Planets Preservation Planning Tool: Plato 2.0

User Manual

VO.8

November 7, 2008

Table of Content

List of Figures

List of Tables

A. Introduction

This document describes how to use the Planets Preservation Planning Tool Plato. Plato implements the Planets Preservation Planning approach which is described in more detail in the next chapter.

So far, preservation planning was a time-devouring ad-hoc procedure in which most steps had to be carried out manually. Plato is a web-based tool that supports and automates the planning process underlying digital preservation endeavours. It integrates distributed services to provide a proactive planning platform for distributed preservation activities. Examples of services integrated in Plato are migration services delivered by CRiB (Conversion and Recommendation of digital Object Formats) and the preservation action services deployed within the EU plan Planets (Preservation and Long-Term Access via Networked Services).

A deployed version of Plato can be found at http://www.ifs.tuwien.ac.at/dp/plato .

Please note: Plato is still ongoing development and will continuously be improved and enhanced. In most cases an update of Plato also comes with a database update which might reset the database's content. However, Plato also supports import and export of all preservation plans developed in Plato. A Preservation plan can be exported as an XML file and downloaded as such.

Working on a preservation plan can seamlessly be continued after importing a plan.

B. The workflow

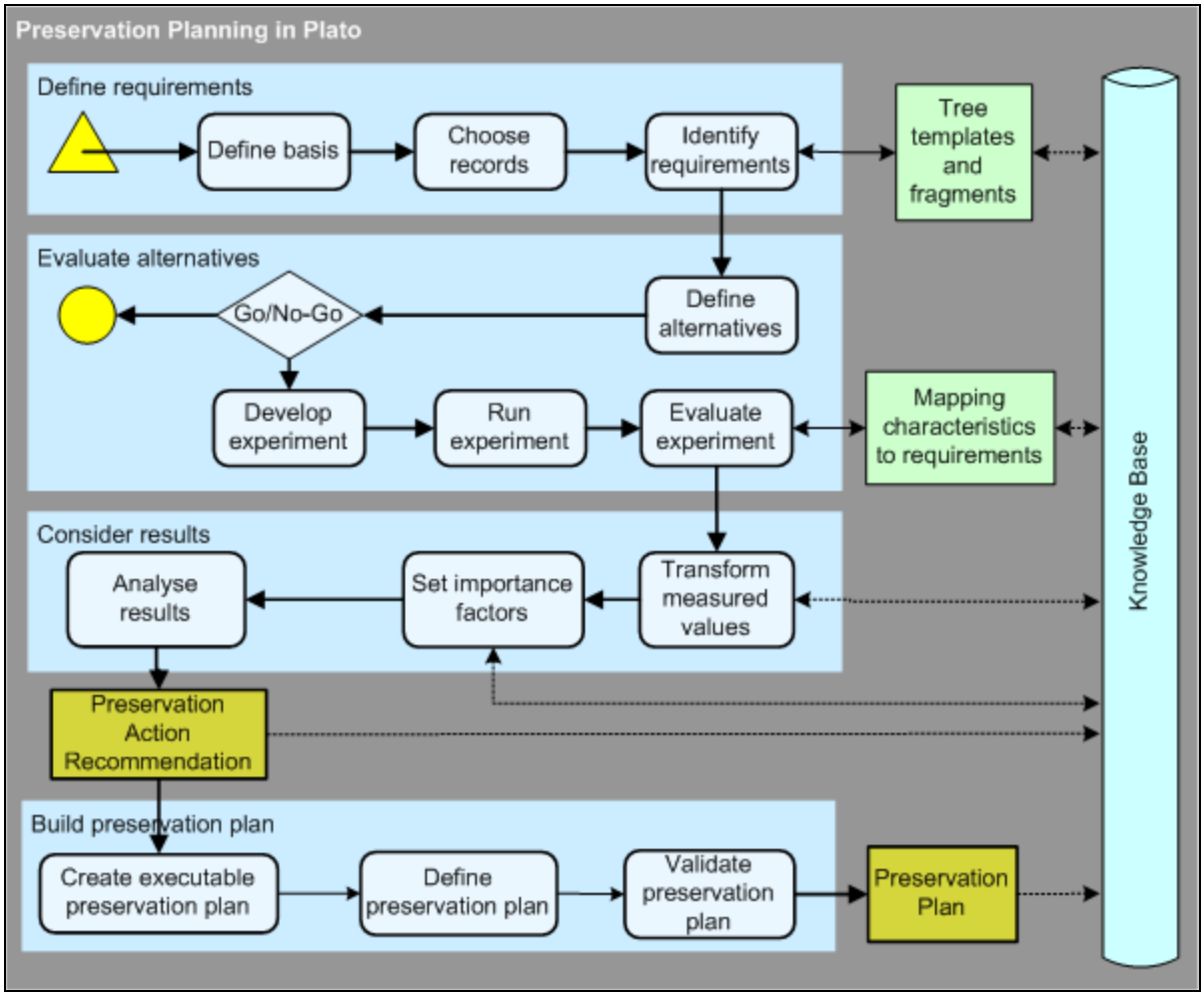
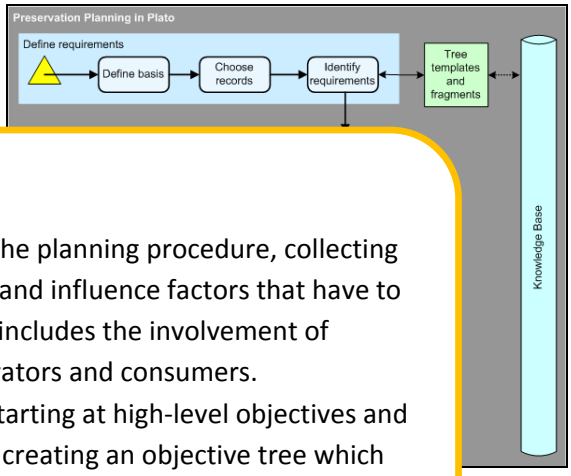


Figure 1 Overview of PLANETS Preservation Planning workflow

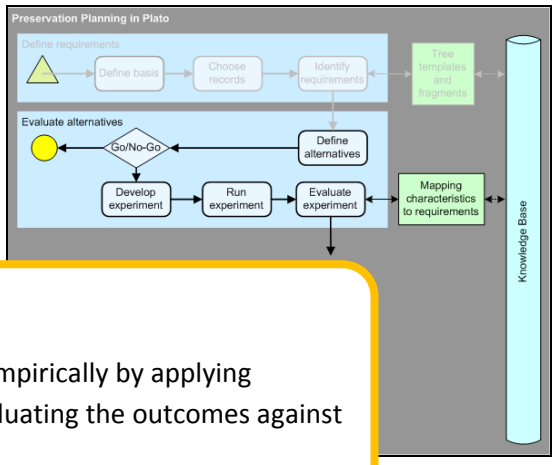
The Planets preservation planning workflow as described in Figure 1 consists of four main stages:

1. Define requirements
2. Evaluate alternatives
3. Consider results
4. Build preservation plan



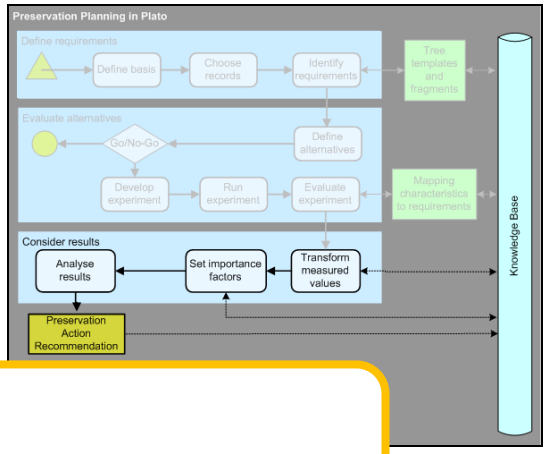
1. Define requirements

Requirements definition is the natural first step in the planning procedure, collecting requirements from the wide range of stakeholders and influence factors that have to be considered for a given institutional setting. This includes the involvement of curators and domain experts as well as IT administrators and consumers. Requirements are specified in a quantifiable way, starting at high-level objectives and breaking them down into measurable criteria, thus creating an objective tree which forms the basis of the evaluation of alternative strategies. Furthermore, as this evaluation would be infeasible on the potentially very large collection of objects, the planner selects representative sample objects that should cover the range of essential characteristics present in the collection at hand.



2. Evaluate alternatives

The evaluation of potential strategies is carried out empirically by applying selected tools to the defined sample content and evaluating the outcomes against the specified requirements.

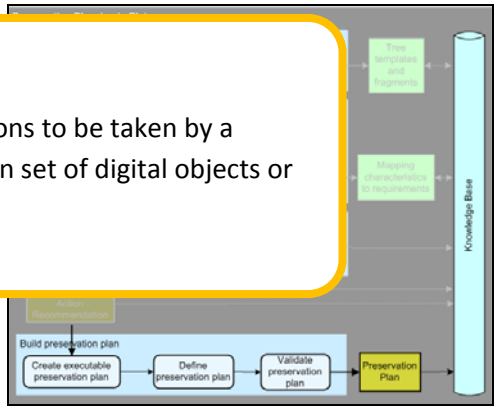


3. Consider results

Analysis of the results takes into account the different weighting of requirements and allows the planner to arrive at a well-informed recommendation for a solution to adopt.

4. Build preservation plan

A **preservation plan** defines a series of preservation actions to be taken by a responsible institution due to an identified risk for a given set of digital objects or records (called collection).



C. About Plato (Planning tool)

Welcome page

Four sections for introducing Plato

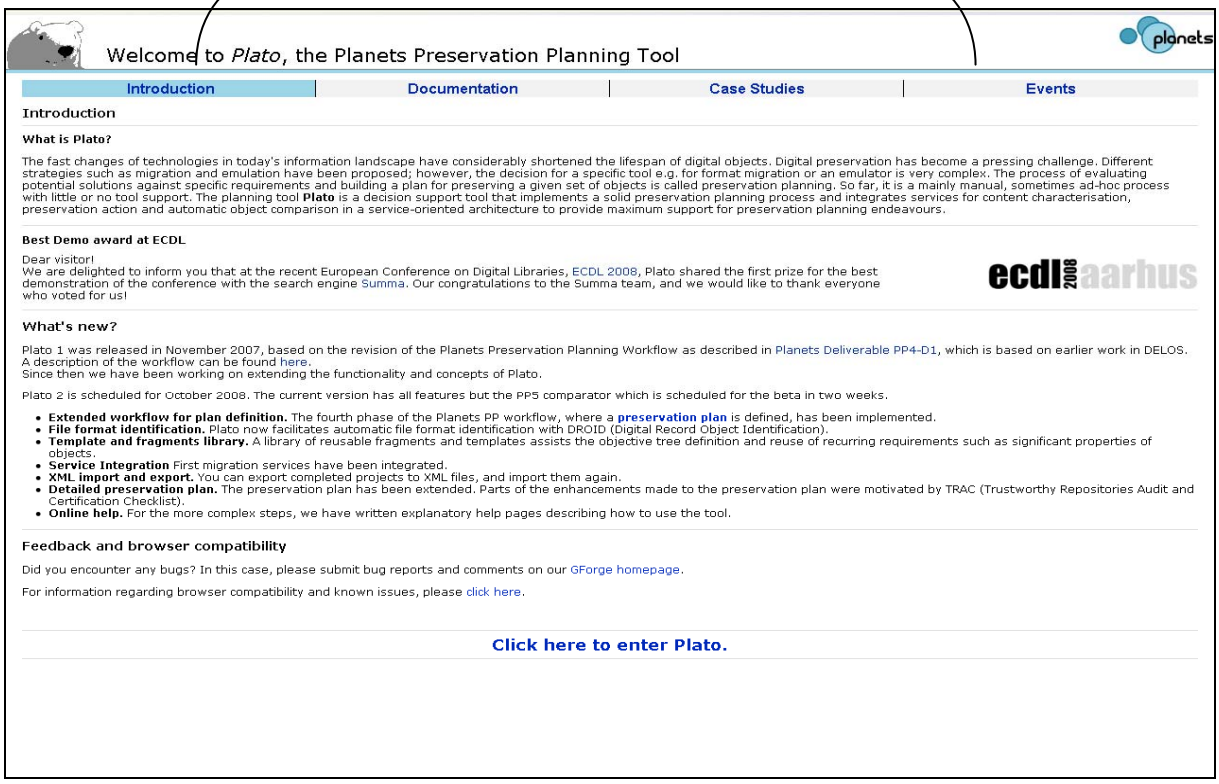


Figure 2 Welcome page

The first page (Figure 2) of Plato is a welcome page presenting detailed information about Plato. The information is subdivided into 4 sections:

1. Introduction	2. Documentation	3. Case Studies	4. Events
• What is Plato • News • What is new? • Feedback and browser compatibility	• Introductory material • Scientific Papers • Plan Deliverables	• Video Games • Interactive Multimedia Art • Electronic Theses and Dissertations • Bitstream Preservation of Digital Photographs	• Upcoming Events • Past Events

Table 1 Information which is on the welcome page

To enter Plato you have to click the link: 'Click here to enter Plato'

The entry to Plato

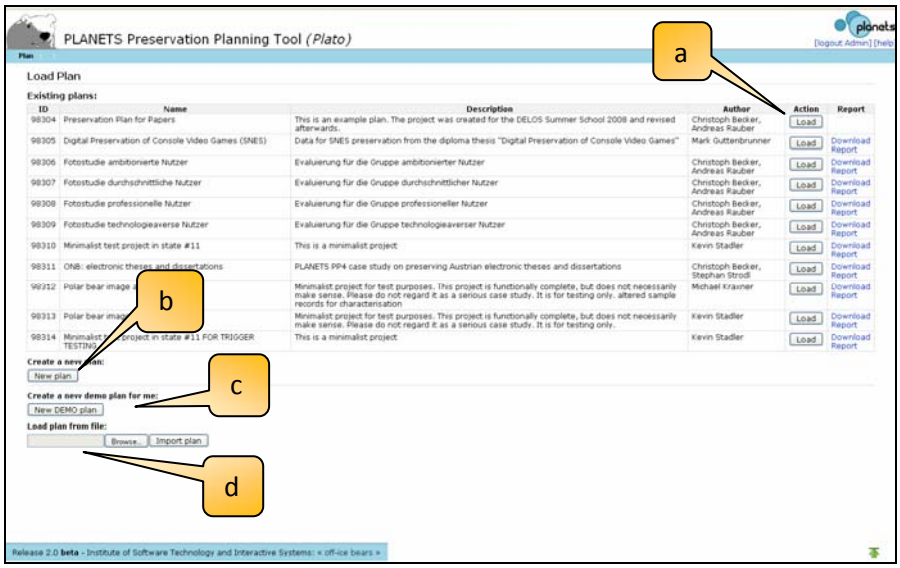


Figure 3 The entry to Plato

The first entry into Plato shown in at Figure 3 gives the opportunity:

- to load an existing plan [see a]
- to create a new plan [see b]
- Create a new DEMO plan [see c]
- Or to load plans from previously exported plans [see d]

Create a new Plan

A new Plan can be started from scratch by either choosing 'New Plan' from the menu or clicking the 'New Plan' button on the 'Load plan' screen.

