTR used to obtain the resulting text. This could mean e.g. the name of the OCR engine used.
    - **Structure**: Hierarchical structure of the document. This could mean e.g. the logical structure of the document in terms of division into chapters and sections and/or the physical structure in terms of pages and page areas.
    - **Text areas**:
        - **Coordinates**: The definition of text areas, lines of text and baselines linking the piece of text to coordinates in the original image (i.e. to the area in the original page image that contains the resulting text).
        - **Text**: The actual text recognized by the OCR / HTR engine.
    - **Orientation**: Orientation of the text in the original image or the angle through which the text needs to be rotated to get normal horizontally flowing text.
    - **Font / style**: Any text style definitions such as the name of the font, size of the font or colors.
    - **Confidence**: The level of confidence that the recognition result (i.e. the text) is actually correct. This must be estimated by the OCR / HTR engine based on how well the text in the document image matches the previously learned (based on training data) pattern of text. The level of confidence could be expressed e.g. on character level, word level or page level.

CC BY

  - **Named Entities**: Indication of named entities present in the text and their location in the text. This means e.g. names of people, locations or dates appearing in the text.
  - **Metadata**: Any additional descriptions concerning the OCR / HTR results.
  - **Other**: Any other interesting and relevant elements / attributes / properties.
- The issues above will be presented in a table for each file format studied. The style of presentation will be (for other formats except PDF) such that:
  - A / B: Indicates that tag B is within tag A.
  - A / B [attr.]: Indicates that tag A has an attribute B.
  - A (boolean): Indicates that A can assume only boolean values.
  - A (a, b, c): Indicates that A can assume only values a, b or c.
- For PDF, the style of presentation will be such that operators (beginning with "/" character) and keywords (not having the initial "/" character) are indicated in the table.

One issue common to all file formats is how to encode the characters appearing in the documents to be transcribed or OCR:ed / HTR:ed. Unicode already contains many characters and for the still missing characters or abbreviation marks there is Medieval Unicode Font Initiative[1] (MUFI) whose objective is to propose missing characters to Unicode and coordinate their allocation in the Private Use Area of the Unicode.

2.2 XML format overview

Due to the fact that many of the file formats to be described in this section are based on XML, a brief overview of the XML format itself is provided. This description is neither meant to serve as a comprehensive reference for the XML format nor to serve as a tutorial; only some basic issues related to XML are presented to make it easier to understand the specifications or schemas of the OCR / HTR formats.

The specification of the XML format is provided by the W3C[2]. XML is a textual file format and has support for Unicode and, thus, characters of various natural languages.

Elements in XML are defined using the `element` tag and their type and name are given using the `type` and `name` attributes. Respectively, **attributes** in XML are defined using the `attribute` tag and their `type` and `name` are given using the type and name attributes.

XML defines some **primitive data types**, for example, decimal, float, boolean, string and dateTime. Decimal is a numeric data type that does not have a fractional part. One data type derived from decimal is integer. Boolean is the data type that allows only two values: true or false (or 1 and 0, respectively). Additional **restrictions on the values** of a primitive data types can be placed in XML, for example, the following expression `minInclusive="0" maxInclusive="1"` in connection with the float primitive data type can be used to restrict the allowed values to be between 0 and 1 (including both 0 and 1). Furthermore, the values of an element can be restricted to a set of given values using the `enumeration`

[1] http://folk.uib.no/hnooh/mufi/
[2] W3C Recommendation, "Extensible Markup Language (XML) 1.0 (Fifth Edition)", 26 November 2008.

CC BY

tag and its `value` attribute. Also, a **default value** for an element can be defined using the `default` attribute in the element tag. These primitive data types along with potential restrictions can be used for the elements and attributes that the user defines in his/her schema file. Also, **derived data types** can be defined by the user using the `simpleType` and `complexType` elements.

The `sequence` element in an element type definition specifies that the parent element must contain certain elements in order. Occurrence indicators `maxOccurs` and `minOccurs` can be used to control the **number of times** an element can occur and a value `unbounded` can be used for the `maxOccurs` indicator if the element can occur arbitrarily many times. By default, attributes are optional. An attribute can be defined to be **mandatory** with the `use` attribute: `use="required"`. A **group of attributes** to be included in complex type definition can be specified using the `attributeGroup` element and `attribute` elements within it (giving both `name` and `type` for each attribute).

Instructions concerning e.g. the meaning and usage of elements can be given inside the XML schema definition using the `documentation` element inside the `annotation` element.

2.3 File formats for recognized text (OCR/HTR)

2.3.1 PAGE XML (Page Analysis and Ground-truth Elements)

General description. The PAGE XML format is an XML-based format for storing OCR-related data. Contrary to many other formats that record only text recognition results (e.g. page content and layout), it also records information about processing steps (binarisation, deskew, dewarping) and layout evaluation metrics and results.

The PAGE format has been introduced in the IMPACT EU project[3] by Pattern Recognition and Image Analysis (PRImA) Research Lab of the School of Computing, Science and Engineering, University of Salford, United Kingdom[4]. According to [5], it is currently used in several projects such as IMPACT Centre of Competence, eMOP, Europeana Newspapers, Transcriptorium and READ. The XML schema files (root.xsd, binarisation.xsd, deskew.xsd, dewarping.xsd, layouteval.xsd, pagecontent.xsd) are available in [6] and the format description is provided in the publication S. Pletschacher, A. Antonacopoulos, "The PAGE (Page Analysis and Ground-truth Elements) Format Framework", Proceedings of the 20th International Conference on Pattern Recognition (ICPR2010), Istanbul, Turkey, August 23-26, 2010, IEEE-CS Press, pp. 257-260[7]. Also, PAGE-related implementations such as libraries in Java and C++ can be downloaded from [8].

[3] http://www.impact-project.eu/
[4] http://www.primaresearch.org/
[5] http://www.digitisation.eu/training/recommendations-for-digitisation-projects/recommendations-formats-standards-recommendations/
[6] http://www.primaresearch.org/schema/PAGE/
[7] http://www.primaresearch.org/www/assets/papers/ICPR2010_Pletschacher_PAGE.pdf
[8] http://www.primaresearch.org/tools/PAGELibraries

CC BY

The Transkribus[9] OCR / HTR tool (related to tranScriptorium and READ projects) can produce results in the PAGE format. Also, a command line tool utilizing the free OCR engine Tesseract and outputting results in the PAGE format is available[10].

Table 1. Tags / attributes / properties of PAGE XML format.

Issue to be indicated	Tag / attribute / property used	Description
Image file	imageFilename	Name of the image file of the page.
Processing	root.xml: Gts / Name	Name of the GTS instance.
	root.xml: Gts / Namespace	Link to the location of the XML-Schema of the GTS resource.
	root.xml: Gts / Data	Link to the XML-instance containing the GTS data.
	root.xml: Gts / Dependencies	
	binarisation.xml: Page / PageImage, PageSkeleton	See e.g. [11].
	binarisation.xml: Patches / Patch / PosX, PosY, Width, Height, PatchImage, PatchSkeleton	
	deskew.xml: DeskewAngle	
	dewarp.xml: Grid	See e.g. [12].
	layouteval.xml: Eval / Profile / ErrorTypeWeights / ErrorTypeWeight / name, weight, allowableWeight, type (split, merge, miss, partial-miss, misclassification, false-detection) [attr.] / Description, RegionTypeWeight RegionTypeWeight / name, weight, allowableWeight, type (split, merge, miss, partial-miss, misclassification, false-detection) [attr.] / Description, RegionTypeWeight, SubTypeWeight SubTypeWeigth / name, weight, allowableWeight, subtype [attr.] /	Contains definitions of page layout analysis evaluation profiles (e.g. weights for different types of errors such as region merges and splits and different types of regions such as text and image). See e.g. [13].

[9] https://transkribus.eu/Transkribus/
[10] http://www.primaresearch.org/tools/TesseractOCRToPAGE
[11] K. Ntirogiannis et al, "An Objective Evaluation Methodology for Document Image Binarization Techniques", The Eighth IAPR Workshop on Document Analysis Systems, 2008.
[12] Po Yang et al, "Grid-Based Modelling and Correction of Arbitrarily Warped Historical Document Images for Large-Scale Digitisation", Workshop on Historical Document Imaging and Processing, 2011.
[13] C. Clausner et al, "Scenario Driven In-Depth Performance Evaluation of Document Layout Analysis Methods", International Conference on Document Analysis and Recognition, 2001.

CC BY

	Description, RegionTypeWeight	
	layouteval.xml: Eval / EvalData / groundTruthFilename, segmentationResultFilename, imageFilename, imageWidth, imageHeight [attr.] / PageObjectResults, BorderResults PageObjectResults / type [attr.] / RawData, Metrics RawData / GroundTruthOverlap, SegResultOverlap, RegionResults, ReadingOrderResults Metrics / NumberOfGroundTruthRegions, NumberOfSegResultRegions, etc.	Contains evaluation results of layout analysis (e.g. recall and precision per region type).
	pagecontent.xml	The rest of the tags appearing in this table are from the pagecontent.xml file.
Structure	Page / TextRegion, ImageRegion, LineDrawingRegion, GraphicRegion, TableRegion, SeparatorRegion, MathsRegion, ChemRegion, MusicRegion, AdvertRegion, NoiseRegion, UnknownRegion	
Text areas, Coordinates	TextRegion / Coords / points [attr.]	List of points in the following format: “x1,y1 x2,y2 …”.
	TextRegion / TextLine / Coords / points [attr.]	
	TextRegion / TextLine / Baseline / points [attr.]	
	TextRegion / TextLine / Word / Coords / points [attr.]	
	TextRegion / TextLine / Word / Glyph / Coords / points [attr.]	
Text areas. Text	TextRegion / TextEquiv / PlainText, Unicode	Text in text region level.
	TextRegion / TextLine / TextEquiv / PlainText, Unicode	Text in text line level.
	TextRegion / TextLine / Word / TextEquiv / PlainText, Unicode	Text in word level.
	TextRegion / TextLine / Word / Glyph / TextEquiv / PlainText, Unicode	Text in glyph level.
Orientation	TextRegion / orientation [attr.]	The angle in degrees the region has to be rotated to correct the skew.

CC BY

	TextRegion / readingOrientation [attr.]	The angle in degrees the baseline of text has to be rotated to correct the skew.
Font / style	TextRegion, TextLine, Word, Glyph / production (printed, typewritten, handwritten-cursive, handwritten-printscript, other) [attr.]	The production attribute can be defined on the TextRegioni, TextLine, Word or Glyph level.
	TextStyle / fontFamily, serif (boolean), fontSize, textColour, bgColour, bold (boolean), italic (boolean), underlined (boolean), subscript (boolean), superscript (boolean), strikethrough (boolean), smallCaps (boolean), letterSpaced (boolean) [attr.]	Example values for the fontFamily attribute are e.g. Arial and Times New Roman. Example values for the textColour and bgColour attributes are e.g. red, blue and black.
Confidence	TextRegion / TextEquiv / conf [attr.]	Confidence value for the OCR result 0…1.
	TextRegion / TextEquiv / index [attr.]	Order of the alternatives defined with multiple TextEquiv / PlainText, Unicode tags. The one with the smallest index is the main text.
Named Entities		
Metadata	Metadata / Creator, Created, LastChange, Comments	Name of creator, time stamps for creation and last change and comments.
Other	Page / ReadingOrder / OrderedGroup / RegionRefIndexed / index [attr.]	Reading order within the page.

2.3.2 ALTO XML (Analyzed Layout and Text Object)

General description. The ALTO XML file format is XML-based. Originally the format was developed in the EU-funded METAe (Meta Data engine) project[14], which was running from September 2000 to September 2003 in the EU 5th Framework programme, in the area of "Digital Heritage and Cultural Content". There were 14 partners from 7 European countries and the USA in the project, e.g. ABBYY Europe among them.

Version 1.0-02 of the ALTO format was released in December 2004 and version 3.1 in January 2016. Content Conversion Specialists (CCS) from the METAe project consortium maintained the ALTO standard until August 2009 and after that the United States Library of Congress[15] ("The ALTO Editorial Board: Collaboration and Cooperation across Borders"[16]) has hosted and maintained it.

According to the Library of Congress web page[17], e.g. the following organizations are using the ALTO XML format in some project: National Library of Australia (Trove), British Newspaper Archive, the United States Library of Congress (Chronicling America) and the National Library of Finland (Digitised Newspapers and Serials). The web page lists plenty of other organizations as well. Furthermore, the European Newspapers

[14] http://cordis.europa.eu/project/rcn/52630_en.html
[15] http://www.loc.gov/standards/alto/
[16] http://library.ifla.org/265/1/177-zarndt-en.pdf
[17] https://www.loc.gov/standards/alto/community/implementers.html

CC BY

project[18] is using ALTO XML (e.g. National Library of Finland is participating in the project) and a METS/ALTO profile[19] has been defined in the project. The docWorks[20] software (of Content Conversion Specialists) is using ALTO as its output format and there are links to several tools related to ALTO XML in [21]. Also, ABBYY FineReader Engine supports ALTO XML starting from the Version 10 Release 2 (December 2010).

The most recent ALTO XML schema is version 3.1 released in 25.1.2016 and available on-line[22]. Descriptions of some ALTO tags Use Cases are available in [23] (version 16.1.2014, ALTO Board) e.g. related to Layout tagging, Structural tagging and Named Entities tagging and there is a discussion forum for open issues[24].

ALTO files consist of three main sections:

- <Description>: Description, e.g. source file information and processing software
- <Styles>: Styles, e.g. descriptions of fonts and paragraphs
- <Layout>: Layout, the actual content e.g. text with positioning with respect to the top-left corner of the page

There is some information available about the <Styles> and <Layout> usage in [25]. Also, information is available in Wikipedia[26].

Furthermore, there is a discussion in [27] concerning the provision of glyph or character level information in ALTO XML format. As of version 3.1, there is only the CC tag (see Table 2) to express character level recognition confidence value, but no support for the provision of glyph or character level recognition variants (contrary to other OCR / HTR formats that have such support). Also, the granularity of the character level confidence value is coarse — it is expressed as an integer between 0 and 9. Therefore, the discussion addresses several aspects of the potential introduction of the glyph level information into the ALTO XML schema and mentions that it is taken to version 3.2 draft schema. Glyph level sample files are available in [28].

Table 2. Tags / attributes / properties of ALTO XML format.

Issue to be	Tag / attribute / property used	Description

[18] http://www.europeana-newspapers.eu/
[19] Europeana Newspapers project, Deliverable "D5.2 Europeana Newspapers METS ALTO Profile (ENMAP) – External Release – Draft". http://www.europeana-newspapers.eu/wp-content/uploads/2014/08/D5.2-Europeana-Newspapers-METS-ALTO-Profile-ENMAP-DRAFT.pdf
[20] http://content-conversion.com/?lang=en#docworks-2
[21] https://github.com/altoxml/documentation/wiki/Software
[22] http://www.loc.gov/standards/alto/v3/alto-3-1.xsd
[23] https://github.com/altoxml/documentation/raw/master/use-cases/ALTO_tags_v1_0.pdf
[24] https://github.com/altoxml/schema/issues
[25] https://www.loc.gov/standards/alto/techcenter/layout.html
[26] https://en.wikipedia.org/wiki/ALTO_(XML)
[27] https://github.com/altoxml/schema/issues/26
[28] https://github.com/altoxml/documentation/tree/master/v3/Glyph

CC BY

indicated		
Image file	Description / sourceImageInformation / filename	
	Description / sourceImageInformation / fileIdentifier	A unique identifier for the image file. This is drawn from MIX.
Processing	Description / OCRProcessing / preProcessingStep, ocrProcessingStep, postProcessingStep	Description of processing steps.
	xProcessingStep / processingAgency	Organization processing the image file. x = pre, ocr or post.
	xProcessingStep / processingDateTime	When the image was processed. x = pre, ocr or post.
	xProcessingStep / processingStepDescription	String describing the processing step performed. x = pre, ocr or post.
	xProcessingStep / processingStepSettings	Freeform description (string) of the processing step settings with sufficient detail (ideally) to enable repeating the processing step. x = pre, ocr or post.
	xProcessingStep / processingSoftware / softwareCreator, softwareName, softwareVersion	Information about the software application used for processing. x = pre, ocr or post.
	Page / PROCESSING [attr.]	A link (xsd:IDREF) to the processing description that has been used for this page.
Structure	Layout / Page	
	LayoutTag / ID, LABEL, TYPE, DESCRIPTION [attr.]	Can be used in ComposedBlock, TextBlock, TextLine and String to indicate e.g. formulas, tables and maps.
	StructureTag / ID, LABEL, TYPE, DESCRIPTION [attr.]	Can be used in ComposedBlock, TextBlock, TextLine and String to indicate e.g. title page, table of contents and chapters.
	Using METS[29]	ALTO can be combined with METS and METS used for indicating specific parts of the ALTO file.
Text areas, Coordinates	TextBlock / HEIGHT, WIDTH, HPOS, VPOS [attr.]	Rectangular area, whose position from the top-left corner of the page is HPOS, VPOS in units defined by the MeasurementUnit element and whose height and width are given by HEIGHT and WIDTH.
	TextBlock / TextLine / HEIGHT, WIDTH, HPOS, VPOS [attr.]	
	TextBlock / TextLine / String / HEIGHT, WIDTH, HPOS, VPOS [attr.]	
	Shape / Polygon / Points [attr.]	List of corner points of the polygon in the form of “x1,y1 x2,y2 …”.

[29] https://www.loc.gov/standards/alto/techcenter/use-with-mets.html

CC BY

	Shape / Ellipse / HPOS, VPOS, HLENGTH, VLENGTH, ROTATION [attr.]	HPOS, VPOS is the center of the ellipse and HLENGTH, VLENGTH the width and height of the ellipse. ROTATION is the rotation angle (in degrees counterclockwise) of the content within the block.
	Shape / Circle / HPOS, VPOS, RADIUS [attr.]	Circular area with center coordinates HPOS, VPOS and radius of size RADIUS.
Text areas, Text	TextBlock / TextLine / String / CONTENT [attr.]	TextBlock = A paragraph of text. TextLine = A line of text. String = A single word.
Orientation	TextBlock / ROTATION [attr.]	ROTATION is the rotation angle (in degrees counterclockwise) of the content within the block.
Font / style	Styles / TextStyle / FONTFAMILY, FONTTYPE (serif, sans-serif), FONTSTYLE (bold, italics, subscript, superscript, smallcaps, underline) [list], FONTCOLOR, FONTSIZE, FONTWIDTH (proportional, fixed) [attr.]	The FONTSTYLE attribute contains a list of whitespace separated strings. The FONTCOLOR attribute is given as XML hexBinary type.
	String / STYLE (bold, italics, subscript, superscript, smallcaps, underline) [list, attr.]	The STYLE attribute contains a list of whitespace separated strings.
Confidence		See [30] for discussion on the calculation of PC, WC and CC.
	Page / PC [attr.]	Page confidence level for the OCR result 0…1. 0 = unsure, 1 = sure.
	String / WC [attr.]	Word/string confidence level for the OCR result 0…1. 0 = unsure, 1 = sure.
	String / CC [attr.]	Character confidence level for the OCR, a list of numbers 0…9. 0 = sure, 9 = unsure.
	Page / ACCURACY [attr.]	Estimated percentage of OCR accuracy 0…100.
	Block / CS (boolean) [attr.]	Correction status: manual correction done or not.
	Block / TextLine / CS (boolean) [attr.]	
	Block / TextLine / String / CS (boolean) [attr.]	
	String / ALTERNATIVE	Alternative for the word.
Named Entities	Tags / NamedEntityTag / ID, LABEL, DESCRIPTION, URI [attr.]	Reference to the NamedEntityTag can be in e.g. TextBlock, TextLine, String / TAGREFS. The LABEL attribute contains the type of Named Entity (e.g. person, location). The URI attribute can contain e.g. a reference to

[30] https://github.com/altoxml/schema/issues/23

CC BY

		GeoNames in case of a location. Named Entity Recognition confidence value could be included in the XmlData part within a pre-agreed tag.
Metadata	Tags / OtherTag / ID, LABEL, TYPE, DESCRIPTION [attr.]	OtherTag could be used to store metadata.
Other	Page / PRINTED_IMG_NR [attr.]	The page number that is printed on the page.
	String / CONTENT, SUBS_TYPE (HypPart1, HypPart2, Abbreviation), SUBS_CONTENT [attr.]	For representing a hyphenated word whose first or second part is in the text line being analyzed.
	Tags / RoleTag / ID, LABEL, TYPE, DESCRIPTION [attr.]	Indication of people involved in the content creation.
	Tags / OtherTag / ID, LABEL, TYPE, DESCRIPTION [attr.]	Any other tag.

