DPF Manager

The open source COMMUNITY

DPF MANAGER

FUNCTIONAL SPECIFICATIONS DOCUMENT

Project acronym: PREFORMA
PREFORMA - Future Memory Standards
PREservation FORMAts for culture information/e-archives
EC Grant agreement no: 619568
EC Call ID: FP7-ICT-2013-11

SEVENTH FRAMEWORK PROGRAMME

PREFORMA

INDEX

1. Document Overview

1.1. Purpose

This document describes the user requirements and functional specifications of the DPF Manager framework and application in PREFORMA project. The specifications are derived from the analysis of the Memory Institution's needs, the analysis of current file validation software, feedback from specialists in image preservation, OAIS model requirements, as well as the PREFORMA specifications.

To describe the procedures involved, the use cases are used. It is the intention here to define functional specifications that describe all possible requirements in the project. For each Memory Institution it can then be determined if they will benefit from implementing all the features or just a subset relevant to them. The same principle will clearly apply in order to explain how the DPF Manager can be integrated inside the OAIS model. In addition of use cases, all functionalities for each component in the Conformance Checker are individually explained.

The operational requirements are also taken into account to ensure the application fulfils all the functionalities with security, privacy, reliability, recoverability and performance.

The main purpose of this documentation is to identify requirements and specifications as well as operational functionalities. The document will provide the specifications for the design deliverable M16.

1.2. Needs analysis

The PREFORMA project has already done a fantastic work capturing and analysing the requirements from the memory institutions and other stakeholders.

To complement this analysis, and to get an even deeper understanding of their current work practices and needs, we have interviewed and sent questionnaires to Memory Institutions and other stakeholders including organizations and companies concerned about preservation issues, such as content producers for Memory Institutions and digitalization companies, were also taken into account.

After analysing the results we have identified different needs depending on the institution's collection size.

Knowing their workflow has helped us to identify which software they are currently using and evaluate their strengths and weaknesses. Memory Institutions that do not use any file validation tools have also provided valuable information.

Our knowledge and the opinion of other specialists in digital image preservation and the TIFF format has helped us to identify the issues and challenges to take into account for long term preservation of digital images.

Finally, we have researched how the DPF Manager would help Memory Institutions using OAIS model.

All this factors are the base of our decisions about the functional specifications and technical design.

1.3. Document structure

Section 1 provides the purpose, analysis of stakeholder needs, and structure of this document as well as references used in the preparation of this document and a list of abbreviations and terminology.

Section 2 provides an introduction to the DPF manager project.

Section 3 introduces the concepts involved in TIFF format and digital image preservation.

Section 4 describes the needs of memory institutions depending on their collection size.

Section 5 describes all the functional and non-functional requirements

Section 6 describes the requirements for the integration in the OAIS model.

Section 7 describes the use scenarios of the application.

Section 8 introduces the DPF manager interfaces.

Section 9 describes the command line interface.

Section 10 describes the web-services interface.

Section 11 describes the DPF Manager graphical user interface.

Section 12 describes the different types of reports the DPF Manager is able to produce.

1.4. Project References

This Section provides a list of the references that were used in preparation of this document.

Reference model for an open archival information system (OAIS) recommended practice CCSDS 650.0-M-2

Trustworthy Repositories Audit & Certification: Criteria and Checklist Version 1.0 - February, 2007

TIFF Revision 6.0 Final — June 3, 1992

Adobe Photoshop® TIFF Technical Notes March 22, 2002

Adobe PageMaker® 6.0 TIFF Technical Notes September 14, 1995

ISO 12234-2:2001 Electronic still-picture imaging -- Removable memory -- Part 2: TIFF/EP image data format

ISO 12639:2004 Graphic technology -- Prepress digital data exchange -- Tag image file format for image technology (TIFF/IT)

XMP specification part 1 - Data model, serializations and core properties. April, 2012

XMP specification part 2 - Additional properties. April, 2012

XMP specification part 3 - Storage in files. May 2013

Exchangeable image file format for digital still cameras: Exif Version 2.3. December, 2012

Dublin Core Metadata Initiative, August 2007

IPTC Standards - Photo Metadata White Paper, 2007 Document Revision 11

1.5. Acronyms and Abbreviations

AIP	Archival Information Package
API	Application Programming Interface
DAM	Digital asset management
DIP	Dissemination Information Package
GLAM	Galleries, Libraries, Archives and Museums
ICC	International Color Consortium
ISO	International Organization for Standardization
OAIS	Open Archival Information System
PDF	Portable Document Format
PDI	Preservation Description Information
QA	Quality Assurance
SIP	Submission Information Package
URL	Uniform Resource Locator
XML	eXtensible Markup Language
XMP	eXtensible Metadata Platform

1.6. Terminology

There are many terms which are used in this reference model and which need to have well defined meanings. These terms are defined in this subsection. When first used in the text, they are shown in bold and are capitalized. Subsequent use employs capitalization only.

Archival Information Package (AIP): An Information Package, consisting of the Content Information and the associated Preservation Description Information (PDI), which is preserved within an OAIS.

Archive: An organization that intends to preserve information for later access and use by a Designated Community.

Associated Description: The information describing the content of an Information Package.

Consumer: The role played by those people or client systems, who interact with OAIS services to find preserved information of interest and to access that information in detail. This can include other OAISes, as well as internal OAIS persons or systems.

Content Data Object: The Data Object, that together with associated Representation Information, comprises the Content Information.

Content Information: A set of information that is the original target of preservation or that includes part or all of that information. It is an Information Object composed of its Content Data Object and its Representation Information.

Context Information: The information that documents the relationships of the Content Information to its environment. This includes why the Content Information was created and how it relates to other Content Information objects.

Data: A re-interpretable representation of information in a formalized manner suitable for communication, interpretation, or processing. Examples of data include a sequence of bits, a table of numbers, the characters on a page, the recording of sounds made by a person speaking, or a moon rock specimen.

Data Submission Session: A delivery of media or a single telecommunications session that provides Data to an OAIS. The Data Submission Session format/contents is based on a data model negotiated between the OAIS and the Producer in the Submission Agreement. This data model identifies the logical constructs used by the Producer and how they are represented on each media delivery or in the telecommunication session.

Descriptive Information: The set of information, consisting primarily of Package Descriptions, which is provided to Data Management to support the finding, ordering, and retrieving of OAIS information holdings by Consumers.

Designated Community: An identified group of potential Consumers who should be able to understand a particular set of information. The Designated Community may be composed of multiple user communities. A Designated Community is defined by the Archive and this definition may change over time.

Digital Migration: The transfer of digital information, while intending to preserve it, within the OAIS. It is distinguished from transfers in general by three attributes:

- a focus on the preservation of the full information content that needs preservation;
- a perspective that the new archival implementation of the information is a replacement for the old; and
- an understanding that full control and responsibility over all aspects of the transfer resides with the OAIS.

Digital Object: An object composed of a set of bit sequences.

Dissemination Information Package (DIP): An Information Package, derived from one or more AIPs, and sent by Archives to the Consumer in response to a request to the OAIS.

Independently Understandable: A characteristic of information that is sufficiently complete to allow it to be interpreted, understood and used by the Designated Community without having to resort to special resources not widely available, including named individuals.

Information: Any type of knowledge that can be exchanged. In an exchange, it is represented by data. An example is a string of bits (the data) accompanied by a description of how to interpret the string of bits as numbers representing temperature observations measured in degrees Celsius (the Representation Information).

Information Package: A logical container composed of optional Content Information and optional associated Preservation Description Information. Associated with this Information Package is Packaging Information used to delimit and identify the Content Information and Package Description information used to facilitate searches for the Content Information.

Information Property: That part of the Content Information as described by the Information Property Description. The detailed expression, or value, of that part of the information content is conveyed by the appropriate parts of the Content Data Object and its Representation Information.

Knowledge Base: A set of information, incorporated by a person or system, that allows that person or system to understand received information.

Long Term: A period of time long enough for there to be concern about the impacts of changing technologies, including support for new media and data formats, and of a changing Designated Community, on the information being held in an OAIS. This period extends into the indefinite future.

Long Term Preservation: The act of maintaining information, Independently Understandable by a Designated Community, and with evidence supporting its Authenticity, over the Long Term.
Management: The role played by those who set overall OAIS policy as one component in a broader policy domain, for example as part of a larger organization.

Memory Institution: An organization maintaining a repository of public knowledge, a generic term used about institutions such as libraries, archives, museums, sites and monument records, clearinghouses, providers of Digital Libraries and data aggregation services which serve as memories for given societies or mankind.

Metadata: Data about other data.

Open Archival Information System (OAIS): An Archive, consisting of an organization, which may be part of a larger organization, of people and systems, that has accepted the responsibility to preserve information and make it available for a Designated Community. It meets a set of responsibilities that allows an OAIS Archive to be distinguished from other uses of the term 'Archive'. The term 'Open' in OAIS is used to imply that this Recommendation and future related Recommendations and standards are developed in open forums, and it does not imply that access to the Archive is unrestricted.

Package Description: The information intended for use by Access Aids.

Packaging Information: The information that is used to bind and identify the components of an Information Package. For example, it may be the ISO 9660 volume and directory information used on a CD-ROM to provide the content of several files containing Content Information and Preservation Description Information.

Preservation Description Information (PDI): The information which is necessary for adequate preservation of the Content Information and which can be categorized as Provenance, Reference, Fixity, Context, and Access Rights Information.

Producer: The role played by those persons or client systems that provide the information to be preserved. This can include other OAISes or internal OAIS persons or systems.

Representation Information: The information that maps a Data Object into more meaningful concepts. An example of Representation Information for a bit sequence which is a FITS file might consist of the FITS standard which defines the format plus a dictionary which defines the meaning in the file of keywords which are not part of the standard.
Another example is JPEG software which is used to render a JPEG file; rendering the JPEG file as bits is not very meaningful to humans but the software, which embodies an understanding of the JPEG standard, maps the bits into pixels which can then be rendered as an image for human viewing.

Semantic Information: The Representation Information that further describes the meaning beyond that provided by the Structure Information.

Structure Information: The Representation Information that imparts meaning about how other information is organized. For example, it maps bit streams to common computer types such as characters, numbers, and pixels and aggregations of those types such as character strings and arrays.

Submission Agreement: The agreement reached between an OAIS and the Producer that specifies a data model, and any other arrangements needed, for the Data Submission Session. This data model identifies format/contents and the logical constructs used by the Producer and how they are represented on each media delivery or in a telecommunication session.

Submission Information Package (SIP): An Information Package that is delivered by the Producer to the OAIS for use in the construction or update of one or more AIPs and/or the associated Descriptive Information.

Transformation: A Digital Migration in which there is an alteration to the Content Information or PDI of an Archival Information Package. For example, changing ASCII codes to UNICODE in a text document being preserved is a Transformation.

2. Introduction

DPF manager is a framework and an application for gaining full control over the technical properties and structure of digital content Data Object intended for Long Term Preservation by Memory Institutions as well as other organizations and companies concerned about preservation issues. These organizations are finding, or will find, that they need tools to ensure that all the digital content they store continues to be effective at representing the whole of the information over time.

Long term preservation of digital image files relies on the use of a stable, open and well documented file format. TIFF is the most widely used and most accepted format for this task. However, TIFF itself is quite complicated and the different standards TIFF/EP (ISO 12234-2:2001) and TIFF/IT (ISO 12639:2004) include many options that are not suitable for long term archival.

In order to consolidate the DPF manager as not only a tool to proof-read a given TIFF file and validate it through and standards but also a tool to ensure the long preservation of a file, therefore, a set of mandatory, optional and unsuitable options will be carefully defined (out of the options defined in the different TIFF standards) in order to guarantee that in future, any image file can be rendered without loss of quality and information, even if hardware and software have changed significantly in the future.

Nowadays there are a lot of software companies offering Digital Asset Management (DAM) software for Memory Institutions in order to control the ingest, management and access to digital data. Such software, although it usually includes a metadata manager, rarely inspects the file content to ensure ISO compliance and long term preservability. DPF manager will be easy to integrate in DAM software to provide this functionality.

Some Archives are using, or trying to use, more specific tools, like JHOVE or FITS, in order to validate a TIFF structure or ISO compliance and report on the issues. Such reports are so large and complex that only experts in digital imaging or a machine is able to analyse its content and find problematic areas. Therefore only Memory Institutions staffed with operators with enough knowledge to understand the reports and take appropriate actions will use these tools. In addition it has to be pointed out that neither of these tools evaluates the digital content suitability for long term preservation.

DPF manager generates a user readable report that can be useful to different audiences, expert and non-expert users, giving a preservability score, format and preservation issues, giving hints and solutions for each problem as well as providing complete information about data content structure. DPF manager report can be generated in an interactive platform like an HTML5 and JavaScript. The combination of the two reports gives enough flexibility to fill in all the scenarios.

Some of the Long Term Preservation activities may conflict with the goal of rapid production and dissemination of digital information. DPF manager aims to reduce the time and effort required to revise file structure, ISO compliant and metadata information as well as giving advice about Data Content preservability.

3. TIFF format and preservation considerations

In the "analog" past, the longevity of documents, being it texts, images, moving images etc. has been defined basically by the properties of the medium or the stability of the support the information has been stored on. While text written on parchment has a longevity way beyond 1000 years, it is well known that for example photographic material starts to deteriorate after approx. 150 years (B/W) or 50 years (for chromogenic colour material). An alleged solution is to copy the original, be it a image or text, periodically to a new carrier to increase its lifetime. Unfortunately every analog copy process introduces new loss because the transfer to a new medium cannot be done without physical limitations, like reduction in sharpness or colour fidelity. At the end, after the n^{th} generation, the original information renders useless due to multiple steps of the introduction errors. In recent times the digital age promises a theoretically unlimited longevity of digital information: A properly done digital copy process can be achieved with zero loss. Thus even the copy in the b^{th} generation will be identical and not distinguishable from the original; they are identical down to the bit-level. Thus, the digital age promises for the first time the unlimited preservation of information. Besides other motivations, this property is an important consideration in the decision to digitize cultural assets. Therefore most digitization projects are done to rescue the assets from further degradation, thus most digital objects are taken from analogue originals (e. g. digital captures of analog photographs) to transform them into the digital domain.