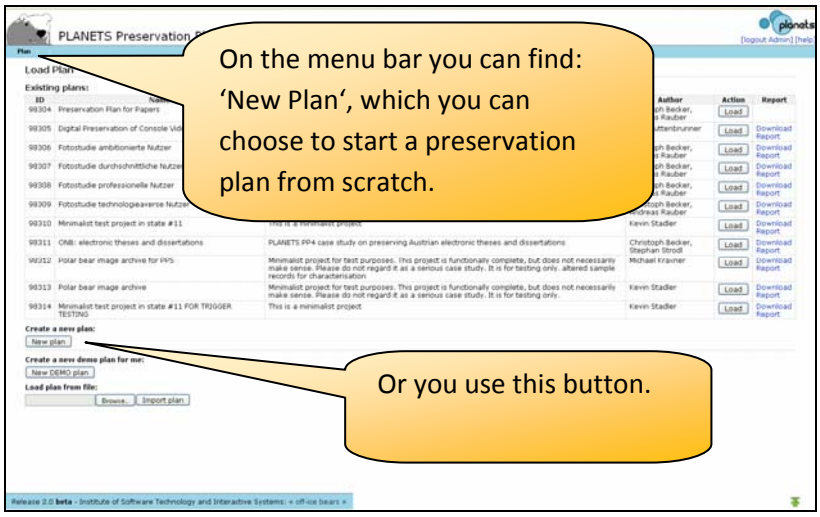


Figure 4 Possibilities to create a Plan

After you have clicked the button for a new plan a new site will open.

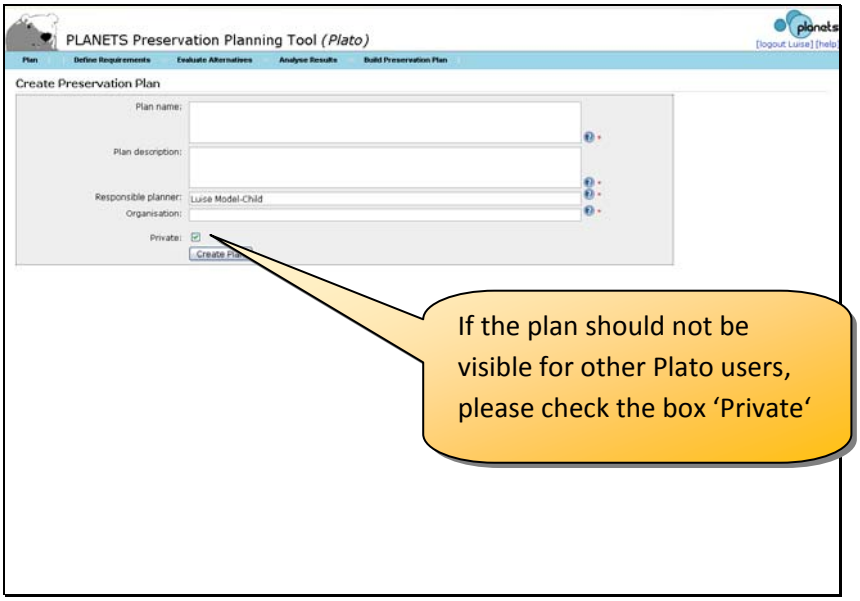


Figure 5 Create Preservation Plan

Name of the text fields	Examples which could be filled in the text fields
Plan name	• Preservation Plan for Papers • Digital Preservation of Console Video Games (SNES)
Plan description	• This is an example plan. The plan was created for the DELOS Summer School 2008 and revised afterwards. • Data for SNES preservation from the diploma thesis "Digital Preservation of Console Video

	Games"
Responsible planner	• Hans Hofman
Organisation	• National Library • Vienna University of Technology

Table 2 Example for the boxes in Create Preservation Plan

After filling out the boxes continue by clicking the button: Create Plan

Load Plan

After logging into Plato, a list of stored plans is provided including the ID of Plan's name, a description and the author(s) name(s) (see **Error! Reference source not found.**). The plan can be loaded with the 'Load' button on the right side. When loading an existing plan, the user will be forwarded to the workflow step corresponding to the plan's state (see Figure 1).

Navigation

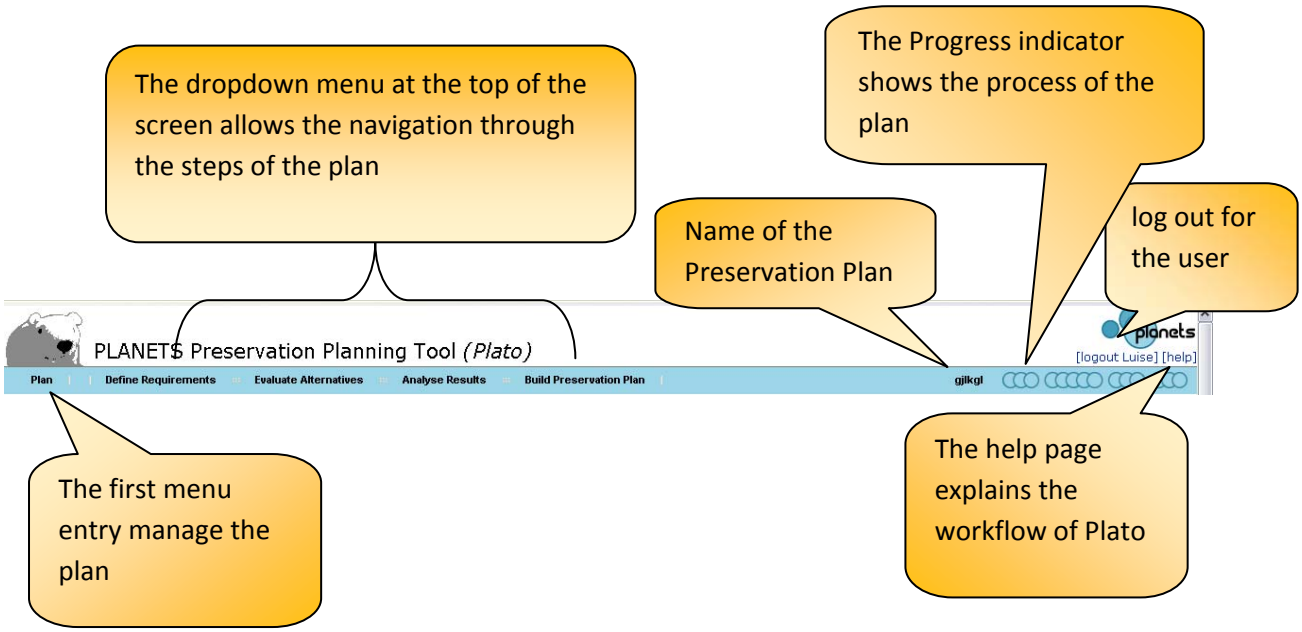


Figure 6 Plato navigation

The dropdown menu at the top of the screen allows in the first entry to manage the plan and in the following entries the navigation through the steps of the plan (see Figure 7).

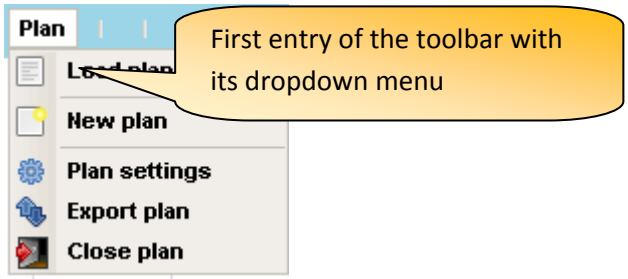


Figure 7 First entry of the toolbar

The first entry of the toolbar manages the plans, it includes:

Plan setting
It provides basic security options that you may use to protect, unprotect and delete plans that are created. There is also the option to upload the final report so that other users are able to see it at the Site: “Load page” (see Figure 3)
These options are only available to the user who created this plan.

Export Plan
(Only available if a plan is opened.
Exports the plan as a XML file.

Close Plan
(Only available if a plan is opened.)
Closes the current plan and redirects to the Create New Plan/Load Plan site

Figure 8 Progress indicator

On the right side of Plato navigation the name of the plan is shown (see Figure 6). A Progress indicator shows the process of the plan (see Figure 8Figure 8). The process indicator shows the plan’s state, how far the plan has progressed. It shows which steps in the workflow have already been completed (dark circles). The user can log out and a help button gives the option to use a help page which provides the workflow.

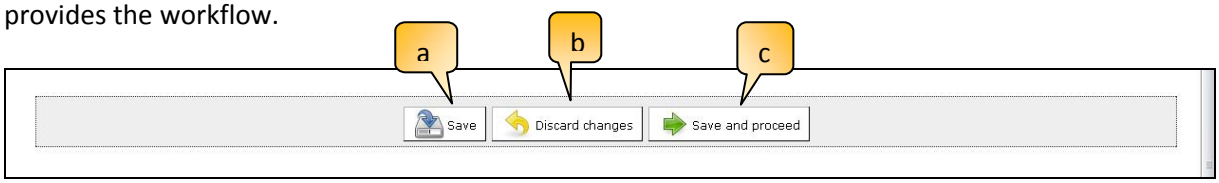


Figure 9 Buttons: Save, Discard changes, save and proceed

Each page of the workflow contains the buttons ‘Save’, ‘Discard changes’ and ‘Save and proceed’ (see Figure 9). The save button (see a) stores the current plan in the database. The ‘discard changes’ button (see c) restores the plan state of the last saving.

It is possible to easily navigate within workflow steps that are completed, i.e. have been passed by clicking ‘Save and proceed’. The plan’s state will be reset if you click either ‘Save’ or ‘Save and proceed’

The next step of the workflow can be reached through ‘Save and proceed’. Plato validates the input and proceeds to the next workflow step if valid. Any validation errors that occurred will be indicated and can be corrected.

D. Workflow steps of Plato

1. Define Requirements

The first phase of the workflow lays out the cornerstones of the planning endeavour. It starts with collecting and documenting the influence factors and constraints on possible actions and procedures,

then describes the set of objects under consideration and finally defines the complete set of requirements to be taken into account.

1.1 Define Basis

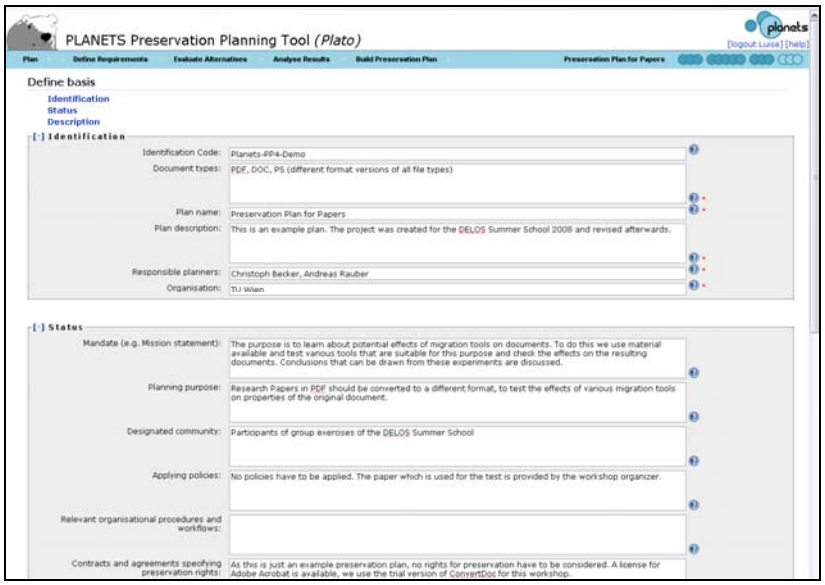


Figure 10 Define Basis

In this step, the preservation planner documents applying institutional policies, legal regulations, and usage criteria that might affect planning decisions for preservation. This may happen in an unstructured form, but preferably these factors are captured in a more formal way making it easier to derive decisions in the respective workflow steps. Examples include policies defining permitted file formats for ingest, and policies related to intellectual property right and legal access regulations. Further important policy elements pertain to characteristics of the preservation action, whether preservation actions that are open source shall be preferred or if just a specific preservation strategy may be applied, such as emulation. This might be possible in cases where the institution doesn't have the copyright and thus any modifications of the digital object are prohibited. Furthermore, the event that led to the planning procedure is documented.

1.1.1 Identification

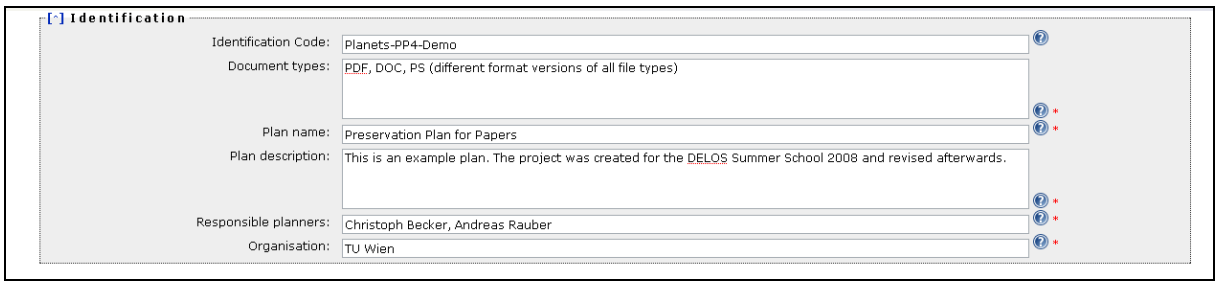


Figure 11 Identification

A preservation plan should be uniquely identified so that it can easily be referred to and retrieved.

Type	Examples
Identification code:	• Planets-PP4-Demo
Document types	• PDF, DOC, PS, (different format versions of all file types) • Digital Data from Cartridges of Super Nintendo Entertainment System (SNES) video games (Binary Streams)
Plan name	• Preservation Plan for Papers • Digital Preservation of Console Video Games (SNES)
Plan description	• This is an example plan. The plan was created for the DELOS Summer School 2008 and revised afterwards. • Data for SNES preservation from the diploma thesis "Digital Preservation of Console Video Games"
Responsible planners	• Christoph Becker • Mark Guttenbrunner
Organisation	• Vienna University of Technology

Table 3 Examples for boxes for Identifikation

1.1.2 Status

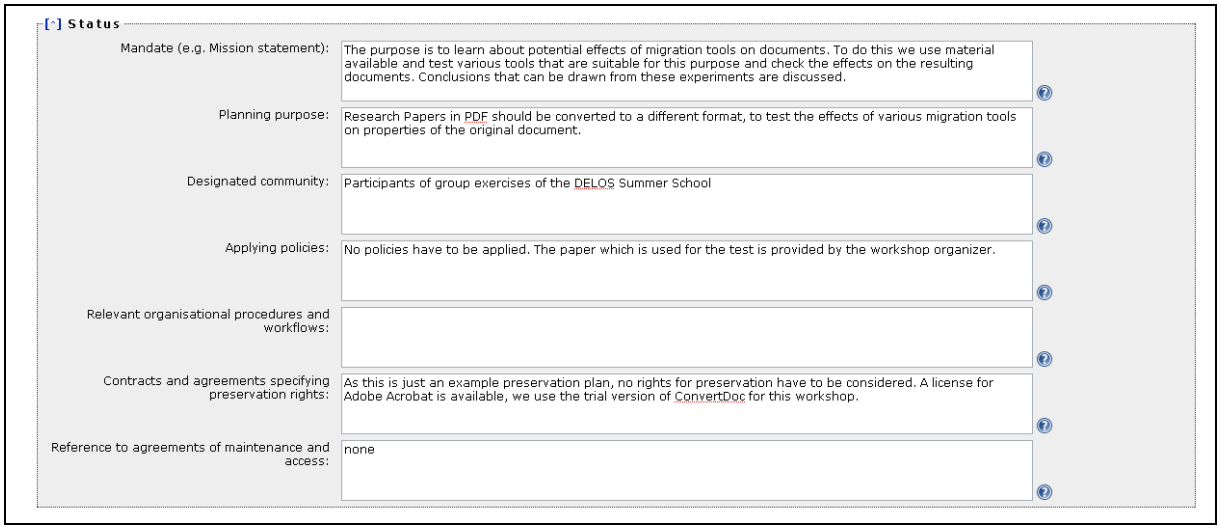


Figure 12 Status

The status of a plan includes both the planning progress – whether a plan is currently being defined, awaiting approval, or already has been deployed and is active – and the triggers which have led to the definition or refinement of the plan.