2.3.3 ABBYY FineReader XML

General description. The ABBYY FineReader XML format is an XML-based format for storing OCR data. It is used by the ABBYY FineReader application[31]. ABBYY FineReader 6.0 XML version 1 is from year 2002 and ABBYY FineReader 10.0 XML version 1 is from year 2011.

The schema for the ABBYY FineReader 10.0 version 1 XML is available in [32]. Furthermore, general information about the tag hierarchies, character attributes and sample XML files are available in [33]. The XML schema is hierarchical so that the description starts from the document level, below which there is the page level. Under the page level, there can be various types of blocks (Text, Table, Picture, Barcode, Separator, SeparatorsBox, Checkmark, GroupCheckmark) and the actual region is then defined under the block using region and rect tags. Description of some of the tags can be found in [34].

Table 3. Tags / attributes / properties of ABBYY FineReader XML format.

Issue to be indicated	Tag / attribute / property used	Description
Image file	page / width, height, resolution [attr.]	Page image width and height in pixels and resolution in ppi.
Processing	document / version, producer	XML version and the name of the XML file producer.
Structure	page / block / blockType (Text, Table, Picture, Barcode, Separator, SeparatorsBox, Checkmark, GroupCheckmark) [attr.]	Various block types are supported. Tags available within the block can depend on the block type.
Text areas, Coordinates	page / block / region / rect / l, t, r, b [attr.]	Left, top, right and bottom coordinates of the rectangle.

[31] https://www.abbyy.com/ocr-sdk/
[32] http://fr7.abbyy.com/FineReader_xml/FineReader10-schema-v1.xml
[33] https://abbyy.technology/en:features:ocr:xml
[34] http://ocrsdk.com/documentation/specifications/xml-scheme-recognized-document/

CC BY

	page / block / text / par / line / l, t, r, b [attr.]	Left, top, right and bottom coordinates of the rectangle surrounding the line.
	page / block / text / par / line / charParams / l, t, r, b [attr.]	Left, top, right and bottom coordinates of the rectangle surrounding the character.
Text areas, Text	page / block / text / par / line / formatting	Text is contained within the formatting tags.
	line / formatting / charParams	
Orientation	page / rotation (Normal, RotatedClockwise, RotatedUpsidedown, RotatedCounterclockwise) [attr.]	
	pageStream / pageElement / text / orientation (Normal, RotatedClockwise, RotatedUpsidedown, RotatedCounterclockwise) [attr.]	
Font / style	paragraphStyle / fontStyle / id, ff, fs, baseFont (boolean), italic (boolean), bold (boolean), underline (boolean), strikeout (boolean), smallcaps (boolean), scaling, spacing, color, backgroundColor [attr.]	ff = font family, fs = font size
	line / formatting / ff, fs, italic (boolean), bold (boolean), underline (boolean), strikeout (boolean), smallcaps (boolean), subscript (boolean), superscript (boolean), scaling, spacing, color, style [attr.]	ff = font family, fs = font size
Confidence	wordRecVariants / wordRecVariant / variantText	
	wordRecVariants / wordRecVariant / wordFromDictionary (boolean) [attr.]	
	charRecVariants / charRecVariant / charConfidence (-1, 0…100) [attr.]	The greater the value of charConfidence, the higher the confidence in the correctness of the recognition. The sum of the character recognition variants does not need to be 100. The value -1 corresponds to undefined confidence value.
	charRecVariants / charRecVariant / serifProbability (0…100, 255) [attr.]	Probability that the character is using a serif font. The value 255 corresponds to undefined serif probability value.
Named Entities		
Metadata		
Other		

CC BY

2.3.4 hOCR

General description. The hOCR format is an XML-based format, but embedded in HTML/XHTML documents. The encoding is done with embedded elements and embedded properties within HTML/XHTML tags: embedded hOCR elements are defined inside the CLASS-attribute of an HTML tag (all hOCR elements are named starting with the string ocr_ or ocrx_) and embedded hOCR properties are defined inside the TITLE-attribute of an HTML tag. However, hOCR could be equivalently represented as pure XML by putting the hOCR elements as XML tags and hOCR properties as XML attributes. The hOCR format re-uses HTML for e.g. encoding style- and font-related information. More information about the hOCR format can be found in [35] and [36].

There are several pieces of software that are using the hOCR format as output format:

- Cuneiform (free OCR software)
- OCRopus[37] (free OCR software)
- Tesseract[38] (free OCR software, latest stable version 3.04.01 from February 2016)

Furthermore, several tools for dealing with the hOCR format are available in [39].

Table 4. Tags / attributes / properties of hOCR format.

Issue to be indicated	Tag / attribute / property used	Description
Image file	ocr_page / image	Defines the name of the image file used as input. Used in ocr_page element.
	x_source	Document source e.g. a URL.
Processing	html_ocr_<engine>	Where <engine> is the name of the OCR engine.
	ocr-system	Using HTML META tag: `<META NAME="ocr-system" CONTENT="name version">`
	ocr-capabilities	The capability to generate certain type of markup should be indicated in the metadata using properties starting with ocrp_ (e.g. ocrp_lang, ocrp_dir, ocrp_poly, ocrp_font, ocrp_nlp). E.g.: `<META NAME="ocr_capabilities" CONTENT="ocrp_lang ocrp_dir">`
Structure	ocr_document, ocr_linear, ocr_title,	Logical structuring elements.

[35] Thomas Breuel, "The hOCR Microformat for OCR Workflow and Results", Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), 23-26 Sept, 2007.
[36] Thomas Breuel (editor), "The hOCR Embedded OCR Workflow and Output Format", March 2010.
[37] https://github.com/tmbdev/ocropy
[38] https://github.com/tesseract-ocr
[39] https://github.com/tmbdev/hocr-tools

CC BY

	ocr_author, ocr_abstract, ocr_part, ocr_chapter, ocr_section, ocr_sub*section, ocr_display, ocr_blockquote, ocr_par, ocr_caption	
	ocr_float, ocr_separator, ocr_textfloat, ocr_textimage, ocr_image, ocr_linedrawing, ocr_photo, ocr_header, ocr_footer, ocr_pageno, ocr_table	Floats.
	ocr_glyph, ocr_glyphs, ocr_dropcap, ocr_chem, ocr_math	Inline representations.
Text areas, Coordinates		To be related to the ocr_page, ocr_carea and ocr_line typesetting elements.
	bbox	Defines the rectangular bounding box for the text in the binarized document image. The format of the bounding box definition is: bbox x1 y1 x2 y2.
	poly	Defines the non-rectangular bounds for the text in the binarized document image. The format of the polygon definition is: poly x1 y1 x2 y2 …
	baseline	Baseline of the line of text expressed as a polynomial in the coordinate system of the bounding box of the line. The format of the polynomial is[40]: baseline p_n p_{n-1} … p_0, where p_i is the coefficient of the i:th order term of the polynomial.
Text areas, Text	HTML DIV or SPAN	Text is contained within HTML DIV or SPAN tags.
Orientation	textangle	Defines the angle of the text relative to the rest of the page.
Font / style	HTML DIV / STYLE [attr.] or HTML SPAN / STYLE [attr.]	Encoded using standard HTML/CSS attributes i.e. style attribute in DIV or SPAN tag and the corresponding style definition (e.g. color, font-size, font-family).
	x_font, x_fsize	OCR-engine specific markup.
Confidence	nlp	Negative log probabilities of the character recognition.
	x_confs	Character confidences; OCR-engine specific.
	x_wconf	Substring confidence; OCR-engine specific.
	cuts	Locations of character segmentation cuts.
	HTML SPAN / INS and DEL elements alt, nlp	HTML SPAN-element in connection with INS and DEL elements is used to represent alternative segmentations / readings (with

[40] https://github.com/tesseract-ocr/tesseract/wiki/FAQ#how-to-interpret-hocr-baseline-output

CC BY

		associated confidence values provided in nlp) of text.
Named Entities		
Metadata		Can be done using META tag of HTML.
Other	order	Defines the reading order of the elements.
	scan_res	The scanning resolution of the image in DPI.
	x_scanner	Scanner representation.
	imagemd5	MD5 checksum of the image file from which this text was recognized.
	lpageno	Page number on the page.

2.3.5 Other OCR / HTR formats

Other OCR / HTR file formats include e.g.:

- **XDOC format**: Output format of ScanSoft. For further information, see ScanSoft, "XDOC Data Format: Technical Specification", Version 4.0, May 1999[41].
- **ORF (OCR results file) format**: OCR result format for the GNU Ocrad OCR program. Contains Comma-separated list of pairs of recognized characters and confidence values. For further information, see [42].

2.4 Other document/text file formats

2.4.1 TEI (Text Encoding Initiative)

General description. The Text Encoding Initiative (TEI) is a consortium that maintains guidelines for the representation of digital texts especially in the field of digital humanities. TEI Guidelines P5[43] is the most recent standard and it was originally released in November 2007 (version 1.0.0); the most up-to-date version of the TEI Guidelines P5 is version 3.0.0 released in 2016-03-29. TEI Guidelines P5 is an XML-based format defining about 500 elements for the (mainly) semantic description of texts. The guidelines enable the expression of both metadata about the document and the structural features of the document (e.g. headings). Basic information about TEI and the guidelines can be obtained from the Wikipedia page[44].

Earlier versions of TEI Guidelines include e.g. P1, which was released in July 1990 and P4, which was the first version to implement XML support and released in June 2002.

For example, the nidaba[45] software (open-source OCR pipeline) encodes the OCR result into an XML document following the TEI P5 Guidelines.

[41] http://www.vividata.com/manuals/core12xdc.pdf
[42] http://www.gnu.org/software/ocrad/manual/ocrad_manual.html
[43] http://www.tei-c.org/Guidelines/
[44] https://en.wikipedia.org/wiki/Text_Encoding_Initiative
[45] https://openphilology.github.io/nidaba/tei.html

CC BY

Digital facsimiles[46] are a concept in TEI that encompasses a collection of images, metadata to identify them and optionally a transcribed version of the images. Thus, facsimiles are suitable for representing OCR / HTR results in TEI files.

Table 5. Tags / attributes / properties of TEI format.

Issue to be indicated	Tag / attribute / property used	Description
Image file	graphic / url [attr.]	Link to image forming part of text or providing and image of it.
	pb / facs [attr.]	Page break (pb) along with a reference to the image of the page (facs).
Processing	teiHeader / encodingDesc	Free-form description of the relationship between the electronic text and the original source[47].
	teiHeader / respStmt / resp, name	Statement of responsibility containing the description of responsibility (resp) or role in production or distribution of the work and the corresponding name (name).
Structure	div / type, n [attr.]	For example: … …
Text areas, Coordinates	surface / ulx, uly, lrx, lry [attr.]	x-coordinate of the upper left corner y-coordinate of the upper left corner x-coordinate of the lower right corner y-coordinate of the lower right corner
	surface / zone / ulx, uly, lrx, lry [attr.]	x-coordinate of the upper left corner y-coordinate of the upper left corner x-coordinate of the lower right corner y-coordinate of the lower right corner
	surface / zone / points [attr.]	Corner points of the polygon defining the 2D area (at least 3 pairs of coordinates).
Text areas, Text	surface / zone / line	Contains the text of a line.
	zone / seg / g	Contains a character or a glyph.
	text / body / p / lg / l	Single text / body of text / paragraph / line group / line of text.
Orientation	lg / style [attr.] / transform:rotateZ(degrees)	Rotation attribute borrowed from the CSS (Cascading Style Sheets) definition. The rotation is given in degrees in the clockwise direction around a point that is by default at

[46] http://www.tei-c.org/release/doc/tei-p5-doc/en/html/PH.html#PHFAX
[47] http://www.tei-c.org/release/doc/tei-p5-doc/en/html/HD.html

CC BY

		the center of the element (text). E.g.: `<lg style="transform:rotate(45deg)">Text</lg>`
Font / style	hi / rendition, rend, style [attr.] / font-size	The rendition attribute contains a pointer to the description of the rendering. The rend attribute contains whitespace-separated tokens. The style attribute contains stylistic information in a formal language.
Confidence	certainty / target, locus, degree [attr.]	OCR recognition confidence. The value of the locus attribute should be "value" and the value of the degree attribute is between 0 (certainly false) and 1 (certainly true).
	div / u / exclude [attr.]	Alternation[48]: providing mutually exclusive transcription alternatives.
	alt / target, mode, weights [attr.]	Probabilities for each alternative really appearing in the text. Mode can have either of the values "excl" or "incl". If mode is "excl", the sum of weights must be 1.
Named Entities	persName	Name of a person.
	placeName	Name of a place.
	orgName	Name of an organization.
	date	
	certainty / target, locus, degree (0…1) [attr.]	Certainty related to the fact that the text is really e.g. a person name i.e. that persName tag is valid. The target attribute should point to the corresponding persName tag and the value of the locus attribute should be "name". The value of the degree attribute is between 0 (certainly false) and 1 (certainly true).
Metadata	xenoData	The tag xenoData enables inserting metadata in non-TEI formats.
Other		

2.4.2 PDF (Portable Document Format)

General description. PDF files differ clearly from the XML-based file formats: they can contain both text and binary data mixed together. The main objective for PDF files is to conserve the presentation (visual appearance) of the document. PDF supports various types of data compression to reduce the size of the files: JPEG and JPEG2000 (from PDF v1.5) for color and grayscale images, CCITT (Group 3 or 4) run-length encoding and JBIG2 (from PDF v1.4) for monochrome images and LZW and Flate (from PDF v1.2) for text, graphics and images. JPEG compression is based on discrete cosine transform and JPEG2000 on wavelet transform (both are lossy compression methods).

[48] http://www.tei-c.org/release/doc/tei-p5-doc/en/html/SA.html#SAAT

The description language of the PDF is based on postfix notation: first the operands are listed and then the operator. PDF supports eight basic types of objects:

- Boolean: `true` or `false`
- Number: e.g. `15` or `2.1`
- Strings: e.g. `(Hello world!)`
- Name: e.g. `/Size`
- Array: e.g. `[ (Hello world!) 15 ]`
- Dictionary: e.g. `<< /Size 15 /Color (blue) >>`
- Stream object: e.g. an image; consists of a dictionary and bytes between the keywords `stream` and `endstream`. The dictionary contains information about the stream indicated with keys such as `/Length` and `/Filter`.
- Null object: indicated with the keyword `null`

Also, a PDF file can contain indirect objects of any type. They are defined by giving them an identifier and the content, e.g.:

```
1 0 obj
(Hello world!)
endobj
```

Then, in other parts of the PDF file, references to the indirect objects can be made using the R operator, e.g.:

```
1 0 R
```

The PDF file structure consists of four parts:

- **Header:** Defining the PDF version, e.g. `%PDF-1.4`
- **Body:** Containing the objects
- **Cross-reference table:** Containing the location of each object within the PDF file, e.g.:

```
xref
0 29
0000000000 65535 f
0000000017 00000 n
0000000340 00000 n
…
```

- **Trailer:** Containing information about the Cross-reference table. `/Size` indicates the number of entries in the Cross-reference table, `/Root` refers to the Document Catalog object, /Info refers to the Document Information Dictionary and `startxref` indicates how far (number of bytes) from the start of the PDF file the Cross-reference table is, e.g.:

CC BY

```
trailer
<</Size 29 /Root 1 0 /Info 2 0 …>>
startxref
96716
%%EOF
```

The structure of the PDF Body is hierarchical: the Document Catalog is a dictionary referring to other objects, e.g.:

```
1 0 obj
<</Type /Catalog /Pages 2 0 R>>
endobj
```

In that example, the Document Catalog refers to `/Pages` dictionary, which in turn refers to individual page objects (`/Kids`), the two page objects (`3 0` and `7 0`) in the example below:

```
<< /Type /Pages /Count 2 /Kids [3 0 R 7 0 R] >>
```

PDF/A[49] is a version of PDF meant for long-term preservation of electronic documents. It is a constrained form of PDF and self-contained not being reliant on information outside the PDF file. Thus, everything needed to display the PDF document always in the same way must be contained in the PDF file itself (e.g. fonts need to be embedded into the PDF file). Also, JavaScript and encryption are forbidden in PDF/A files and device-independent color must be used (either ICC profiles or CIE Lab color specifications).

PDF/A has **three versions**:

- PDF/A-1 (ISO 19005-1) specified in 2005
- PDF/A-2 (ISO 19005-2) specified in 2011
- PDF/A-3 (ISO 19005-3) specified in 2012

PDF/A-1 is based on PDF version 1.4 and PDF/A-2 and PDF/A-3 are based on PDF version 1.7. The only difference between PDF/A-2 and PDF/A-3 is that PDF/A-3 enables embedding of arbitrary file formats inside the PDF file. Explicit association must be made between the embedded file and the PDF / object / structure. However, there are no requirements in the PDF/A-3 specification concerning the long-term usability of the embedded files. The embedded files should not be rendered by a PDF/A-3 standards conforming reader, but the possibility to extract the embedded files should be provided.

Furthermore, in addition to the three different versions of PDA/A, there are **different levels of conformance**[50]:

- a (Full conformance): Tagged PDF (i.e. the logical structure of the document must be defined).

[49] https://en.wikipedia.org/wiki/PDF/A
[50] http://www.alliancegroup.co.uk/pdf-a.htm

CC BY

- b (Basic conformance): Rendered visual appearance must be preserved, not the logical structure of the document.
- u (Intermediate conformance): In addition to conformance b, all text in the document must have Unicode equivalents.

PDF/A-1 file can have level a or b conformance and PDF/A-2 and PDF/A-3 can have level a, b or u conformance.

A searchable PDF is a document, which contains an image of the text (e.g. scanned page) and additionally the recognized or transcribed text (e.g. recognized using OCR) thus enabling searches to be carried out based on the content of the document. The text is made invisible for the human viewer by rendering it using text rendering mode 3 "Neither fill nor stroke text (invisible)" (see PDF 1.7 specification, section "5.2.5 Text Rendering Mode") defined using the Tr operator. Simply changing the rendering mode to some other (e.g. rendering mode 2: "Fill, then stroke text.") results in both the image of the text and the actual text being rendered on top of each other.

More information about PDF file format can be found in the specification document (Adobe Systems Incorporated, "PDF Reference", sixth edition, Version 1.7, November 2006[51]). Also, the PDF format has been published as an open standard in 2008 (ISO 3200-1:2008). Furthermore, information concerning specifically PDF/A can be found in the corresponding specification document ("ISO 19005-3:2012 Document management -- Electronic document file format for long-term preservation -- Part 3: Use of ISO 32000-1 with support for embedded files (PDF/A-3)" [52]) which is not, however, available for free. Originally the PDF format was specified by Adobe Systems and made available in 1993.

There are many **tools** for dealing with PDF files. Some basic tools for e.g. decompressing object streams (e.g. text) include qpdf[53], for extracting text from PDF (text that has been written as ASCII or Unicode; not text in bitmaps) PDFMiner[54] (pdf2txt) and for visualizing the structure of the pdf PDFBox[55]. Also, OCR tools can support PDF by making it possible to export the OCR results in PDF format (e.g. ABBYY OCR: PDF or PDF/A export[56]).