3.1. Sustainability Factors

However, as we well know, despite the fact that digital data could be essentially preserved forever; there are some major hurdles to overcome.

On one hand, all digital data carriers have a very limited lifespan due to technical obsolescence and/or unstable materials – they are on a physical level analog. This problem can be overcome by a early enough periodical migration onto new data carriers. This process is known as bitstream preservation.

On the other hand, also the file formats which are used to represent the digital data (e. g. digital images) may become obsolete over time. Any file format basically defines the logical structure and meaning of the bits within the bitstream and thus it is essential for correct interpretation and proper rendering of the coded data. Both the data carrier migration and the format migration are addressed in the OAIS reference model. Unfortunately a file format or parts of its logical structure and definition can become obsolete. As a result the information in the file is rendered useless, even-though the bit stream is still properly readable.

A format migration is more complex than creating a plain copy of a bit stream, by copying it to a new data carrier. The existing file format – the logic structure and its content – must be read, translated and written to a new data container. In such a process e.g. important metadata can be easily lost, due to improper transformation into new code which leads to files that can't be rendered correctly. Therefore it is necessary to use a stable and proper file format for long-term preservation of digital data. The stability of a format is determined by the following criteria:

- the format has to be well documented
- the format should be in wide used
- The format should not contain proprietary or patented elements (algorithms etc.). In other words, it should be an open format.
- the format should be as simple as possible
- no interlinkage with external data is allowed (e.g. fonts or any other resource)
- In addition to that it must be given, that the file-format is capable to store the relevant information without significant compromises. In the case of image files this could for example be the quantisation depth for correct tonal reproduction, that shall be bigger than 8bit/channel for most applications.

3.2. Digital Image Preservation

Digital images do not exist – only digital data representing an analogue image. In order to get an image, this data has to be rendered – that is printed on a support or displayed on a screen etc.

Usually, in the near future after a digital image has been created, there are means to render the image on an appropriate device. However, if we consider time frames Galleries, Libraries, Archives and Museums (GLAM) institutions have to keep information for much longer periods of time. We consider long term preservation of digital images as the task to be able to render the image data in a correct way well beyond the timeframe of 10 years, possibly even an infinite period of time.

A prerequisite for any digital long term preservation is the so called bitstream preservation: the bits have to be kept over time in a way that they still can be deciphered in the future. Usually bitstream preservation follows the OAIS reference model of digital archives which is based on a migration model. The bits have to be copied to a new data carrier before the old data carrier becomes unreadable due to aging or technological obsolescence. This copy process has zero tolerance to errors. Therefore complicated copy procedures with redundant copies and checksums have to be used.

However the major problems are the file formats. The same series of bits may have many different meanings depending on its position within a byte stream. The byte 01000010 can represent the character “B”, the integer 66 using the base 10, can be part of a floating point number etc. Even if it is known that it represents an integer, what is the meaning of this number 66 at the given position with the bitstream?

A file format defines for every position of a series of bits its meaning.

It has to be noted that meaning at a specific position may be dependent of the values at another position within the bitstream. Thus the bitstream must be read in a certain, not necessarily sequential, order for getting a meaningful interpretation. Thus the bitstream also contains some information which is used to interpret other parts of the bitstream. In order to guarantee the long term preservation of digital data, the knowledge about the meaning of the bits has to be preserved in the same way as the actual image data itself. Thus, it is obvious, that for any digital long term preservation, the file format must conform to the following requirements:

1. The information about the file format – that is the meaning of the bits – has to be openly available and published
2. There must be no patents or other proprietary elements used within the file format
3. There must be no loss of information when converting image data into the long term preservation format.[1]
4. The format should be well adopted and have a widespread use
5. The format should be as simple as possible for the given task

Digital images can be divided into three major parts (where the last is optional):

1. The actual image data
2. Technical metadata describing the structure of the image data. These are essential in order to be able to interpret and render the image data
3. Other Metadata describing supplementary information such as the creator, the data, the camera used, contextual information etc. These, while helpful for many purposes, are in most cases not necessary for rendering a correct image. However for the best possible quality rendering, technical specifications of the camera are necessary to correct e.g. lens distortions.

Most often the metadata area is called the image header because traditionally the metadata have been stored at the beginning of the file. However in many file formats such as the TIFF, the position of the metadata within the file is not fixed.

3.3. Tiff format

One of the most widespread formats used to represent image data is the TIFF format. TIFF (Tagged Image File Format) was originally created by Aldus Corporation in 1986. TIFF is a flexible, adaptable file format for handling images and data within a single file, by including various header tags for size, channel definition, image-data arrangement, applied_image compression and others, define the technical configuration of the image. The flexibility allows for many different variants and can include further on metadata which follows other format definitions such as IPTC-data, EXIF-data or ICC-data. It also allows adding proprietary elements and supports many different compression schemes, even JPEG. There are some options within the TIFF standard, which are rarely used and not supported by most applications. Further on some application remove metadata that is not used by the application without notice of the user.

[1] There are many image formats which do reduce the size of the file by removing redundancy within the image data. Lossy file formats such as the JPEG to modify the image data in an irreversible way which will result in the permanent loss of information and even may introduce visible artifacts.

The TIFF format is completely open in the sense that the complete specification has been published. The standard was finalized in 1992 and the full documentation can be found at https://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf .

3.3.1. TIFF-Baseline and TIFF-Extensions

The TIFF reference manual distinguishes between the Baseline TIFF and TIFF-Extensions. The "Baseline TIFF is the core of TIFF, the essentials that all mainstream TIFF developers should support in their products", as the TIFF documentation states.

3.3.2. TIFF/EP (ISO 12234-2:2001) for Electronic Photography

TIFF/EP was designed to fulfil the requirements of the camera manufacturers. The TIFF/EP has extensions and modifications to the TIFF-Baseline and TIFF-Extensions standard that makes it incompatible for a standard TIFF viewer and includes many technical metadata (camera parameters) which is necessary for the conversion process. In fact the conversion algorithms are proprietary. Therefore TIFF/EP is totally unsuitable for long term preservation.

3.3.3. TIFF/IT (ISO 12639, 2004) for Image Technology

The library of congress describes the TIFF/IT as follows[2]:
"ISO 12639:2004 specifies a media-independent means for prepress electronic data exchange using a tag image file format (TIFF). [...] The FP subtype provides a mechanism for creating a package that includes separate image layers (of types CT, LW, etc.) to be combined to create the final printed image."

Thus, the purpose of this format the transmission of prepared images (e.g. advertisements) for integration into print publications (e.g. magazines).

3.4. Proposal for a new standard

As described above, the TIFF format is quite complex and parts of the original definition have become obsolete, while new not formally standardized additions have been made. It would be easily possible to create a TIFF file that is fully conformant to the TIFF Revision 6.0 specifications but would be virtually useless because no existing renderer would be able to open and render it[3]. Since a digital archive has the goal that the file can be rendered in a indefinite but possibly

[2] See Sustainability of Digital Formats: Planning for Library of Congress Collections, http://www.digitalpreservation.gov/formats/fdd/fdd000072.shtml

[3] It should be noted that it would be possible but quite time consuming and difficult to develop a new renderer that would open any conforming TIFF file.

far future, a simplistic approach is necessary, or to paraphrase Albert Einstein: “Everything should be made as simple as possible, but not simpler” That is, a TIFF file suitable for long term archiving should use only the minimal set of tags that is necessary to allow a correct future rendering of the data and to represent the essential descriptive metadata.

In addition, a TIFF file suitable for long term preservation should only contain one IFD and thus only one image.

We therefore propose a subset of TIFF which is fully compatible with the TIFF standard but marks some tags as mandatory, some as optional and some as forbidden in order to guarantee the correct rendering in the future. In analogy to PDF/A format we propose to call this Format TIFF/Archive or TIFF/A.

3.4.1. Do we need a new standard?

Do we really need yet another image file format standard? Having so many “standard” image file formats such as TIFF, JPEG, JPEG2000, GIF, PNG just to name a few of the well-known variants, this is an important question! However, the answer may be somewhat surprising:

TIFF/A is not a new image file standard, since it builds completely on an existing de facto standard which is the TIFF format with some of the widely used more recent (than 1992) “private” extensions such as the XMP tag. Thus any existing conformant modern TIFF reader will be able to open a TIFF/A file and render it correctly without problems. However, since the TIFF format is very complex and offers many options, a subset of the de facto TIFF specification has to be defined for archiving. The features that a TIFF file would allow are categorized for the TIFF a with mandatory, optional and forbidden. Thus the TIFF enforces some stricter rules on how a TIFF file for archiving must be constructed. In this sense, TIFF/A is not a new file format but a version of the TIFF format that is suitable for long term archival.

4. Analysis of memory institutions needs

The PREFORMA project did conduct a series of workshops to determine the needs of the memory institutions, producing a set of requirements for the suppliers to deliver on.

To take that work further, and to get a better understanding of how different memory institutions tackle the digital preservation challenge, we designed and distributed a questionnaire with specific questions about type and volume of digital assets, current capabilities and limitations, and future needs.

The full result of the analysis of the answers is included as an annex at the end of this document, but we have included a few key numbers here:

- **88% of memory institutions use the TIFF format for digital preservation of still-images:** This data provides additional validation of the correctness of the PREFORMA requirement to focus on the TIFF format.

- **70% of memory institutions do not use any type of software to check that their images are well formatted and conform to a specific standard:** The DPF Manager will help those memory institutions by performing automated and periodical checks on new and existing files.

- **68% of memory institutions do not have any metadata validation process in place:** The DPF Manager can automatically validate the metadata found in digital images, so it will help memory institutions automate the metadata validation process.

- **Memory institutions outsource the digitalisation process of 28% of their collection** The DPF Manager will provide a set of APIs and interfaces so external companies can check the results of the digitalisation process before sending the digital files to the memory institutions.

- **66% of memory institutions are aware of the OAIS reference model, but only 42% use it:** By making it easier to check files at ingestion time and during other parts of their lifecycle, the DPF Manager will help memory institutions follow the recommended processes in the OAIS reference model.

Thanks to our experience working in several other digital preservation projects, and based on the responses to the questionnaires, we have decided to segment our target users in three different groups based on the size of their operations (archive size and expected growth rate).

In this section we describe each target group, the particular challenges they face, and what DPF manager has to provide to satisfy their needs.

4.1. Small memory institution

Characteristics:

- They have an average of 14,000 images in their collection
- Their collection grows at an average rate of 2,500 images per year.
- They do most of the digitalisation process in-house
- A majority of them are aware of the OAIS reference model, but they don't apply it.
- They work with TIFF and JPEG images
- They don't have any file structure and standards conformance checking process in place
- They don't have any metadata validation process in place
- They don't have any structured process for archiving purposes, or a very basic one.
- They are not aware of patent and licensing issues.

DPF Manager will offer them these functionalities:

- Facilitate the implementation of the OAIS reference model processes
- Easy to use Graphical User Interface
- Automatic and periodical validation of file structure, standard adherence and metadata
- Simple and comprehensive reports, ready to be read and understood by any user, regardless of their technical knowledge level.
- Provide attached an advanced report, so they will be able to consult with an expert.
- An application with a simple and guided user interface, avoiding any possibility of human error on its operation.
- An application with no external dependencies and easy step-by-step installation.
- Web access to a cloud open service for processing a reduced number of files.

4.2. Medium memory institution

Characteristics:

- They have an average of 600,000 images in their collection
- Their collection grows at an average rate of 25,000 images per year.
- They outsource most of the digitalisation process
- A majority of them are aware of the OAIS reference model and apply it.
- They work mostly with TIFF images
- They don't have any file structure and standards conformance checking process in place
- They don't have any metadata validation process in place
- They have structured procedures for archiving purposes.
- They are aware of patent and licensing issues.

DPF Manager will offer them these functionalities:

- Facilitate the implementation of the OAIS reference model processes
- Automatic and periodical validation of file structure, standard adherence and metadata
- Reduced and comprehensive reports and advanced reports, all in one.
- Possibility to configure the behaviour of the application.

- Possibility to access the application using several interfaces:
    - Command-line.
    - Graphic User Interface application.
    - Web application.
- Batch processing to be able of add a large number of files.
- OAIS compliant procedure, easily integrable to the current workflow.

4.3. Large memory institution

Characteristics:

- They have an average of 23.5 million images in their collection
- Their collection grows at an average rate of 1.5 million images per year.
- They do most of the digitalisation process in-house
- A majority of them are aware of the OAIS reference model and apply it.
- They work mostly with TIFF images
- Half of them don't have any file structure and standards conformance checking process in place
- Most of them have a metadata validation process in place
- Medium/large number of staff, with technical and archival knowledge.
- They have structured procedures for archiving purposes.
- They are aware of patent and licensing issues.

DPF Manager will offer them these functionalities:

- Reduced and comprehensive reports and advanced reports, all in one.
- Advanced preservation advice, complementary to the reports
- Possibility to configure the behaviour of the application, component a component.
- Possibility of access to the application over several interfaces:
    - Command-line.
    - Graphic User Interface application.
    - Web application.
- Batch process to be able of add a large number of files.
- OAIS compliant procedure, easily integrable to the current workflow.

5. Requirements

We have used the information collected from the questionnaires to complement the information provided by PREFORMA, to create the list of requirements that the DPF manager should meet.

The requirements are divided between functional (system specific behaviours and functions) and non-functional (how the system should operate) requirements

5.1. Functional Requirements

5.1.1. DPF Manager

The DPF Manager MUST:

- Verify whether a file has been produced according to the specifications of a standard file format,
- Verify whether a file matches the acceptance criteria for long-term preservation by the memory institution,
- Report in human and machine readable format which properties deviate from the standard specification and acceptance criteria, and
- Perform automated fixes for simple deviations in the metadata of the preservation file
- Perform periodic checks on the files in an archive

5.1.2. Shell

The Shell component of the DPF Manager MUST:

- Facilitate the validation of files at four moment in the life cycle of a digital document, identified in the use cases of the challenge brief, i.e. validating files at creation time, transfer time, digitisation time and migration time,
- Allow for automating the procedures for checking, reporting and fixing preservation file,
- Allow for configuring fully automated, periodical checks,
- Allow for batch processing of extensive file sets,
- Allow for configuration of additional components in particular implementation checkers, policy checkers and reporters for other preservation file formats that are developed in the PREFORMA ecosystem,
- Allow for use by non-expert users, and
- Be operational in a closed zone with no Internet access.
- Expose a set of API to be integrated into other programs/systems.
- Provide live output information about current process to give a user operation feedback when process a lot of files.
- Provide a log output with information about process executed in order to audit the process and detect errors.