Type	Examples
Mandate (e.g. Mission statement)	• The purpose is to learn about potential effects of migration tools on documents. To do this we use material available and test various tools that are suitable for this purpose and check the effects on the resulting documents. Conclusions that can be drawn from these experiments are discussed.
Planning purpose	• Research Papers in PDF should be converted to a different format, to test the effects of various migration tools on properties of the original document. • The library has the legal obligation to preserve every published console video game like national libraries are obliged to preserve publications on paper and offer possibilities to display these games to the public.
Designated community	• Participants of group exercises of the DELOS Summer School • The target audience is visitors of the library. It is not necessary to publish the collection online. Access to games from the library collection to experience the games original look & feel should be possible for the public. Access to original media shall not be necessary to avoid damage to rare specimen.
Applying policies	• No policies have to be applied. The paper which is used for the test is provided by the workshop organizer. • For legal reasons only games physically in the possession of the library are preserved. This is because of copyright restrictions. The archive doesn't hold the copyright for the rest of the collection.
Relevant organisational procedures and workflows	• Team / the committee concerned • Plan has to be presented and approved by the vice board
Contracts and agreements specifying preservation rights	• No rights for preservation have to be considered. Licenses for both Adobe Acrobat and ConvertDoc are available. The institution has the necessary right to change the object, which might be necessary for migration.
Designated community	• Participants of group exercises of the DELOS Summer School • The target audience is visitors of the library. It is not necessary to publish the collection online. Access to games from the library collection to experience the games original look & feel should be possible for the public. Access to original media shall not be necessary to avoid damage to

	rare specimen.

Table 4 Examples for the boxes in Status

1.1.3 Description

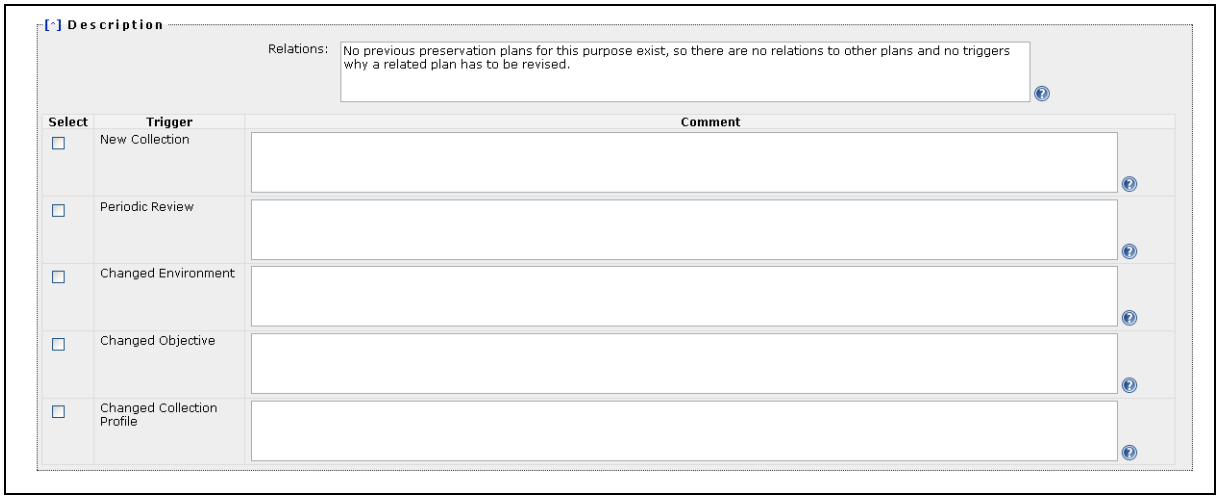


Figure 13 Description

New Collection

This is the most common event, where a preservation plan is created from scratch for a new collection, for which no plan was previously defined.

Changed Collection Profile

Changes in the collection profile of an existing collection may require a revision of existing preservation plan. Examples for changes in the collection profile are newly accepted object formats or significant changes in collection size. It is the responsibility of technology watch functions to ensure that these triggers are actually fired; the corresponding events should then be recorded in the planning documentation.

Changed Environment

The environment of a preservation plan consists of the technical environment, the designated communities and the host institution. Changes in the environment can lead to a change in preferences, for example with respect to the system context in which a preservation action needs to operate. They might also imply a change in factors which influence existing preservation plans, for example changed prices for hardware or software. Other changes are the availability of new preservation strategies or obsolescence of object formats which are used in an existing preservation plan. Changes in the environment require a revision of existing preservation plans, while the objectives for the evaluation usually will remain unchanged.

Changed Objective

Changes and developments in the environment can change the objectives for preservation evaluation over time. In this case it is necessary to evaluate existing preservation plans against changed objectives. Examples for these changes are technology developments or changes in high-

level policies or legal obligations that have an impact on preferences and objectives. Changes in the designated community may also effect the goals and objectives.

Periodic Review
Periodical reviews of existing preservation plans are needed to continually verify the appropriateness of plans, and to improve and further develop existing plans. A periodic review should re-iterate the planning activity taking into account new developed preservation strategies, and seek to verify and potentially improve existing plans.

Type	Examples
Relations	• No previous preservation plans for this purpose exist, so there are no relations to other plans and no triggers why a related plan has to be revised. • No previous attempts to preserve these kinds of documents have been made.
New Collection	
Periodic Review	
Changed Objective	
Charged Collection Profile	

Table 5 Examples for the boxes for Description

1.1.4 Policies

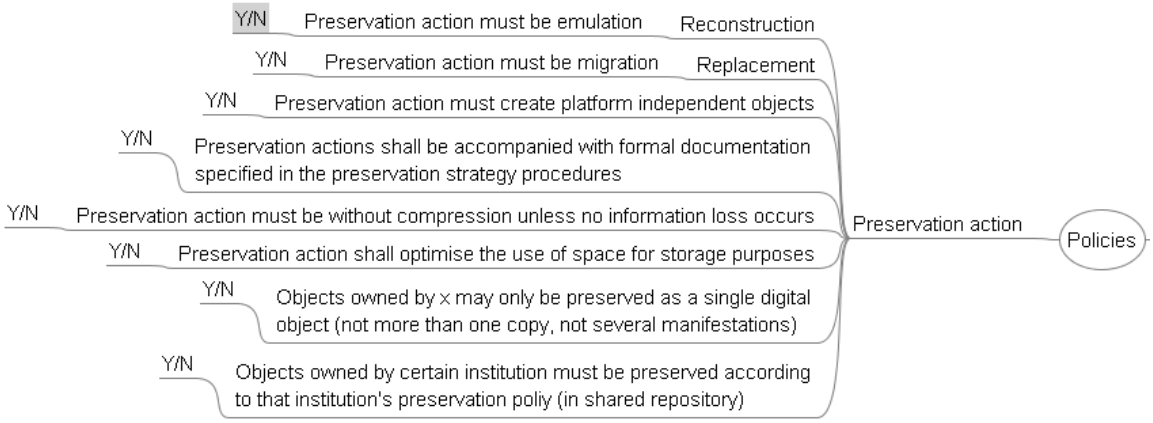
Why policies?
Organizations differ in many ways, type, size, direction, just to to name a few. That's why apart from technical and intellectual properties of digital objects also the strategy, policy, goals and constraints of the institution are an integral part of the preservation plan.

Usually organizations have created documents describing their policies, strategies, workflows, plans, and goals to provide guidance.

The policies template
In the course of the Planets project a conceptual model of organizational digital preservation policies and strategies has been created. It incorporates relevant organisational characteristics and strategic directions to support the planning process in digital preservation projects.
Relevant policies have been structured in tree form which can be used as a template for defining your organizational policies.

How to create a policy tree
Based on the policy tree you can model you policies by replacing the leaves, which denote the scale of the policy, by their actual values. You should remove policies you don't have in place by simply removing the node forming the policy.

In the tree, policies have been grouped into several levels, as for instance shown in the screenshot below. The screenshot shows policies pertaining to *Preservation Action*. Leaves in the tree denote scales the particular policy can be measured in. Policy *Preservation Action must be emulation* can be answered either *Yes*, or *No*.



After all scales have been replaced by their actual values, the policy tree might look like follows:

Upload the policy XML file

The workflow step 'Define Basis' allows to upload the policy tree which has been saved as Freemind file (.mm).

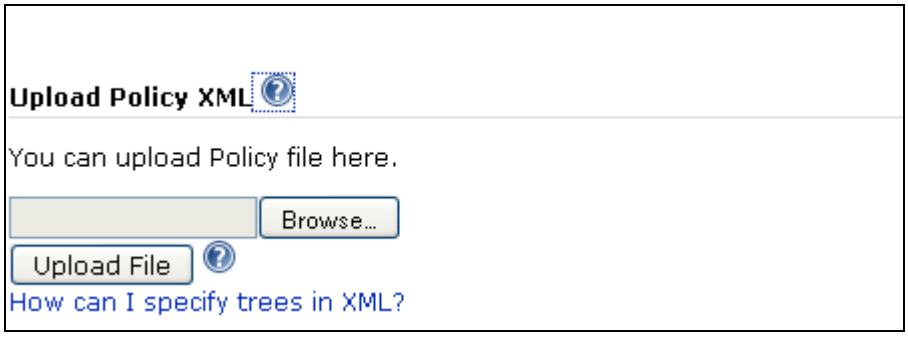


Figure 14 Upload Policy XML

After the tree has been uploaded (by pressing the button 'Upload File') Plato displays the tree and allows you to store the policies with the preservation project.

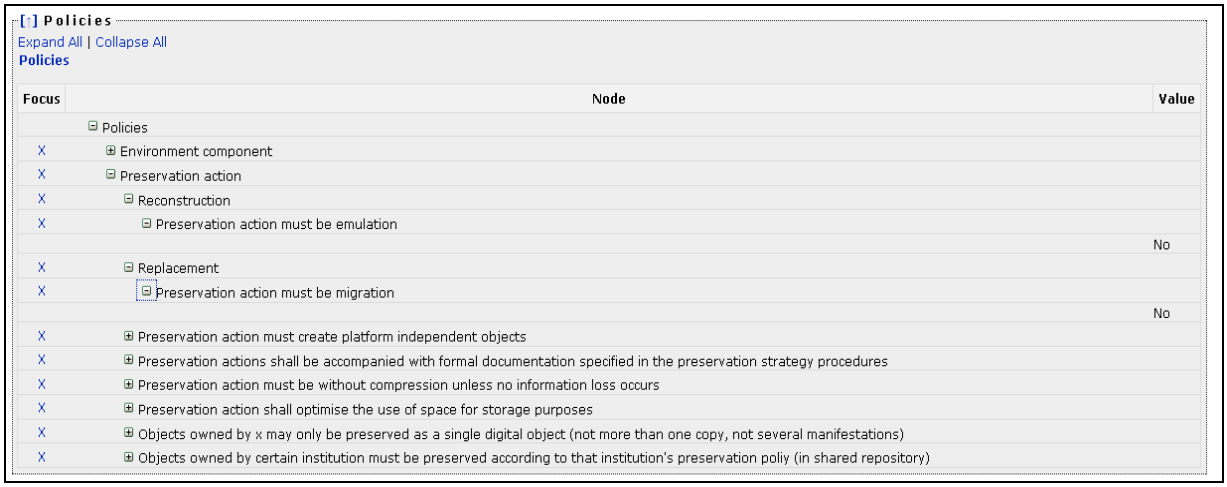


Figure 15 Policies

1.2 Define Sample Records

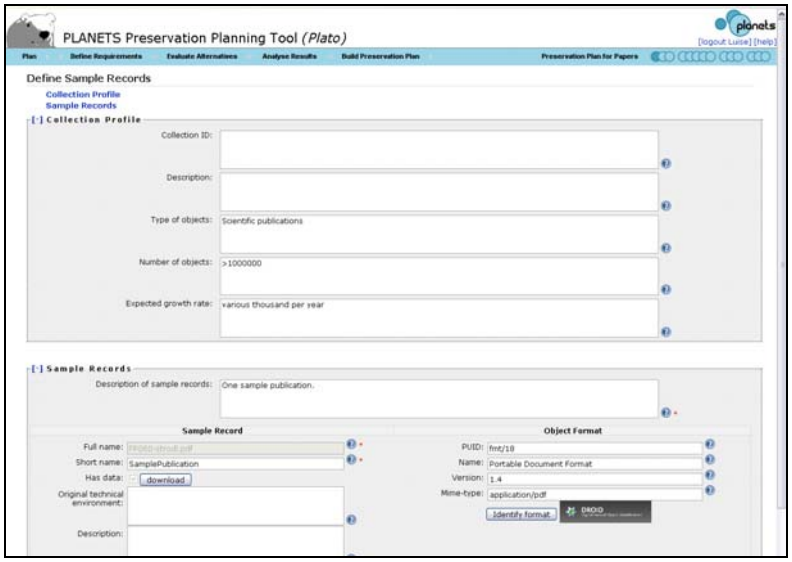


Figure 16 Define Sample Records

The second step describes the set of objects that forms the scope of the current plan, and selects a subset of representative objects for experimentation. A general description of the characteristics of the set of objects, called *collection*, includes basic properties such as the size of the collection, the class of objects and the object formats they are currently represented in. While this can be done in a manual descriptive way, a formal representation is desirable.

1.2.1 Collection Profile

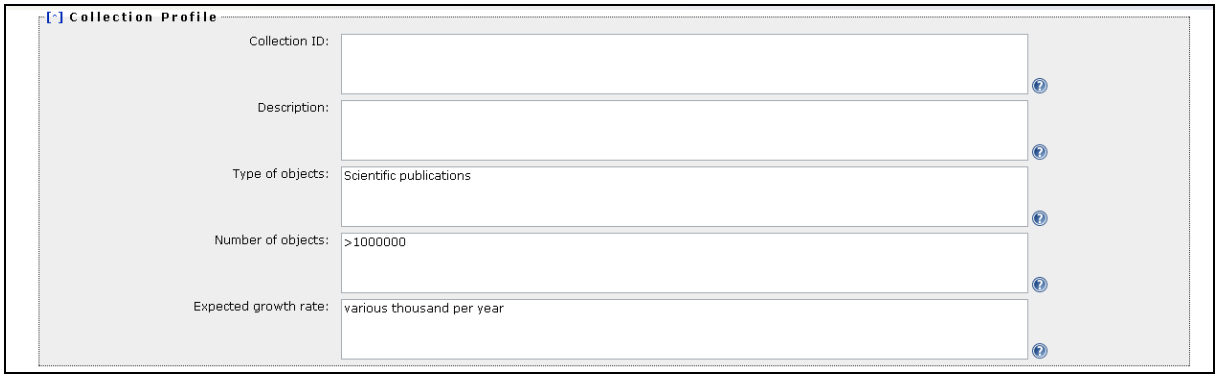


Figure 17 Collection Profile

Collection profiling tools can provide automated descriptions of the technical characteristics of objects. Characteristics of interest include object formats, file sizes and their variation within the collection, but also aspects such as an assessment of the risks of each object type and each object, thus leading to a risk profile of the collection. As a complete evaluation of the quality of preservation action tools is infeasible on the potentially very large collection of objects, the planner selects representative sample objects that should cover the range of essential characteristics present in the collection at hand. To reduce effort to a minimum, this subset should be as small as possible.