A general **use case** for PDF/A files with respect to embedding other file formats is given in the report "The benefits and risks of the PDF/A-3 file format for archival institutions. An NDSA report", February 2014[57]: the PDF document contains human-readable content and equivalent machine-readable content is embedded in the PDF file in e.g. XML format. This would require using special tools or implementing own tools to utilize the embedded data since a PDF/A-3 compliant viewer needs to be able to present only the primary document — not the embedded files. Several examples about embedded machine-readable content in

[51] http://www.adobe.com/content/dam/Adobe/en/devnet/acrobat/pdfs/pdf_reference_1-7.pdf
[52] http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=57229&commid=53674
[53] http://qpdf.sourceforge.net/
[54] http://www.unixuser.org/~euske/python/pdfminer/
[55] https://pdfbox.apache.org/2.0/commandline.html
[56] http://www.ocr4linux.com/en:documentation:pdf_export_keys
[57] http://lcweb2.loc.gov/master/gdc/lcpubs/2013655115.pdf

CC BY

PDF/A files are given in the same report on page 7 and a use case concerning invoices embedded as XML in PDF/A-3 files is given on page 9. Furthermore, starting from page 11 several PDF/A-3 usage scenarios are described and analyzed.

Table 6. Keywords and operators of PDF format.

Issue to be indicated	Keywords / operators used	Description
Image file	stream, endstream	Embedded in the PDF file as XObject (`/Type /XObject`) between `stream` and `endstream` keywords with specific type of compression (e.g. `/Filter /CCITTFaxDecode`).
Processing		Could be indicated in the metadata.
Structure	/H1, /H2, /H3, etc. (headings) /P (paragraphs) /TOC, /TOCI (table-of-contents) /L, /LI (lists) /Table, /TR, /TD (tables)	See PDF 1.7 specification, section "10.7.3 Standard Structure Types" and, in general, PDF 1.7 specification, section "10.7 Tagged PDF". Logical structure is stored separately from the content to be displayed.
Text areas, Coordinates	Tm T*	Text matrix given as an operand to the Tm operator i.e. `a b c d e f Tm`, where a=scale x, b=shear x, c=shear y, d=scale y, e=offset x and f=offset y. `T*` operator moves to the start of the next line.
Text areas, Text	stream, endstream BT, ET Tj	Text is stored as content streams between `stream` and `endstream` and BT (begin text) and ET (end text). It can be in compressed format (e.g. `/FlateDecode` indicates that variable-length Lempel-Ziv adaptive compression cascaded with adaptive Huffman coding" has been used, see PDF 1.7 reference page 71). After decompression, the text can be contained as such as an operand of the `Tj` operator or as character references to the `CMap` table elsewhere in the PDF file.
Orientation	/Rotate Tm	`/Rotate` key in the `/Page` dictionary for the rotation of the entire page.

CC BY

		Text rotation by the angle θ (counterclockwise) can be accomplished using the Tm operator: `cos(θ) sin(θ) –sin(θ) cos(θ) 0 0 Tm`.
Font / style	Tf Tr RG rg	Choose font and font size, rendering mode and color (for stroking and non-stroking operations). Invisible selectable and searchable text (e.g. OCR result text) can be accomplished e.g. using text rendering mode 3 “Neither fill nor stroke text (invisible)” (PDF 1.7 specification, section “5.2.5 Text Rendering Mode”).
Confidence		
Named Entities		

Metadata	`<<`
	`  /Title (title)`
	`  /Subject (subject)`
	`  /Keywords (keywords)`
	`  /Author (author)`
	`  /CreationDate (cdate)`
	`  /ModDate (mdate)`
	`  /Creator (creator)`
	`  /Producer (producer)`
	`  ...`
	`>>`
		Document Info Dictionary (since PDF version 1.1). See PDF 1.7 specification, section “10.2.1 Document Information Dictionary”.

	`/applicationname <<`
	`  /LastModified (date)`
	`  /Private <<`
	`  dictionary`
	`  >>`
	`>>`
		PieceInfo Dictionary (since PDF version 1.3) indicated with `/PieceInfo` in the page object. See PDF 1.7 specification, section “10.4 Page-Piece Dictionaries”.
		
		applicationname = Name of the application, whose metadata this is.
		date = Date of last modification of the document by this application.
		dictionary = Private application data dictionary in the form of names and values.

	`/A <<`
	`  /O /UserProperties`
	`  /P [`
	`  << /N (n1) /V (v1) >>`
	`  << /N (n2) /V (v2) >>`
	`  ...`
	`  << /N (nk) /V (vk) >>`
	`  ]`
	`>>`
		Object Data (User Properties): (since PDF version 1.6). Metadata related to specific objects inside the PDF, e.g.: `/N (Book Number) /V 1234`. See PDF 1.7 specification, section “10.6.4 Structure Attributes”.

	`<<` `  /Type /Metadata`	XMP (Extensible Metadata Platform)

CC BY

	/Subtype /XML /Length xxx >> stream ... <xmp:CreateDate>xxx </xmp:CreateDate> <xmp:ModifyDate>xxx </xmp:ModifyDate> ... endstream	metadata (since PDF version 1.4). XML-based format for metadata. In metadata stream dictionary, the following definitions must be made: /Type /Metadata /Subtype /XML. XMP specification contains multiple schemas e.g. Dublin Core. See PDF 1.7 specification, section "10.2.2 Metadata Streams".
Other		

2.4.3 Other formats

Other formats for storing text and formatting include e.g.:

- **Open Document Format for Office Applications (ODF)** or OpenDocument: This is actually a zip archive, whose file extension has been changed (e.g. to .odt). Inside the zip archive, there are several XML files and folders. The main file inside the archive is the content.xml file, which contains the actual document content (text, excluding any binary data such as images). The OpenDocument format is used by e.g. OpenOffice suite of applications.
- **Word Document format (DOCX)**: This is also a zip archive containing XML files and folders. Most of the document content is contained in the file document.xml. DOCX format is used by Microsoft Office 2007 and later versions.
- **DjVu**[58] (in Wikipedia[59]): This is a file format intended for scanned documents and as a replacement for the PDF format.

Even for TIFF files, functionality called TIFF IFilter[60] can be installed in e.g. Windows 7 operating system to index TIFF files (for search) based on their textual content. The textual content is extracted using OCR.

2.5 Summary of file formats

In general, the file formats reviewed in this document describe the textual content, layout and styles used in the page. Also, in some cases, OCR / HTR process related information can be provided in the files, such as values concerning the text recognition confidence and additional tags describing specific words in the text such as Named Entities.

Viewers. There are pieces of software for viewing files of the formats studied in this report. For example, the PAGE Viewer[61] (of the Pattern Recognition & Image Analysis Research Lab of the University of Salford, Manchester) can open and display files in PAGE XML, ALTO XML, ABBYY FineReader XML and hOCR formats.

[58] http://djvu.org/
[59] https://en.wikipedia.org/wiki/DjVu
[60] https://technet.microsoft.com/en-us/library/dd755985(v=ws.10).aspx
[61] http://www.primaresearch.org/tools/PAGEViewer

CC BY

The hOCRImageMapper[62] application is meant for viewing hOCR files only and BnlViewer[63] application is meant for viewing METS/ALTO files.

Converters. There are pieces of software for converting between OCR / HTR file formats. For example, an ABBYY to ALTO converter[64] written in PHP5, hOCR to PDF converters written in Python and in Java[65] (links on the page), ABBYY to TEI converter[66] written in PHP, ALTO/ABBYY to PAGE converter[67], hOCR (text + jpegs) to PDF converter[68] and ALTO to/from hOCR converter[69].

Comparison of formats. A brief comparison of OCR formats is available in [70]. It addresses hOCR, ALTO XML and ABBYY FineReader XML formats from the point of view of how certain things are expressed in each format (e.g. bounding boxes and confidence values). Also, release dates for various versions of the formats are mentioned in it.

The Impact Centre of Competence in Digitisation has given recommendations on formats and standards useful in digitization[71]. In addition to recommending specific formats for OCR results files (ALTO, PAGE; alternative UTF-8 plain text), it gives recommendations on master file formats for long-term preservation for still images (TIFF; alternative JPEG2000), textual documents (TEI, PDF/A; alternative UTF-8 plain text), descriptive metadata (DCMES, MODS; alternative MARC21), structural metadata (METS) and administrative metadata (PREMIS, MIX, TextMD).

Furthermore, the Succeed EU project has reviewed 17 existing guidelines / recommendations[72] concerning 1) File formats for master and delivery files, 2) Metadata formats (descriptive, structural, administrative and 3) Other formats (OCR output, linguistic resources, tools packaging, other). More specifically, the following guidelines / recommendations have been reviewed in the deliverable:

1. IMPACT project recommendations
2. JISC Digital Media Guidelines
3. Recommendations of the Bibliothèque nationale de France
4. New York State Archives — Imaging Production Guidelines

[62] https://mlichtenberg.wordpress.com/2014/12/23/hocrimagemapper-a-tool-for-visualizing-hocr-files/
[63] https://sourceforge.net/p/bnlviewer/home/Home/
[64] https://github.com/ironymark/AbbyyToAlto
[65] http://xplus3.net/2009/04/02/convert-hocr-to-pdf/
[66] http://able.myspecies.info/abbyy-xml-tei-xml
[67] http://www.primaresearch.org/tools/PAGEConverterValidator
[68] https://github.com/tmbdev/hocr-tools#hocr-pdf
[69] https://github.com/UB-Mannheim/ocr-fileformat
[70] https://github.com/kba/ocr-xsl/blob/master/OCR-Format-Comparison.md
[71] http://www.digitisation.eu/training/recommendations-for-digitisation-projects/recommendations-formats-standards-recommendations/
[72] Succeed-project deliverable "D4.1 Recommendations for metadata and data formats for online availability and long-term preservation", 16.1.2014. http://www.succeed-project.eu/sites/default/files/deliverables/Succeed_600555_WP4_D4.1_RecommendationsOnFormatsAndStandards_v1.1.pdf

CC BY

5. The NARA Technical Guidelines for Digitizing Archival Materials for Electronic Access
6. NISO Framework of Guidance for Building Good Digital Collections
7. California Digital Library Guidelines for Digital Objects and File Format Recommendations
8. DFG guidelines on digitization
9. The Getty Research Institute online resources
10. Universal Photographic Digital Imaging Guidelines
11. Federal Agencies Digitization Initiative Guidelines
12. Technical Guidelines for Digital Cultural Content Creation Programmes (MINERVA project)
13. The National Digital Newspaper Program — Technical Guidelines for Applicants
14. Images for web delivery — standards, image capture standards, metadata for images created by the National Library of Australia
15. University of Virginia Library — community digitization guidelines
16. Image specifications and Functional Requirements for Citation Capture (PubMed Central Back Issue Scanning Project)
17. Picture Queensland Image Digitisation Manual 2007

The Succeed project review found out that only 11 out of the 17 recommendations indicated a format for the OCR results: ALTO XML (5 recommendations), TEI (3 recommendations), ASCII (3 recommendations) and Unicode (2 recommendations) appeared in the recommendations. The PDF format appeared, not in the OCR results file formats, but in the delivery file formats being the second most common format after the JPEG.

Out of the main formats reviewed in this document (PAGE XML, ALTO XML, ABBYY FineReader XML, hOCR, TEI and PDF), the one clearly differing from the rest is PDF. Its purpose is different compared to the rest: it serves as a format for presenting something to the end-user and conserving the visual appearance of the presentation. Additionally, it can contain OCR results in the form of text enabling copy-pasting text from the document and textual searches within the document. Thus, it bundles the images and the text together into one file that is easy to transfer. Other types of bundles can also be envisaged, for example, the National Library of Finland is offering the OCRed content of the historical newspapers in a bundle consisting of three types of data: 1) Metadata, 2) ALTO XML and 3) raw text[73]. The ALTO XML file is created by the Content Conversion Specialists' (CSS) DocWorks program.

Graphical illustrations of the XML schemas for PAGE XML, ALTO XML and ABBYY FineReader XML can be found in the appendices (Appendix I: Visualization of the PAGE XML schema, Appendix II: Visualization of the ALTO XML schema and Appendix III: Visualization of the ABBYY FineReader XML schema).

Furthermore, a simple example of a manually transcribed page can be found in image, PDF, PAGE XML, ALTO XML and TEI formats in the appendices (Appendix IV.1: Example document page image, Appendix IV.2: Example document page PDF text, Appendix IV.3: Example document page PAGE XML, Appendix IV.4:

[73] T. Pääkkönen et al, "Exporting Finnish Digitized Historical Newspaper Contents for Offline Use", D-Lib Magazine, Volume 22, Number 7/8, July/August 2016. http://www.dlib.org/dlib/july16/paakkonen/07paakkonen.html

CC BY

Example document page ALTO XML and Appendix IV.5: Example document page TEI). The PDF, PAGE XML, ALTO XML and TEI files were generated by exporting the transcription results from the Transkribus tool version beta 0.8.7. Additional examples can be found on the Internet, for example, in [74] OCR processing results of the ABBYY Recognition Server can be found in ABBYY FineReader XML and ALTO XML formats.

3 Implications on systems at the National Archives of Finland

3.1 General

This section starts by providing information about the **state-of-the-art** of relevant systems and processes used at the National Archives of Finland and describes also the changes in the state-of-the-art that are foreseen to occur in the near future. The state-of-the-art section includes a description of the outside world as well in the form of listing existing external data sources that are or could be relevant.

Next, **requirements and possibilities** related to potential future existence of OCR:ed / HTR:ed document text are identified and clarified. The purpose is to figure out what kind of functionality could exist if automatic OCR / HTR of the document images were possible.

Finally, the **implications** of the fulfillment of the listed requirements on processes, systems and processing needed are tackled. The purpose is to try to identify what might need to be changed as a result of the introduction of the automatic OCR / HTR into the digitization process.

3.2 Technology and terminology overview

Some relevant technological and methodological concepts and terms are briefly explained in this section. The purpose is to explain on a general level the relevant background, but not to get stuck into small differences in the definition of terms.

3.2.1 Data and text mining

Data mining is the process of finding interesting patterns in data. The process can be divided into several phases e.g. according to the **CRISP-DM**[75] (CRoss-Industry Standard Process for Data Mining) process: business understanding, data understanding, data preparation, modelling, evaluation and deployment. Within the CRISP-DM process, issues such as handling of missing values, feature extraction and feature selection need to be tackled. Data mining approaches themselves can be divided into **descriptive** (**unsupervised**) and **predictive** (**supervised**) techniques. Clustering and association rule mining fall into the group of descriptive techniques and regression and classification into the group of predictive techniques. The objective of the **regression analysis** is to build (learn) a model to predict the values of a continuous variable based on other variables (continuous or discrete) and the objective of the **classification analysis** is to build (learn) a model to predict the values of a discrete variable based on other variables (continuous or discrete).

[74] https://abbyy.technology/en:features:ocr:alto
[75] Pete Chapman et al, "CRISP-DM 1.0: Step-by-step data mining guide", 2000

CC BY

The term **text mining** refers to the fact that the data being mined is in (unstructured) textual form. Techniques analogous to data mining (of numerical data) can be identified within text mining: for example, **text categorization** (classification) and **text clustering**. An example of text categorization is the classification of an incoming email message as spam or not spam (based on previously learned classification model). In order to perform e.g. text categorization, the unstructured textual data (documents) needs to be converted into a more suitable format first. Such a format could be based on the counts of words appearing in documents (e.g. Term Frequency – Inverse Document Frequency approach). **Natural Language Processing** (**NLP**) as a term overlaps partially with text mining. Tasks such as tokenization, stemming, lemmatization and part-of-speech tagging fall into NLP. However, Named Entity Recognition i.e. the process of finding and naming entities, such as names of people or locations, appearing in text can fall into text mining or Natural Language Processing.

3.2.2 Indexing and searching

Search engine indexing refers to the act of collecting and processing data to facilitate conducting fast searches in e.g. a document collection. An **inverted index** can be the end-result of indexing: for each (search) term, an inverted index lists references to all documents, in which the term appears. Additionally, the inverted index can list the position of the term in each document.

3.2.3 Social metadata and crowdsourcing

With **social metadata**, we mean the metadata provided by normal users and usually voluntarily. **Crowdsourcing**, on the other hand, refers to the act of harnessing normal users to carry out some task, e.g. the provision of metadata. Thus, social metadata can be obtained by crowdsourcing.

3.2.4 Ontologies

With **controlled vocabulary**, we mean a set of agreed terms. A **taxonomy** is an extension of a controlled vocabulary in a way that the controlled vocabulary is organized into a hierarchy (i.e. it has parent-child relationships). Furthermore, a **thesaurus** is taxonomy with more information (e.g. associative relationships in addition to parent-child relationships).

Ontology describes the structure of certain domain in terms of **concepts** and **relations** between concepts. Concepts are also called classes and classes can have subclasses (subclasses are actually a result of relations between classes). Relations can be between classes or between objects and they can be symmetric or transitive. **Instances of classes** are also called objects or individuals or entities. Ontology together with instances of classes forms a **knowledge base**. The terminology related to ontologies is not always consistent between different sources; relations are often designated as properties and properties are further divided into two groups: 1) Data type / attribute / simple properties and 2) Object / relationship / complex properties. The former group relates a class or an instance to a literal (e.g. integer value or string value) and the latter group relates classes/instances to classes/instances.

CC BY

There are various ways to define ontologies. In particular, **RDF**[76] (Resource Description Framework), **RDFS**[77] (Resource Description Framework Schema) and **OWL**[78] (Web Ontology Language) enable more and more specific description of ontologies. Actually, RDF is both and **abstract model** and a **vocabulary** for expressing ontologies. As an abstract model, it states that data shall be described using triples consisting of subject, predicate and object. RDF as such does not take stand on the actual **serialization format or file format** in which the RDF data is stored; there are several file formats available to store RDF data, e.g. **XML**, **Turtle**[79] (Terse RDF Triple Language) and **N3**. Turtle is a human-readable format, in which namespaces for URIs can be defined like `@prefix ex: <http://example.org>` and triple statements can be written like `ex:John ex:hasAge 30 .` i.e. by terminating the triple with a period. As a vocabulary, RDF gives some terms that can be used to describe classes or instances e.g. `rdf:type` can be used to express that John is a person (`ex:John rdf:type ex:Person`). `rdf:type` itself is an instance of `rdf:Property`. Other RDF terms include e.g. `rdf:Description`, `rdf:about`, `rdf:Bag`, `rdf:Seq` and `rdf:Alt`. RDFS is also a vocabulary and adds more on tops of RDF, terms such as `rdfs:subClassOf`, `rdfs:subPropertyOf`, `rdfs:range` and `rdfs:domain`. For example, we could say that `ex:hasAge rdfs:domain ex:Person` and `ex:hasAge rdfs:range xsd:integer` meaning that subjects of the predicate (property) `ex:hasAge` are instances of class `ex:Person` and objects of the predicate (property) `ex:hasAge` are integers. Furthermore, OWL provides still more possibilities for description and it comes in three flavors (in the order of increasing complexity): OWL Lite, OWL DL (Descriptive Logic) and OWL Full. OWL enables e.g. expressing property restrictions of two different kinds: 1) value constraints and 2) cardinality constraints. A value constraint refers to the range of values of the property related to certain class and a cardinality constraint refers to the number of values a property can take related to certain class. The restrictions are defined using `owl:Restriction` and `owl:onProperty` and they are applicable to both data type properties and object properties.

Other vocabularies include e.g. **CHANGE**[80] (Change Vocabulary Specification), which is a lightweight spatio-temporal vocabulary defining terms such as `Merge`, `Split` and `Namechange` and **TISC**[81] (Open Time and Space Core Vocabulary Specification), which define spatial properties (e.g. `nearest`, `northOf`, `areasize`, `touches`) and temporal properties (e.g. `happensAt`, `existenceBeginsAt`). Furthermore, **SKOS**[82] (Simple Knowledge Organization System) is a vocabulary that enables representing controlled vocabularies, taxonomies and thesauri. It defines terms such as skos:prefLabel, skos:altLabel, skos:broader and skos:narrower. Thus, it is possible to say e.g. `ex:Computer skos:broader ex:Laptop` meaning that computer is a more general concept than laptop.