The Shell component SHOULD:

- Generate a summary report when multiple files are checked.

5.1.3. Implementation Checker

The Implementation checker component of the DPF Manager MUST allow for checking compliance with the TIFF standard specifications as defined in:

- ISO (2001). Electronic still-picture imaging — Removable memory — Part 2: TIFF/EP image data format. ISO/TC 42, ISO 12234-2:2001
- ISO (2004). Graphic Technology -- Prepress digital data exchange -- Tag image file format for image technology (TIFF/IT). ISO/TC 130. ISO 12369:2004

The first verifications that the Conformance Checker will have to do are related to the file structure, and other basic aspects of the file.

First of all, it has to verify that the image does not have any type of compression, this is a main PREFORMA requirement, and a fail in this part has to report a high risk warning.

Then, it has to check the internal structure, according to the Baseline 6.0, and we further suggest to check additional aspects like inappropriate metadata or empty spaces inside the file structure, i.e.,

The next step is validating the fulfilment of the ISO standards: TIFF/EP and TIFF-IT of the files. In addition we will check the file under the specifications of the new standard proposed: TIFF-A, more restrictive but more preservable than the other previous two.

Finally, the implementation checker should to rate the quality of a file in terms of preservation, to provide a comprehensive conclusion of the analysis.

This checker phase has to be configurable by a configuration file structured in XML, to guarantee the possibility of reproducing specific configuration in time (day by day), and space (in different memory institution that are interchanging files).

5.1.4. Policy Checker

The Policy checker component of the DPF Manager MUST allow for checking technical parameters of files, based on the acceptance criteria defined by the memory institution, including technical metadata for still images

Users must be able to specify their own acceptance criteria, ensuring normalization between all the archive files.

5.1.5. Reporter Checker

The Reporter component of the DPF Manager MUST provide:

- A machine readable report, including preservation metadata for each file checked and allowing external software agents to further process the file.
- The machine readable report will be produced using a standard XML format, implemented by all Conformance Checkers in the PREFORMA ecosystem, which allows the reported module to combine output from multiple checker components in one report. At the end of the design phase (February 2015), a proposal for such standard output format will be made by the consortium.
- A human readable report, assessing the preservation status of a batch of files as a whole, reporting to a non-expert audience whether a file is compliant with the standard specifications, and addressing improvements in the creation/digitisation process.

It will generate a report with 3 different levels of information:

- General Score: just to show a rate of quality in preservation
- Non-expert report: a brief comprehensive abstract with the result of the analysis, the conflictive points and advice about the file.
  Expert report: advanced descriptions of all the problems or warnings detected with advice about how to solve them.

5.1.6. Metadata Fixer

The Metadata fixer component of the DPF Manager MUST allow for performing fully automated fixes of incongruities in the metadata embedded in the file, based on the report of the implementation checker. Such automated fixes may include:

- Making embedded technical metadata conform with the properties contained by the preservation file
- Normalising embedded administrative metadata about the preservation file.

This checker phase has to be configurable by a configuration file structured in XML, to guarantee the possibility of reproduce specific configuration in time (day by day), and space (in different memory institution that are interchanging files).

5.2. Non-functional requirements

5.2.1. Availability

The DPF Manager shall be available all the time. This consideration is important in a client - server configuration where a lot of clients are trying to check multiple files. In this case the system has to respond to all the requests.

5.2.2. Audit trail

The DPF Manager shall log all the internal messages and outputs, including errors. This will make it possible to audit the system behaviour when necessary.

5.2.3. Deployment

The DPF Manager MUST allow for deployment in different infrastructures and environments:

- Website: The PREFORMA toolset will be deployed at the website where you can upload files and have them checked. The website demonstrates the functionalities of the tool and will be used by the PREFORMA project to show the tool to the EC.
- Evaluation framework: The PREFORMA toolset will be deployed in the DIRECT evaluation infrastructure. This deployment allows for testing the tools in the PREFORMA project.
- Stand-alone: The PREFORMA toolset should allow for deployment as an executable to be downloaded and installed on a workstation. This ensures that the tool can be used by institutions and producers without any central IT infrastructure.
- Client-server: The PREFORMA toolset should allow for deployment on a server that enables sharing the use of the tool via multiple workstations. This enables a scalable use of the tool in digital repositories.
- Integration in legacy systems: The PREFORMA toolset should allow for integration into proprietary legacy systems, adding functionality for validating files.

5.2.4. Documentation

The source code must include comprehensive documentation which allows for automated generation of the internal API of the application.

5.2.5. Extensibility

The source code must be build in a way that makes the system easily extensible by the suppliers and/or third parties.

5.2.6. Interoperability

The DPF manager must interface with other software systems via API's.

5.2.7. Modularity

The source code MUST be built in a modular fashion for improved maintainability

5.2.8. OAIS integration

The DPF Manager application is intended for use within the OAIS Reference Framework:

- The DPF Manager MUST enable implementation of the following OAIS functions:

    - Quality Assurance at Ingest, validating (QA results) the successful transfer of the SIP to the temporary storage area.
    - Generate AIP at Ingest, transforming one or more SIPs into one or more AIPs that conform to the Archive's data formatting standards and documentation standards.
    - Archival Information Update at Ingest, providing a mechanism for validating the contents update (repackaging, transformation) of the Archive.
    - Additionally, the application must allow Producers to check whether a file conforms to the technical criteria before submission of a file to an OAIS Archive.

- The DPF Manager MUST support the following use cases (OAIS document lifecycle):

    - In Creation time: validating files by the producer before transfer to the memory institution (cf. government agencies producing public documents) .
    - At Transfer time: validating files by the memory institution when ingested in a digital repository (cf. libraries monitoring the preservation status of publications deposited in their repository) .
    - In Digitisation time: validating files as quality control of files delivered by a digitisation company or by staff. (cf. museums that perform quality control on the digital representations and documentation they produce) .

5.2.9. Open source & licensing

All software developed in the project MUST be open source and licensed under “GPL v3 or later” and “MPL v2 or later”. Any third party library or component MUST have a license that is compatible with the project license.

5.2.10. Platform independence

The DPF Manager MUST be built for portability between different platforms.

5.2.11. Performance

Due the amount of files that large memory institutions use to archive, and ingest in their daily tasks, one priority requirement is the performance processing large number of files concurrently.

Taking this in consideration, DPF manager has to be optimized to process in parallel any number of files, where every one of these files can be enough big by itself.

Also, the process has to do background analysis, letting the staff to work, waiting for an asynchronous result, but advising of the process status, in order to know if there was any problem.

It is important to notice that, even if the time in batch processing is not a problem, the result has to be quickly enough to allow memory institutions to ingest all the files they need to add to their archives.

5.2.12. Recoverability

During an operation could be an error in the hardware infrastructure. DPF manager should be able to recover and resume the process that was doing before the hardware error.

5.2.13. Reliability

DPF Manager has to operate without failures. This includes ensuring accurate data input, error-free state management, and non-corrupting recovery from detected failure conditions.

All configuration files are in XML format, that has to be validated with and standard. The PDF Manager has to be designed to ensure this reliability with a good architectural infrastructure a Built-In Application Health Checks and consistent error handling.

The reliability of the application also depends on the use of software development best practices, such as the use of test and quality assurance tools.

5.2.14. Scalability

The DPF Manager should be designed in such a way that it is easily scalable (both vertically and horizontally)

5.2.15. Security

One of DPF manager functionalities is the ability to use the tool in a client-server mode. In this mode all the communication is done over a network. Being aware of the security and privacy risks in this kind of communication is mandatory to implement an authentication/authorization system that checks who can access the server and ensure that this communication is done with privacy.

These may involve providing a DPF manager with a validation system that control server access and the use of a data encryption algorithm.

5.2.16. Testability

Unit tests shall be provided, together with the integration with a Continuous Integration service and test files. Developers should be able to run these tests to make sure the system performs as expected. The system should include the necessary documentation to help new contributors use and extend the tests suites.

6. OAIS integration

The Open Archival Information System, usually referred to as the OAIS model, is a reference model that has been widely accepted by the digital preservation community as a key standard for digital repositories.

An OAIS Archive, consisting of an organization of people and systems, which has accepted the responsibility to preserve information and make it available for a Designated Community. Where the information being maintained is deemed to need Long Term Preservation. Long Term is long enough to be concerned with the impacts of changing technologies, including support for new media and data formats, or with a changing user community.
The OAIS model is also of interest to those organizations and individuals who create information that may need Long Term Preservation and those that may need to acquire information from such Archives.

There are mandatory responsibilities that an organization must carry out in order to operate an OAIS Archives. The OAIS shall:

- Negotiate for accept appropriate information from Producers
- Obtain sufficient control of the information provided to the level needed to ensure Long Term Preservation.
- Determine which is the Designed Community who should be able to understand the information provided.
- Ensure that this information is independently understandable, without needing special resources such assistance of producers.
- Follow documented policies as procedures with ensure that the information is preserved
- Make the preserved information available to the Designated Community.

The DPF Manager shall help the OAIS archive to ensure that responsibilities.

6.1. OAIS responsibilities

6.1.1. Negotiate to accept information

The organization operating an OAIS should have establish some criteria that aid in determining the type of information that is willing to, or it is required to, accept.
The DPF Manager shall be configured to ensure that the information meets all OAIS internal standards. These criteria may include, among others, whether the information is ISO compliant, internal information structure, information representation characteristics.
To help with the ingest negotiation process the application shall generate a result in a compressible form for users or systems in machine-machine negotiations.

Pro-active negotiation

This negotiation can be done pro-actively by producers in creation time. When a producer generates content that has to be deposited in an OAIS Archive, with a particular acceptance criteria, they can check these technical properties when the information is made available.

In digitalization time this pro-active negotiation can be also applied. Archives check the technical properties of digital representations of collection items, internally or externally produced, if they meet the requirements specified in the digitization tender.

In this scenery an Archive can provide to those producers and standalone app with specific organization acceptance criteria in order to let the producer to know the results of and ingest event before it is done. Another way to provide the producers of this functionality is using a client or web interface that interacting with a servers in the organization with specific configuration.

The DPF Manager shall permit this kind of pro-active checks and interactions in aid of reduce negotiation effort and time.

Ingest negotiation

In a Data submission session, after a Submission Agreement, between the producer and manager, multiple submission information packages (SIP) should be checked in a temporary storage area. A submission information package could contain different kind of content with different organization.

A DPF Manager command line interface can be used in this case as part of Commands and utilities in OAIS Common Services used in this case by the Quality Assurance function.
In this transfer time the application shall handle different source inputs as, single files, files in directories, URL's, compressed files, container formats, bitstream as well as provide an effective way to process or understand the report of this inputs. Global report shall be created to give an overall result of ingest result and the most common problems. Also a specific report for every file check shall be generated.

When OAIS Generate a new AIP at ingest form one or more SIPs transformation, the DPF manager shall conform the archive's data formatting standards and documentation standards.

6.1.2. Obtain sufficient control for preservation

The OAIS must assume sufficient control over the Content Information and Preservation Description Information so that it is able to preserve it. The factors that influence in assuming the control are categorized, as follows.

- Copyright implications, intellectual property another legal restrictions.
- Authority to modify representation information
- Agreement with external organizations.

OAIS should ensure that its subsequence actions to preserve the information and make it available conform to these rights and limitations.

Preservation process

DPF Manager shall validate is any metadata information in the content conflict with the Submission Agreement transfer rights between producer and Manager as well as report if there is any legal restriction or legal issue in data structure that could be in conflict with proprietary rights.

This information is crucial and establishes the guidelines for ingestion and rules for dissemination or duplication the information when necessary.

Dissemination process

During dissemination process DPF Manager shall inform if there is any personal data in metadata to ensure consistency with the user rights also come under this type of operations.

Also shall check if all metadata meets the Dissemination Agreement rights accorded between Archive and Consumers.

6.1.3. Determine Designed Community

The submission of Content Information requires a determination as to who the expected Consumers, or Designed Community, of this information will be. The OAIS Archive should make the effort to ensure that this community can understand the information.

Designing a target community help to decide what kind of information should be provided in support of the Representation Information and the Preservation Description Information.

The DPF Manager shall check content information to ensure that the content information can be understood by Consumers. Content information following standards should improve the likelihood that the information will be understandable.

6.1.4 Information is independently understandable

Digital Content Information and PDI need adequate Representation Information to be Independently Understandable to the Designated Community. This Representation Information has to be as complete as possible otherwise some of the information may be understandable only for a few specialists and be lost when they are no longer variable.

DPF Manager shall inform about information structure and content in order to generate the Representation in appropriate way guaranteeing futures disseminations, reproductions, modifications.

The DPF Manager shall extract all metadata associate with the Content Information and confirm that's trustful to complete the Description Information.

6.1.5. Follow established preservation policies and procedures

It is essential for an OAIS to have documented policies and procedures for preserving its AIP's and it should follow those procedures.

OAIS Archives need tools for monitoring and auditing all the activity, interactions, submission because apply the preservation plans when they are needed is essential to avoid being caught with very costly system maintenance, emergency system replacements, and costly data representation transformation.

DPF manager shall generate a log with all the results. This log can be audit to execute a preservation plan if there is any issue.

Preservation Planning functions include evaluating the content of the Archive and periodically recommending archival information updates, recommendation migrations of current Archives holding, developing recommendations for Archives standards and policies, providing periodic risk analysis reports.

DPF manager shall be configured to be executed in automated form in a regular period of time to audit the AIP's. An XML file with all the configuration has to be provided in order to automate these operations. DPF report shall supply a preservation index to determine if an AIP needs to be migrated.

6.1.6. Make the information available

By definition, an OAIS makes the Content Information in its AIPs visible and available to its Designated Communities. In this process Archive and Consumer accord in a Dissemination Agreement how these access should be. During this dissemination one or more AIP has to be transform in one or more DIP.