Type	Examples
Collection ID	• COLLECTION-12-2008
Description	• Physical collection of Cartridges for the Nintendo SNES which have been transferred to binary streams.
Type of objects	• Scientific publications • Digital Data from Cartridges of Super Nintendo Entertainment System (SNES) video games (Binary Streams).
Number of objects	• >1000000 • 1000
Expected growth rate	• Slow. Occasionally new games can be added to the library's collection.

Table 6 Examples for the boxes in Collection Profile

1.2.2 Sample Records

Figure 18 Sample Records

For these samples, an in-depth characterisation is performed, describing the significant properties and their technical characteristics such as their name and provenance, the file format, and specific risk factors. The uploaded sample record can be identified by clicking 'Identify format' on the page. For identification Plato uses the software tool DROID which it calls and displays the output: PUID (Pronom Unique Identifier), Name, Version and Mime-type.

Type	Examples
Description of sample records	• One sample publication
Sample Record: Full name	• Very large publication containing > 20 figures.
Sample Record: Short name	• Very large publication
Sample Record: Original technical environment	
Sample Record: Description	• This is our larges publication file we have in our collection. It contains a lot of figures, tables and equations.
Object Format: PUID	• Fmt/18
Object Format: Name	• Portable Document Format
Object Format: Version	• 1.4
Object Format: Mime-type	• application

Table 7 Examples for the boxes in Sample Records

1.3 Identify Requirements

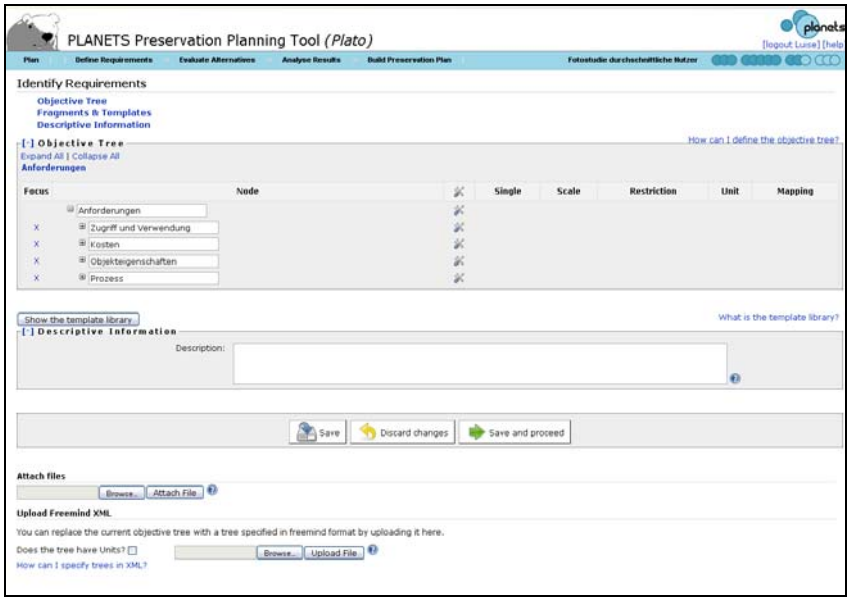


Figure 19 Identify Requirements

Requirements definition is the heart of preservation planning. It is the basis for the decisions to be taken and documents the priorities and preferences of the institution. Requirements are collected from the wide range of stakeholders and influence factors that have to be considered for a given institutional setting. This may include the involvement of curators and domain experts as well as IT administrators and consumers.

1.3.1 Objective Tree

The requirements are specified in a quantifiable way, starting at high-level objectives and breaking them down into measurable criteria, thus creating an *objective tree* which forms the basis of the evaluation of alternative strategies. This step has proven to be the most critical and complicated stage in the planning procedure. An incomplete requirement specification leads to a skewed evaluation and potentially wrong decisions. On the other hand, curators tend to exhibit a reluctancy to quantify their preferences, and especially try to avoid questions such as *What is the loss I am willing to accept? which are of central importance. The complexity involved in specifying goals and breaking them down to concrete, quantifiable criteria is a considerable challenge. However, through tterative refinement of abstract goals such as I want to preserve these objects exactly as they are towards more concrete requirements (The size needs to be unchanged) we ultimately arrive at mea surable criteria such as The image width, measured in pixel, needs to be unchanged. As requirements need to be negotiated between the stakeholders, a common approach is to define the requirements in a workshop setting where as many stakeholders as feasible are involved. On a practical level, two tools have been very useful: post-it notes and mind-mapping software. While post-it® notes and flip charts as classical tool to support brainstorming activities have the benefits of allowing everyone to act at the same time, mind maps provide the better overview of the current state of requirements for all participants and allow a moderator to channel the discussion process. Often, a combination of both tools is the most productive approach. While the resulting objective trees usually differ through changing preservation settings, some general principles can be observed. At the top level, the*

objectives can usually be organised into four main categories – characteristics of the objects, the records, and the process, and requirements on costs.

Object characteristics describe the visual and contextual experience a user has by dealing with a digital object. These characteristics are often referred to as significant properties. A common way of describing them is to consider the five aspects “Content”, “Context”, “Structure”, “Appearance”, and “Behaviour”.

The tree contains the requirements for preserving a collection of static web pages containing documents and images. The branch Behaviour is divided into three different groups of criteria: deactivate, preserve, and freeze. This reflects the preferences of the archive that some functionality, such as menu navigation, is needed for properly accessing the web pages, while most active content shall be disabled or frozen. For example, visitor counters shall be preserved in the state they had at the moment of ingest.

Recently, several plans such as INSPECT have presented detailed analysis of the significant properties of different categories of objects, including vector images, moving images, e-Learning objects, and software. These can provide a very valuable input to this aspect of requirements specification. On the other hand, the automated characterisation of the sample objects defined in the previous step further supports the analysis of their significant technical properties.

- Record characteristics describe the foundations of a digital record, the context, interrelationships and metadata.

- Process characteristics describe the preservation process itself, for example the procedure of migrating objects. These characteristics include the complexity of applying preservation action tools or their performance and usability, but equally should cover aspects such as documentation or the degree of validation. The definition of process characteristics is particularly dependant on the specific context the preservation process is taking place. The technical environment may effectuate specific requirements on the interoperability of tools, while institutional policies or legal regulations may enforce specific licensing requirements or require a particular degree of automated documentation. Thus the institutional and technical context and constraints posed by it have to be considered carefully.

- Costs have a significant influence on the choice of a preservation solution, but are inherently hard to quantify. Ultimately the Total Cost of Ownership (TCO) is the guiding figure for deciding whether or not a preservation strategy meets the needs of an institution within the constraints of its budget. Instead of providing a single numeric criterion which is extremely complex to quantify, costs might also be defined as infrastructure characteristics, putting an emphasis on cost factors instead of the resulting figures for cost estimates. These cost factors can then be further broken down to cover hardware, software, and staff costs.

The objective tree thus documents the individual preservation requirements of an institution for a given partially homogeneous collection of objects. The tree as such is entirely independent of the strategy employed, be it migration or emulation. It is of vital importance that it is concerned solely with the *problem space* and does not specify solutions such as *We want to migrate to PDF/A*, unless

these decisions have been made already on a higher level. While such specifications are sometimes brought forward in the requirements workshops, they commonly can be traced back to the reasons underlying them, such as preferences for transforming objects to standardised, widely supported file formats and deactivation of active content. The decision to migrate to PDF/A using a specific tool right be the right one; however, without proper documentation of the reasons and the evaluation leading to it the recommendation cannot be considered trustworthy. The tree shown in Figure 2 contains a branch named *technical characteristics*. In this specific case, the institutional policy constrained the class of preservation action to be considered to migration; emulation was not an option. Thus the requirements describe in a very specific form the desired characteristics of the target format the objects should be kept in. These characteristics together form a *risk assessment* of the format and become a central part of evaluating applicable tools and strategies.

An essential step of requirements definition is the assignment of measurable effects to the criteria at the leaf level of the objective tree. Wherever possible, these effects should be objectively measurable (e.g. e per year, frames per second, bits per sample). However, in some cases, (semi-) subjective scales need to be employed. For example, the quality of documentation that is available for a file format or a tool should not be judged by the number of pages alone; instead, a subjective scale such as *excellent, good, average, poor, very poor* could be used. Similarly, the *openness* of documentation of a file format could be one of *fully standardised*; *openly published*, but not standardised by a recognized body; and *proprietary*.

The assignment of measurable effects to criteria can also align them with characteristics that can be automatically extracted from objects to automate the evaluation procedure. Existing software tools such as JHove allow automated extraction of some of the basic properties of common object formats; the eXtensible Characterisation Languages provide an in-depth description of the complete informational content of an object in an abstract representation. These descriptions can be used to derive properties to be measured, and support the automated comparison of these properties when migrating the objects to different formats.

Ongoing case studies revise and extend the previously conducted evaluation studies, build concrete preservation plans for specific collections of objects, and cover new types of objects that have not been evaluated yet in a variety of institutional settings.

The experience which is accumulated through carrying out planning activities and requirements definition can be shared between institutions easily through the supporting software, which contains a knowledge base of recurring fragments of objective trees and templates that can be used as a starting point.

The outcome of the first phase is a complete documentation of the planning context, the collection of objects at question, and the specific requirements that form the basis for the evaluation of alternative action paths.

1.3.2 Fragments & Templates

Figure 20 Button Show the Fragments & Templates

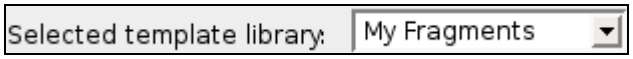
If the template library should be shown, the button 'Show the template library' must be pressed.

Templates and fragments
To allow you the (partial) reuse of an objective tree which you have already defined within a former project, or simply to assist you in creating a nicely structured objective tree from scratch, Plato provides the template-tree facility.

At the moment, there are four template libraries available: The "Public" libraries are visible to all Plato users. Any user can edit the content of the library and share it with other users. The "My"-libraries are only visible to one user each and can be used for re-using (parts of) objective-trees between the user's projects without publishing them to others. (Later versions of Plato will also contain template libraries for sharing within an institution.)

- **Templates** should be used to store complete objective trees, be it from former case studies or just rough tree-layouts which you might want to re-use for other projects. When selecting a template, your project's current objective tree will be replaced by the template, which you can then customize to your needs.
- **Fragments** on the other hand can be used to store and organize snapshots of parts of an objective tree for partial reuse. They can be inserted at any spot in your project's objective tree.

You can switch between the different template libraries by using the dropdown-box:



(Depending on the number of nodes contained in the template-tree displaying the newly selected library might take up to a second when switching between libraries. Below the dropdown-box a spinning wheel will be visible as long as loading is in progress.)

Using a template
If you want to start building your objective tree by using one of the templates provided in the template library, simply click the Load-Button next to the template of your choice.

Figure 21 Template Tree

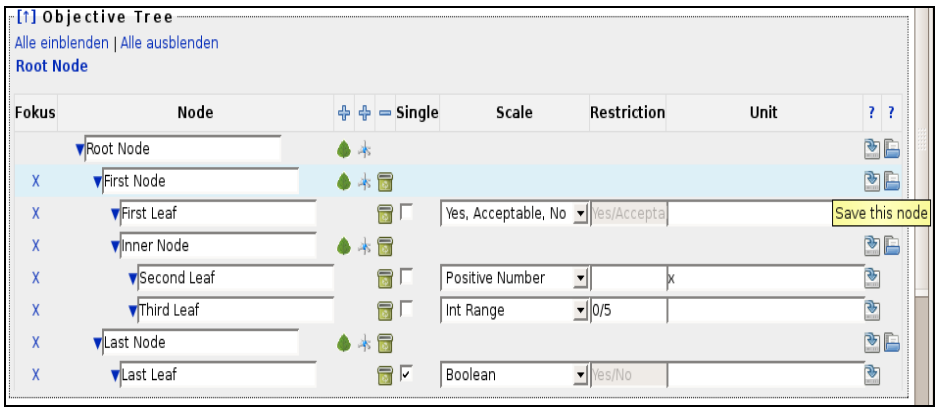
After confirming that you want to discard your current objective tree it will be replaced by a copy of the selected template and you can start customizing it to your needs.

Saving a template

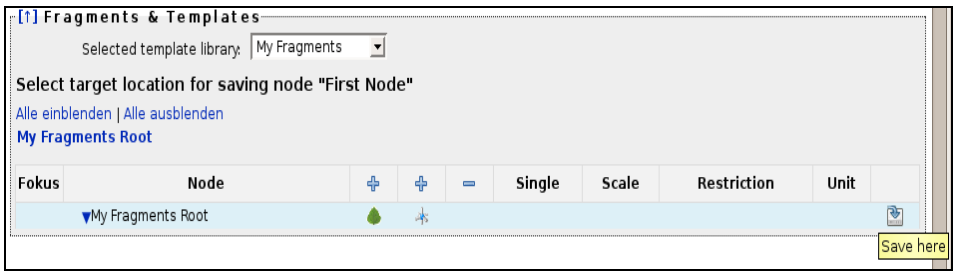
Saving your objective tree to the library as a template is not possible yet.

Saving a fragment

If you want to save a node to the template library, first make sure that your currently selected library is in fact a fragments and not a template library. If so, click the Save-Icon (), which can be found at the right end of your objective tree:



The template-tree will now display the same icon next to each of its nodes:

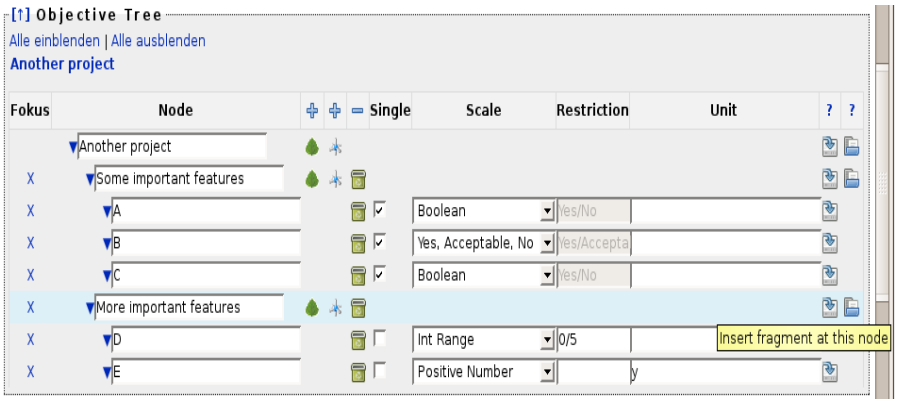


Use the dropdown-box to select the appropriate fragments library and click the save-icon next to the node where you want to save the previously selected node from your objective tree:

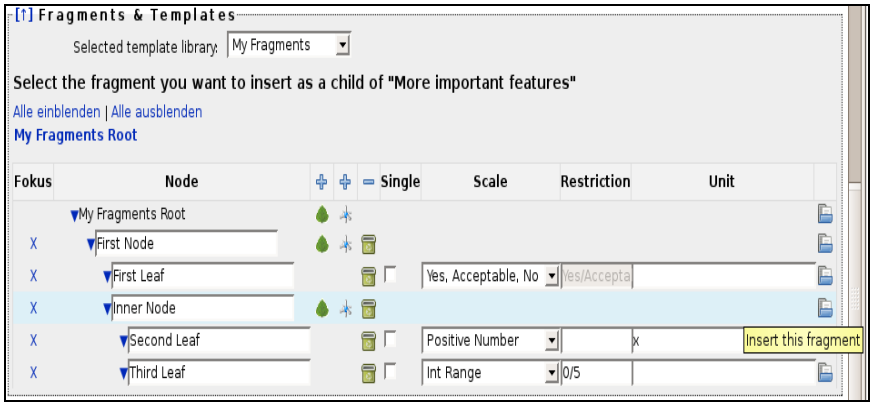
A copy of the previously selected node, including its child-nodes and leaves as well as their scales and restrictions, is now stored in the library.