[76] https://www.w3.org/RDF/
[77] http://www.w3.org/TR/rdf-schema/
[78] https://www.w3.org/OWL/
[79] http://www.w3.org/TR/turtle/
[80] http://linkedearth.org/change/ns/
[81] http://observedchange.com/tisc/ns
[82] http://www.w3.org/TR/skos-reference/

CC BY

Finally, the knowledge expressed as RDF triples can be queried using SPARQL (SPARQL Protocol and RDF Query Language). An example about SPARQL query could be

```
PREFIX ex: <http://example.org>
SELECT ?person
WHERE {
          ?person ex:hasAge 30 .
}
```

which would return persons having age 30.

3.2.5 Description standards and metadata standards

There are many **description standards**, **metadata standards** and associated **file formats**.

ISAD(G)[83] (General International Standard Archival Description) is a description model that defines elements to be used in archival descriptions. In total, it gives 26 elements of which six are essential (reference code, title, creator, date(s), extent of the unit of description and level of description). There are also **Finnish archival description and indexing rules** from the year 1997 that are consistent with the ISAD(G) standard and the correspondence between the description rules and the FINMARC format is indicated in the documents[84]. **ISAAR (CPF)**[85] (International Standard Archival Authority Record For Corporate Bodies, Persons and Families) is a description model standardized by the International Council on Archives (ICA). It provides guidelines for the description of authority data. **RDA**[86] (Resource Description and Access): Indicates what kinds of description entities are used and what kind of data is stored in the description entities. It replaces ISBD (International Standard Bibliographic Description). RDA can be used e.g. with MARC21 file format. **EU Core Vocabularies**[87] are metadata standards developed by ISA (Interoperability Solutions for European public Administrations) programme running in 2010-2015 and consist of Core Person Vocabulary, Registered Organisation Vocabulary, Core Location Vocabulary and Core Public Service Vocabulary. E.g. the Core Person Vocabulary encompasses fundamental characteristics of a person (e.g. name and date of birth). The EU Core Vocabularies do not stipulate certain representation format (file format).

Dublin Core (DC) is a metadata standard that can be used to describe web resources or physical resources (e.g. books). DCMES (Dublin Core Metadata Element Set) defines 15 basic metadata elements (e.g. Title, Publisher, Date). **METS**[88] (Metadata Encoding and Transmission Standard) provides an XML-based metadata standard for holding descriptive, administrative and structural metadata. The purpose of METS is to enable maintaining digital objects within repositories and exchanging them between repositories. The METS

[83] http://www.ica.org/en/isadg-general-international-standard-archival-description-second-edition
[84] http://www.arkisto.fi/fi/arkistojen-kuvailu--ja-luettelointisaeaennoet (in Finnish)
[85] http://www.ica.org/en/isaar-cpf-international-standard-archival-authority-record-corporate-bodies-persons-and-families-2nd
[86] https://www.loc.gov/aba/rda/
[87] http://ec.europa.eu/isa/ready-to-use-solutions/core-vocabularies_en.htm
[88] http://www.loc.gov/standards/mets/

CC BY

standard is maintained by the Library of Congress. **EAD**[89] (Encoded Archival Description) is an XML-based standard for describing archives (compare MARC for describing bibliographic materials), their finding aids i.e. organization of collections of archival materials. There are several versions of EAD, e.g. EAD 2002 and EAD3. The root element in EAD 2002 and EAD3 is <ead>. In EAD 2002, the root element can contain <eadheader>, <frontmatter> and <archdesc>, whereas in EAD3 the root element can contain <control> and <archdesc>. In EAD3, the hierarchical structure of the material can be expressed using e.g. <c01>, <c02>, <c03> etc. tags wrapped within each other. **EAC-CPF**[90] (Encoded Archival Context – Corporate bodies, Persons and Families) is an XML-based standard for recording information about the creators of archival materials (corporate bodies, persons and families). It conforms to the model of ISAAR (CPF). The format is used in the archives domain (e.g. in the National Archives of Finland). The EAC-CPF standard is maintained by the Society of American Archivists and Berlin State Library. EAC-CPF can be used together with EAD to refine information provided by EAD. In EAC-CPF, instances are defined using the <eac-cpf> element, which contains <control> and <cpfDescription> elements. The latter element (<cpfDescription>) contains the actual description of the corporate body, person or family. Examples about the usage of EAC-CPF are available e.g. in [91]. **MARC21**[92] (MAchine Readable Cataloguing) consists of five communication formats: MARC 21 Format for Authority Data, MARC 21 Format for Bibliographic Data (subset: MARC 21 LITE Bibliographic Format), MARC 21 Format for Holdings Data, MARC 21 Format for Classification Data and MARC 21 Format for Community Information. It is used in the library domain (nationally and internationally). A MARC21 record consists of three parts: leader, directory and variable fields. MARC was originally released in 1967-68 and the Finnish version FINMARC was introduced at the end of 1970's. **MODS**[93] (Metadata Object Description Schema)is an XML-based metadata format and includes a subset of MARC fields. Some top-level MODS elements include e.g. <titleInfo>, <name> and <typeOfResource>. Examples of metadata in MODS format are available in [94] e.g. for a digitized book. **PREMIS**[95] **(**PREservation Metadata: Implementation Strategies) is a metadata standard for the long-term preservation of digital objects. It contains five different entities: Intellectual, Object, Event, Rights and Agent. **MIX**[96] (NISO Metadata for Images in XML Schema) an XML-based metadata format for technical data concerning images. An example about MIX metadata is available in [97]. **ADDML**[98] (Archival Data Description Markup Language) is a standard for describing a collection of plain text files and it has been developed by the National Archives of Norway. SÄHKE2[99] is a Finnish regulation or standard concerning the processing, management and preservation, disposal and preservation of born-digital documents. It stipulates what kind

[89] https://www.loc.gov/ead/
[90] http://eac.staatsbibliothek-berlin.de/
[91] http://eac.staatsbibliothek-berlin.de/tag-library/examples.html
[92] https://www.loc.gov/marc/
[93] http://www.loc.gov/standards/mods/
[94] https://www.loc.gov/standards/mods/userguide/examples.html
[95] http://www.loc.gov/standards/premis/
[96] http://www.loc.gov/standards/mix/
[97] http://www.loc.gov/standards/mix/instances/test_mix10.xml
[98] http://www.arkivverket.no/arkivverket/Arkivbevaring/Elektronisk-arkivmateriale/Standarder/ADDML
[99] http://www.arkisto.fi/fi/palvelut/julkisen-hallinnon-saehkoeiset-palvelut/saehke-maeaeraeykset (in Finnish)

CC BY

of metadata should be generated and stored when processing documents and gives XML schema for storing the metadata.

3.3 State-of-the-art and beyond (at National Archives of Finland)

3.3.1 External data sources

This section describes some of the external data sources that are either currently used at the National Archives of Finland or that could be used in the future. In a way, the vocabularies presented in the previous section can also be considered to be external data sources, but here more extensive sources of data are presented. **Finto**[100] is a Finnish thesaurus and ontology service, which offers many of the ontologies described below (e.g. YSO, KOKO and SAPO) and even more. It is maintained by the National Library of Finland. There is also a web page[101] of Finto that contains links to ontologies under development (e.g. SAPO) in addition to ready ontologies.

Names of people. People can be uniquely identified using e.g. **ISNI** (International Standard Name Identifier), which is governed by ISNI-IA (International Agency). The searchable ISNI database is available in [102]. The identifiers are unique and permanent and alternative spellings for the names can be provided in the database. **VIAF**[103] (Virtual International Authority File) is an authority file formed by linking national authority files and operated by the Online Computer Library Center (OCLC). Several national libraries as well as ISNI are participating in the VIAF project. The base source for ISNI is VIAF. Another identification systems is **ORCID** (Open Researcher and Contributor ID), which is meant to identify scientific authors and contributors. ORCID is a subset of ISNI and part of ISNI identifier range is reserved for ORCIDs. ORCIDs are maintained by ORCID Inc. and it is possible to search by ORCID or name in the database[104]. Both ISNI and ORCID are 16 characters long consisting of digits 0-9 (except for the last character in the ID, which may also be X) in groups of 4 characters separated by hyphens. **Names of Finnish communities** are also available in [105] in RDF/XML format. The number of entries is approximately 40k.

Names of places. The **Place Name Register**[106] of the National Land Survey of Finland contains data about 800k place names in Finland (e.g. a coordinate point). These are current place names, not historical. **GEOnet Names Server**[107] (GNS) contains e.g. coordinates (in WGS84) of places. **GeoNames**[108] contains over 11M placenames, coordinates (in WGS84) and e.g. elevation and population as well as relations such as children and neighbours. **SUO**[109] (Finnish Geo-ontology) contains classes related to human-constructed (e.g. city) and natural (e.g. lake) places and it is based on the Finnish Place Name Register and GEOnet Names

[100] https://finto.fi/en/
[101] http://dev.finto.fi/en/
[102] http://isni.oclc.nl
[103] https://viaf.org/
[104] https://orcid.org/orcid-search/search
[105] http://finto.fi/cn/fi/
[106] http://www.maanmittauslaitos.fi/digituotteet/nimisto
[107] http://geonames.nga.mil/gns/html/
[108] http://www.geonames.org/
[109] http://seco.cs.aalto.fi/ontologies/suo/

CC BY

Server (GNS). It has hierarchical, topological (e.g. overlaps, touches) and part-whole relationships. **SAPO**[110] (Finnish Spatio-Temporal Ontology) is an ontology time-series of Finnish municipalities between years 1865 and 2010. It is available in [111] in RDF/Turtle format. It indicates times of start of existence and end of existence for places and borders as WGS84 polygons. It uses e.g. TISC and SKOS terms. SAPO version dated as 26.6.2014 contains 2010 preferred terms in Finnish and 558 preferred terms in Swedish. More information on SAPO can be found in the report concerning current status of SAPO and possibilities for extending it[112].

General index terms or keywords. YSA[113] (General Finnish Thesaurus) covers all fields of knowledge and contains most common terms; in total, it has about 34k terms of which about 6k are geographical names. It has links to the corresponding Swedish thesaurus **ALLÄRS**[114] (General Swedish Thesaurus) which has also about 34k terms and to YSO. YSA utilizes SKOS vocabulary to indicate e.g. broader and narrower terms (`skos:broader` and `skos:narrower`) and the corresponding Swedish term (`skos:exactMatch`). Also, it has links to related concepts (`skos:related`). **YSO**[115] (General Finnish Upper Ontology) is an extension of YSA and it is based on YSA and ALLÄRS. It has about 29k concepts and is trilingual (Finnish, Swedish and English). It uses various relationships such as hierarchical parent-child relationships, part-whole relationships and associations and has also links to Library of Congress Subject Headings (LCSH). In comparison to YSA, YSO has more detailed analysis of meanings of the concepts (e.g. one concept in YSA might appear as two concepts in YSO), complete hierarchy between concepts and unique identifiers (URIs) for all concepts, YSA, ALLÄRS and YSO are available for download in both RDF/XML and RDF/Turtle formats. **KOKO**[116] is a collection of Finnish core ontologies including YSO and more specific ontologies. It has about 51k terms in Finnish, about 33k terms in Swedish and about 34k terms in English. **WordNet** is a database for English language containing various relations between words such as hypernyms/hyponyms and meronyms/holonyms. The data is available also in RDF format[117]. Based on the original English WordNet, versions for various languages have been created. For example, **FinnWordNet** (i.e. database for Finnish language) is available via on-line search interface[118] and the data can be downloaded[119] as well. Also, Swedish WordNet is available via on-line search interface[120].

[110] http://seco.cs.aalto.fi/ontologies/sapo/
[111] http://dev.finto.fi/sapo/en/
[112] J. Väätäinen et al, "Sapon nykytila, ylläpito ja laajennusmahdollisuudet", Kansalliskirjasto, 2015. (in Finnish) https://www.doria.fi/bitstream/handle/10024/113684/SAPOn%20nykytila%201_2015%20raportti_3182015.pdf?sequence=2
[113] https://finto.fi/ysa/en/
[114] https://finto.fi/allars/en/
[115] https://finto.fi/yso/en/
[116] https://finto.fi/koko/en/
[117] http://wordnet-rdf.princeton.edu/
[118] http://www.ling.helsinki.fi/cgi-bin/fiwn/search
[119] http://www.ling.helsinki.fi/kieliteknologia/tutkimus/finnwordnet/lataa.shtml
[120] http://www2.lingfil.uu.se/swordnet_test/

CC BY

Material description. A metadata vocabulary for the description of materials is available in [121]. It includes terminology from both ISBD and RDA.

Other. Other data sources include e.g. DBpedia[122], [123], which has Wikipedia content in structured form (as RDF data). Also, the Finnish n-grams from historical newspapers between years 1820 and 2000 are available by decade[124]. Furthermore, the Language Bank of Finland[125] and the Swedish Language Bank[126] contain further language resources.

3.3.2 Systems of the National Archives of Finland

Digitized material. The total amount of digitized material at the National Archives of Finland is around 45 million pages. The material originates from many different periods of time from the year 1316 onwards. Also, the material is written in several languages: mostly either in Finnish or Swedish, but also Russian sometimes.

Currently the digitized material is stored for both long-term preservation and for end-user viewing purposes in both **TIFF** format and **JPEG** format. The TIFF files are meant only for long-term preservation and the JPEGs for end-user viewing. For JPEGs, there are two versions: **a high-quality JPEG** (which has the same number of pixels as the corresponding TIFF) and **a low-quality JPEG** (which has one third of the number of pixels in both horizontal and vertical directions of the corresponding TIFF); the corresponding points per inch values are 300 ppi and 100 ppi. Each of the different versions of the document image is put to a separate folder in the directory structure of the storage medium: tiff, jpeg and thumbs for TIFFs, high-quality JPEGs and low-quality JPEGs, respectively. Multi-page image files are not allowed. Out of these image files, the high-quality JPEG would be used as a starting point for the OCR / HTR process. For long-term preservation, image files are packaged to a TAR archive so that all files created (as a result of digitization) within one hour are packaged together.

Searching and viewing the digitized material. Currently, there are several ways or parallel systems for searching and viewing the digitized material at the National Archives of Finland. **Finna**[127] provides a faceted search for the material and has facets for limiting the search results based on e.g. domain, organization, type of material, author and language. The search is based on descriptive metadata and it targets both digitized and analog material. Finna does not currently use full-text when carrying out the search. **Astia**[128] is a system for searching and ordering of material and it enables limiting the search temporally (year range) and, furthermore, keyword suggestions are provided in a list based on what is typed in the search box (keywords starting with the same text that is typed in the search box are suggested). Suggestions are also

[121] http://finto.fi/mts/fi/ (in Finnish)
[122] http://wiki.dbpedia.org/
[123] https://en.wikipedia.org/wiki/DBpedia
[124] http://urn.fi/urn:nbn:fi:lb-2015041001 (in Finnish)
[125] https://www.kielipankki.fi/language-bank/
[126] https://spraakbanken.gu.se/eng
[127] https://www.finna.fi/?lng=en-gb (in English)
[128] https://astia.narc.fi/astiaUi/search.php (in Finnish)

CC BY

provided after the search has been carried out. **Vakka**[129] enables searching by name, free-text or index word (also partial words) and temporal limitations can be imposed (year range). **Digital Archives** search[130] supports search by keywords (also partial keywords) and accessing the material. **AARRE**[131] is specifically for searching military documents and search by keywords (also partial keywords) is supported.

Metadata. Currently metadata is stored in various places: in **Vakka**, in **Digital Archives**, in **AARRE** and **inside image files** (TIFFs, JPEGs). For example, index lists indicating finer than archival unit level structure of the material are stored in Digital Archives and technical metadata (e.g. scanner make and model) plus some brief descriptive metadata are stored in the image file. In the image file metadata there is also an identifier of the form NNNNNNN.KA, which can be used to retrieve metadata from the Digital Archives. The access rights metadata is stored in Digital Archives: it could be on image file level but in practice it is on archival unit level. Also, the structure of the archival material including links to image files is stored in the Digital Archives.

3.3.3 Foreseen future changes

Metadata. AHAA will be a metadata service common for many Finnish archives, which have decided to adopt it. The main objective of AHAA is to enable the management of descriptive metadata and index data of the material. In AHAA, it will be possible to provide metadata to archival material in the form of index terms based on KOKO[132] and place names based on SAPO[133]. Also, it will be possible to manage the restrictions of use and display (but not user access rights) and handling of multiple manifestations of certain archival material will be supported.

A centralized agent/actor/authority database. There is a document describing potential plans concerning a centralized authority database[134]. However, currently the authority data is created within the AHAA system and ISAAR (CPF) and RDA are taken into account in the authority data stored in AHAA.

Long-term preservation. There will be a system for long-term preservation of digital material called PAS. The PAS specification includes e.g. the listing of file formats suitable and acceptable for long-term preservation[135]. For text, CSV, ePUB, XHTML, XML, HTML, ODF, PDF/A and plain text are listed as acceptable formats. For images, DNG, JPEG, JP2, PNG and TIFF are accepted. Most OCR / HTR file formats studied in this document are based on XML/HTML, thus, in that sense they would be directly acceptable for the PAS long-term preservation system.

[129] http://www.narc.fi:8080/VakkaWWW/EtuSivu.action (in English)
[130] http://digi.narc.fi/digi/?lang=en_US (in English)
[131] http://kronos.narc.fi/aarre/aarre.php
[132] https://finto.fi/koko/en/ (in English)
[133] https://onki.fi/en/browser/overview/sapo (in English)
[134] KDK, "KDK:n tietoarkkitehtuuriryhmän nimitietopalvelua koskeva selvitys", V1.0, 2016. (in Finnish) http://www.kdk.fi/images/tiedostot/KDK_Nimitietopalveluselvitys_2016.pdf
[135] KDK, "Säilytys ja siirtokelpoiset tiedostomuodot", v.1.4.0, 19.1.2016. http://www.kdk.fi/images/tiedostot/KDK-PAS-tiedostomuodot-v1.4.pdf

CC BY

The submission of data to PAS and retrieval of data from PAS are based on **Submission Information Packages (SIPs)** and **Dissemination Information Packages (DIPs)**, respectively[136], [137]. Data to be submitted to PAS is packaged as a ZIP file with a mets.xml main metadata file (adhering to the METS format) at the root of the package. SFTP protocol is then used to transfer the SIP to PAS. As a result, an **Archival Information Package (AIP)** is created at the PAS service based on the SIP. The AIP contains additional metadata in comparison to SIP e.g. in the form of event information related to the reception of the package (in the PREMIS metadata format). Afterwards, it is possible to update the data in PAS incrementally including in and **incremental SIP** only those files that have changed. When data residing in PAS is needed, which should happen only seldom e.g. when the use copies of the archival material have been lost, a DIP is requested from PAS. Thus, requesting of DIPs is not directly related to the process of an end-user requesting and viewing archival material. The request is carried out using HTTP-based REST API and it is possible to download either the whole package as ZIP or only the metadata as METS.

Metadata for long-term preservation. One part of the PAS system specifications is the description of METS profile and listing of technical metadata formats accepted for each file format. In general, METS contains e.g. the following metadata sections[138]:

- **Descriptive metadata** <dmdSec>: Reference to external metadata (outside the METS document) or embedded metadata (within the METS document).
- **Administrative metadata** <amdSec>: This section can contain four types of metadata: 1) technical metadata (<techMD>), 2) IPR metadata (<rightsMD>), 3) source metadata (<sourceMD>) and 4) digital provenance metadata (<digiprovMD>).
- **File section** <fileSec>: Lists all the files making up the digital version of the object (e.g. thumbnails, master image files, files containing textual transcriptions of the images).
- **Structural map** <structMap>: Defines the hierarchical structure of the digital object and has links to the files in the <fileSec> of the METS document.