During these process DPF manager shall be used for auditing submissions and ensure Archive dissemination standards policies and procedures to support the preservation objective of the OAIS.

Moreover, DPF manager shall be used to advise if any metadata has private personal information that has to be removed in the DIP.

6.2. Digital migration

In the OAIS model Digital Migration is a transfer of information in order to preserve it. This digital migration can be motivated by different factors such as improvement in cost-effectiveness, new consumer-service requirements or media decay.

Digital Migrations are time consuming, costly, and expose the OAIS to greatly increased probabilities of information loss. Therefore, an OAIS has a strong incentive to consider Digital Migration issues and approaches.

The risk is higher when the migration involves changing the bit sequences, in this scenario something may go wrong in the process and some unintended changes to bits may take place.
A Digital Migration where there is some change in the bits of the Packaging Information is called Repacking, by the other hand, is a Transformation when there is some change in the Content Information or PDI bits while attempting to preserve the full information content.

From an OAIS functional perspective, migrations involving repackaging or transformations are orchestrated by the Administration sub-function called Archival Information Update.

Archival Information Update requests a Dissemination Information Packages (DIPs) from the Access function. It then processes DIPs to create Submission Information Packages (SIPs) and feeds these to the Ingest function to turns these into new AIPs for storage by the Archival.

During the migration process DPF Manager shall be used by de Data Management to obtain the Descriptive Information used in the generation of DIP. If Generate DIP function requires any Data Content transformation here DPF Manager shall validate these transformations. Later, when the SIP is ingested the Quality Assurance function shall use the DPF manager to ensure that generated AIP complies with archives data formation and documentation standards as well as extracting descriptive information.

6.3. OAIS functional Entities

The OAIS model defines six functional entities:

- Ingest Functional Entity
- Archival Storage Functional Entity
- Data Management Functional Entity
- Administration Functional Entity
- Preservation Planning Functional Entity
- Access Functional Entity

Each entity has a well-defined role inside the OAIS model and there are some of these entities where the DPF Manager could be integrated.

6.3.1. Ingest Functional Entity

There are two components inside the ingest entity where the DPF manager could be used.

The Quality Assurance function

This function validates (QA results) the successful transfer of the SIP to the temporal storage area. Here the DPF manager could validate the file with the institution criteria acceptance.

The Generate AIP function

This function transforms one or more SIPs into one or more AIPs this may involve file format conversion, so DPF manager could validate the new files generated, or reorganization of the Content Data. The Generate AIP also may be request to produce the Descriptive Information that completes the AIP. DPF manager report contains all the internal file structure and metadata that it is adequate to fill the Descriptive Information. The report generated by the DPF manager is also suitable to be sent to the Audit Submission function to ensure that the Content Information is understandable and usable by the Designed Community.

6.3.2. Administration Functional Entity

Inside the Administration Functional entity there are two components that could indirectly benefit of the DPF manager use.

Establish Standards and Policies function

This function is responsible for establishing and maintaining the Archives system standards and policies. The use of a general XML configuration file with the criteria acceptance of a file fills in the functionalities of these component where it could have an XML configuration file for each file format that the institution manage.

Audit Submission

The audit submission must verify that the quality of the data meets the requirements of the Archive and the review committee. In this case the report generated by the DPF manager could be suitable for auditing the submission.

6.3.2. Access Functional Entity

Generate DIP

This function accepts a dissemination requests, retrieves the AIP from Archival Storage, and moves a copy of the data to a temporary storage area for further processing which could include conversion between different data type or output formats. Then, the DPF manager could be used in order to validate that these transformation has been done following the required criteria’s.

6.4. Archive interoperability

In general one OAIS is not interoperable with another, however, there are a number of reasons that some level of interoperability may be desirable, motivated for example by Users, Producers or Management.

Consumers may wish common schemas in Package structures, global site access,
Producers may wish to have a single depository for all their products, common SIP schema for submission to different Archives.

Managers want to reduce the cost through sharing expensive hardware, software, preservation efforts.
There are different degrees of interactions between Archives:

- Cooperating
- Federated
- Shared Resources

6.4.1. Cooperating archives

Archive with potential common producers, common submissions standards and common dissemination standards, but no common finding aids.

This interaction is based on standards agreements among two or more Archives. Cooperating archives have related communities of interest, so they have standardized their submission and dissemination methods for the benefit of Consumers and Producers.

DPF Manager shall be configured through an xml file. This xml configurations files shall be validated and shared between these institutions in order to audit the submission and dissemination in the same way.

6.4.2. Federated archives

Archives with both a Local Community (The original Designed Community served by the Archive) and a global community which has interest in the holding of several OAIS Archives and has influenced those Archives to provide access to their holding in a common way.

This Global access is accomplished through addition of a standard ordering and dissemination mechanist.

In this case the application shall check the content information in dissemination time in order to standardize the dissemination package in these communities. Sharing the same configuration file could be a solution.

6.4.3. Archive with shared functional areas

Archives that have entered into agreement with other archives to share resources.
The motive for this may be share expensive resources such file managements system, peripheral devices for ingestion or dissemination information, complicated transformations between SIP, AIP, DIP.

This requires various standards internal to the Archive (such as ingest-storage and access-storage interface standards).

DPF Manager shall be configured in a client-server mode, all archive here are sharing the same Quality Assurance function control for ingestion.

7. Use scenarios

This section describes how the DPF Manager can be used in different scenarios, from standalone to client-server and other different cases.

For each scenario a description is provided, together with an example of how the DPF manager should be used.

7.1. Stand-alone

In the standalone mode, users should have a self-contained application, meaning that it should be able to work without requiring any external connection or other software.

The shell can run in standalone mode with two different interfaces: command line and a Graphical User Interface (GUI).

To run the shell from the command-line, a user will be able to execute it with the following command:

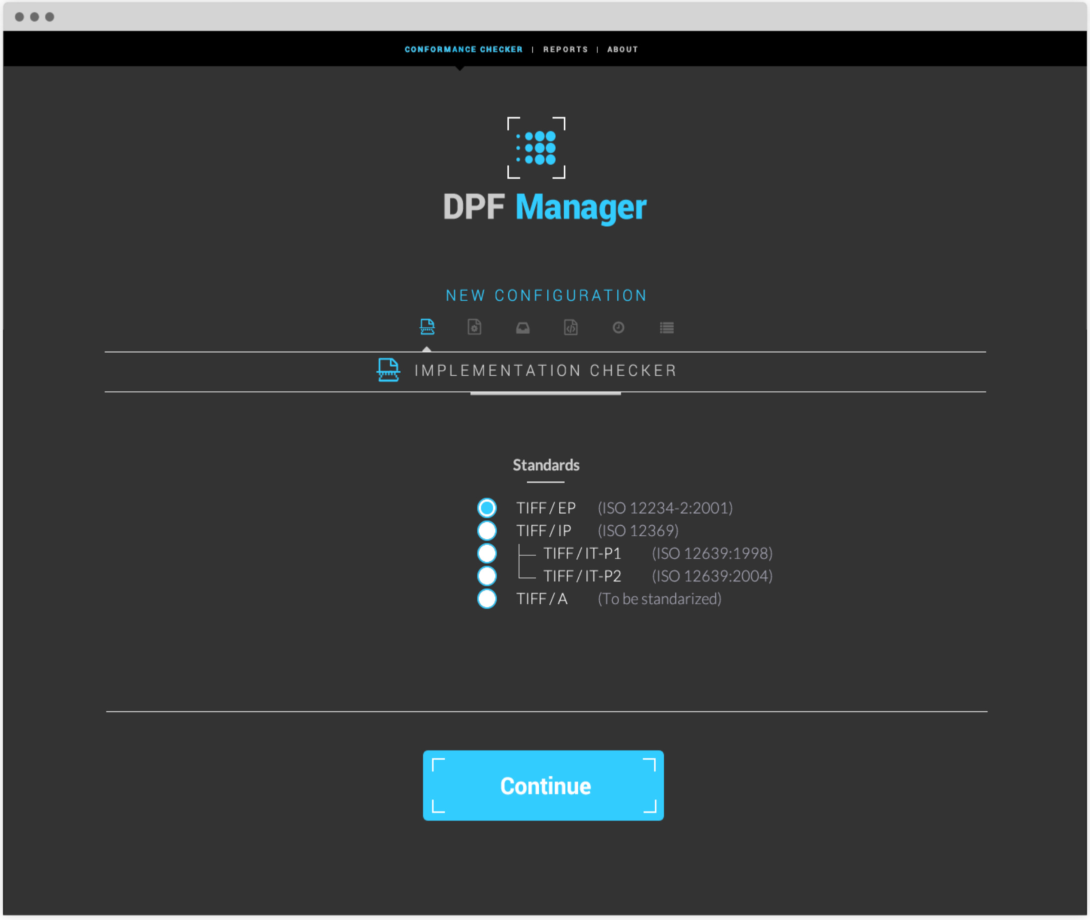
```
c:\>dpfmanager.exe        -files=/archive/*.tiff        -config=config.xml        -reports_folder=/tmp/reports -fixed_files_folder=/tmp/files
```

To launch the GUI, the user can execute the following command (or they could add a shortcut to their desktop, so they don't need to open the command line at all):

```
c:\>dpfmanager.exe
```

In this case there is no need to define any parameters or configuration files from the command line, as everything will be managed by the GUI.

Below is an example of the GUI screen where the user can configure the standards they want to check a file against (implementation checker configuration):

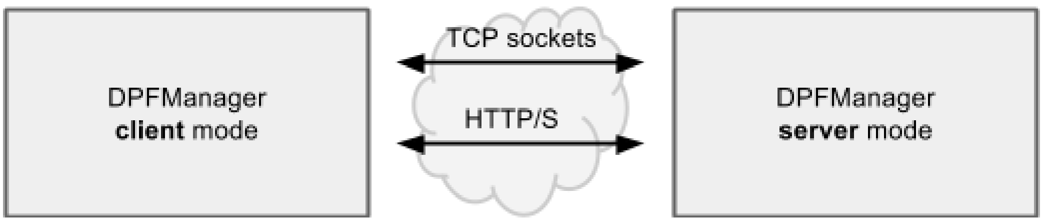


7.2. Client-server

DPF Manager can work both as a client or as a server, depending on the options passed to the command line or specified in the configuration file.

A DPF Manager started in client mode is able to communicate with one or multiple DPF Manager started in server mode.

The communication can be either via TCP sockets or HTTP/S:



In client mode, the DPF Manager needs to know how to locate the DPF Manager servers. The command line options are very similar to the standalone mode, but adding the address and port of the server:

```
client:\>dpfmanager.exe  -server=80.45.32.45  -port=80  -files=/archive/*.tiff  -config=config.xml -reports_folder=/tmp/reports -fixed_files_folder=/tmp/files
```

On the server side, users will need to launch the DPF Manager in server mode:

```
server:\>dpfmanager -mode=http_rest_server -port:80 -default_config=config.xml
```

7.3. Web application

The DPF Manager can be integrated with other applications via the shell or a sockets/HTTP interface, it can easily be used as part of a web application that gives users the possibility to upload a file, get it checked and see the resulting report online:

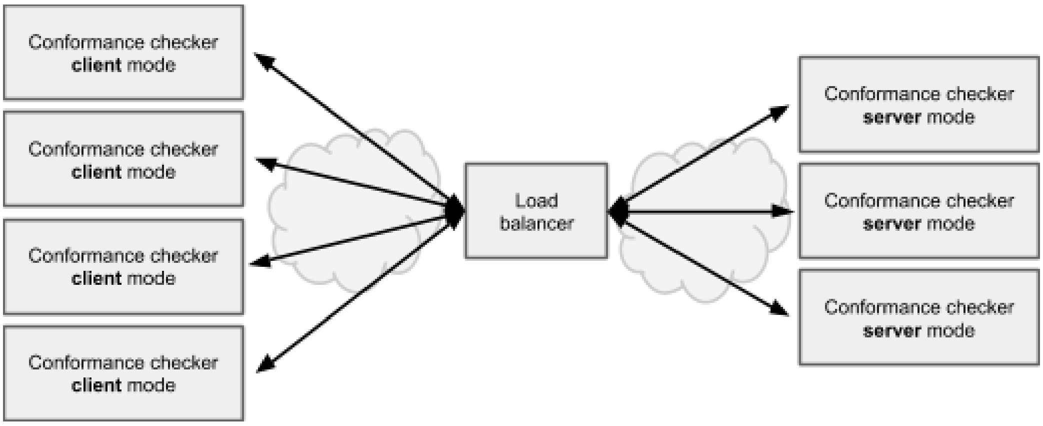
In this scenario, the integration between the web application and the DPF Manager can be done through:

- The command line API.
- The DPF Manager in server mode, accepting requests from the web application.
- The web application using the internal Conformance Checker API.

7.4. High availability

For big memory institutions with large archives and multiple staff members validating files, a simple client-server architecture with a single DPF Manager acting as a server may not be enough, as it may become overloaded with requests.

As the DPF Manager in server mode can use the TCP or HTTP protocol, it is very easy to deploy multiple DPF Manager acting as servers behind a proxy or a load balancer, so they can accept as many simultaneous requests as needed, without requiring complicated configurations on the client side:



7.5. Periodical checks

Archivists must be able to assign a task to theDPF Manager to check files at regular intervals.

This can be done through any of the DPF Manager user interfaces (GUI, command line client and network server). To give an example, here we show how it would be done using the command line:

```
c:\>dpfmanager  -mode=cron  -frequency='0  0  *  *  *'  -files=/archive/*.tiff  -
config=config.xml -reports_folder=/reports -fixed_files_folder=/files
```

For demonstration purposes only, we have used a syntax similar to the CRON task runner found in most Unix/Linux operating systems. In the example above, the DPF Manager would check all the TIFF files in the /archive folder every day at midnight.

7.6. External producer

When working with external producers, memory institutions have three different options:

1. The external producer uses the standalone version of the DPF Manager.
2. The external producer runs the DPF Manager as a client, communicating with a DPF Manager running as a server inside the memory institution infrastructure.
3. The memory institution offers a website where external producers can upload files and get them checked.