Inserting fragments from the fragments library into your objective tree

If you want to insert a fragment from the template library into your objective tree, click the Insert-Icon () next to the node where you want to insert the fragment. (Insertion operations are only possible at inner nodes, not at leaves.)



The fragments library will now display the same icon next to each of its nodes:



Use the dropdown-box to select the appropriate fragments library and click the insert-icon next to the node which you want to copy into your objective tree:

The selected node from the library, including its child-nodes and leaves as well as their scales and restrictions, was copied into your objective tree.

1.3.3 Descriptive Information

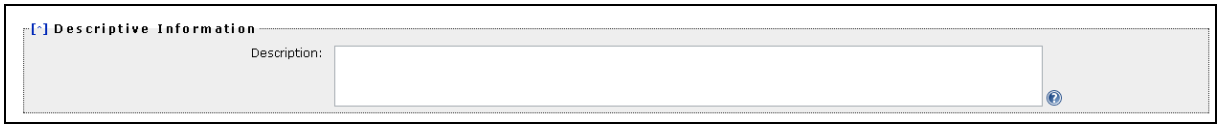


Figure 22 Descriptive Information

Type	Examples
Description	• Although we convert to PDF/A, which is a standard and has several, tools that check for PDF/A compatibility, we have to investigate requirements such as 'Footnotes' and 'Number of pages'. This is because the tools (alternatives) can be buggy.

Table 8 Examples for the boxes in Descriptive Information

2. Evaluate Alternatives

The second phase of the planning workflow evaluates potential actions in a quantitative way by applying them to the previously defined sample content and analysing the outcomes with respect to the requirements specified in the objective tree. This empirical evaluation procedure results in an evidence base that underlies the decisions to be taken in the successive phases.

2.1 Define Alternatives

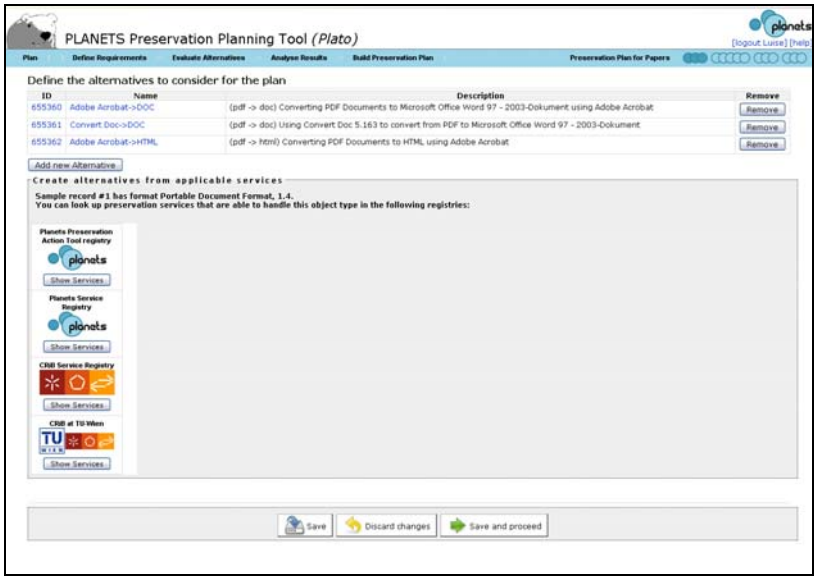


Figure 23 Define the alternatives to consider for the plan

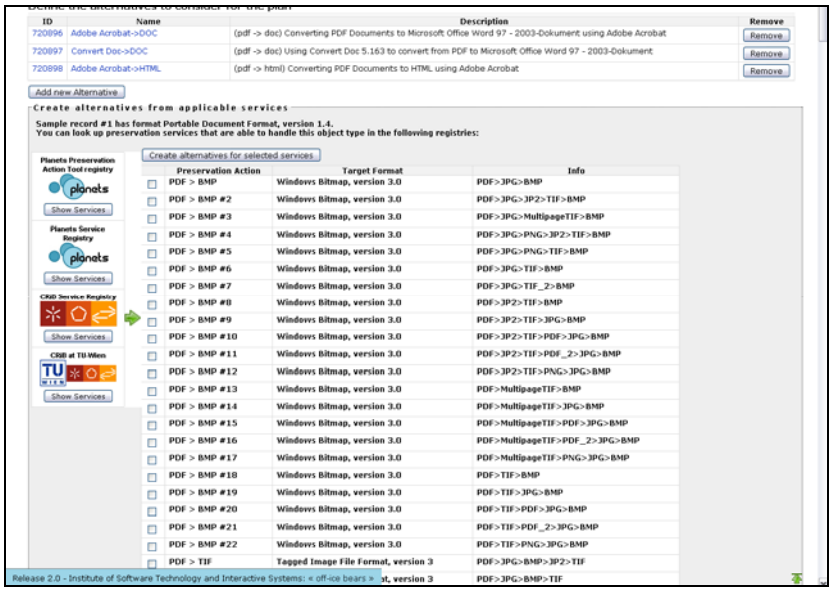


Figure 24 CRIB Service Registry: list of potential alternatives

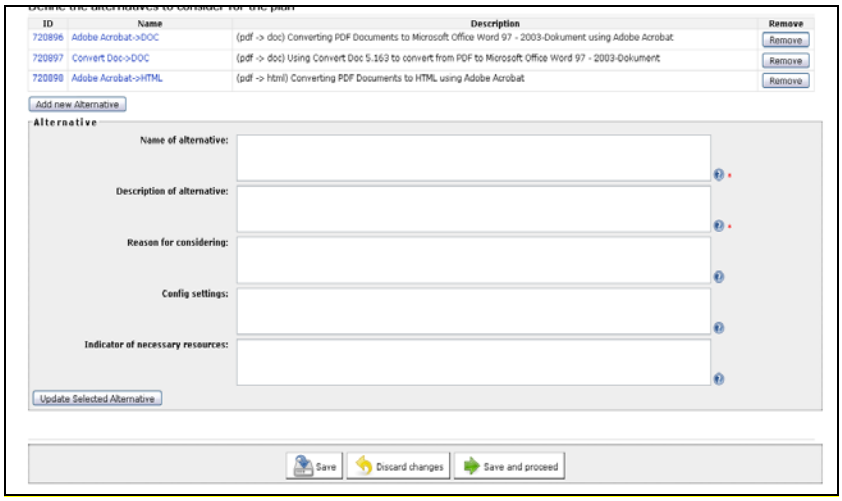


Figure 25 Add new Alternative

The natural first step of evaluation is to define the possible courses of actions to be taken into consideration. A variety of different strategies might be applicable; for each alternative action, a complete specification of the entailed steps and the configuration of the software tool employed is desired. The discovery of potential actions that are applicable varies in complexity according to the type of content. Often, this implies an extensive search phase, investigating which tools are applicable to the type of objects at hand. Registries holding applicable preservation action tools can be consulted for reference and are potentially very beneficial to support the search. The outcome is a *shortlist* of potential candidates for performing preservation actions, which will be evaluated empirically during the next steps.

2.2 Go/No-Go

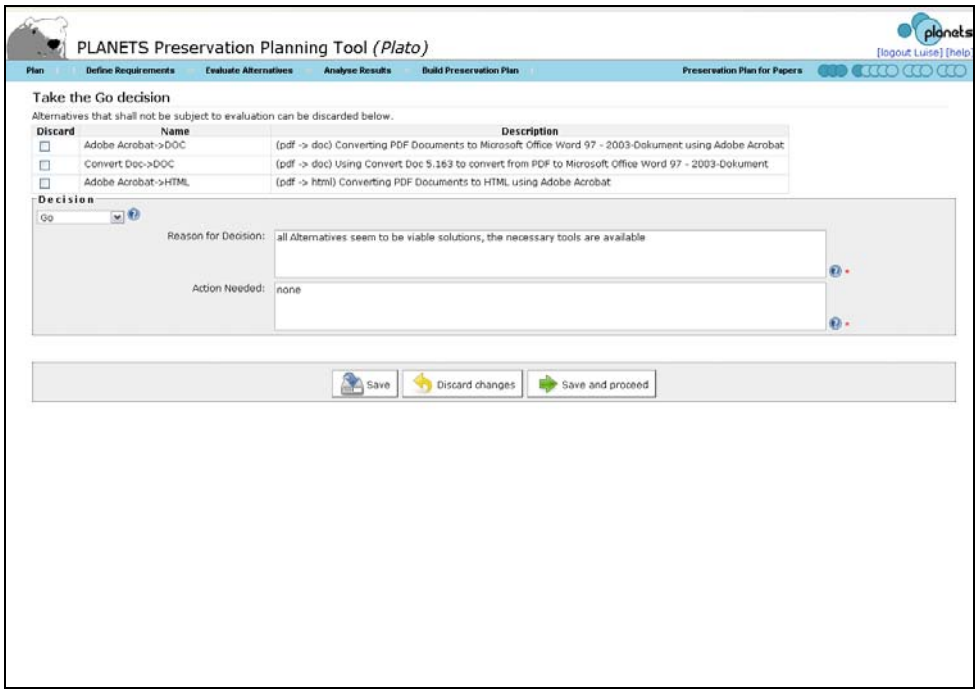


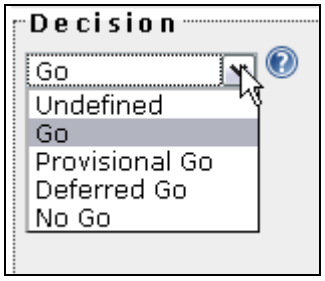
Figure 26 Take the Go decision

Before continuing with the experimentation procedure, this step reconsiders the situation at hand and evaluates whether it is feasible and cost-effective to continue the planning procedure. In cases where the evaluation is considered infeasible or too expensive, a reduction of candidate tools might be necessary. In order to proceed to the next workflow step, the user has to take the GO decision.

Type	Examples
Reason for Decision	• All Alternatives seem to be viable solutions, the necessary tolls are available
Action Needed	• none

Table 9 Examples for the boxes in Take the Go decision

2.2.1 Take the go decision



2.3 Develop Experiments

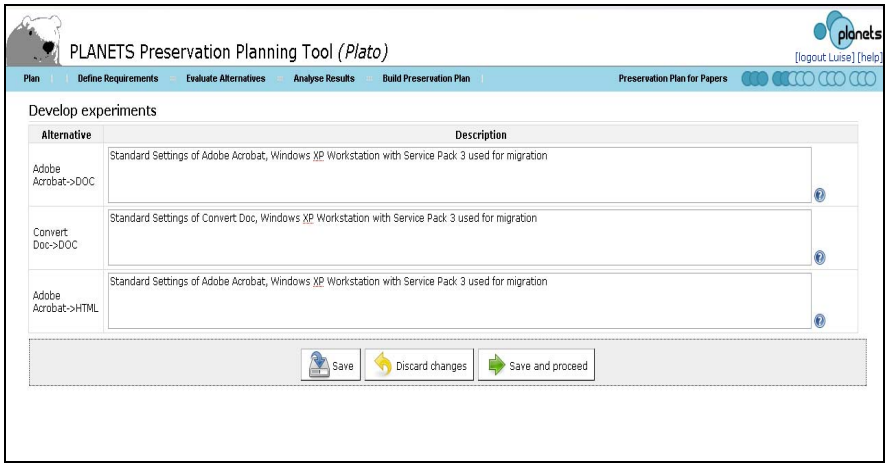


Figure 27 Develop Experiments

This step defines and documents the configuration of the tools on which experiments are carried out, and thus builds the basis for experiment execution in the next step. This includes setup procedures, a documentation of the hard- and software environment, and additional steps needed to carry through the evaluation of experiments.

Type: Alternative	Examples for Description
Alternative: Adobe Acrobat -> DOC	• Standard Settings of Adobe Acrobat,

	Windows XP Workstation with Service Pack 3 used for migration
Alternative: Doc-> DOC	• Standard Settings of Convert Doc, Windows XP Workstation with Service Pack 3 used for migration
Alternative: Acrobat -> HTML	• Standard Settings of Adobe Acrobat, Windows XP Workstation with Service Pack 3 used for migration

Table 10 Examples for the boxes in Develop Experiments

2.4 Run Experiments

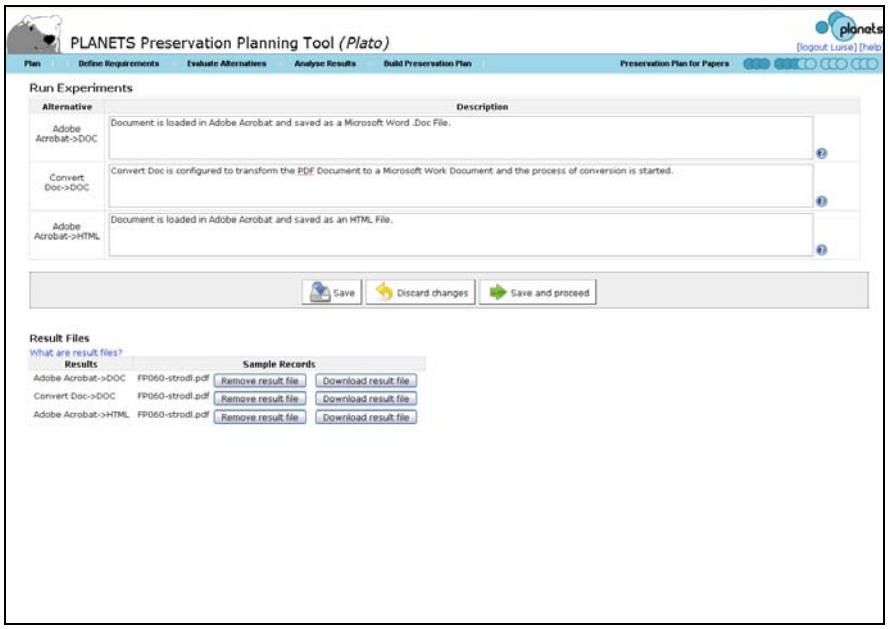


Figure 28 Run Experiments

In this step, all considered candidate tools for preservation actions are applied to the complete set of sample objects that have been defined in the first phase. This produces a series of experiment results that can be analysed and are stored for future evidence. In the case of object conversion, this means that the resulting output files shall be stored for further reference. When evaluating emulators, a documentation detailing the experience of rendering of the object is needed.