Which METS elements are required is specified in the KDK PAS documentation[139]. The KDK METS profile is common for all Submission Information Packages (SIPs) and Archival Information Packages (AIPs). Also, Dissemination Information Package (DIP) is formed according to KDK METS profile with the exception that it can alternatively contain only the metadata (no actual material/documents). Additionally, PAS events are added to the DIP in the format of PREMIS events. Different types of metadata (e.g. PREMIS and MIX) are wrapped inside the METS file. The definition of the metadata format must be stored in the MDTYPE- or OTHERMDTYPE-attribute of the <mdWrap>-element. In case of descriptive metadata, allowed values for the MDTYPE attribute are MARC, MODS, DC, EAD, EAC-CPF, LIDO, VRA and DDI and allowed values for the OTHERMDTYPE attribute are EN15744 and EAD3. In case of technical metadata, allowed values for the

[136] KDK, ”Aineistojen ja niiden metatietojen paketointi pitkäaikaissäilytykseen”, V1.5.0, 19.1.2016. (in Finnish) http://www.kdk.fi/images/tiedostot/KDK_metatiedot_ja_aineiston_paketointi_v1.5.pdf
[137] KDK, ”PAS-palvelun rajapinnat”, V1.0.0, 19.1.2016. (in Finnish) http://www.kdk.fi/images/tiedostot/KDK_rajapinnat-1.0.0.pdf
[138] http://www.loc.gov/standards/mets/METSOverview.v2.html
[139] KDK PAS, “Aineistojen ja niiden metatietojen paketointi pitkäaikaissäilytykseen”, V1.5.0, 19.1.2016.

CC BY

MDTYPE attribute are PREMIS:OBJECT and NISOIMG and allowed values for the OTHERMDTYPE attribute are e.g. VideoMD, AudioMD and ADDML. The format of the technical metadata depends on the file format of the actual content. For JPEGs and TIFFs, the metadata schema for the mandatory technical metadata is MIX[140] (Metadata for Images in XML Schema) e.g. for EXIF data. For CSV and plain text files, the metadata schema is ADDML. However, some metadata elements of MIX are not recommended to be used (e.g. <fileSize>, <FormatDesignation>) because they overlap with similar elements in PREMIS (i.e. the information should be defined using PREMIS). Thus, PREMIS is used for e.g. indicating the file format (<premis:formatName>) and file format version (<premis:formatVersion>) and file integrity using hashing (<premis:fixity>).

Searching and viewing the digitized material. Finna is already in use as one system for accessing the digitized material and it will continue to do so. In the future, Finna harvests its metadata from AHAA using OAI-PMH[141] (Open Archives Initiative Protocol for Metadata Harvesting) to its own metadata database. OAI-PMH uses HTTP GET and POST and the responses are in XML format (OAI-PMH supports any metadata format encoded in XML; EAD3 and EAC-CPF formats are used in AHAA to offer the metadata for Finna). All search requests in Finna are targeted to the already harvested metadata (i.e. metadata residing in the databases of Finna). When the user types a keyword in the user interface of Finna, a drop-down list appears giving suggestions for the search keyword. There is a search API[142] in Finna: the search request is made using HTTP protocol and the results are given in JSON format. Finna uses in its implementation[143], [144] VuFind[145] (open source) user interface (UI) for search/browsing portal, Apache Solr for metadata indexing and as a search engine, Record Manager (in-house tool) for metadata harvesting and Voikko[146] for Finnish linguistics.

Finally, some materials of the National Archives of Finland are also available via the **Europeana** portal[147]. **Formula**[148], [149] is a service maintained by the National Library of Finland. Its purpose is to enable the delivery of e.g. material metadata to the Europeana portal. Formula harvests the metadata using OAI-PMH protocol and converts it to ESE[150] (Europeana Semantic Elements) or EDM[151] (Europeana Data Model; ESE is a subset of EDM) format and act as repository for metadata harvesting of the Europeana. However, the

[140] http://www.loc.gov/standards/mix/
[141] https://www.openarchives.org/pmh/
[142] https://www.kiwi.fi/pages/viewpage.action?pageId=53839221 (in English)
[143] http://www.doria.fi/bitstream/handle/10024/98935/Erkki%20Tolonen%20-%20Finna%20ja%20ontologiat-1.pdf?sequence=2
[144] https://www.kiwi.fi/download/attachments/51841966/20141023 Finna and Ontologies - Nordlod-Final.pptx?version=1&modificationDate=1414153274497&api=v2
[145] http://vufind-org.github.io/vufind/
[146] http://voikko.sourceforge.net
[147] http://www.europeana.eu/portal/en
[148] http://www.kdk.fi/index.php/fi/europeana-ja-muut-hankkeet/formula (in Finnish)
[149] https://www.doria.fi/bitstream/handle/10024/74744/Europeana_21.3.2012_Sainio.pdf?sequence=1
[150] http://pro.europeana.eu/page/ese-documentation
[151] http://pro.europeana.eu/page/edm-documentation

CC BY

delivery of the materials of the National Archives of Finland to the Europeana portal has not been based on Formula service, but on Archives Portal Europe[152].

3.4 Requirements and possibilities (at National Archives of Finland)

- What kind of requirements are there at the National Archives of Finland concerning the OCR / HTR document formats and where do the requirements come from?
- What kind of requirements are there concerning the usage of full-texts of the documents (if they are available in the future)?

3.4.1 Potential use cases

This section describes future envisaged requirements in the form of use cases. The purpose of listing the use cases is to be able to determine how well the reviewed OCR / HTR file formats fulfill the requirements and to identify the implications of the use cases on processes and other systems. Thus, these use cases are **not meant** to be implemented in the CO:OP project — rather, they refer to a situation in the future.

Table 7. Use cases related to the OCR / HTR issues.

#	Priority	Use case description
1) Process		
1.1		It SHALL be possible to automatically trigger • the execution of the OCR / HTR process • the execution of the Named Entity Recognition process once the digitized document is available.
1.2		It SHALL be possible to manually trigger • the execution of the OCR / HTR process • the execution of the Named Entity Recognition process.
		It SHALL be possible to automatically trigger • the re-execution of the OCR / HTR process • the re-execution of the NER process when the algorithms are better or more (or more suitable) training data is available. (Thus, sufficient metadata about the OCR / HTR process itself should be stored in this case for each document to be able to decide if the re-execution of the OCR / HTR process would lead to increase in the quality of the material. Also, manual curation carried out for the original OCR / HTR material should be taken into account.)
		It SHALL be possible to automatically send a curation request after • automatic OCR / HTR has finished • normal user has provided a document transcription • automatic Named Entity Recognition has finished • normal user has provided Named Entity tags.
		It SHALL be possible to store text recognition confidence values.
2) Multi-user scenarios		

[152] https://www.archivesportaleurope.net/

CC BY

		It SHALL be possible to handle conflicts due to simultaneous editing by several users. This concerns editing of e.g. the transcription text and Named Entities. (One possibility is to lock the material for editing.)
		It SHALL be possible to cross-check the transcriptions and Named Entity tags between users (e.g. by increasing the level of confidence of the transcription if several users provide the same transcription or the same Named Entity tag).
3) Long-term preservation of documents		
		The OCR / HTR text file format SHALL have future support (e.g. be used by several other organizations).
		The OCR / HTR text file SHALL have a link to the TIFF file from which the text was recognized. (There could be a link in the OCR / HTR text file to the TIFF file or e.g. METS XML file could have links / pointers to both OCR / HTR text files and the corresponding TIFF files.)
4) Information service		
		It SHALL be possible to carry out searches based on the full-text of the document.
		It SHALL be possible to leave out words that have likely not been correctly recognized from the search engine index (confidence values are too low).
		It SHALL be possible to give as input (by the user) a threshold value for the word confidence when conducting a search.
		It SHALL be possible to carry out a search defined with a Finnish word also in Swedish and the other way around. Also, a search defined with an English word SHALL be carried out in Finnish and Swedish.
		It SHALL be possible to carry out a search with a more general concept.
5) Automatic metadata generation		
		It SHALL be possible to automatically generate metadata from the OCR / HTR full-text.
6) Viewing / presentation		
		It SHALL be possible to view the digitized document (e.g. TIFF or JPEG) and OCR:ed / HTR:ed text side by side (without editing i.e. read-only; on-line).
		It SHALL be possible to download and view the original documents and the transcriptions (on the user's own computer; off-line). (This could be fulfilled e.g. by bundling the original documents and the transcriptions into a PDF file.)
7) Transcription (crowdsourcing-based or official)		
		It SHALL be possible to manually provide transcription for the document image. Therefore it SHALL be possible to link parts of the transcription (e.g. words, sentences) to specific areas (defined by coordinates) of the document image (e.g. TIFF or JPEG) file. (These transcriptions could then serve as e.g. training data for automatic text recognition.)
		It SHALL be possible to edit an already manually transcribed document and an already automatically recognized text (i.e. curate the transcription or recognition results).
8) Validation tool for digitization		

CC BY

		It SHALL be possible to extract and recognize the (printed / written) page numbers of the document and present them to the end-user.
9) Named Entity Recognition (NER)		
		It SHALL be possible to manually provide Named Entity tags to the document transcriptions. This SHALL be possible for both archive employees (official NER) and normal users (crowd-sourcing based NER).
		It SHALL be possible to edit the already manually provided Named Entity tags and the already automatically recognized Named Entities (i.e. curate the Named Entity tags).
		It SHALL be possible to automatically recognize Named Entities in the document transcription.
		It SHOULD be possible to store a confidence value for each Named Entity tag i.e. a value indicating the probability that the corresponding word is really of the indicated Named Entity type.
		It SHALL be possible to provide links between Named Entities in the archival documents and Named Entity databases (thus, possibly disambiguating the Named Entities).
10) Restrictions of use		
		It SHALL be possible to automatically recognize potential restrictions of use for the digitized material (e.g. based on temporal coverage of the material).
		It SHALL be possible to provide assistance for the blackening of words in case of restricted material. (Names of people are already personal information as such and can have implications on restrictions of use.)
11) Social metadata		
		It SHALL be possible for normal users to provide tags (either from a controlled vocabulary or any tags) for documents.
		It SHALL be possible for normal users to bookmark documents for later use (login would be necessary in this case).
		It SHALL be possible for normal users to provide ratings for documents. (ratings could indicate the subjective quality, relevance or interest of the documents)
		It SHALL be possible for normal users to provide links to external resource for documents.

3.5 Potential implications (at National Archives of Finland)

3.5.1 Potential implications on processes

Digitization process and OCR / HTR. As long as the resolution of the input image is good enough for OCR / HTR, the digitization itself would not be affected by the introduction of an OCR / HTR file format. However, in the prioritization of the material for digitization, the suitability of the material for automatic OCR / HTR (i.e. how well automatic OCR / HTR work for certain material) could be taken into account as one criterion. Furthermore, the following issues should be considered:

- When and by whom would the OCR / HTR process be executed for a scanned document?
- When and by whom would the OCR / HTR result (recognized text) be curated?

CC BY

- When and by whom would the OCR / HTR process be re-executed?
- Based on what information would the decision be made about the re-execution?
- When and by whom would any other procedures such as Named Entity Recognition and keyword extraction be executed and their results curated?

3.5.2 Potential implications on existing or future systems

The introduction of an OCR / HTR file format and associated processing can have implications on several existing or future systems and this section addresses them.

The characteristics of the archive material (i.e. material from various periods of time and written in various languages) have several implications on OCR / HTR issues. For example, the training data for each document to be processed should be carefully chosen so that the characters used in the training material are similar to the characters used in the material to be recognized. Furthermore, because the confidence levels of the OCR / HTR for the text would likely depend a lot on the age of the material (the writing of older documents is more obscure than of the more recent documents), bigger part of search hits (for search based on OCRed / HTRed text) would be more recent material rather than older material. This could give the false impression that the search keyword does not appear so often in older material even though it really appears, but it just has not been automatically recognized in older material. Also, the vocabulary used and variations in spelling can differ between older and more recent material: this can have implications on e.g. automatic Named Entity Recognition algorithms, if the algorithms are trained with old material, but applied to more recent material. The multilingual nature of the material would benefit from automatic translation of search query words so that search would be automatically carried out in each language; this could be accomplished with multilingual vocabularies / ontologies, which record correspondence between words in e.g. Finnish and Swedish.

AHAA. A new manifestation is needed in AHAA metadata system for OCR:ed / HTR:ed material stored in any of the reviewed file formats. Also, there should be a link backwards from the OCR / HTR manifestation to the image file from which the text was recognized. Furthermore, if metadata automatically extracted from the full-text of the document or social metadata (provided by normal users) is stored in AHAA, they might need to be differentiated from the official metadata somehow (since they might be less reliable) or curated (which, however, would require additional effort).

PAS. If the OCR / HTR results are put to PAS (long-term preservation), links to OCR / HTR files should be put to e.g. METS structMap section. Furthermore, the same issues arise for PAS as for AHAA concerning the automatically extracted metadata and social metadata: it should be decided, where in PAS metadata descriptions is such metadata stored if it is stored at all. Also, event-related metadata concerning the creation of the OCR / HTR file should be stored in PREMIS metadata format. If a lot of human and computational effort has been expended on OCR / HTR, collecting social metadata and automatic extraction of metadata, it would be useful to be able to restore that data in case it is lost from back-end systems.

CC BY

Finna. There could be several implications on Finna. For example, there could be need to enable using the full-text (from the OCR / HTR process) in the search. This would require the full-text to be indexed into the Finna search engine database and it would clearly increase the indexing burden in comparison to indexing only the metadata. This could also call for modifications in the interfaces of Finna: the currently used OAI-PMH interface is meant for harvesting metadata to Finna, but harvesting the whole full-text would be a different issue. Also, if OCR / HTR confidence values are to be used in the search so that the end-user can provide a threshold value for the confidence, even bigger changes would be needed in the indexing and search functionality. Furthermore, there could be implications on the user interface of Finna: e.g. it would be good to be able to allow full-text search or limit the search to metadata only to keep the number of search hits manageable and to decrease the computational load of the servers. In the user interface, this could be accomplished with new features in faceted search. Multilingual search could also be relevant when more material (e.g. full-text) is available; this would mean that the search is carried out automatically using both the given Finnish word and the corresponding Swedish word and could be accomplished using dictionaries or ontologies (having mapping between corresponding concepts). Finally, functionality for displaying the automatically recognized text in connection with the scanned document (e.g. JPEG) would be needed.

3.5.3 Potential new systems / functionality needed

This section describes what kind of systems / functionality would need to be introduced to fulfill the requirements described previously. Also, links to relevant publications and already existing solutions are given.

E.g. the following systems / functionality would be needed:

Text recognition (OCR / HTR) system. This should encompass e.g. functionality for providing ground truth data (i.e. manual transcriptions of documents), functionality for manual and automatic selection of ground truth data for each material to be OCR:ed / HTR:ed and functionality for storing ground truth data and text recognition results in the chosen OCR / HTR file format.

The special case of **tabular document images** (i.e. documents that consist of tables that have been filled in) has been addressed e.g. in [153]. In that paper, several images containing the same tabular form are first registered (i.e. aligned to each other with respect to translation and scaling) and then the blank form is recovered by taking a median of several images at each pixel location. After that, the pixels containing handwriting for each image are detected based on the recovered blank form and the individual images. In [154], the concept of **waypointing** is addressed. Waypointing means the detection of points where something changes in a series of document images (either on the document level or on the document field level). In case of tabular document images, the **regions of interest** (e.g. document header and footer) are manually indicated and automatic comparison of the fields (i.e. content of the regions of interest) is carried out to point out to the user which of the fields have changed between consecutive tabular document images. This

[153] R. T. Clawson et al, “Extraction of Handwriting in Tabular Document Images”, Family History Technology Workshop, 2012.
[154] K. Bauer, “Better Historical Document Indexing Using Waypointing and ROI Data”, 2013.

CC BY

kind of approach implies that the user needs to manually input to the system only the information that has changed.

Named Entity Recognition (NER) system. This can encompass recognition of entities such as names of people, names of locations (cities, countries, etc.), names of organizations, expressions of time (years, months, etc.) and quantities and is part of **Information Extraction** (IE). We should differentiate between two types of people / organizations: 1) those participating in or being responsible for the production / creation of specific archival material and 2) those just appearing the archival documents (e.g. in a list of people travelling in a ship). The main difference is the fact that for the former case, the disambiguation of the names is more relevant and realistic whereas for the latter case, the disambiguation of the names can be very difficult or even impossible.

Tokenization (i.e. splitting the document full-text into words or equivalent entries) is the first preprocessing step that is needed for Named Entity Recognition. For highly inflected languages, such as Finnish, **stemming or lemmatization** (or normalization) of words is a necessary prerequisite for successful Named Entity Recognition; this means that the inflection suffixes are basically removed from words. There are several stemmers / lemmatizers for the Finnish language, e.g. Omorfi[155], Lingsoft Fintwol[156], Snowball stemmer[157] and FinnPos[158]. Other preprocessing steps can include e.g. part-of-speech tagging. Named Entity Recognition can be **rule-based** relying on hand-crafted rules or **machine learning based** relying on training data. These require functionality for manual provision of recognition rules and functionality for provision of ground truth data (i.e. manually indicating Named Entities), respectively.

A highly significant feature for Named Entity Recognition is the initial capital letter of a word[159], however, in older texts there might be capitalizations even when the word is not a proper name or vice versa. Also, part-of-speech (POS) tags can be used as features for NER, however, it is unclear how well the part-of-speech taggers perform on old texts. Some features used in Named Entity Recognition are described and evaluated in [160]. The features include those based on local knowledge such current, preceding and subsequent token and their generalizations such as suffixes, prefixes and shapes (e.g. encoded as Xxx and Xxx-xx). Also, features based on or utilizing external knowledge such as part-of-speech tags, word clustering information and various gazetteers (e.g. Wikipedia and DBPedia). The paper [161] brings up similar issues related to features, but also addresses other issues related to Named Entity Recognition such as the representation schemes for text segments (e.g. BILOU and BIO). For historical material, whose text has

[155] https://github.com/flammie/omorfi
[156] http://www2.lingsoft.fi/cgi-bin/fintwol
[157] http://snowballstem.org/
[158] M. Silfverberg et al, "FinnPos: An Open-Source Morphological Tagging and Lemmatization Toolkit for Finnish", 2016. https://users.ics.aalto.fi/tpruokol/papers/silfverberg2016finnpos.pdf
[159] C. Grover et al, "Named Entity Recognition for Digitised Historical Texts", Language Resources and Evaluation, 2008.
[160] M. Tkachenko et al, "Named Entity Recognition: Exploring Features", Proceedings of KONVENS 2012.
[161] L. Ratinov et al, "Design Challenges and Misconceptions in Named Entity Recognition", Proceedings of the 13th CoNLL, 2009.

CC BY

been automatically recognized, there are some challenges for NER, e.g. impact of OCR / HTR errors[162], impact of historical spelling variants and impact of names of people appearing without context (e.g. in tables). Also, e.g. in lists of people, characters reminiscent of quotation marks (“) are sometimes used to denote that the title (profession) of a person appearing in the previous line should be used in the current line as well and, thus, this information should be taken into account.

Some existing Named Entity Recognizers include e.g. FiNER (a rule-based NER for the Finnish language) and SweNER[163]. A Named Entity Recognition study[164] conducted for the historical newspaper material of the National Library of Finland has concluded that with the tested NER tools (FiNER and SeCo’s ARPA), about half of Named Entities can be recognized even in the presence of OCR errors. The study also utilized existing sources of place names, e.g. the Finnish Place Name Registry, the Finnish spatio-temporal ontology (SAPO), repository of old maps and associated places and a name registry of places in historic Karelia. Furthermore, a low precision of NER (in one evaluation) for place names was attributed to the fact that many names were both person and place names. For person names, the Virtual International Authority File was used in one evaluation. Finally, Levenshtein distance was used for fuzzy matching as an attempt to circumvent issues related to OCR errors. Another study[165] utilized the Stanford NER tagger and evaluated it for French, German and Dutch newspaper material. In [166], TEI `placeName` and `persName` tags are used for manually annotated (which could be based on automatic NER pre-tagging) material to encode the information about specific Named Entities. Furthermore, references to other external databases for place and person names can be provided in connection with the tags mentioned. In [167], a set of 100 pages of historical newspaper text was manually annotated with respect to Named Entities for both French and Dutch and 200 pages for German. The Stanford NER tagger was used and the output was stored in ALTO format utilizing the `NamedEntityTag`. The Named Entity Recognition evaluations carried out yielded precision values of around 0,94-0,95 for Dutch and 0,7-0,84 for French and recall values of around 0,56-0,76 for Dutch and 0,58-0,83 for French.