The second and third options have the advantage that the memory institution could update the policies in the policy checker without having to send an updated version of the DPF Manager to the external producers.

7.7. Software developers

Our proposed architecture makes it very easy for software developers to integrate the DPF Manager framework in their products.

They have two ways of doing so:

1. They can package the standalone version of the DPF Manager as part of their software installation, and use it through the command line client (in a way that is completely transparent to the end users)

2. If they are developing their software using the same programming language as the DPF Manager, they could decide for a deeper level of integration, by integrating the DPF Manager framework source code as a library and using the DPF manager internal API, without needing to interact via the command line:

my_own_software.java

```
import EasyInnova.DPFManager.*;

DPFManager dpfm = new DPFManager(options):
Report r = dpfm.check('file1.tiff');
```

7.8. Integration into the OAIS model

With all the scenarios showcased above, our architecture allows the DPF Manager to be used in multiple ways, either manually or automatically and from a single computer to a networked environment.

It is therefore an architecture that allows for integration into the OAIS model at every required step of the process (check at creation, transfer, digitisation and migration times), whether it is done manually or integrated into an OAIS model management software.

8. DPF Manager interfaces

The previous section illustrated how the DPF manager can be used in different scenarios. The following sections in the document will introduce the different interfaces that DPF Manager provides to users.

Our design of the DPF Manager includes three different interfaces: a command line API, a web service API, and a graphical user interface (GUI).

The following table shows the target audience for each interface:

	Human users	Other software/services
Command line	✓	✓
Web service		✓
GUI	✓	

Sections 9, 10 and 11 of this document introduce and explain how each interface works.

9. Command line API

9.1. Introduction

The command line API is aimed at human users and integrations with legacy systems and other software.

It provides the same functionality as the Graphical User Interface and the Web service API.

9.2. Command line options reference

All the functionality of the DPF Manager is accessible through the command line API. The options available in application executable are described in the table below:

Option	Description
no option	When the dpfmanager is called without any option, the program will start in GUI mode
-help	List all the available commands, with a short explanation for each one
-info	Returns a list of all the Conformance Checkers that are available through this Shell, and a structured description of what each Conformance Checker can do
-list	Shows a list of the files that have been checked until now, with a summary of the result. This includes showing the results of periodical checks
-limit	Maximum number of results returned by the -list option (e.g. 100)
-page	If there are more results than -limit, request the x set of results
-files	Path to the file or group of files (using wildcards) that the user wants to check
-config	Path to the configuration file
-reports_folder	Path to the folder where the generated report/s will be put
-fixed_files_folder	If the metadata fixer is invoked and as a result the original file is modified or a duplicate file is created, this option allows the user to define the path to the folder where these files will be put
-server **-port**	When the DPF Manager is acting in client mode (-mode option), it needs to know the location (IP address or name of the server and port) of the application acting in server mode.
-mode	If unspecified, the shell will start in standalone mode.

	The mode option can have three values: - **Client**: the DPF Manager will start in client mode, sending requests to a DPF Manager running in server mode. The options -server and -port are mandatory in this mode - **Http_rest_server**: the DPF Manager will start in server mode, exposing an HTTP REST API - **Cron**: the DPF Manager will start in cron mode and perform periodical checks of the files specified with the -files option. The frequency of the checks can be specified with the -frequency option
-frequency	If the DPF Manager is started in cron mode (periodical checks), this option specifies how often the checks should be performed. The syntaxis is the same as the one used by the popular CRON service found in most Unix/Linux operating systems. As an example, a value of '0 0 * * *' means that the files will be checked every day at midnight.
-default_config	If the DPF Manager is started in server mode, the users can define a default configuration. This configuration can be overwritten on each request that the server receives, if the client sends their own configuration. If the client doesn't send any configuration file, the default will be used.

10. Web service API

10.1. Introduction

The web service API is available when the DPF Manager is run in server mode (see Use scenarios section of this document).

It exposes an HTTP REST API that can be used with any client that supports HTTP requests.

10.2. End-point reference

Below is a list of all the API endpoints, together with a description of the data expected and the data returned. The full API specification is included in the appendixes.

HTTP verb and end point	Description
GET /conformance_checkers	Get a list of all the Conformance Checkers accessible through this shell. **Input**: no parameters required **Output**: returns a list of all the Conformance Checkers accessible through this Shell, together with a structured description of their capabilities (what the implementation checker, policy checker, reporter and metadata fixer can do)
POST /conformance_checks	Request a check for a single or multiple files. **Input**: list of files to be checked, and the configuration for the implementation checker, policy checker, reporter and metadata fixer. **Output**: returns the data needed by the users to access the final report (checking large files can take some time, so the checking process is asynchronous).
GET /conformance_checks	Returns a list of all the previous validations and their result. **Input**: no parameters required, optional parameters include pagination of the results. **Output**: a list of all the files that have been checked until now, together with the result of the check.
GET /results/{request_id}	Returns the result for a given validation request. **Input**: id of the request, as returned by the POST

	/conformance_checks call. **Output**: returns a reference to the report/s generated for that request (if the validation process has finished), and to the modified files if the metadata fixer was invoked.
GET /status/{request_id}	As the validation process is asynchronous, the users can check the status of the request at any time to see if it is still ongoing or has already finished. **Input**: id of the request, as returned by the POST /conformance_checks call. **Output**: the status of the request (ongoing, finished), and extended information (e.g. link to the results if the request has finished)
GET /tasks	Returns a list of the periodical checks that are running on the system **Input**: no parameters required. **Output**: the list of periodical checks that the DPF manager has been requested to run
DELETE /tasks/{task_id}	Deletes a task (periodical check). **Input**: id of the task to delete **Output**: the result of the operation (success or failure)

11. Graphical User Interface

11.1. Introduction

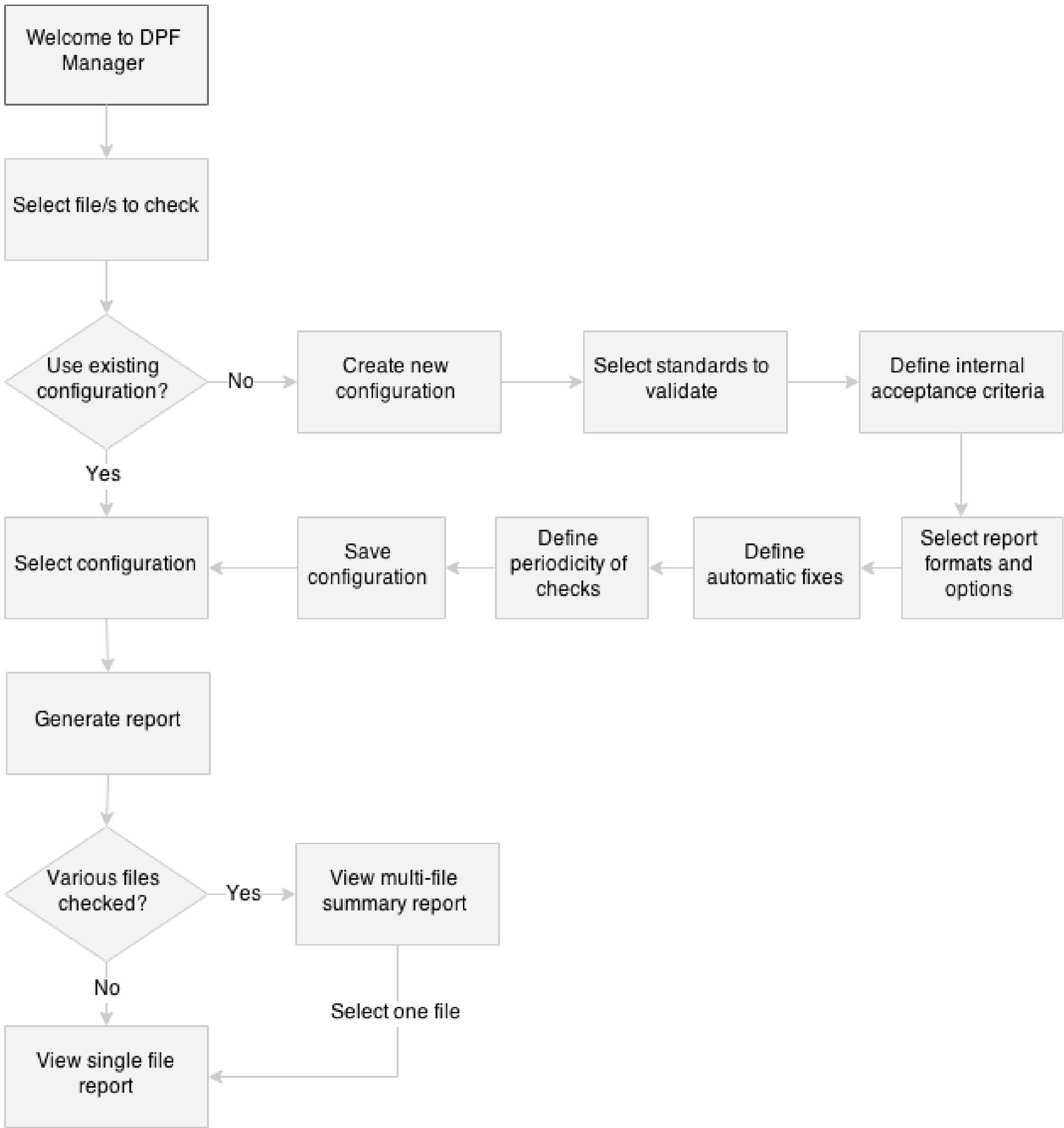
The DPF Manager GUI is targeted at users who prefer to work with a visual tool rather than through the command line, independently of their level of expertise.

The GUI guides users through the validation process, from specifying which standards should be used, to defining the rules of the policy checker, the formats of the reporter and the options of the metadata fixer. It also allows users to view the reports from inside the DPF Manager itself

The DPF Manager GUI can be used in standalone mode, as well as a client in a client-server environment.

11.2. Workflow

Here is the workflow of the GUI:

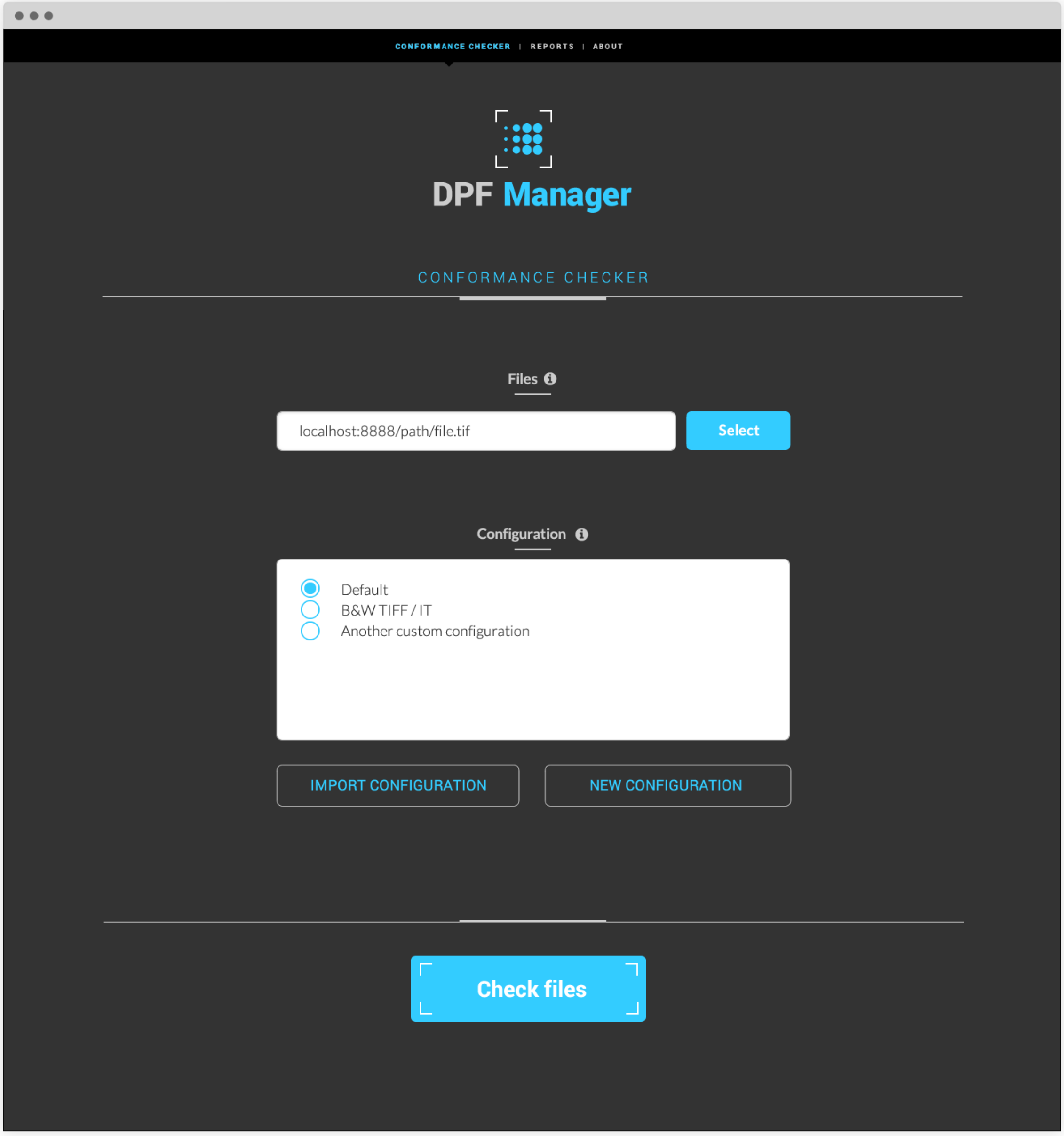


11.3. Screens

11.3.1. Home screen

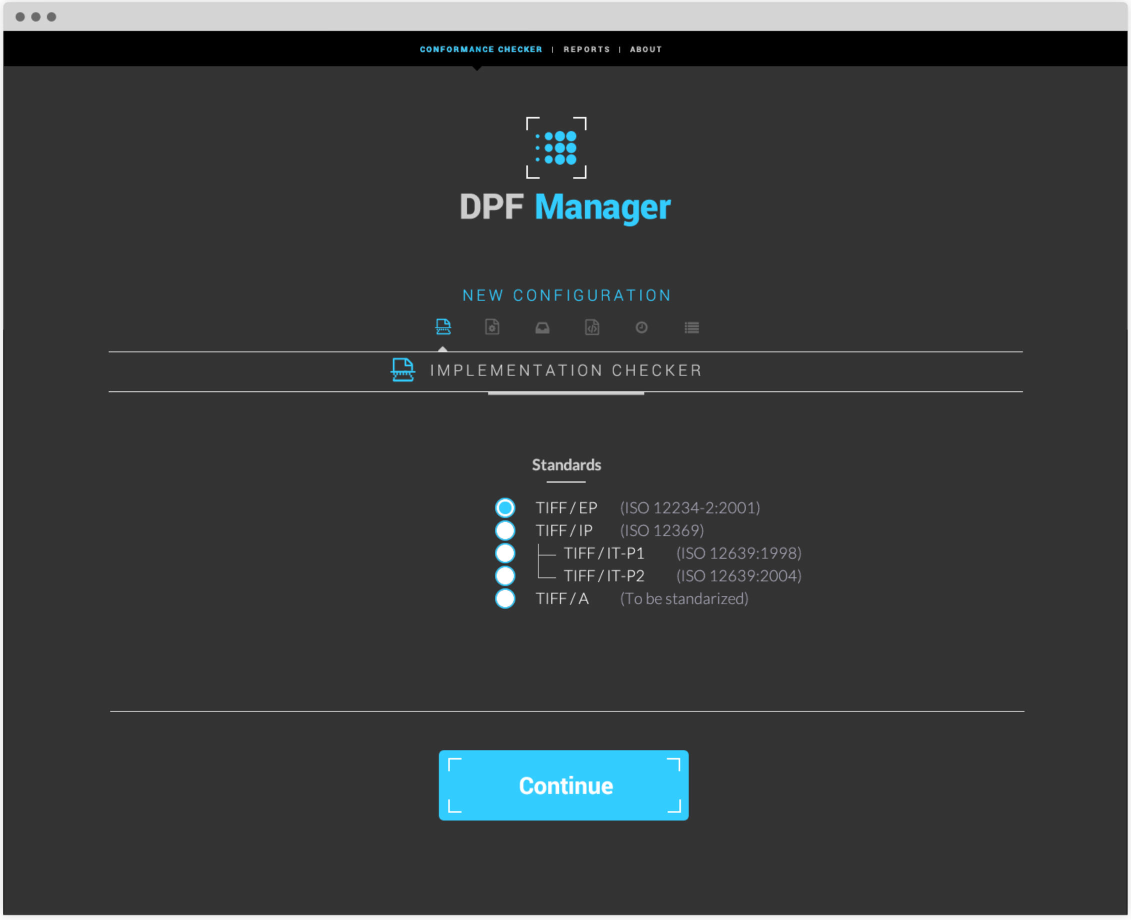
The home screen is the main screen of the application, the screen users see when they open it.

In this screen the users can select the file/s to check, and the configuration to use. In case no configuration exist, or the user wants to create a new one, they can do so from this screen.



11.3.2. New configuration: implementation checker

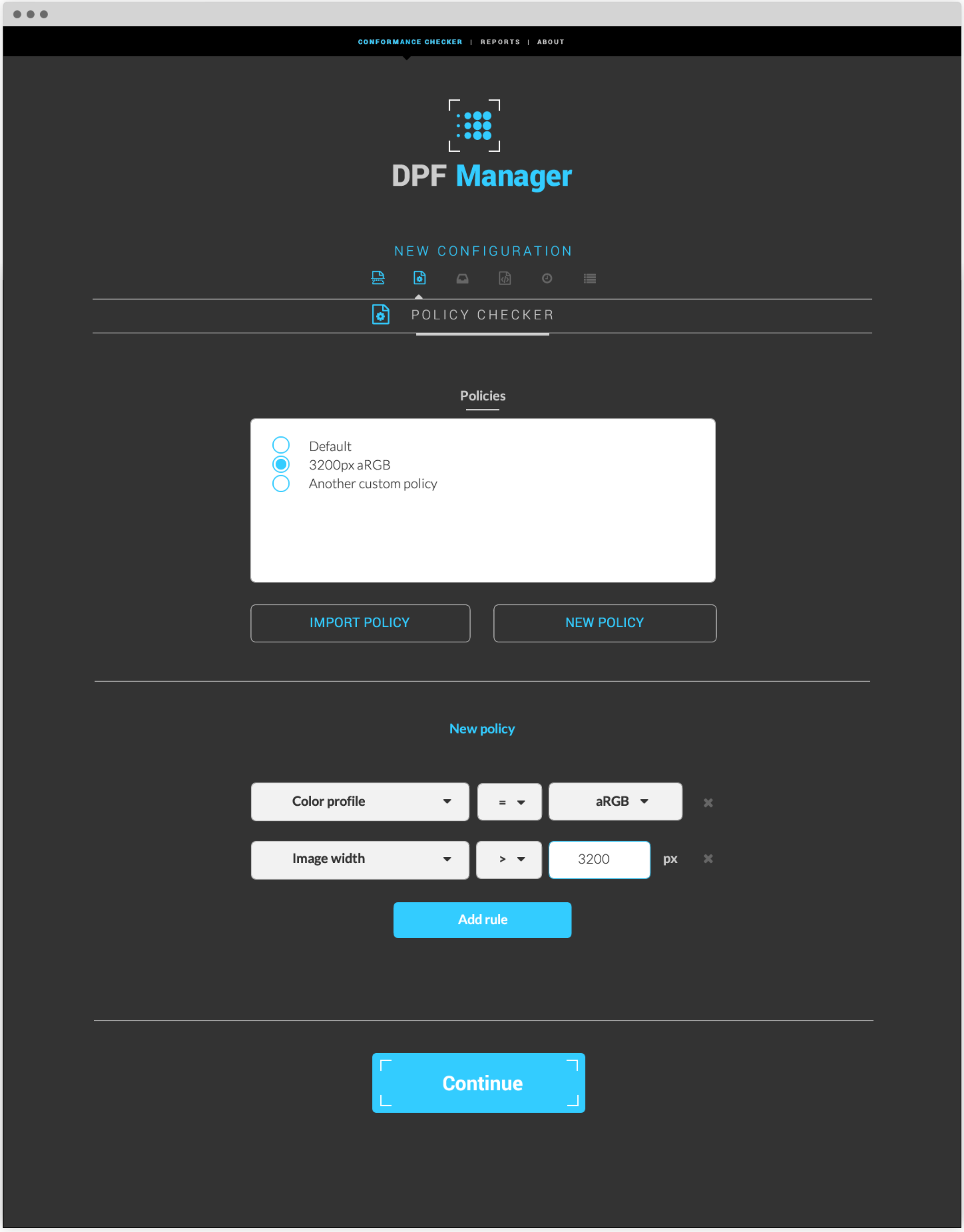
The first step when creating a new configuration is to define which standards the selected files will be validated against:



11.3.3. New configuration: policy checker

After defining the standards, the users can define the acceptance criteria based on their own internal policies.

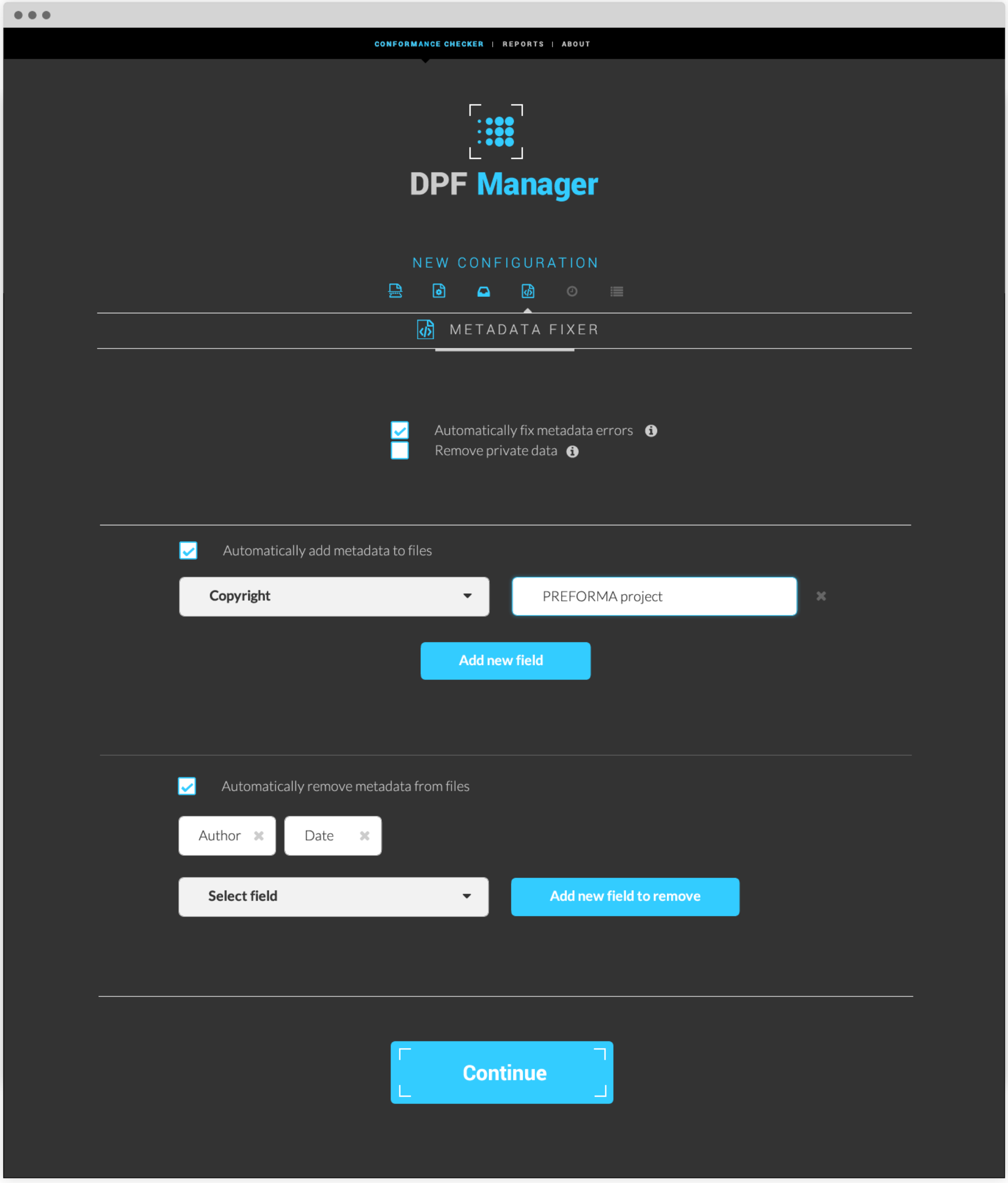
They can create a new set of rules, or load an existing one. Each rule is composed of a field name, an operator, and a value:



11.3.4. New configuration: metadata fixer

In case a file does not conform to a standard, the user can enable the automatic fixing of simple issues.

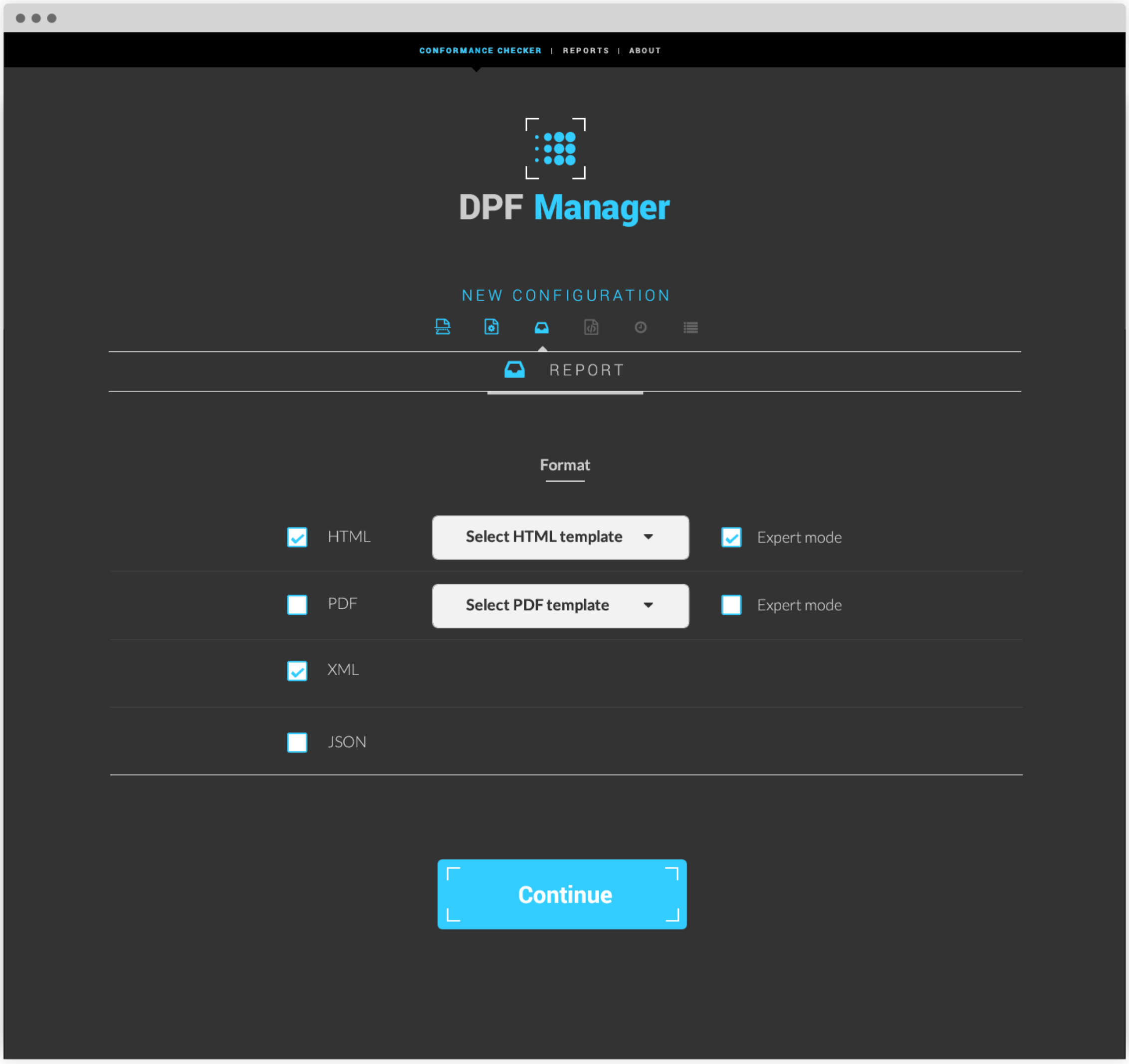
Users can also define which metadata should be automatically added to the checked files, and which metadata should be removed:



11.3.5. New configuration: reporter

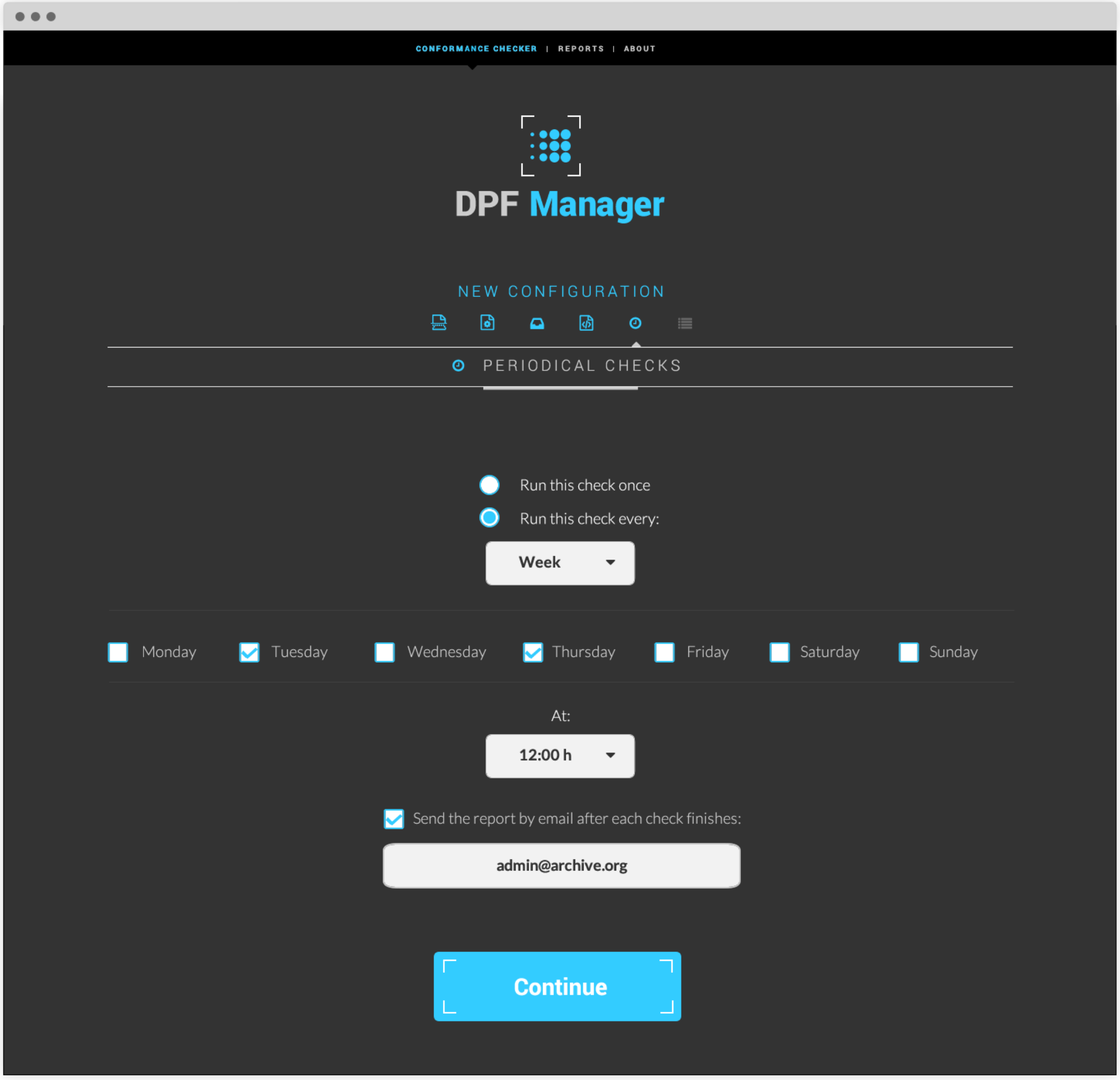
In this screen users can select the format/s of the report that will be generated for each checked file.

For the HTML and PDF reports, users will be able to select between expert and non-expert reports (the difference is the level of detail displayed in the report), and whether they want to use a custom template. Custom templates can be used to produce reports with the logos (and other visual elements like typography or color schemes) of the institution or company using the DPF manager:



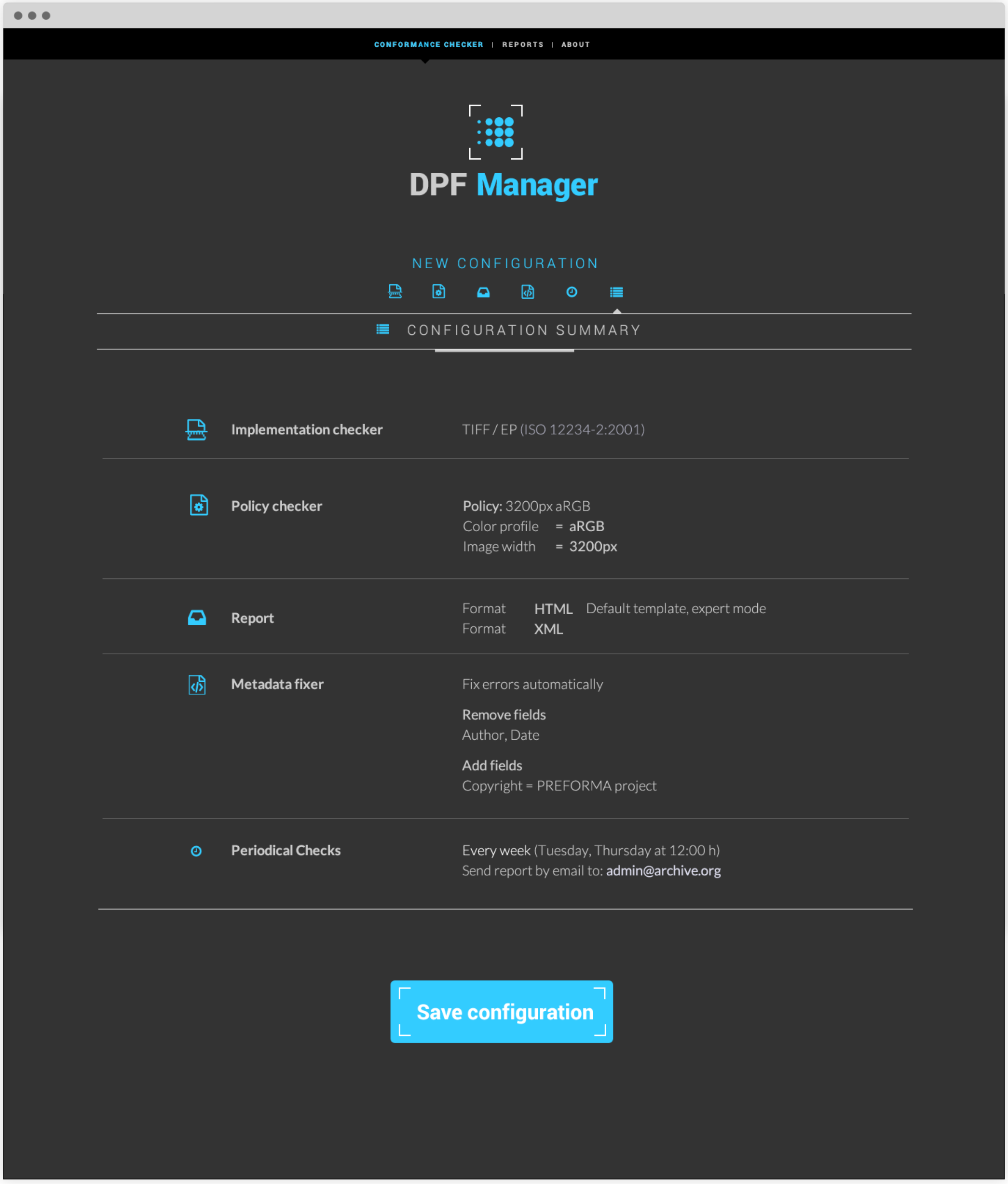
11.3.6. New configuration: periodical checks

When configuring the validation process, users can define whether the process should run once, or whether it should run periodically (e.g. every weekday at 12am).
If the checks are run periodically, then users have the option to request that the resulting report for each check is sent to a list of email addresses.



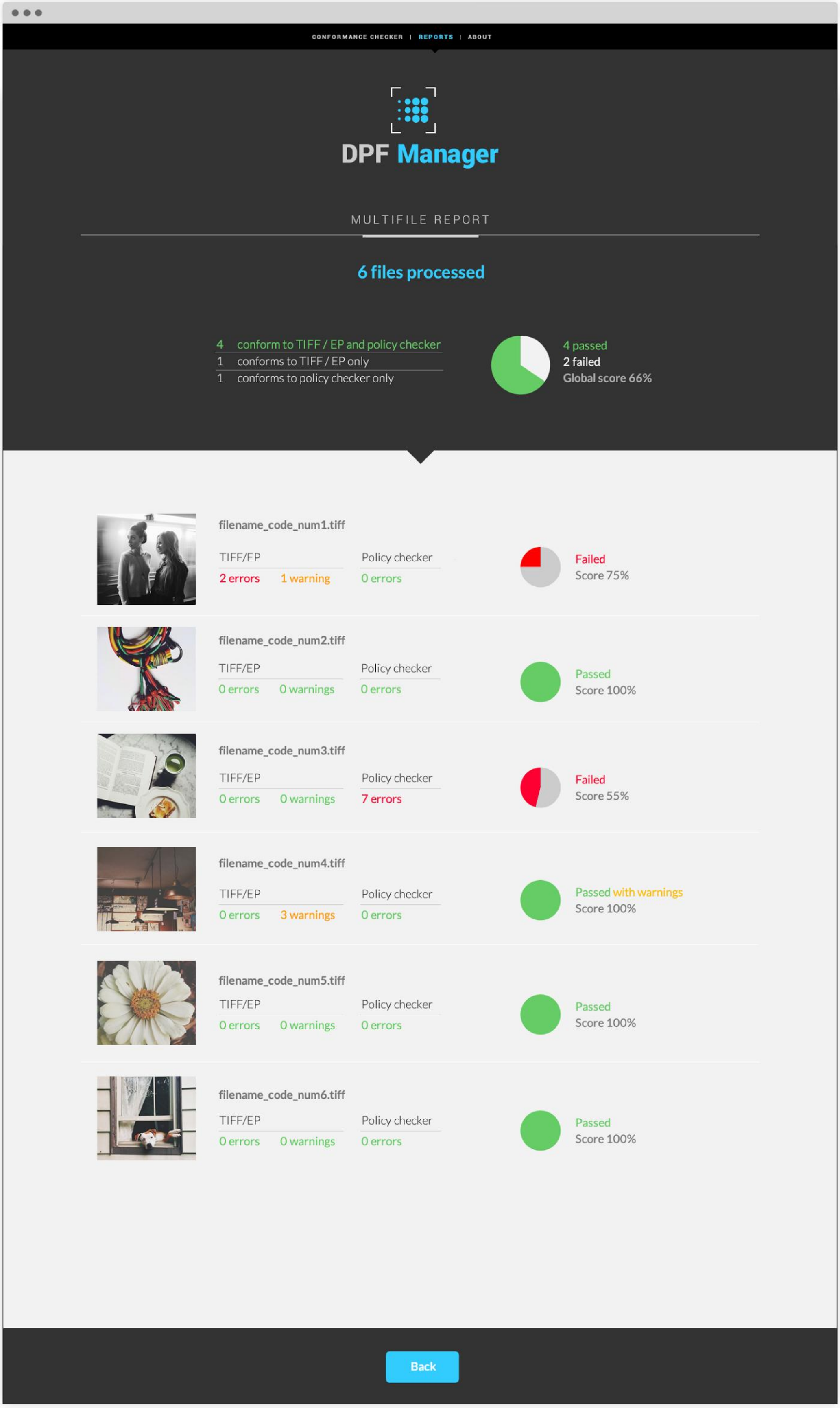
11.3.7. New configuration: summary & save

In this screen the GUI shows a summary of all the options that the users have selected during the configuration process. This configuration can be saved for later use:



11.3.8. Multiple-file report

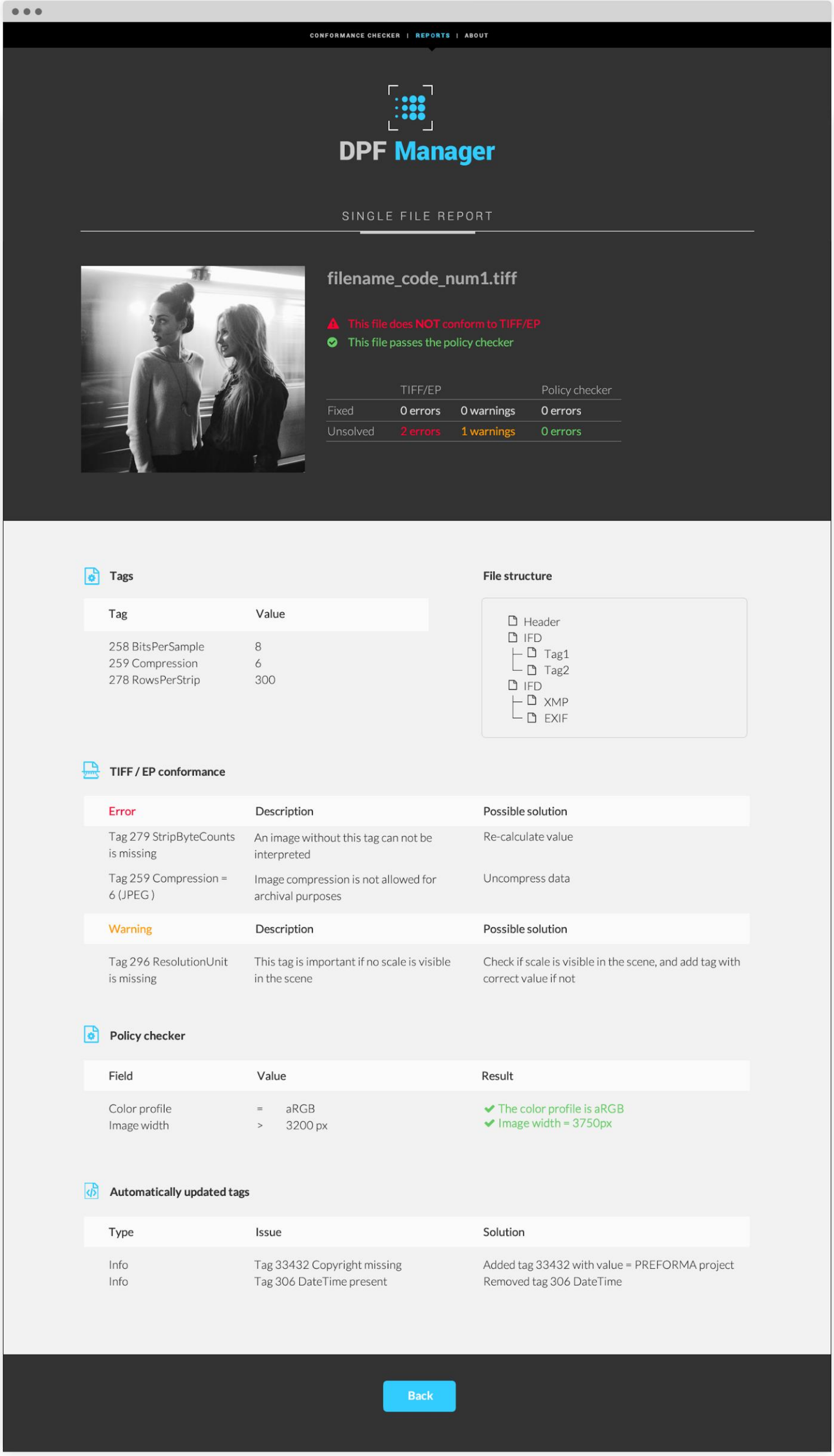
When the user has selected more than one file in the home screen, the DPF Manager GUI will show first a summary of the results of each individual file. With this information, users can see at once how the process went for all the files.


CONFORMANCE CHECKER | REPORTS | ABOUT
DPF Manager
MULTIFILE REPORT
6 files processed
4 conform to TIFF / EP and policy checker
1 conforms to TIFF / EP only
1 conforms to policy checker only
4 passed
2 failed
Global score 66%
filename_code_num1.tiff
TIFF/EP
2 errors 1 warning
Policy checker
0 errors
Failed
Score 75%
filename_code_num2.tiff
TIFF/EP
0 errors 0 warnings
Policy checker
0 errors
Passed
Score 100%
filename_code_num3.tiff
TIFF/EP
0 errors 0 warnings
Policy checker
7 errors
Failed
Score 55%
filename_code_num4.tiff
TIFF/EP
0 errors 3 warnings
Policy checker
0 errors
Passed with warnings
Score 100%
filename_code_num5.tiff
TIFF/EP
0 errors 0 warnings
Policy checker
0 errors
Passed
Score 100%
filename_code_num6.tiff
TIFF/EP
0 errors 0 warnings
Policy checker
0 errors
Passed
Score 100%
Back

11.3.9. Single file report

A single report is produced for each file the DPF Manager checks. The content of the report will change according to the preferences set in the configuration (expert or non-expert reports).

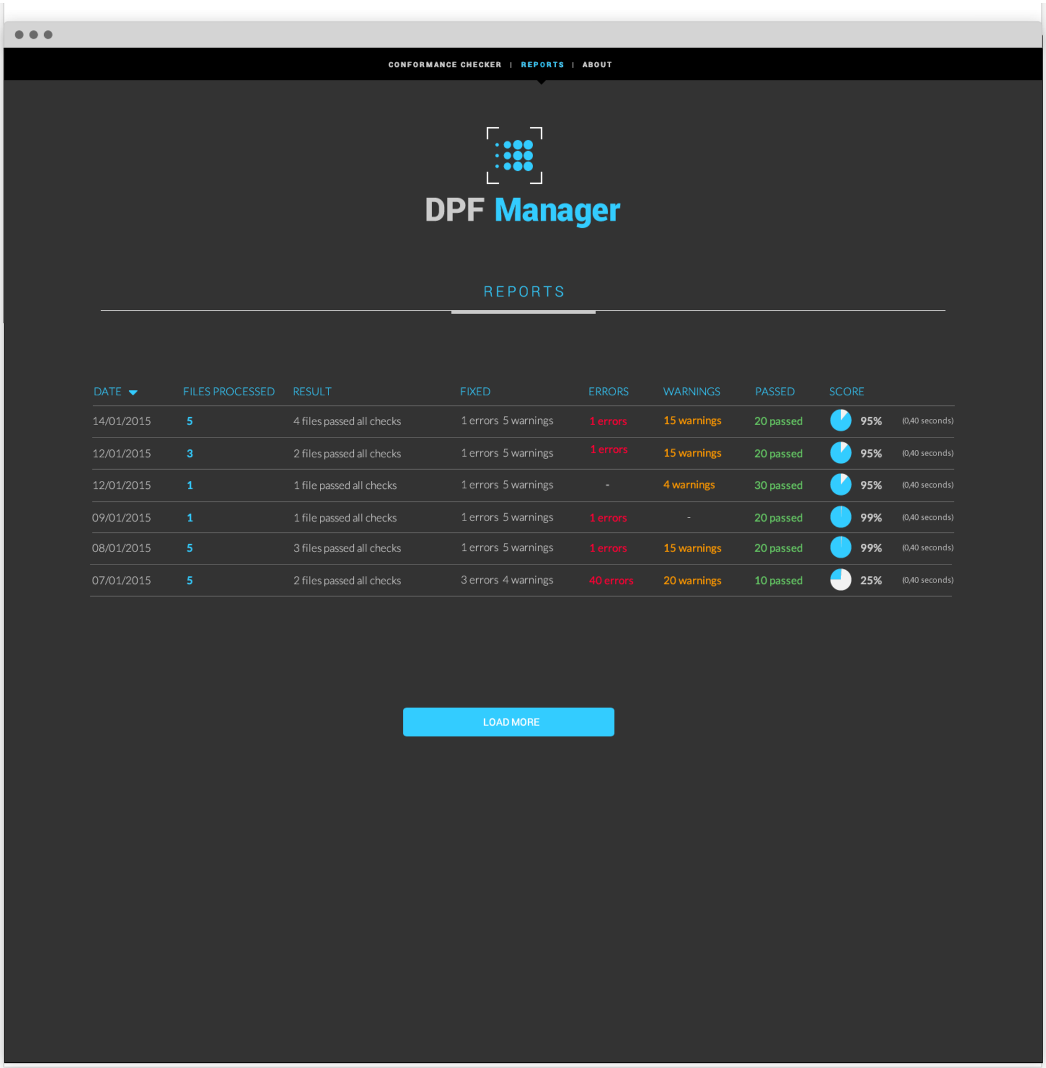
Each file is assigned a conformity score that makes it very easy to see how far a file is from being conformant. This score depends on the number of errors that the system has identified during the implementation and the policy checking process. The highest the number of errors, the lowest the score will be:


CONFORMANCE CHECKER | REPORTS | ABOUT
DPF Manager
SINGLE FILE REPORT
filename_code_num1.tiff
This file does NOT conform to TIFF/EP
This file passes the policy checker
TIFF/EP
Policy checker
Fixed
0 errors
0 warnings
0 errors
Unsolved
2 errors
1 warnings
0 errors
Tags
Tag
Value
258 BitsPerSample
8
259 Compression
6
278 RowsPerStrip
300
File structure
Header
IFD
Tag1
Tag2
IFD
XMP
EXIF
TIFF / EP conformance
Error
Description
Possible solution
Tag 279 StripByteCounts is missing
An image without this tag can not be interpreted
Re-calculate value
Tag 259 Compression = 6 (JPEG)
Image compression is not allowed for archival purposes
Uncompress data
Warning
Description
Possible solution
Tag 296 ResolutionUnit is missing
This tag is important if no scale is visible in the scene
Check if scale is visible in the scene, and add tag with correct value if not
Policy checker
Field
Value
Result
Color profile
= aRGB
The color profile is aRGB
Image width
> 3200 px
Image width = 3750px
Automatically updated tags
Type
Issue
Solution
Info
Tag 33432 Copyright missing
Added tag 33432 with value = PREFORMA project
Info
Tag 306 DateTime present
Removed tag 306 DateTime
Back

11.3.10. Checking & report history

This screen shows a list with all the checks that have been performed until now and their result.

This screen is particularly useful when users set a check to be executed periodically (e.g. every Thursday at 12pm), as they can see the results for each execution.



12. Reporting

12.1. Introduction

The end goal of the DPF Manager is to let users know if a file/s conforms to a given standard and to a set of rules (acceptance criteria) defined by each memory institution.To do that, DPF Manager generates a report for each file.

The reports can be generated in four different formats, XML and JSON (optimal for machine-to-machine communication) and HTML and PDF (for human consumption).

The XML and JSON reports have the same structure but differ in the syntax used (XML or JSON syntax).

The HTML and PDF reports can be created with different levels of detail, aimed at expert and non-expert users. They can also be customized via templates, so the reports match the corporate image of the institution (logo, color scheme, etc…).

When DPF Manager checks more than one file, the system creates a single report for each requested file, and collects part of that information into a summary report (multiple file report).

12.2. Multiple file report

The goal of the multiple file report is to show compliance statistics for all the requested files in a single document, allowing the user to get a good idea of how the process went without having to open each one of the individual reports.

This report can be generated in the four formats specified before (XML, JSON, HTML and PDF).

12.2.1. Summary

The summary shows the following information:

- Number of files checked
- Number of files that conform to the selected standard
- How many errors have been detected, and how many have been automatically fixed (if the metadata fixer was enabled).
- A global score (an average of the individual scores for each file)

12.2.2. Information per file checked

After the summary the global report includes a summarized view of the results for each individual file. This information includes:

- Thumbnail and location (path) of the file checked
- Number of errors and warnings detected, and how many have been automatically fixed
- How many tests were passed successfully
- A score indicating the level of compliance

12.3. Single file report

The single file report contains all the information needed to understand the results of the validation process for a single file.

It includes a brief summary with the number of errors found, and a detailed report of the file structure and metadata and its compliance with the selected standard and the memory institution acceptance criteria, as well as the corrections applied by the metadata fixer.

12.3.1. Summary

The summary section displays the thumbnail and path of the image, to make it easier to identify the file the report belongs to.

It also includes the number of errors found during the validation process, and how many were automatically corrected by the metadata fixer.

The last element in the summary is a numeric score (from 0 to 100%) that shows how close the file is to being conformant.

12.3.2. Tags

The report includes a list with all the tags (and its values) found on the metadata of the TIFF file. This list of tags is aimed at experienced users. As most of the compliance errors will be based on the information contained in the metadata tags, this list will make it easier for expert users to understand the errors found in the document.

12.3.3. File structure

The TIFF file format does not have a flat structure. It can have containers of data that are linked between them.

This section of the report shows users what the structure of a given TIFF file is.

12.3.4. Standard/s compliant

This table shows the errors and warnings found by the implementation checker. An error is something that prevents the file from conforming to a selected standard (e.g. the image data in the TIFF file is compressed).

A warning usually refers to information that has not been found on the file, that is not mandatory by the standard, but that is highly recommended for preservation purposes. A file with no errors and one or more warnings is compliant with a standard, but some elements could be improved.

For each error and warning found, the report informs users of the cause of the error/warning, and also provides a list of possible fixes.

12.3.5. Acceptance criteria performance

Each memory institution can define their own acceptance criteria. This section reports on the results of checking a file against that criteria.

The report lists all the criteria, and for each one it reports on whether the file matches that criteria or not.

12.3.6. Automatic fixes

One of the components of the Conformance Checker is the metadata fixer, which can:

1. Apply automatic fixes, always preserving the Information Representation, in order to improve the sustainability of the file.
2. Add metadata to an existing file
3. Remove metadata from an existing file.

This section of the report shows the results of the metadata fixer. It displays the errors that were found, and the fixes applied. It also lists the tags added and removed from the file.

13. Additional considerations

The Functional Specifications document has described the need for a new TIFF standard, the functional and non-functional requirements and how the proposed solution will work.

One of the goals of the project is to open source the code of the project under a GPLv3+ and MPLv2+ license, to ensure that:

- End users have no restriction or legal risk when running the DPF Manager
- Developers can take the code and either add new functionality, integrate it with their systems, or create a new piece of work based on the existing functionalities.

In order to create and grow an open source community around the project, and to foster collaboration between all interested parties, we have created an “Open Source Best Practices and Dissemination Plan” document. The document details the infrastructure, people, processes and targets we will put in place to achieve this goal.

As the licensing decisions play an important role in any open source project, we have sought legal advice from id law partners, a firm specialised in Open Source licenses. We have included in the documentation the “Engagement letter” and the “Intellectual Property Rights report”. The report should be of interest to any party that may want to use and/or extend the DPF Manager.

Even though the DPF Manager is an open source project, its future depends on our capability to create a self-sustaining business around it. Every open source project needs resources to keep it alive and this one is no different. We have also included as part of the documentation the “Business Commercialisation Plan” that outlines the strategy Easy Innova will follow to create a set of revenue-earning services around the open source DPF Manager.