Type: Alternative	Examples for Description
Alternative: Adobe Acrobat -> DOC	• Standard Settings of Adobe Acrobat, Windows XP Workstation with Service Pack 3 used for migration
Alternative: Doc-> DOC	• Standard Settings of Convert Doc, Windows XP Workstation with Service Pack 3 used for migration

Alternative: Acrobat -> HTML	• Standard Settings of Adobe Acrobat, Windows XP Workstation with Service Pack 3 used for migration

Table 11 Examples for the boxes in Run Experiments

2.4.1 Results Files

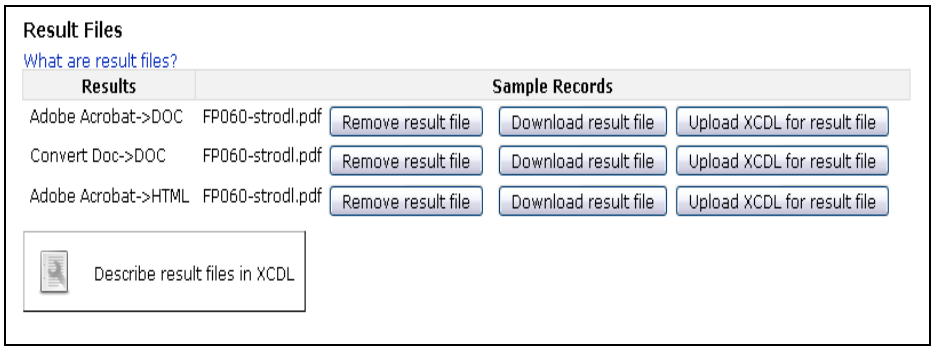


Figure 29 Results Files

2.5 Evaluate Experiments

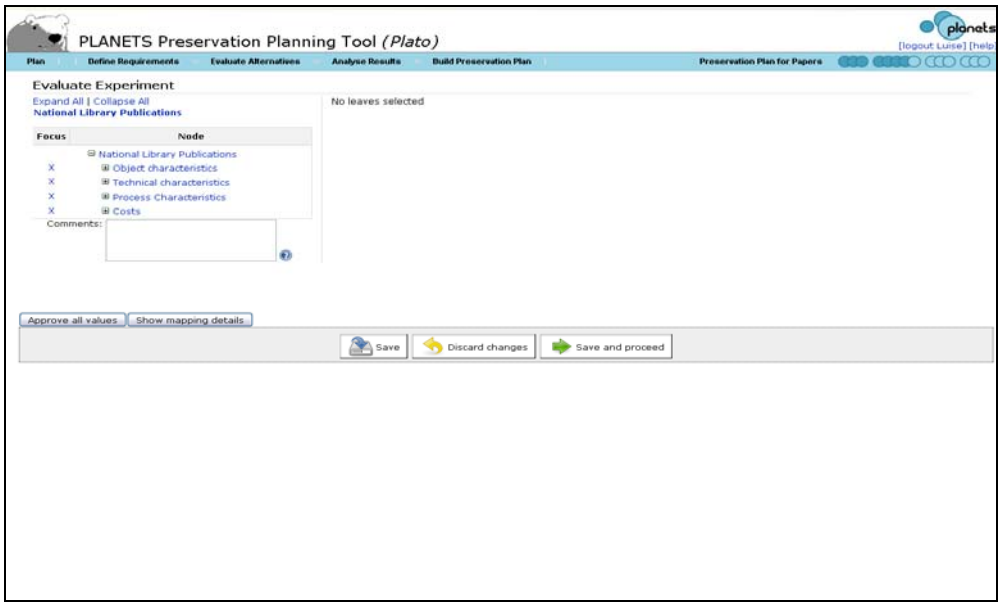


Figure 30 Evaluate Experiment

The evaluation of experiments is based on the requirements specified in the objective tree. All criteria on the leaf level of the objective tree are evaluated, taking into account the empirical evidence resulting from the experiments conducted. Each *Preservation Action Tool* is evaluated through applying it on *Sample Objects* in a controlled experiment. This creates an *Experiment Result* that constitutes part of the evidence base. A *Criterion* is a measurable *Requirement*. It can be associated with a tool (*Tool Criterion*) or vary with every object a tool is applied to (*Object Criterion*). In the latter case, it can be mapped to an *Object Property*. These properties are measured of the

original *Sample Object* and the *Experiment Result*, and the obtained values are compared through a comparison metric. Tool criteria, on the other hand, are associated with a *Tool Property* and evaluated in a *Tool Evaluation*. For example, the previously mentioned criterion *image width unchanged* is an *object criterion* which can be measured by characterisation tools such as JHove or XCL and compared automatically for each result of an experiment. Similarly, the relative file size of objects can be measured automatically per object. The relative file size averaged over the sample objects would then be used as evaluation value for the corresponding criterion. In other cases, information might be obtained from registries or inserted manually. For example, the judgment of quality of documentation, or the degree of adoption of a file format, can be queried in registries such as PRONOM, or judged by the preservation planner. Documenting the evaluation of experiment results completes the empirical evidence base for decision making and concludes the second phase of the preservation planning workflow.

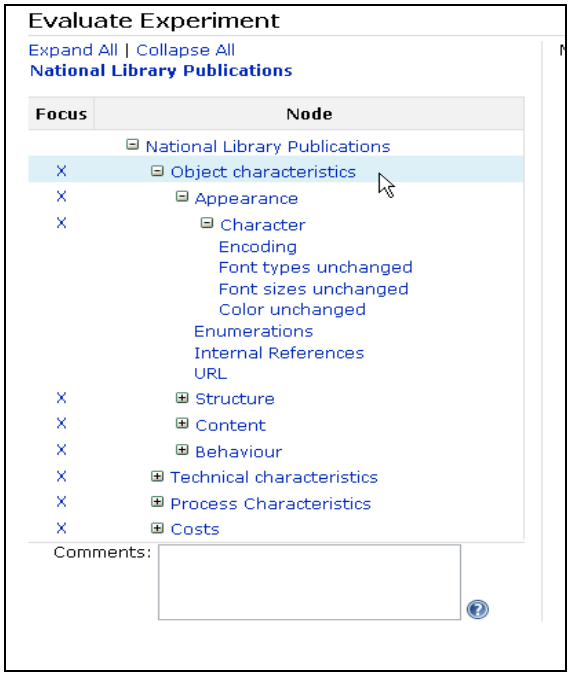


Figure 31 Requirements to evaluate

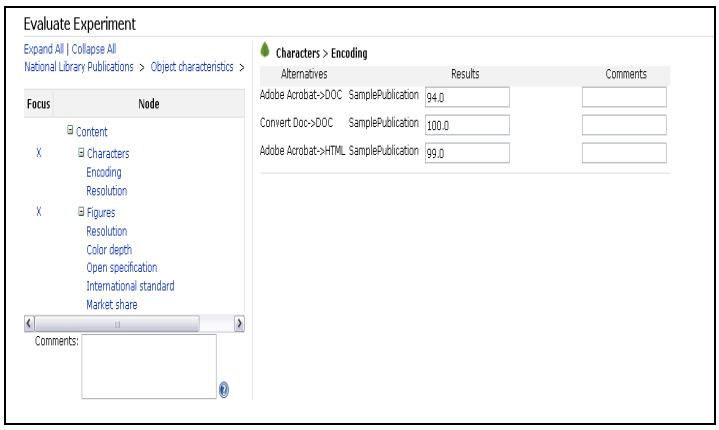


Figure 32 Evaluation of requirement 'Encoding'

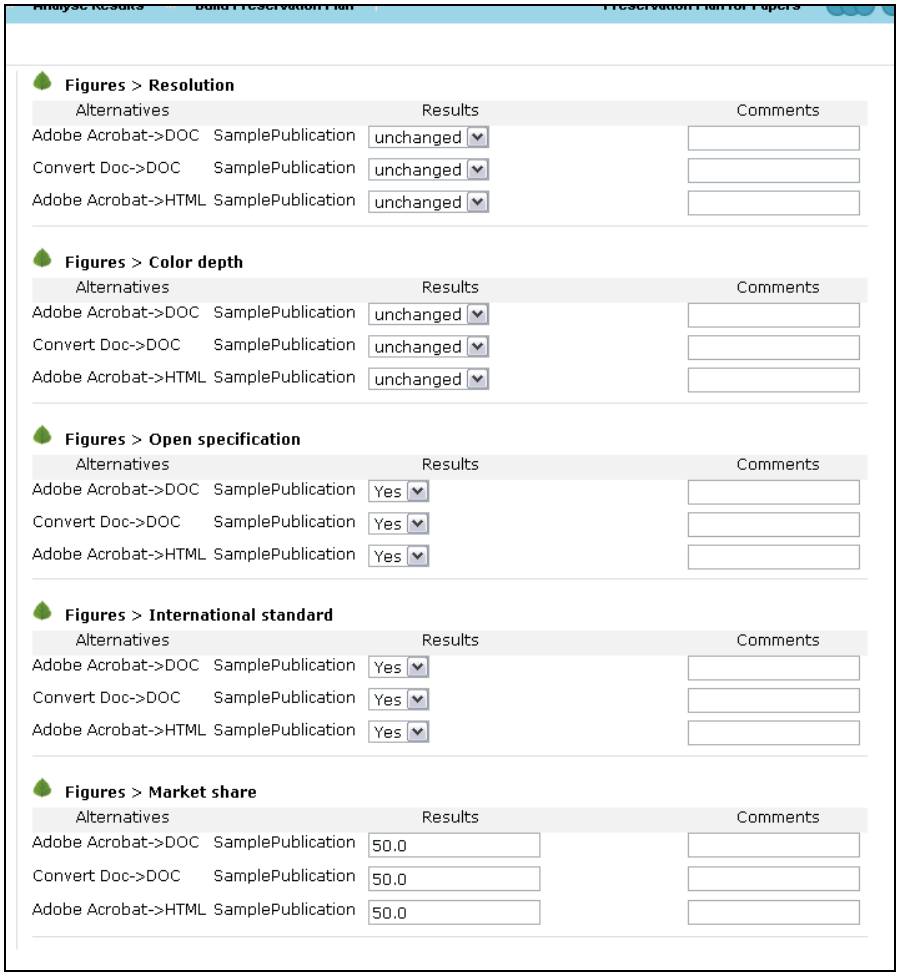


Figure 33 Evaluation showing different scales

3. Analyse Results

3.1 Transform Measured Values

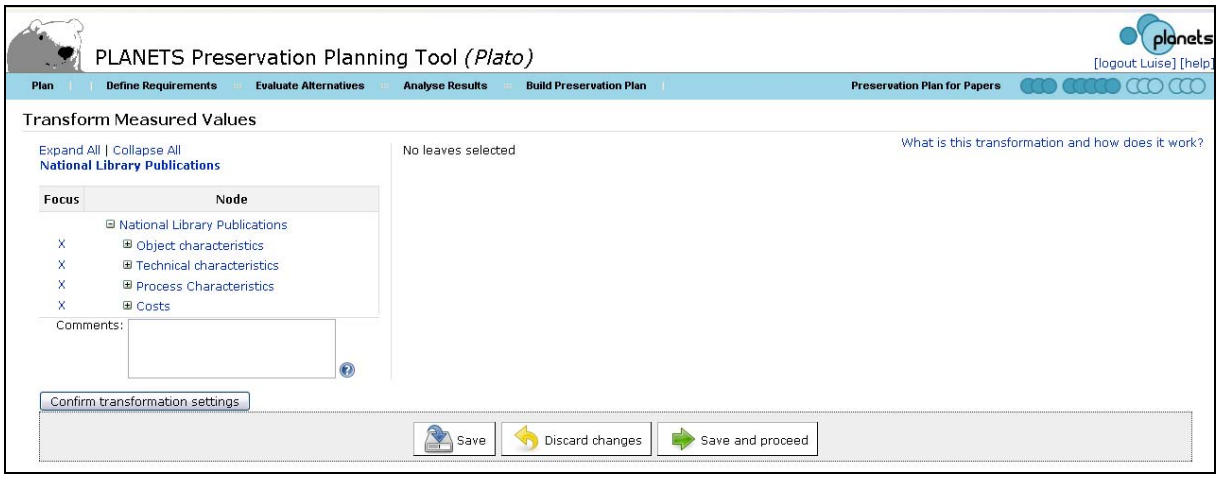


Figure 34 Transform Measured Values

The result of the previous step is an objective tree fully populated with evaluation values for all criteria. However, the measurements in this tree are of varying scales and thus cannot be aggregated

and compared directly. Thus, transformation rules are defined which result in a mapping from all possible measurement scales to a uniform target scale. This scale usually consists of real numbers ranging from 0 to 5. The lowest value 0 denotes an unacceptable result, while 5 is the best possible evaluation value. Corresponding to the scales employed, we can distinguish two types of transformation settings: numerical thresholds and ordinal mappings.

- For ordinal values, a mapping is defined for each possible category, resulting in a value between 0 and 5 to be assigned. For a boolean scale, *Yes* might be mapped to 5, whereas *No* will often be mapped to a low value. In this case, a decision has to be made whether the negative result *No* should be acceptable or not, i.e. mapped to 1 or to 0.

- For numeric values, thresholds are defined for each integer number from 1 to 5. All numbers below the lowest threshold (or above the highest, in case of descending order) will then be transformed to 0. The calculation of values between the threshold is usually done using linear interpolation.

In both cases, the definition of *acceptance criteria* is an essential step, where decision makers have to clearly specify the constraints they are willing to accept. This further provides a gap analysis which clearly points out both strengths and limitations of the candidates under evaluation.

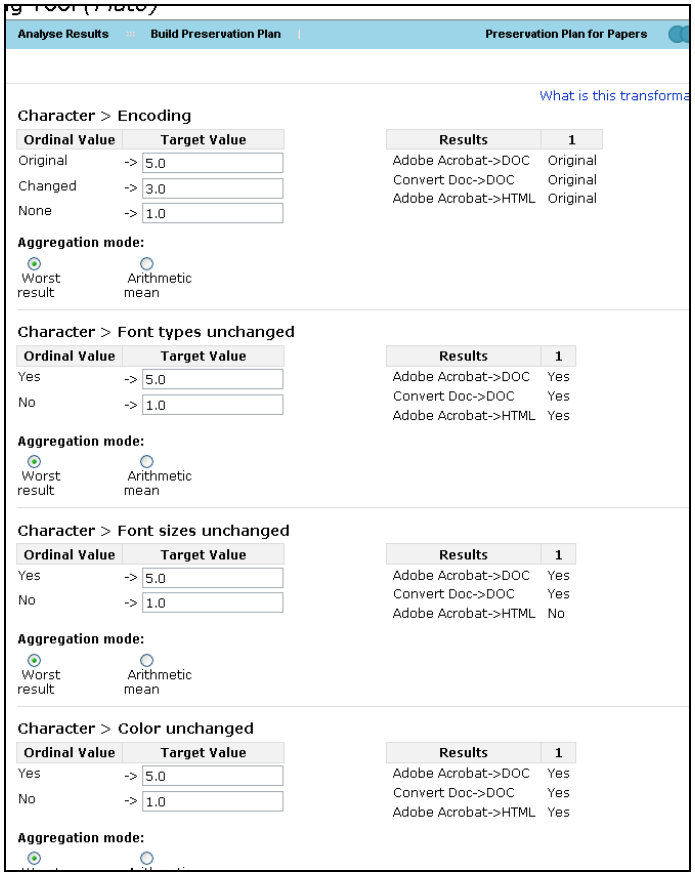


Figure 35 Transformation tables for different requirements

3.1.1 Why Transformation?

Values in the requirements tree are measured in different units (seconds, euro, bits and goodness values). To be able to aggregate the different values they have to be transformed to a uniform scale. Experience has shown that a scale with the resolution of discrete values 0-5 with 0 being an unacceptable value and 5 the best possible result works very well.