[162] K. J. Rodriquez et al, “Comparison of Named Entity Recognition tools for raw OCR text”, Proceedings of KONVENS 2012.
[163] D. Kokkinakis et al, “HFST-SweNER — A New NER Resource for Swedish”, 2014. http://www.lrec-conf.org/proceedings/lrec2014/pdf/391_Paper.pdf
[164] Kimmo Kettunen et al, “Modern Tools for Old Content — in Search of Named Entities in a Finnish OCRed Historical Newspaper Collection 1771-1910”, Proceedings of LWDA, 2016.
[165] C. Neudecker et al, “Large-scale refinement of digital historic newspapers with named entity recognition”, IFLA Newspapers/GENLOC Pre-Conference Satellite Meeting, 2014. http://www.ifla.org/files/assets/newspapers/Geneva_2014/s6-neudecker_faber_wilms-en.pdf
[166] C. Thomas et al, “Standardized Information on historical Proper Names in Digital Full Text Transcriptions. Crowdsourcing ref=”[ID]”s for <placeName> and <persName> tags in the corpora of the German Text Archive / Deutsches Textarchiv”, DCH2015 Berlin. http://www.deutschestextarchiv.de/files/DCH2015_Berlin_Crowdsourcing-ref-ids-in-the-DTA-2.rtf.pdf
[167] C. Neudecker, “An Open Corpus for Named Entity Recognition in Historic Newspapers”, Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), 2016. http://www.lrec-conf.org/proceedings/lrec2016/pdf/110_Paper.pdf

CC BY

NER as such can be a preprocessing step for other tasks[168] such as ontology population, relation extraction and text classification. More information about NER in digital humanities can be found e.g. in the presentations of a workshop "Named entity recognition in digital humanities" held in 2015[169].

Automatic metadata generation system. This kind of system could include functionality related to e.g. **keyword extraction**. For keyword extraction, the Java-based Maui[170] can e.g. assign terms with a controlled vocabulary and extract keywords. A Python-based library RAKE[171] (Rapid Automatic Keyword Extraction) is also for keyword extraction. A keyword extraction tutorial[172] shows the basic usage of both Maui and RAKE. In [173], the issue of **automatic generation of structural and descriptive metadata** for scanned document material for aiding content access is addressed. The OCRed text is stored in DjVu XML file and used as input for the metadata generation. Also, the DjVu XML file contains structural information (page, paragraph, line, word) and coordinates of the bounding box of each word. Various types of features (style features, semantic and linguistic features, structure and context features, font features) are extracted based on the DjVu XML file. The metadata generation is carried out in both rule-based and machine-learning based (using Support Vector Machine) way for volume, issue and article level metadata. Levenshtein distance is used to tackle the OCR errors when matching strings. In [174], **bibliographical and quantitative metadata** was extracted from METS and ALTO files of OCRed material. The metadata include e.g. number of words, illustrations and tables and content types. Furthermore, visualizations of the metadata with respect to e.g. time are shown in the paper. Some scripts used for extracting the metadata are available in [175].

Other text mining / NLP tasks. There are also other text mining and NLP tasks that could be executed for textual material. Some are based on previous processing steps such as Named Entity Recognition. In [176], other types of Information Extraction tasks in addition to NER are listed and described: **Co-reference Resolution (CO)**, **Relation Extraction (RE)** and **Event Extraction (EE)**. For OCRed newspaper material, some possibilities for text mining, such as article segmentation/extraction are pointed out in[177]; also, an example

[168] D. Kokkinakis et al, "HFST-SweNER — A New NER Resource for Swedish", Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), 2014. http://www.lrec-conf.org/proceedings/lrec2014/pdf/391_Paper.pdf

[169] https://kitwiki.csc.fi/twiki/bin/view/FinCLARIN/KielipankkiTapahtumaNimentunnistusDigitaalisissaIhmistieteissaTyopaja

[170] http://entopix.com/maui/

[171] https://github.com/zelandiya/RAKE-tutorial

[172] https://www.airpair.com/nlp/keyword-extraction-tutorial

[173] Xiaonan Lu et al, "A Metadata Generation System for Scanned Scientific Volumes", JCDL'08, 2008.

[174] Jean-Philippe Moreux, "Innovative Approaches of Historical Newspapers: Data Mining, Data Visualization, Semantic Enrichment: Facilitating Access for various Profiles of Users", 2016. http://www.euklides.fr/blog/altomator/EN-DM/article_en2.pdf

[175] https://github.com/altomator/EN-data_mining

[176] J. Piskorski et al, "Chaper 2: Information Extraction: Past, Present and Future" in T. Poibeau et al (eds.), "Multi-source, Multilingual Information Extraction and Summarization", 2013.

[177] Kimmo Kettunen, "Tekstinlouhinnan mahdollisuudet Digin historiallisessa sanomalehtiaineistossa", slides, 2015. (in Finnish)

CC BY

visualization of a news map is show in the slides. In [178], the task of flood detection from newspaper archives is considered. The approach proposed includes the usage human labor harnessed with Amazon's Mechanical Turk to generate reference data. GeoNames.org is used for detecting names of places and other tools include Python and NLTK (Natural Language ToolKit), Scikit Learn and NumPy. Furthermore, the calculation of TF-IDF (Term Frequency – Inverse Document Frequency) is used in the summarization part to select the most representative text snipped for a news article. **Automatic categorization** of newspaper sections is addressed in [179]. A manual analysis of the sections of some historical newspapers is carried out first and a test data set created. Then, an attempt is made to automatically identify for each page which of the five types of sections considered it contained. One approach in the paper was based on page-level word lists (i.e. word counts on page level). The paper [180] addresses the task of **text reuse detection**. Several different text similarity measures are analyzed in the paper and classified into three different categories of measures: content similarity, structural similarity and stylistic similarity. The content based similarity measures treat the texts e.g. as sequences of characters and calculate in various ways how similar the strings or substrings appearing in two texts are. Structural similarity measures can be based on e.g. stopword n-grams or part-of-speech n-grams. In [181], text reuse detection is pointed out as one task for historical newspaper data to detect the sharing of certain texts in several newspapers. **Record linkage** between various historical sources is one relevant task that can provide more information related to specific entities appearing in one source. In [182], the Noordelijke Monsterollen Databases containing lists of crew members for ships from the time period between years 1803 and 1937 and the archives of the National Library of the Netherlands have been linked. The text for the newspaper archives has been obtained using OCR and ships appearing in a text having the same name have been disambiguated using the last name of the captain, type of the ship and the year. In [183] the objective was to do **topic classification** (multi-label) based on oral history interviews converted into text using Automatic Speech Recognition (ASR). Free text and thesaurus-based approaches were used together. The thesaurus contained both part-whole relations and is-a relations. Some issues to tackle noted in the study included recognition errors (e.g. word error rates between 15% and 50%) and variations in recognition errors based on speaker; these issues have analogies in OCR / HTR (e.g. variations in HTR rates based on writer). The approach adopted for topic classification was based on term frequencies and inverse document frequencies and, additionally, leveraged temporal information about the instant of time in the interview that certain topics occurred.

[178] A. Yzaguirre et al, "Newspaper archives + text mining = rich sources of historical geo-spatial data", 9th Symposium of the International Society for Digital Earth (ISDE), 2016.
[179] R. B. Allen et al, "Automated Processing of Digitized Historical Newspapers beyond the Article Level: Sections and Regular Features" in G. Chowdhury et al (eds.), ICADL 2010, LNCS 6102, 2010.
[180] D. Bär et al, "Text Reuse Detection Using a Composition of Text Similarity Measures", Proceedings of COLING 2012, December 2012.
[181] T. Pääkkönen et al, "Exporting Finnish Digitized Historical Newspaper Contents for Offline Use", D-Lib Magazine, Volume 22, Number 7/8, July/August 2016. http://www.dlib.org/dlib/july16/paakkonen/07paakkonen.html
[182] A. C. Bravo-Balado, "Linking historical ship records to newspaper archives", Master's Thesis, 2014.
[183] J. S. Olsson et al, "Improving Text Classification for Oral History Archives with Temporal Domain Knowledge", SIGIR'07, 2007.

CC BY

Image analysis. In addition to the analysis of textual data (created as a result of manual transcription or automatic OCR / HTR process), the analysis of images appearing in the document material can be relevant. In [184], an image analysis approach to detecting image genres in historical newspapers is adopted. The image features used are based on eigenvector representation of the images. The image genres considered are portrait, text characters, exterior views of buildings, full body, half body and group of people and the number of images used is only 60. Clustering based on Self-Organizing Map and K-means is carried out for the manually extracted images and then classifiers based on back-propagation and simulated annealing are trained for the data and evaluated. Also, image retrieval using query-by-example approach is also analyzed. In [185], automatic annotation of images in historical archives is considered. The images can be e.g. heraldic shields. Features based on color, shape, texture and text are mentioned and an algorithm for determining the weights related to combining distance measures based on various features is proposed.

Social metadata provision system. The results of an extensive study related to social metadata in the LAM (Libraries, Archives and Museums) sector is provided in three reports[186], [187], [188]. The study has carried out a site review for 76 sites that have relevance in the LAM sector and that supported some social media features. Out of the 76 sites, about 50% were from USA. A more detailed site review was carried out for a subset of 24 sites. Also, a survey was sent to site managers (42 responses received) in October-November 2009. Some observations mentioned in the reports are:

- Comments, tagging and RSS were the most common social media features.
- More than 50% of the sites used a controlled vocabulary and of those 64% used Library of Congress Subject Headings (LCSH).
- Most sites index user-contributed metadata, however, only 39% of survey respondents incorporate the user-contributed metadata into their own metadata workflows.
- 50% of sites edit user contributions before they are posted.
- Social media features offered by the site in the order of how frequently they are offered: comments, tagging, RSS, annotations, user profiles, user-contributed images, bookmarks, reviews, user-compiled lists, edit text, user awareness (who's logged on), form sub-groups, user recommendations, other, user-contributed video, collaborative filtering and synchronous chat.

Crowd-sourcing system. This kind of system can include functionality for e.g. providing and correcting transcriptions of text and functionality for providing and correcting Named Entity tags. In [189], [190], the **OCRUI**

[184] R. B. Allen et al, "What to Do With a Million Pages of Digitized Historical Newspapers?", iConference 2010.
[185] Xiaoyue Wang et al, "Annotating Historical Archives of Images", JCDL'08, June 16-20, 2008.
[186] Karen Smith-Yoshimura et al, "Social Metadata for Libraries, Archives and Museums. Part 1: Site Reviews", September 2011.
[187] Karen Smith-Yoshimura et al, "Social Metadata for Libraries, Archives and Museums. Part 2: Survey Analysis", December 2011.
[188] Karen Smith-Yoshimura et al, "Social Metadata for Libraries, Archives, and Museums. Part 3: Recommendations and Readings", March 2012.
[189] http://blogs.helsinki.fi/fennougrica/2014/02/21/ocr-text-editor/
[190] "OCRUI Interface for the correction of OCR text material". http://blogs.helsinki.fi/fennougrica/files/2014/03/VanHemel_webinar.pdf

CC BY

crowd-sourcing system developed at the National Library of Finland is described. It consists of front-end implemented in JavaScript and back-end implemented in Python. It basically enables editing ALTO XML files containing the results of automatic OCR process and exporting the corrected results as text, ALTO XML or PDF.

Indexing and searching system. To index the textual content of the recognized documents, a file format (full-text document file format) specific **parser** is required to, in the simplest case, ignore the XML tags and just index the actual textual content of the document. **Apache Tika**[191] seems to have functionality for extracting both metadata and text from many different file formats[192]. However, in a more advanced case, the confidence values for text recognition and Named Entity Recognition could be taken into account. **Apache Lucene**[193] is an open-source indexing and search library. It supports various types of queries in its Query Parser Syntax[194], e.g. targeting specific fields, wildcard searches (? and *), fuzzy searches (based on edit distance), proximity searches and range searches. **Apache Solr**[195] and **Elasticsearch**[196], [197] are search platforms on top of Apache Lucene.

Finally, various issues related to digital archives and data analysis can be seen e.g. in topics of recent conferences and workshops, such as the workshop of "Computational Archival Science: digital records in the age of big data" in IEEEE International Conference on Big Data, 2016[198].

4 Conclusions

As a summary, the following general level characterizations can be associated with each of the analyzed file formats:

- **PAGE XML, ALTO XML, ABBYY FineReader XML, hOCR, TEI:** Files can be opened and edited with normal text editors (e.g. Notepad), but the pages that the formats describe cannot be rendered without special (viewing) tools. In principle, some parts of a PDF file can also be edited with normal text editors, however, PDF files are more complicated in structure than XML-based files and they can also contain (compressed) binary objects.
- **ALTO XML:** A widely used and recommended format specifically for storing OCR results.
- **TEI:** Very extensive and general format containing a lot of tags. Allows expressing e.g. the certainly related to the recognition of Named Entities.
- **PAGE XML:** Has support for providing ground truth data related to various OCR processing steps (e.g. deskew, binarization) and evaluation metrics.

[191] http://tika.apache.org/
[192] http://tika.apache.org/1.13/formats.html
[193] https://lucene.apache.org/
[194] https://lucene.apache.org/core/2_9_4/queryparsersyntax.html
[195] http://lucene.apache.org/solr/
[196] https://www.elastic.co/products/elasticsearch
[197] https://github.com/elastic/elasticsearch
[198] http://dcicblog.umd.edu/cas/ieee_big_data_2016_cas-workshop/

CC BY

- **hOCR:** The textual content and styling can be viewed using standard HTML tools (web browsers). Used by some free OCR tools as output format.
- **ABBYY FineReader XML:** Used by ABBYY FineReader tools (engine) as output format.
- **PDF:** Can contain both the original document image files based on which the OCR / HTR was carried out and the recognized text (bundling them together). The pages can be viewed and text searches conducted using normal PDF tools (e.g. Adobe Acrobat Reader). **PDF/A** is a specific type of PDA file in which e.g. dynamic content and external references are forbidden.

For detailed information about each of file formats, the reader is encouraged to read the corresponding subsection under the sections 2.3 and 2.4.

The following conclusions concerning other issues can be drawn:

- Extensive evaluations would be useful to gain practical information about how well text recognition (OCR / HTR), Named Entity Recognition and any subsequent processing (e.g. automatic metadata generation, event detection) work for certain type and age of material and what are the main reasons for errors.
- Basic evaluations combined with calculations/estimations would be useful for any system dimensioning purposes e.g. to determine how much computational power is needed to run text recognition, Named Entity Recognition and any subsequent processing. This would also provide input for the issue of whether it is feasible to re-execute text recognition.
- Storing several recognition variant words in the OCR / HTR output files can be useful for e.g. evaluation of the OCR / HTR algorithms and for semi-automatic text recognition. In the latter case, the human user would utilize the recognition variants to rapidly choose the correct one. However, trials would be useful to determine how usable for the human curator the provision of the recognition variants would be in comparison to not providing them.
- There are many sub-issues mentioned in this document (e.g. social metadata, indexing and searching, text mining for historical documents) and references for further information given for them. Each of them would be worth a separate, more extensive study (including some trials or evaluations).

Furthermore, there are many references in this document, but to highlight just a few due to the fact that they seem especially relevant for certain topic, the following ones can be mentioned:

- Succeed-project deliverable “D4.1 Recommendations for metadata and data formats for online availability and long-term preservation”, 16.1.2014.
- “The benefits and risks of the PDF/A-3 file format for archival institutions. An NDSA report”, February 2014.
- Karen Smith-Yoshimura et al, “Social Metadata for Libraries, Archives, and Museums”, Parts 1, 2 and 3, 2011-2012.

CC BY

Finally, Table 8 provides a list of issues that should be taken into account related to the introduction of the OCR / HTR file format and for some of the issues gives a comment.

Table 8. List of issues to be taken into account.

Item nbr	Issue	Possible comments
OCR / HTR file format		
1	Selection of format for storing the OCR:ed / HTR:ed text.	ALTO XML is a widely used and recommended format, both nationally and internationally. It has support for many relevant features (e.g. confidences, manual correction status indication, Named Entities) and updated versions of the ALTO XML have appeared. Therefore, ALTO XML could be used as OCR / HTR output format. However, the development and introduction of new versions of the ALTO XML should be followed, since they can contain new features that should be adopted (e.g. glyph level information).
2	How to indicate the writer in the (HTR) recognized text? (corresponding to the concept of font in OCR)	ALTO XML has tags for indicating the style of the text for printed text (e.g. FONTFAMILY, which is of type string). For handwritten text, maybe the tag FONTFAMILY could be reused (since its values are not restricted) or one possibility could be to use OtherTag to express the writer.
3	Decision concerning the provision of PDF as a downloadable, viewable (off-line) and searchable alternative bundling together the document images (several pages) and texts.	It would be useful to provide an evaluation of PDF size vs. subjective quality of the PDF for various kinds of archive material. I.e. to choose a suitable level of image compression for the document images contained in the PDF.
OCR / HTR file storage		
4	Decision of where to store the OCR / HTR results file: 1) In back-end system serving the end-users 2) In long-term preservation system (PAS)	The long-term preservation system (PAS) is mainly intended for receiving each document once, storing it and being requested for the document only under special circumstances (e.g. when the document in the back-end system has been lost). Thus, recurrent requests for and updates to (OCR / HTR) documents would cause a lot of additional load defeating the original and main purpose of the long-term preservation system. Therefore, the OCR / HTR documents could be edited (e.g. curated) in the back-end system and, if necessary, put into the long-term

CC BY

		preservation system once.
Links to/from OCR / HTR file		
5	Addition of a manifestation type to AHAA for the OCR:ed / HTR:ed text.	
6	Linking the OCR / HTR results file to the OCR / HTR input image file in METS file.	fileSec and structMap sections of the METS file should list the corresponding image and text files.
7	Linking the OCR / HTR results file to the OCR / HTR input image in OCR / HTR results file.	This can be done in ALTO XML using the element Description / sourceImageInformation / filename that should contain the name of the input image.
OCR / HTR, NER and subsequent processing functionality		
8	Automatic OCR / HTR functionality.	
9	Functionality for: A) providing ground truth data (layout and text) (official or crowd-sourcing-based) B) curation of OCR / HTR results.	
10	Automatic NER functionality.	
11	Functionality for: A) providing ground truth data (Named Entities) (official or crowd-sourcing-based) B) curation of NER results.	
12	Functionality beyond OCR / HTR and NER (e.g. event detection, automatic metadata generation, record linkage, automatic document categorization)	First, various types of additional functionality should be analyzed in detail, some references for specific studies are provided in this report. Then, preliminary evaluations of how well the algorithms work on real data could be carried out and based on that a subset of functionality to be implemented could be chosen.
Process-related		
13	Decision of when to update the OCR / HTR results file in the back-end system or in the long-term preservation system.	Curation should automatically be stored in the OCR / HTR results file in the back-end system.
14	Decision of when and based on what a re-OCR / re-HTR action would be triggered.	
Metadata		
15	Decision of what to allow for social metadata tags: 1) Anything 2) Only terms from a controlled vocabulary	As a starting point, it could be useful to limit the tags to a controlled vocabulary (e.g. the Finnish KOKO ontology). However, an evaluation of how well the controlled vocabulary covers various kinds of archival document topics could be useful.
16	Decision of when to start showing and using the social metadata tags with the material: 1) Show/Use immediately after somebody has tagged	Alternative 3) carries along additional burden of work and partially defeats the original purpose of social metadata tagging (i.e. using external workforce to carry out tasks for which there are