3.1.2 Transformation tables

For every leaf in the tree a transformation table has to be created:

Ordinal Values, Yes/Acceptable/No
For every ordinal or Yes/Acceptable/No value a numeric value has to be assigned. For an ordinal value that is considered better a higher numeric value has to be assigned than to an ordinal value that is considered worse. For an ordinal value that makes an alternative completely unacceptable '0'.

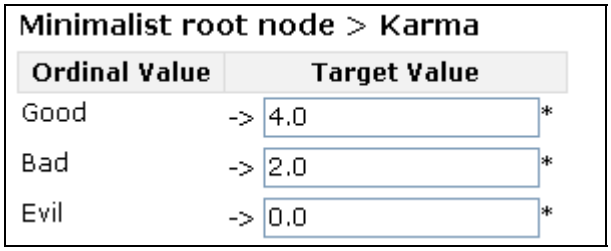


Figure 36 Transformation tables for ordinal scale

Numeric Values (Positive Number, Positive Integer, Int Range)
For threshold values numeric values are assigned. By selecting the appropriate 'Threshold stepping' it can be chosen, if the transformation should be linear (interpolating between the values) or if steps should be used.

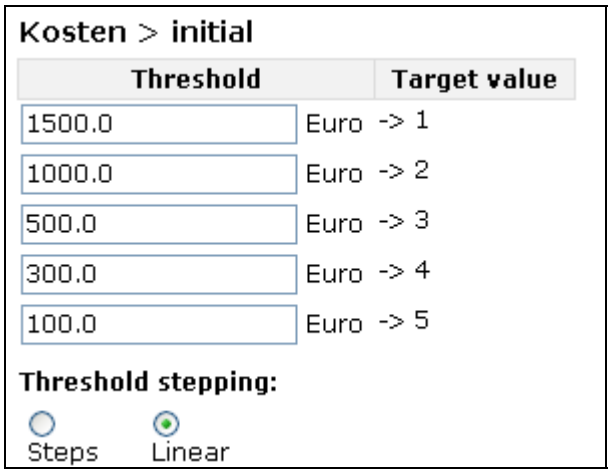


Figure 37 Transformation table for numeric scale

With "Steps" selected for Threshold Stepping, a measured value of e.g. 65 would get the same transformed value as 60 ('3'). If you choose "Linear" the value will be interpolated between the transformation values, so a value of '3,5' would be assigned.

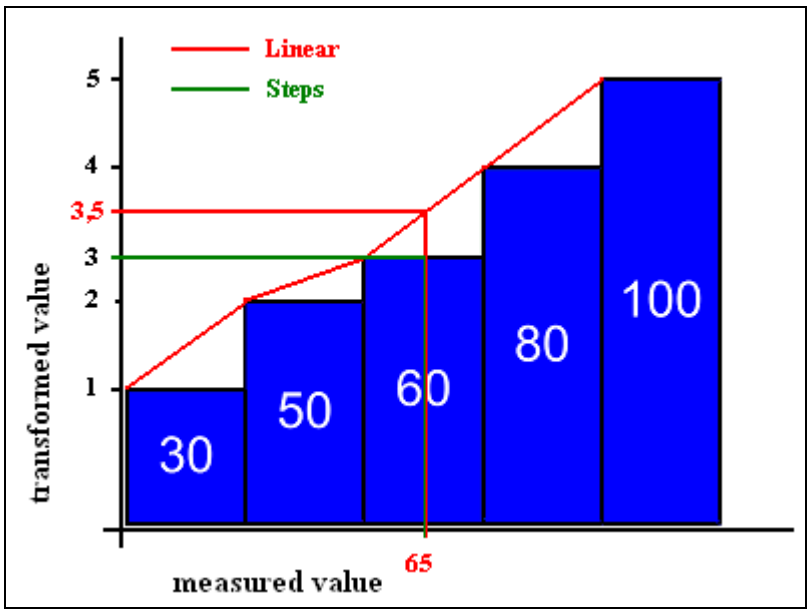


Figure 38 Interpolation of measured values

The measured values from the evaluation are shown next to the transformation tables.
To help you select the correct numeric values for the transformation, all the measured results are shown in a table to the right of the transformation table. For every alternative the sample records (numbered one to number of sample records) are shown in one line.

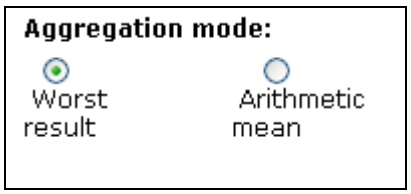
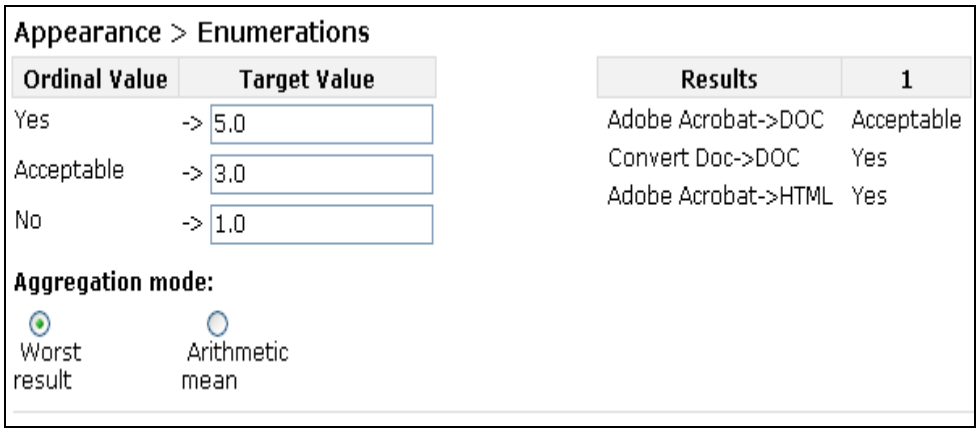


Figure 39 Aggregation mode for transformation

3.1.3 Examples for transformations for ordinal values and numeric values

Transformation of ordinal values
In this example the enumerations in a document are evaluated. If the enumeration is available and like it is in the original the value 'yes' is assigned. For an enumeration that is still supported but changed, 'acceptable' would be selected. If the enumeration is not supported anymore 'no' is chosen for this values.
The transformed values are '5' for 'yes', as this is the best possible result, '3' for acceptable and '1' for 'no'. We did not assign '0' to 'no', as it is not a knock-out criterion in our scenario.
Next to the transformation table we can see the results for every alternative for the one sample record we have in our fictional plan.
As an aggregation method 'Worst' is selected. If we would have more than one sample record and we have 'acceptable' for one record and 'yes' for every other record, the final value for this alternative would still be '3', as 'acceptable' is the worst result of all records for this alternative.



Transformation of numeric values

This example shows how many MegaBytes per second an alternative is able to handle. The measured values are again shown next to the transformation table. '0.05' MByte/second is our best result, so we assigned '5' to this value. '0.01' was the worst result, so '1' is assigned. The other values are evenly distributed between these values.

'Steps' is selected for Threshold stepping, so if an alternative would be able to handle 0.019 MByte/second, the resulting value would still be '1'.

As Aggregation Method "Arithmetic Mean" has been selected. If we would have two sample records, 0.01 MByte/second for one of the records and 0.02 MByte/second for the other record the resulting transformed value would be the arithmetic mean of '1' and '2' -> '1.5'.

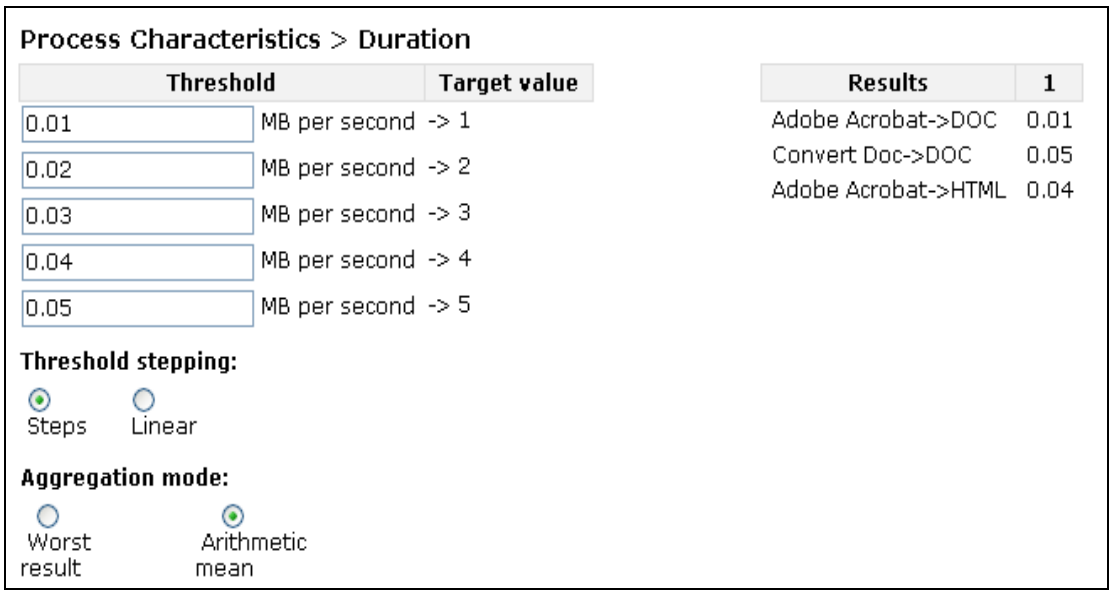


Figure 40 Transformation of numeric values

3.2 Set Importance Factors

This section explains the automatic weight balancing of objective trees in Plato.

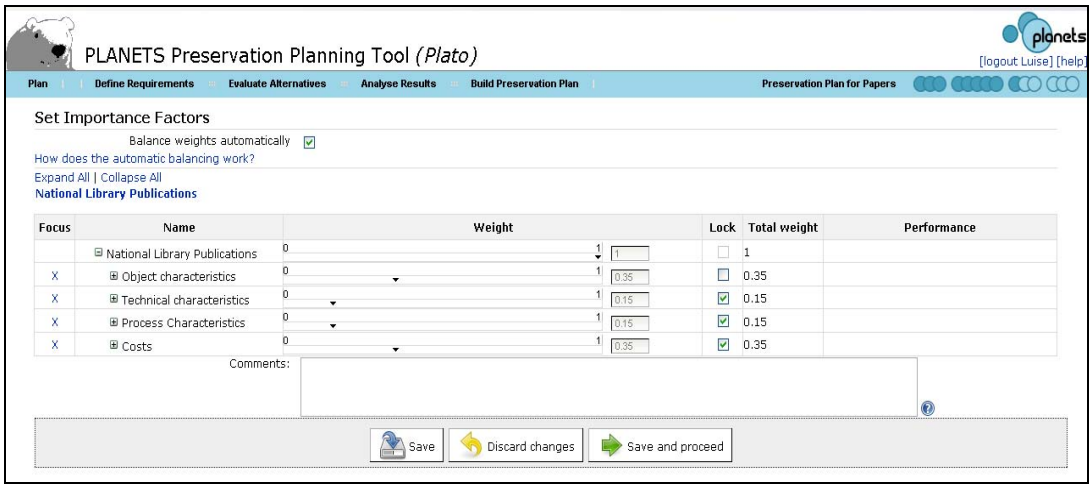


Figure 41 Set Importance Factors

When you reach this step the first time, Plato will automatically balance the relative importance of all nodes in the tree equally.
Using the tree table editor, you can then set the relative importance factors of all siblings in a branch for all levels.
The sum of weights on each level must always be equal to 1; this will be validated before you can proceed.
The column 'total weight' provides the overall influence of a node on the root value. For example, a node with weight 0.5 that is child of a node with weight 0,5 which is directly under the root will have a total weight of 0.5*0.5=0.25.

Automatic balancing
If the flag 'Balance weights automatically' is checked, Plato will activate an automatic balancing that relieves you of checking the sum yourself. Just use the sliders to increase weights of nodes that you deem important. The remaining weights will be balanced automatically. Each weight becomes locked once you adjusted it yourself, so that it is not rebalanced afterwards. This is indicated by the 'lock' column.

Example
Assume you have a root node with four nodes - A,B,C,D. When you reach the page, all weights are set to 0.25.
Now you adjust the weight of node A to 0.4 - the remaining weights are set to 0.2 each and node A is locked.
Now you adjust the weight of node D to 0.4. The effect is that node D becomes locked, too, while B and C are set to 0.1 each. Node A is not affected, because it is already locked.

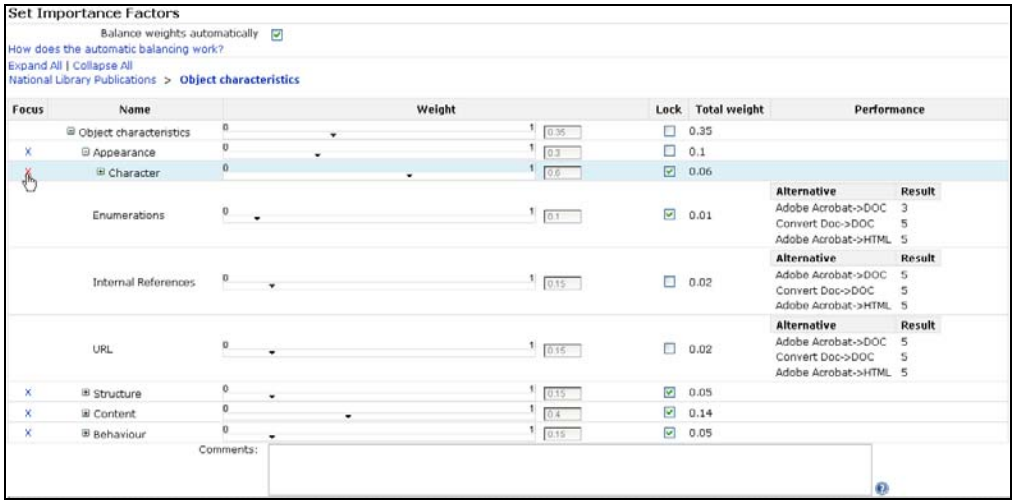


Figure 42 One example in Set Importance Factors

3.3 Analyse evaluation results for Preservation Plan for Papers

The final step of the evaluation phase considers the complete evidence base of information produced during the previous phases of the workflow and analyses the performance of the candidate components in the experiment evaluation to arrive at a conclusion and recommendation for the best tool to employed, and the corresponding configuration. Alternatives are ranked by their evaluation values which are aggregated over the tree hierarchy using two different methods.