CC BY

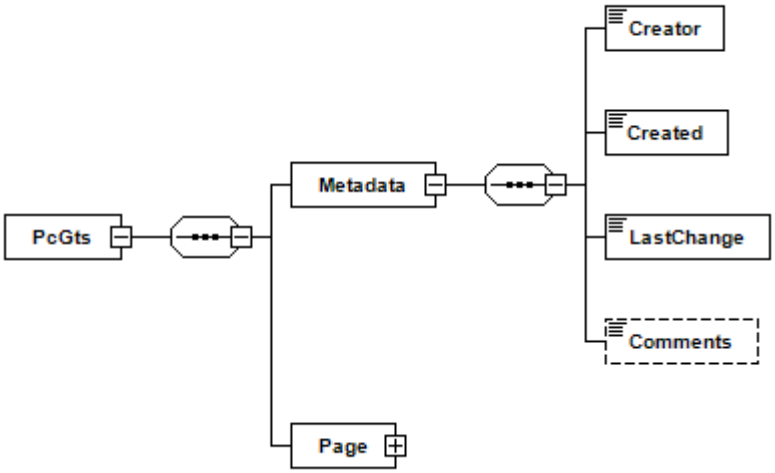
	2) Show/Use after peer-control (e.g. two people provide the same tag) 3) Show/Use after curation (by e.g. employees of the archive)	not enough internal resources). Alternative 2) might be too restrictive because several people might not tag the same data.
17	Decision of how to use the social metadata tags: A) Enable searching (e.g. via a facet, keeping the social metadata tags separate from official metadata) B) Display when viewing the material	
18	Decision of what to allow for automatically extracted metadata tags: 1) Anything 2) Only terms from a controlled vocabulary	The automatic metadata extraction algorithms could propose e.g. keywords that differentiate most certain document from the rest of the documents.
19	Decision of where to store social metadata tags in: A) AHAA: using which field and how to indicate that this is social metadata and not official metadata). B) PAS (if stored there): using which metadata format and how to indicate that this is social metadata and not official metadata. C) Finna.	
20	Decision of where to store automatically extracted metadata tags in: A) AHAA: using which field and how to indicate that this is automatically extracted metadata and not official metadata). B) PAS (if stored there): using which metadata format and how to indicate that this is automatically extracted metadata and not official metadata. C) Finna.	
21	Functionality for social metadata provision e.g. in Finna.	Is Finna the right place to provide social metadata?
22	Addition of OCR / HTR event to PREMIS metadata in PAS.	Event related to the OCR / HTR process execution.
23	Decision of whether to try to link the recognized Named Entities (at least for part of the Named Entities) to some person / organization database.	In most cases, there is not enough information in the archival document to do that i.e. to disambiguate the person / organization.
Registration and login		
24	Decision of whether to require registration & login for e.g. social metadata contributors and text transcribers.	Requiring registration & login could increase the quality of the metadata and transcriptions provided. However, requiring only minimal amount of personal information to be provided or enabling the usage of some already existing

CC BY

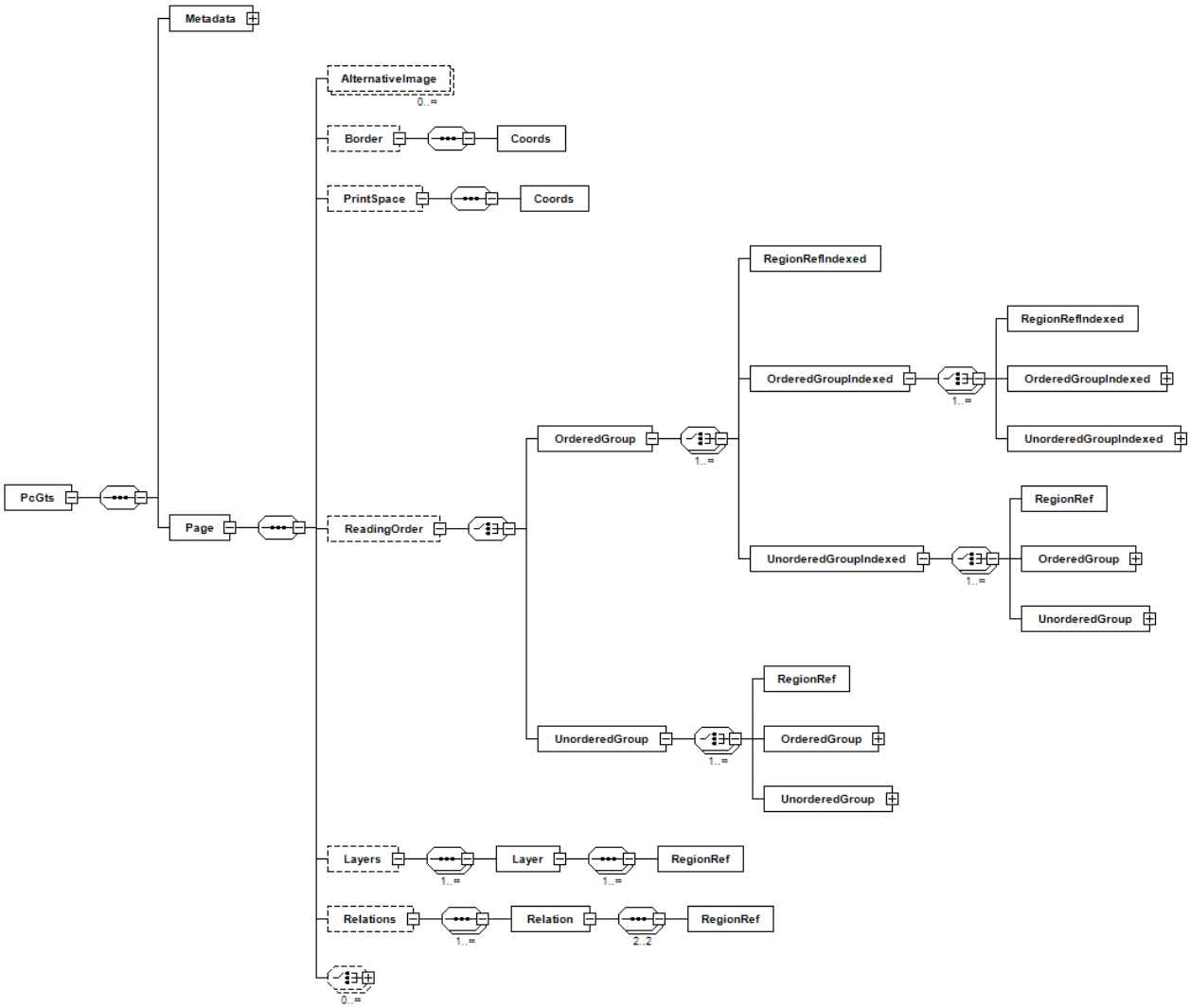
		user accounts for the login would be useful. Also, CAPTCHAs could be used to prevent robot registrations.
25	Decision of how to provide the login.	E.g. providing the login using some already existing service for which users typically have an account.
Search and access		
26	Decision of how to harvest and index the full-text to Finna.	Current metadata harvesting protocol for Finna is OAI-PMH. Is a new protocol needed for full-text harvesting? The recognized text should be parsed from the OCR / HTR results file (which contains e.g. coordinate information as well). Optionally including a threshold value for the confidence indicating whether to include a word in the index or not.
27	New facets for Finna search: A) Automatic metadata B) Social metadata C) Full-text D) Various Named Entities (persons, locations, organizations)	This would enable limiting the search to certain type of data.
28	New functionality for Finna search: A) Giving confidence threshold as a parameter	The confidence values for text recognition need to be stored in the Finna search index for this to work.
29	Automatic translation of search keywords in Finna.	Partially this could be accomplished using vocabularies / ontologies that define the words / concepts in several languages.
30	Automatic search query expansion to more general concepts in Finna.	Partially this could be accomplished using ontologies that define a hierarchy for concepts.
31	Europeana and harvesting, searching and accessing.	Similar issues concern Europeana as Finna (issues listed above). However, for simplicity, Europeana is listed here only as one issue.

CC BY

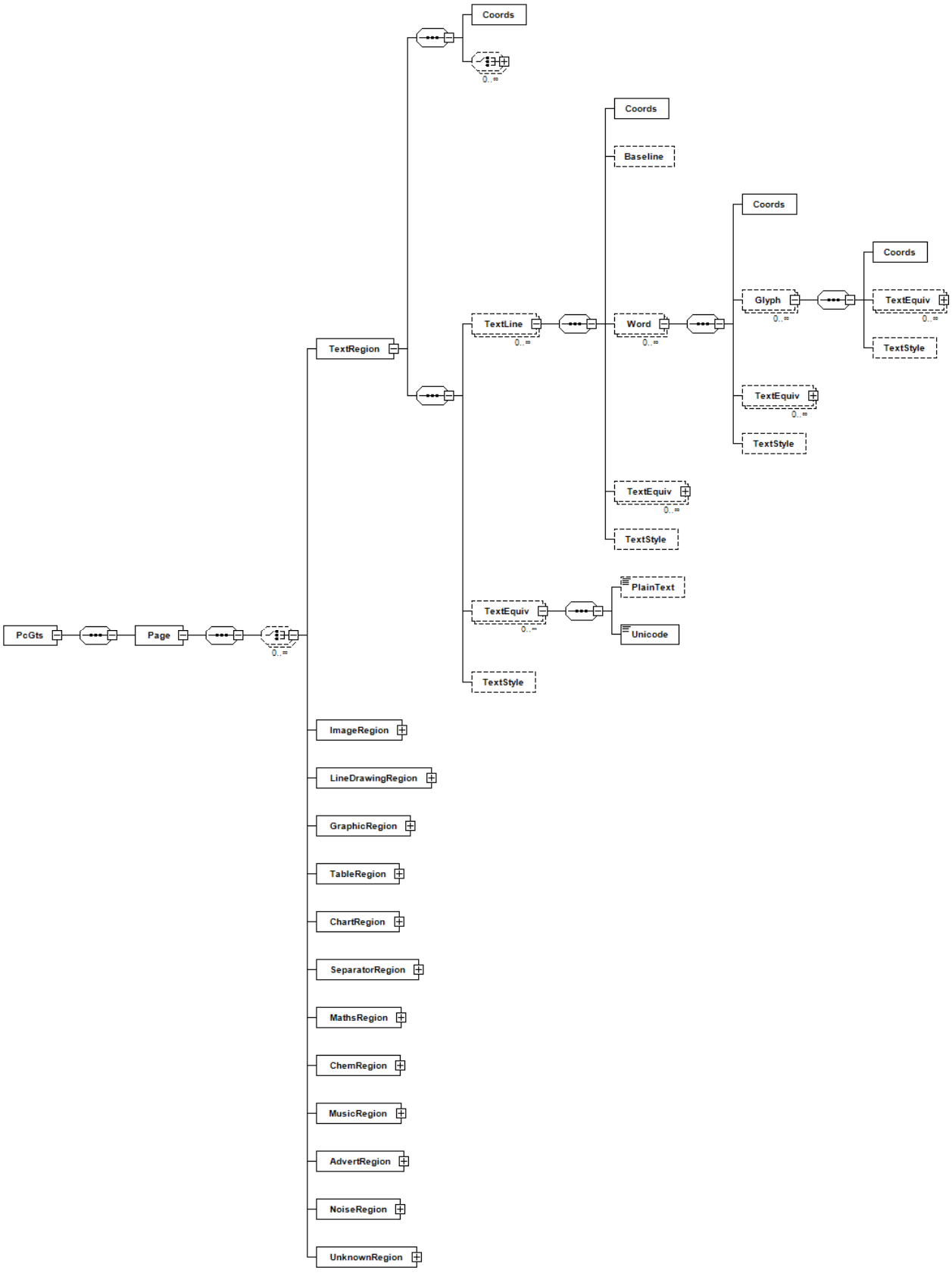
Appendix I: Visualization of the PAGE XML schema



CC BY

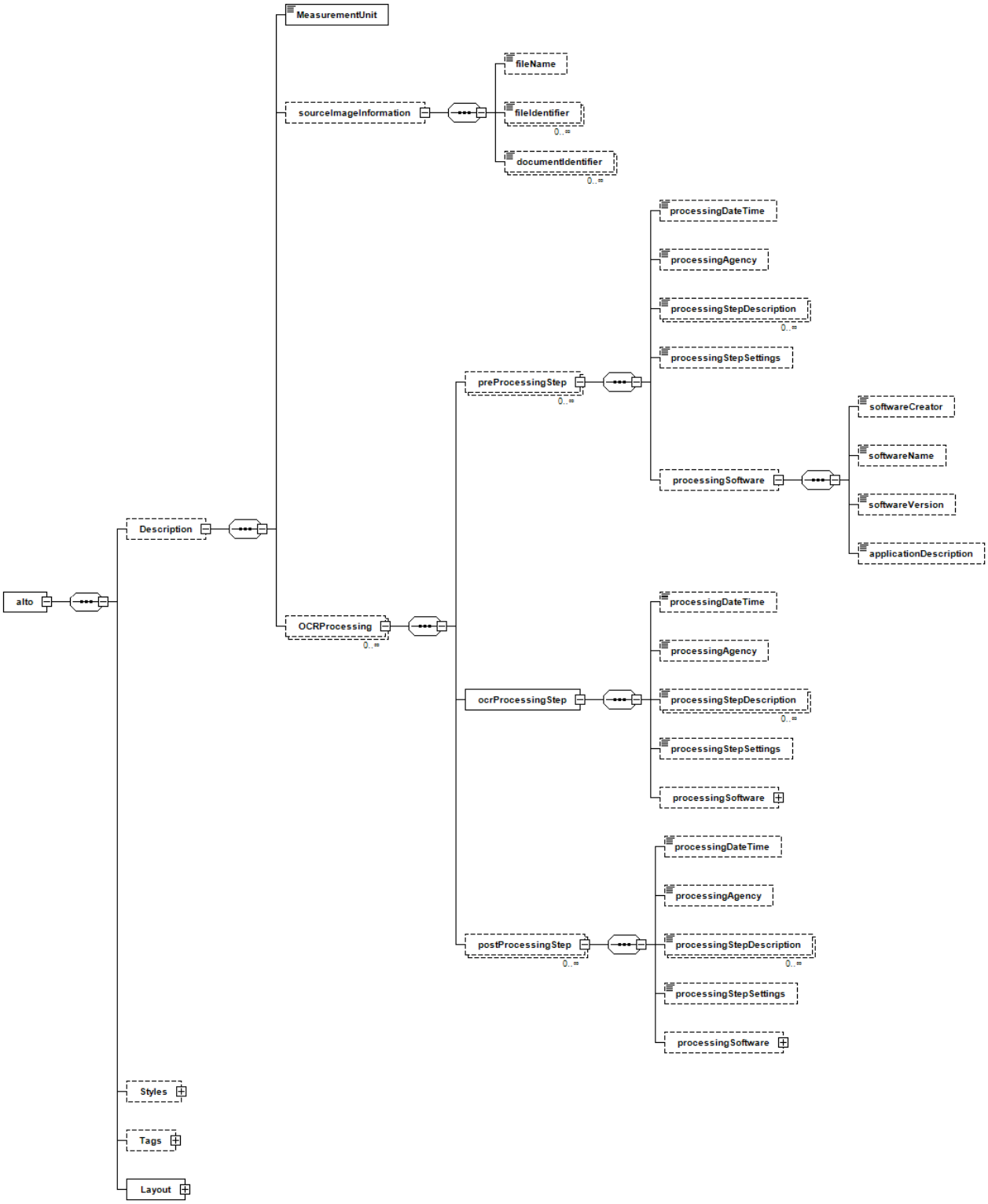

Metadata
AlternativeImage
0..∞
Border
Coords
PrintSpace
Coords
RegionRefIndexed
RegionRefIndexed
OrderedGroupIndexed
OrderedGroupIndexed
UnorderedGroupIndexed
OrderedGroup
1..∞
PcGts
Page
ReadingOrder
RegionRef
OrderedGroup
UnorderedGroup
UnorderedGroupIndexed
RegionRef
OrderedGroup
UnorderedGroup
Layers
Layer
RegionRef
Relations
Relation
RegionRef
2..2

CC BY

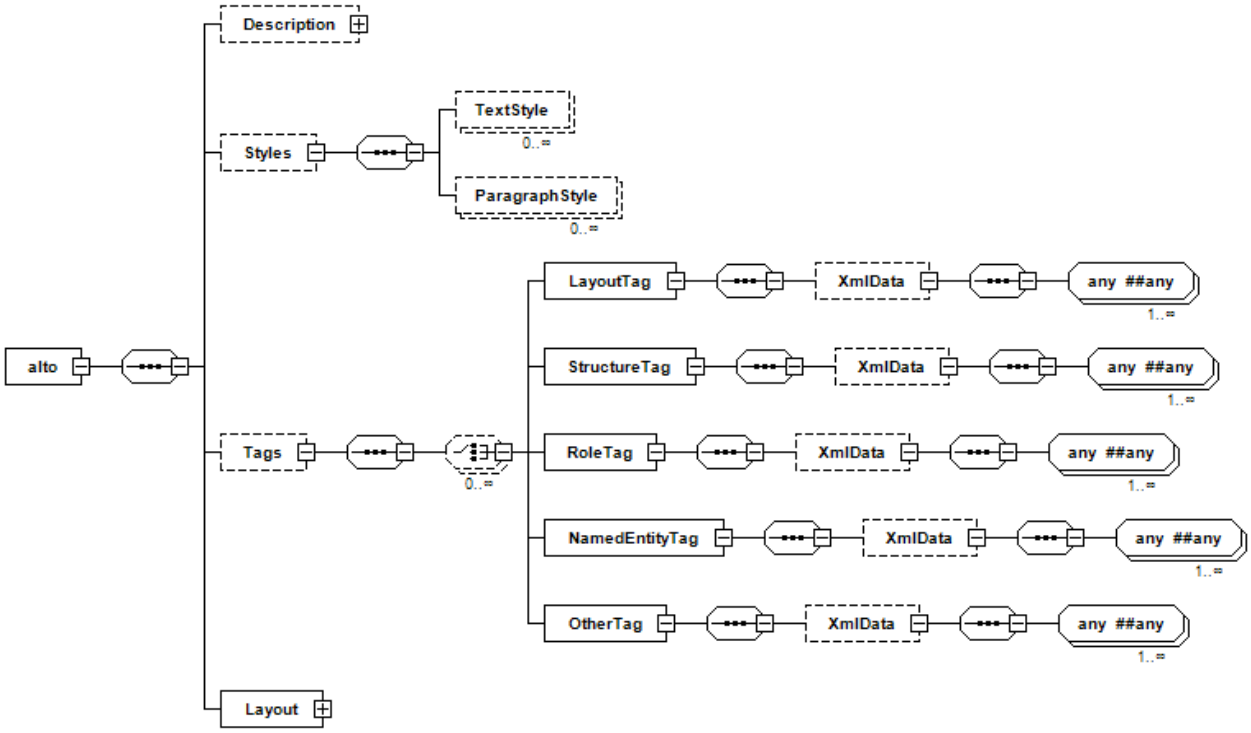

Coords
0..∞
Coords
Baseline
Coords
Coords
Glyph
0..∞
TextEquiv
0..∞
TextStyle
TextLine
0..∞
Word
0..∞
TextRegion
TextEquiv
0..∞
TextStyle
TextEquiv
0..∞
TextStyle
PlainText
TextEquiv
0..∞
Unicode
PcGts
Page
0..∞
TextStyle
ImageRegion
LineDrawingRegion
GraphicRegion
TableRegion
ChartRegion
SeparatorRegion
MathsRegion
ChemRegion
MusicRegion
AdvertRegion
NoiseRegion
UnknownRegion