- Weighted multiplication is used to filter alternatives which exhibit unacceptable evaluation values at the criterion level, as these have been mapped to a target value of 0 during transformation and thus result in a total performance of 0.
- On the remaining alternatives, weighted addition is used to directly compare the performance of tools on all levels of the tree hierarchy.

The analysis and comparison of the alternatives considered can be guided significantly by a graphical visualization as provided by the planning tool described in Section 6. As a result of the evaluation, the preservation planner makes a decision and recommendation for a tool to be selected. The method allows for the selection of multiple components that are considered to be complementary. For example, many conversion tools for electronic documents have problems with entirely preserving the layout as it was displayed in the original environment, whereas migrating a document to an image loses the future potential for full-text search access. In some cases it might be desirable to combine both approaches and thus select multiple tools for the incorporation into a preservation system. As an essential element of the recommendation, the reasons underlying it are documented, together with the expected effects of applying this strategy on the set of objects at hand. For example, it may be known that the easy *editability* of objects will be lost as a direct cause of converting them to a format such as PDF/A. As this might not be a requirement, or not be assigned

significant weight, it might not influence the decision in a significant way. However this reasoning needs to be documented as part of the decision making procedure.

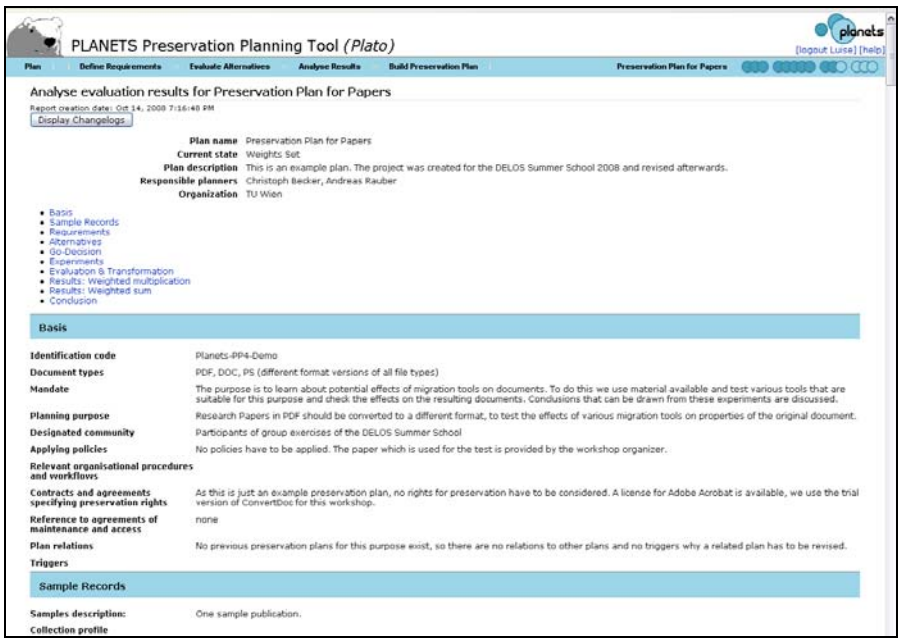


Figure 43 Analyse evaluation results for Preservation Plan for Papers

Figure 44 Go Decision for Plan

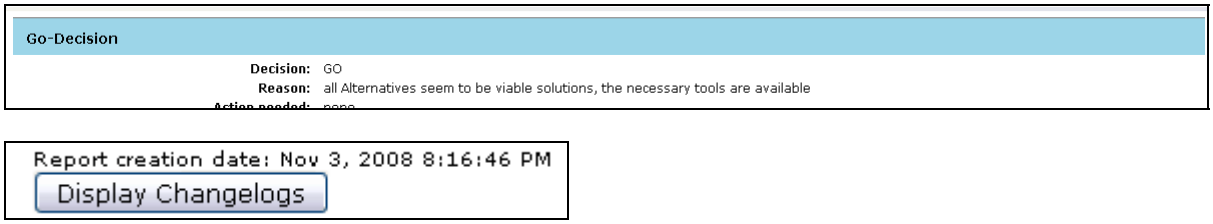


Figure 45 Display Changelogs

The change logs stored with the preservation plan can be displayed by clicking on 'Display Chang'logs'. The change logs are displayed next to the respective fields or group of fields that have been altered.

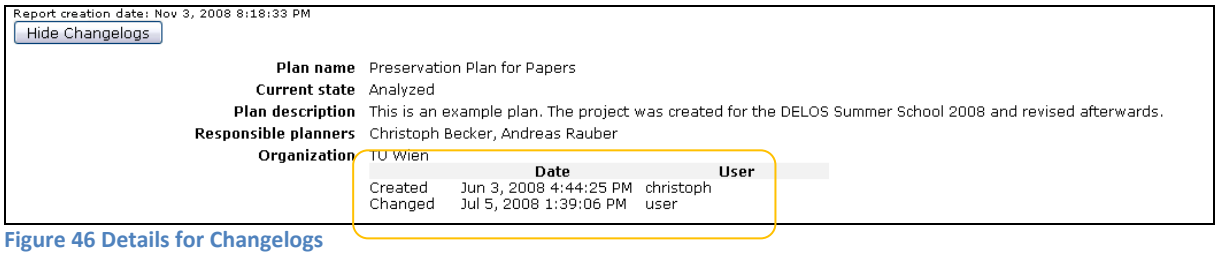


Figure 46 Details for Changelogs

4. Build Preservation Plan

In the fourth and final phase of the planning workflow, a preservation plan is created, based on the decision for a preservation action. It specifies a series of concrete steps or actions, along with organisational responsibilities, rules and conditions for executing the preservation action on the collection.

A ***preservation plan*** defines a series of preservation actions to be taken by a responsible institution due to an identified risk for a given set of digital objects or records (called collection). The Preservation Plan takes into account the preservation policies, legal obligations, organizational and technical constraints, user requirements and preservation goals and describes the preservation context, the evaluated preservation strategies and the resulting decision for one strategy, including the reasoning for the decision. It also specifies a series of steps or actions (called **preservation action plan**) along with responsibilities and rules and conditions for execution on the collection. Provided that the actions and their deployment as well as the technical environment allow it, this action plan is an executable workflow definition.

4.1 Create Executable Plan

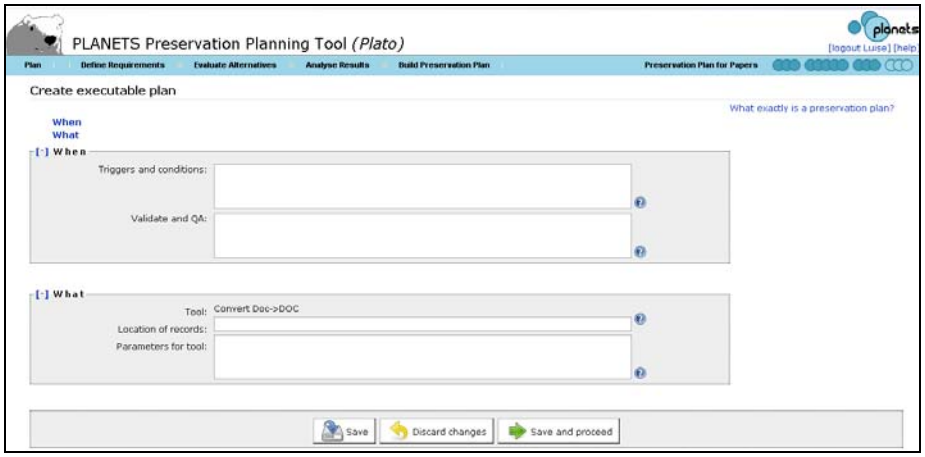


Figure 47 Create Executable Plan

This step of the workflow defines the triggers for the execution and the conditions under which the preservation action will be carried out. Hard- and software requirements as well as dependencies to other systems are documented. To enable the execution of the preservation plan, tool settings and details about the location of the collection on which the action is to be performed are defined. To quality assure the performed actions, a subset of the criteria used for evaluating solutions can be selected. These criteria will then be extracted using characterisation services on the objects after the

performed action and validating that defined thresholds of these criteria are met. The necessary documentation that has to be recorded for performing the action is also defined in this step.

4.1.1 When

Triggers and conditions for execution Hard-and software requirements, other dependencies

Type	Examples
Triggers and conditions	
Validate and QA	

Table 12 Examples for the boxes in Create Executable Plan / When

4.1.2 What

- Executable Preservation Plan (e.g. executable workflow, programme) that will execute the preservation action on the digital objects and automated mechanisms for validating results of preservation action [includes <ID>]
- Other actions needed (reporting/documenting…)

Type	Examples
Location of records	
Parameters for tool	

Table 13 Examples for the boxes in Create Executable Plan / What

4.2 Define Preservation Plan

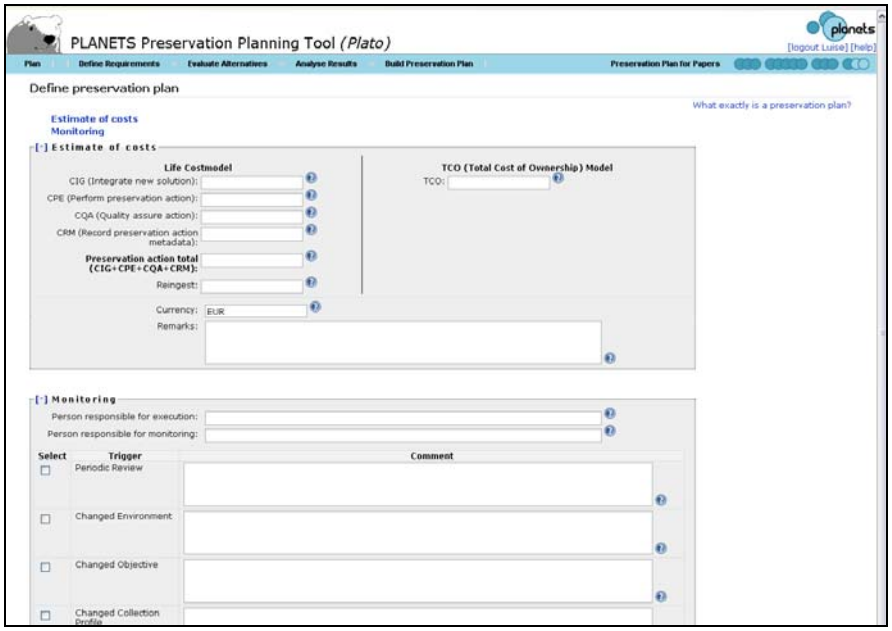


Figure 48 Define preservation plan

While most parts of the preservation planning workflow take care of the technical aspects of the preservation plan, this step mainly defines the organizational details.

Cost factors influence the decision on a specific alternative. In this step, a more detailed calculation of costs using an approved cost model is performed. Cost models that can be used are for example Life2 [2] or the Total Cost of Ownership (TCO) model. While an estimate of the costs may be fine for evaluating the alternatives, the costs have to be determined as accurate as possible in this step.

The assignment of responsibilities is also documented in this step. Monitoring the process of applying the preservation actions has to be done by a different role than executing the preservation plan. It also has to be monitored if an event occurs that makes it necessary to reevaluate the plan. Possible triggers for this are either a scheduled periodic review, changes in environment such as new available tools detected through technology watch, changed objectives (changed target community requirements) or a changed collection profile (e.g. new objects in the collection). Another possible trigger is that certain thresholds on the evaluation are no longer met by applying the preservation action.

4.3 Validate Preservation Plan

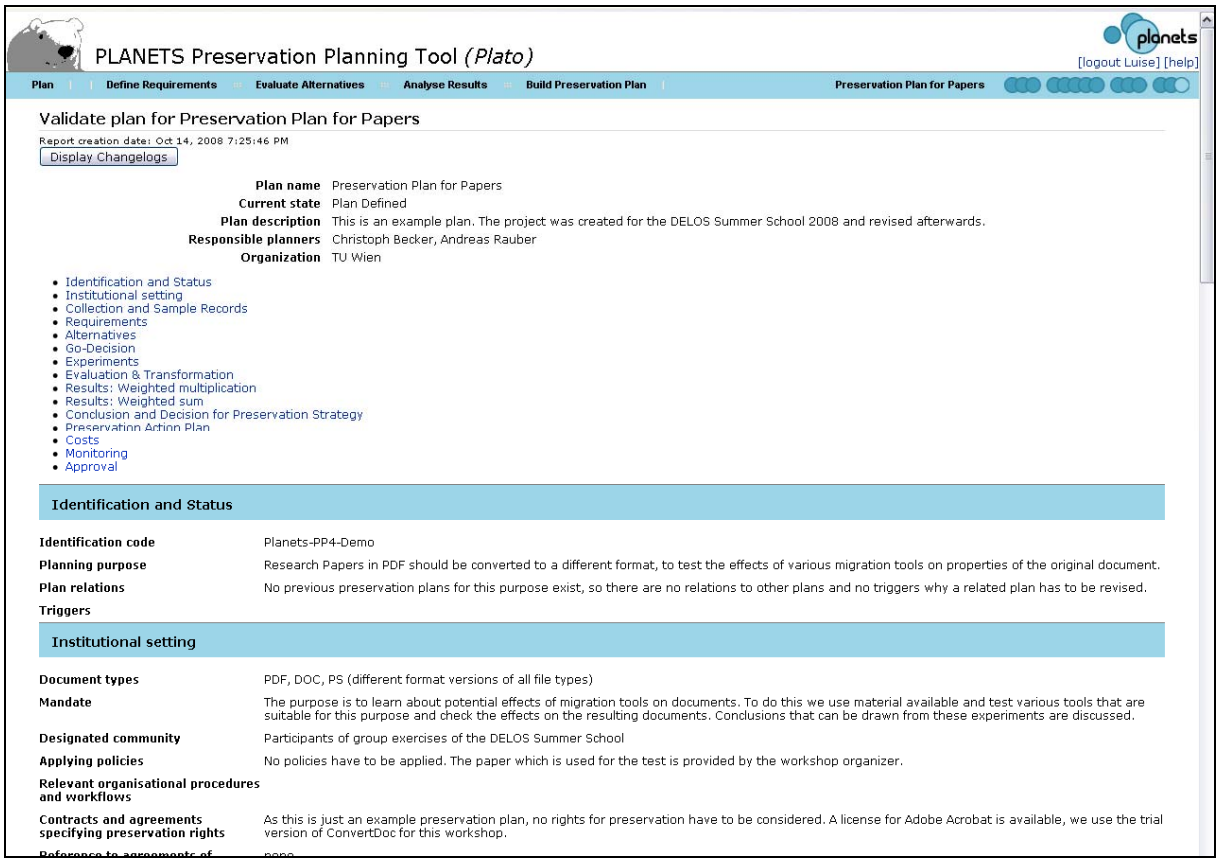


Figure 49 Validate plan for Preservation Plan for Papers

In the final stage the whole documentation about the preservation plan has to be reviewed. The process of evaluating different alternatives and taking a decision for a recommended tool based on

the evaluation and the creation of the preservation action plan, the documentation about the basic framework of the institution and parameters under which the plan is valid are verified again.

Tests on a defined set of sample objects are performed in this step to check the validity of the preservation plan and the preservation action plan. Finally the validated plan has to be approved by the person responsible for approval. Once the plan is approved, no more changes on the plan should be done without revising the whole plan.