CC BY

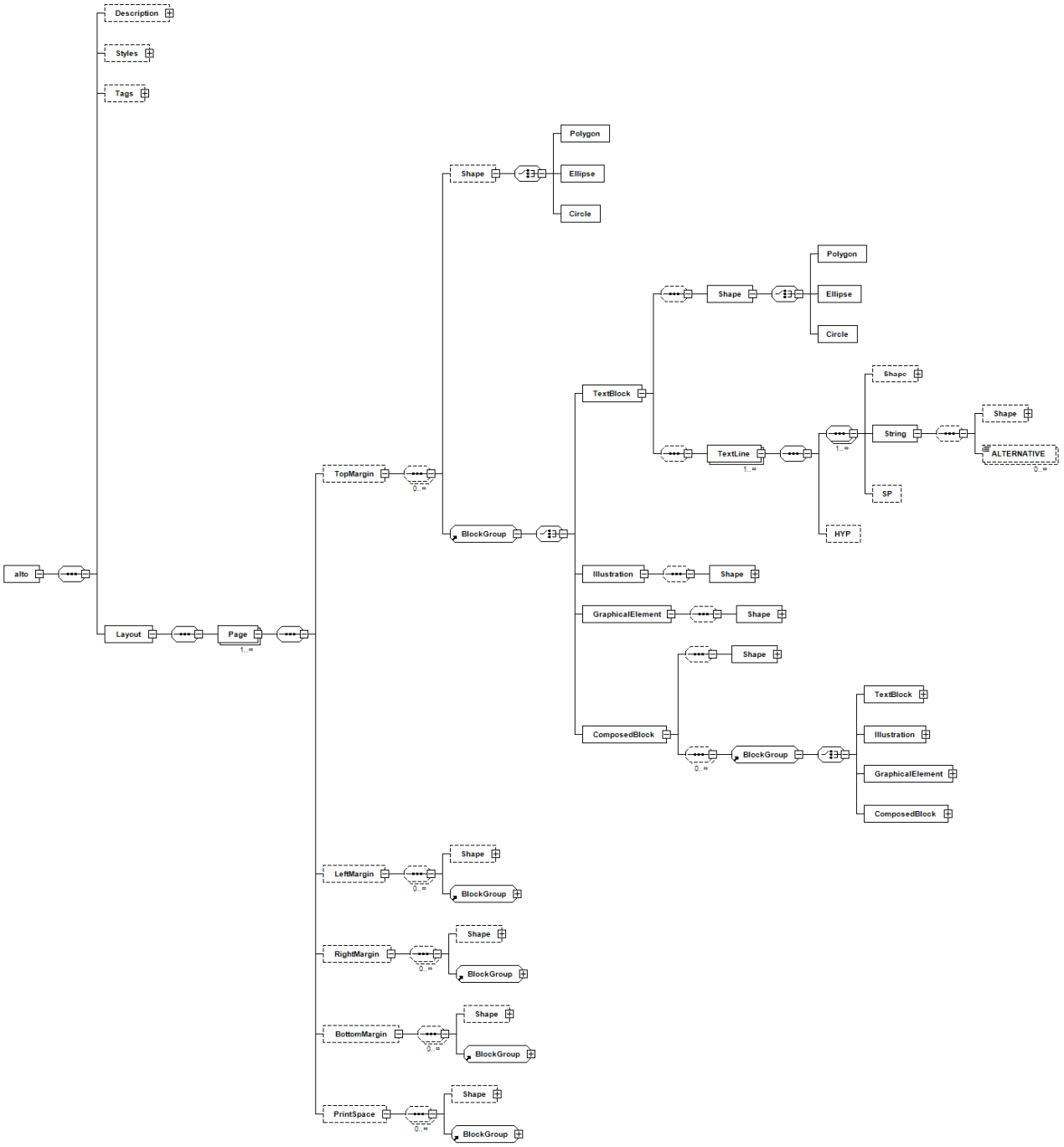
Appendix II: Visualization of the ALTO XML schema



CC BY


Description
Styles
TextStyle
0..∞
ParagraphStyle
0..∞
alto
Tags
0..∞
LayoutTag
XmlData
any ##any
1..∞
StructureTag
XmlData
any ##any
1..∞
RoleTag
XmlData
any ##any
1..∞
NamedEntityTag
XmlData
any ##any
1..∞
OtherTag
XmlData
any ##any
1..∞
Layout

CC BY


Description
Styles
Tags
alto
Layout
Page
TopMargin
Shape
Polygon
Ellipse
Circle
BlockGroup
TextBlock
Shape
Polygon
Ellipse
Circle
TextLine
Shape
String
Shape
ALTERNATIVE
SP
HYP
Illustration
Shape
GraphicalElement
Shape
Shape
ComposedBlock
BlockGroup
TextBlock
Illustration
GraphicalElement
ComposedBlock
LeftMargin
Shape
BlockGroup
RightMargin
Shape
BlockGroup
BottomMargin
Shape
BlockGroup
PrintSpace
Shape
BlockGroup

CC BY

Appendix III: Visualization of the ABBYY FineReader XML schema

CC BY

Appendix IV.1: Example document page image

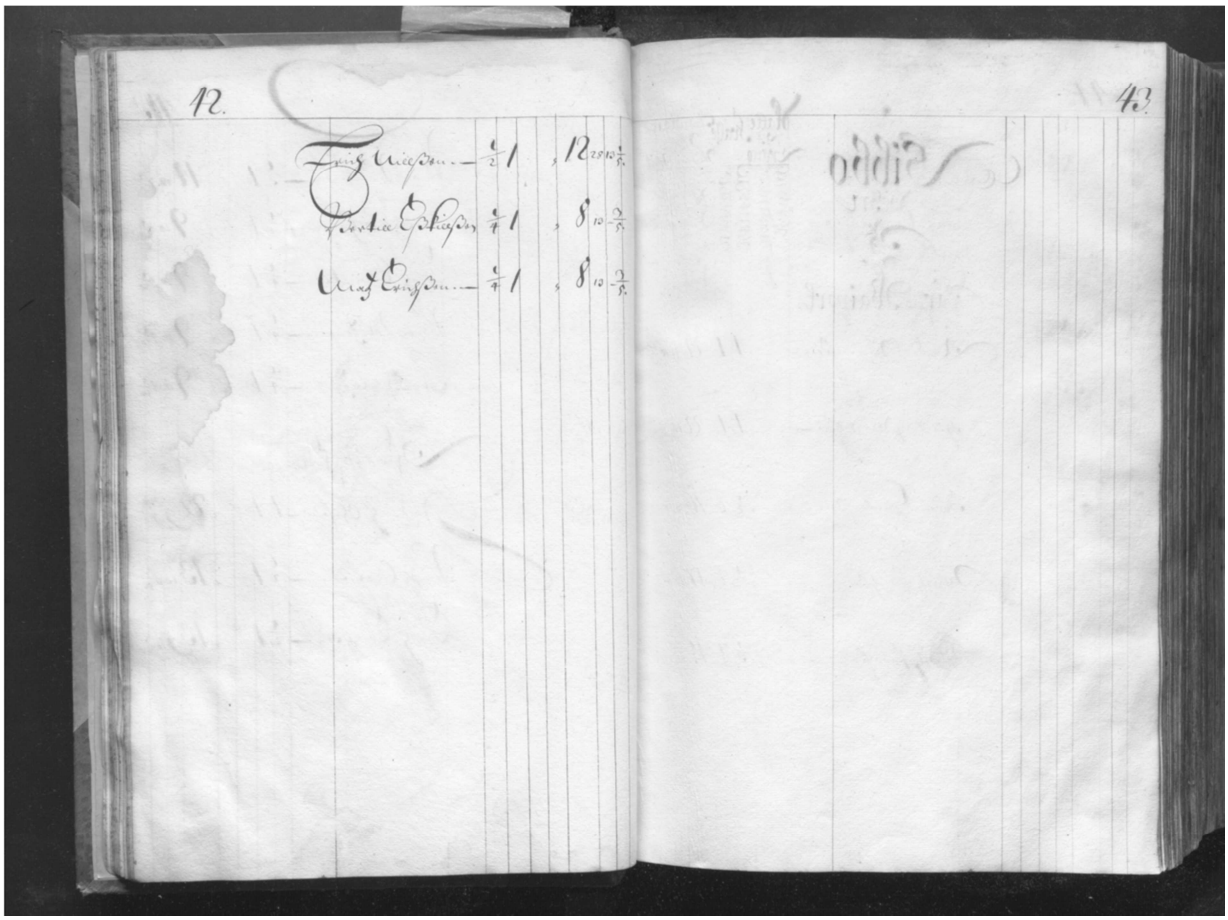


Figure 1. Source of the image: Kansallisarkisto, Läänintilit, Yleisiä asiakirjoja, Uudenmaan ja Hämeen läänin maakirja 1682-1682 (6957a)

CC BY

Appendix IV.2: Example document page PDF text

42. 43.

Erich Nillßon — 1/2 1 „ 12 28 13 1/5.

Bertill Eßkillßon 1/4 1 „ 8 13 — 3/5.

Matz Erichßon .—„ 1/4 1 „ 8 13 — 3/5.

CC BY

Appendix IV.3: Example document page PAGE XML


```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>
- <PcGts xsi:schemaLocation="http://schema.primaresearch.org/PAGE/gts/pagecontent/2013-07-15
http://schema.primaresearch.org/PAGE/gts/pagecontent/2013-07-15/pagecontent.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://schema.primaresearch.org/PAGE/gts/pagecontent/2013-07-15">
  - <Metadata>
      <Creator>TRP</Creator>
      <Created>2016-04-20T10:11:32.925+07:00</Created>
      <LastChange>2016-05-31T18:06:20.230+02:00</LastChange>
    </Metadata>
  - <Page imageHeight="4335" imageWidth="5822" imageFilename="43805229.jpg">
    - <ReadingOrder>
      - <OrderedGroup caption="Regions reading order" id="ro_1463541207774">
          <RegionRefIndexed regionRef="region_1461643036633_652" index="0"/>
          <RegionRefIndexed regionRef="region_1461643039613_653" index="1"/>
          <RegionRefIndexed regionRef="region_1461643046087_654" index="2"/>
        </OrderedGroup>
      </ReadingOrder>
    - <TextRegion id="region_1461643036633_652" custom="readingOrder {index:0;}">
        <Coords points="853,360 1083,360 1083,528 853,528"/>
      - <TextLine id="line_1461643051687_655" custom="readingOrder {index:0;}">
          <Coords points="879,448 1053,490 1045,524 871,482"/>
          <Baseline points="872,477 1046,519"/>
        - <TextEquiv>
            <Unicode>42.</Unicode>
          </TextEquiv>
        </TextLine>
      - <TextEquiv>
          <Unicode>42.</Unicode>
        </TextEquiv>
      </TextRegion>
    - <TextRegion id="region_1461643039613_653" custom="readingOrder {index:1;}">
        <Coords points="1284,541 3024,541 3024,1493 1284,1493"/>
      - <TextLine id="line_1461643056960_656" custom="readingOrder {index:0;}">
          <Coords points="1424,745 2908,703 2968,769 2941,791 2894,735 1425,780"/>
          <Baseline points="1425,775 2896,730 2945,788"/>
        - <TextEquiv>
            <Unicode>Erich Nillßon — 1/2 1 „ 12 28 13 1/5.</Unicode>
          </TextEquiv>
        </TextLine>
      - <TextLine id="line_1461643061500_657" custom="readingOrder {index:1;}">
          <Coords points="1484,1040 2883,1022 2973,1092 2951,1119 2871,1055 1484,1075"/>
          <Baseline points="1484,1070 2873,1050 2954,1115"/>
        - <TextEquiv>
            <Unicode>Bertill Eßkillßon 1/4 1 „ 8 13 — 3/5.</Unicode>
          </TextEquiv>
        </TextLine>
      - <TextLine id="line_1461643066850_658" custom="readingOrder {index:2;}">
          <Coords points="1451,1328 2383,1331 2915,1315 2981,1395 2953,1416 2900,1347 2384,1366 1451,1363"/>
          <Baseline points="1451,1358 2384,1361 2902,1342 2957,1413"/>
        - <TextEquiv>
            <Unicode>Matz Erichßon .—„ 1/4 1 „ 8 13 — 3/5.</Unicode>
          </TextEquiv>
        </TextLine>
      - <TextEquiv>
          <Unicode>Erich Nillßon — 1/2 1 „ 12 28 13 1/5. Bertill Eßkillßon 1/4 1 „ 8 13 — 3/5. Matz Erichßon .—„ 1/4 1 „ 8 13 — 3/5.</Unicode>
        </TextEquiv>
      </TextRegion>
    - <TextRegion id="region_1461643046087_654" custom="readingOrder {index:2;}">
        <Coords points="5269,329 5473,329 5473,533 5269,533"/>
      - <TextLine id="line_1463541180851_16" custom="readingOrder {index:0;}">
          <Coords points="5291,466 5461,486 5456,521 5286,501"/>
          <Baseline points="5287,496 5457,516"/>
        - <TextEquiv>
            <Unicode>43.</Unicode>
          </TextEquiv>
        </TextLine>
      - <TextEquiv>
          <Unicode>43.</Unicode>
        </TextEquiv>
      </TextRegion>
    </Page>
</PcGts>
```


CC BY

Appendix IV.4: Example document page ALTO XML


```
<?xml version="1.0" encoding="UTF-8"?>
- <alto xsi:schemaLocation="http://schema.primaresearch.org/PAGE/gts/pagecontent/2013-07-15
http://schema.primaresearch.org/PAGE/gts/pagecontent/2013-07-15/pagecontent.xsd http://www.loc.gov/standards/alto/ns-v2#
http://www.loc.gov/standards/alto/alto-v2.0.xsd" xmlns:page="http://schema.primaresearch.org/PAGE/gts/pagecontent/2013-07-15"
xmlns="http://www.loc.gov/standards/alto/ns-v2#" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  - <Description>
      <MeasurementUnit>pixel</MeasurementUnit>
    </Description>
    <Styles/>
  - <Layout>
    - <Page WIDTH="5822" HEIGHT="4335" PHYSICAL_IMG_NR="1" ID="Page1">
        <TopMargin WIDTH="5822" HEIGHT="9999999" HPOS="0" VPOS="0"/>
        <LeftMargin WIDTH="9999999" HEIGHT="-9.999999E6" HPOS="0" VPOS="9999999"/>
        <RightMargin WIDTH="5822" HEIGHT="-9.999999E6" HPOS="0" VPOS="9999999"/>
        <BottomMargin WIDTH="5822" HEIGHT="4335" HPOS="0" VPOS="0"/>
      - <PrintSpace WIDTH="-9.999999E6" HEIGHT="-9.999999E6" HPOS="9999999" VPOS="9999999">
        - <TextBlock language="" WIDTH="230" HEIGHT="168" ID="region_1461643036633_652" HPOS="853" VPOS="360">
          - <Shape>
              <Polygon POINTS="853,360 1083,360 1083,528 853,528"/>
            </Shape>
          - <TextLine WIDTH="174" HEIGHT="76" ID="line_1461643051687_655" HPOS="879" VPOS="448" BASELINE="524">
              <String WIDTH="174" HEIGHT="76" ID="string_line_1461643051687_655" HPOS="879" VPOS="448" CONTENT="42."/>
            </TextLine>
          </TextBlock>
        - <TextBlock language="" WIDTH="1740" HEIGHT="952" ID="region_1461643039613_653" HPOS="1284" VPOS="541">
          - <Shape>
              <Polygon POINTS="1284,541 3024,541 3024,1493 1284,1493"/>
            </Shape>
          - <TextLine WIDTH="1544" HEIGHT="88" ID="line_1461643056960_656" HPOS="1424" VPOS="703" BASELINE="791">
              <String WIDTH="1544" HEIGHT="88" ID="string_line_1461643056960_656" HPOS="1424" VPOS="703"
                CONTENT="Erich Nillßon — 1/2 1 „ 12 28 13 1/5."/>
            </TextLine>
          - <TextLine WIDTH="1489" HEIGHT="97" ID="line_1461643061500_657" HPOS="1484" VPOS="1022" BASELINE="1119">
              <String WIDTH="1489" HEIGHT="97" ID="string_line_1461643061500_657" HPOS="1484" VPOS="1022"
                CONTENT="Bertill Eßkillßon 1/4 1 „ 8 13 — 3/5."/>
            </TextLine>
          - <TextLine WIDTH="1530" HEIGHT="101" ID="line_1461643066850_658" HPOS="1451" VPOS="1315" BASELINE="1416">
              <String WIDTH="1530" HEIGHT="101" ID="string_line_1461643066850_658" HPOS="1451" VPOS="1315"
                CONTENT="Matz Erichßon .—„ 1/4 1 „ 8 13 — 3/5."/>
            </TextLine>
          </TextBlock>
        - <TextBlock language="" WIDTH="204" HEIGHT="204" ID="region_1461643046087_654" HPOS="5269" VPOS="329">
          - <Shape>
              <Polygon POINTS="5269,329 5473,329 5473,533 5269,533"/>
            </Shape>
          - <TextLine WIDTH="170" HEIGHT="55" ID="line_1463541180851_16" HPOS="5291" VPOS="466" BASELINE="521">
              <String WIDTH="170" HEIGHT="55" ID="string_line_1463541180851_16" HPOS="5291" VPOS="466" CONTENT="43."/>
            </TextLine>
          </TextBlock>
        </PrintSpace>
      </Page>
    </Layout>
</alto>
```


CC BY

Appendix IV.5: Example document page TEI

```
<?xml version="1.0" encoding="UTF-8"?>
<TEI xmlns="http://www.tei-c.org/ns/1.0">
  <teiHeader>
    <fileDesc>
      <titleStmt>
        <title type="main">Copy of NAF_504_Original_Scans_Training_Set</title>
      </titleStmt>
      <publicationStmt>
        <publisher>tranScriptorium</publisher>
      </publicationStmt>
      <sourceDesc>
        <bibl>
          <publisher>TRP document creator: guenter</publisher>
        </bibl>
      </sourceDesc>
    </fileDesc>
  </teiHeader>
  <facsimile xml:id="facs_376">
    <graphic url="43805229.jpg"/>
    <surface lry="5822" lrx="4335" uly="0" ulx="0">
      <zone type="TextRegion" xml:id="facs_376_region_1461643046087_654" points="5269,329 5473,329 5473,533 5269,533">
        <zone type="Line" xml:id="facs_376_line_1463541180851_16" points="5291,466 5461,486 5456,521 5286,501"/>
      </zone>
      <zone type="TextRegion" xml:id="facs_376_region_1461643036633_652" points="853,360 1083,360 1083,528 853,528">
        <zone type="Line" xml:id="facs_376_line_1461643051687_655" points="879,448 1053,490 1045,524 871,482"/>
      </zone>
      <zone type="TextRegion" xml:id="facs_376_region_1461643039613_653" points="1284,541 3024,541 3024,1493 1284,1493">
        <zone type="Line" xml:id="facs_376_line_1461643056960_656" points="1424,745 2908,703 2968,769 2941,791 2894,735
          1425,780"/>
        <zone type="Line" xml:id="facs_376_line_1461643061500_657" points="1484,1040 2883,1022 2973,1092 2951,1119
          2871,1055 1484,1075"/>
        <zone type="Line" xml:id="facs_376_line_1461643066850_658" points="1451,1328 2383,1331 2915,1315 2981,1395
          2953,1416 2900,1347 2384,1366 1451,1363"/>
      </zone>
    </surface>
  </facsimile>
  <text>
    <body>
      <pb n="376" facs="#facs_376"/>
      <p facs="#facs_376_region_1461643046087_654">
        <lg>
          <l facs="#facs_376_line_1463541180851_16">43.</l>
        </lg>
      </p>
      <p facs="#facs_376_region_1461643036633_652">
        <lg>
          <l facs="#facs_376_line_1461643051687_655">42.</l>
        </lg>
      </p>
      <p facs="#facs_376_region_1461643039613_653">
        <lg>
          <l facs="#facs_376_line_1461643056960_656">Erich Nillßon — 1/2 1 „ 12 28 13 1/5.</l>
          <l facs="#facs_376_line_1461643061500_657">Bertill Eßkillßon 1/4 1 „ 8 13 — 3/5.</l>
          <l facs="#facs_376_line_1461643066850_658">Matz Erichßon .—„ 1/4 1 „ 8 13 — 3/5.</l>
        </lg>
      </p>
    </body>
  </text>
</TEI>
```

CC BY