

Dokumentation intranda TaskManager 2.0

Technische Dokumentation

Revision	Datum	TaskManager-Version	Änderungen
1	25.03.2014	2.0	Initialer Entwurf
2	02.05.2014	2.0	Überarbeitung, Integration Screenshots
3	16.05.2014	2.0	Integration Internet Archive Download
4	09.02.2015	2.0	Integration ALTO conversion, J-Download, J-Import
5	02.10.2015	2.0	Integration JHOVE-Validierung
6	12.10.2015	2.0	Integration LayoutWizzard
7	13.10.2015	2.0	Integration Storage Balancer
8	10.11.2015	2.0	Integration Epub Generierung

Inhaltsverzeichnis

1	Einführung	3
2	Was ist der intranda TaskManager	3
3	Nutzeroberfläche	3
3.1	Dashboard	3
3.2	Suche	6
3.3	Plugin-Seiten	7
3.3.1	Aktive Jobs innerhalb der Warteschlange	8
3.3.2	Abgeschlossene Jobs	8
3.3.3	Fehlerhafte Jobs	8
3.3.4	Job-Aktionen	10
4	Job-Plugins	14
4.1	JPEG-2000-Generierung mittels Lurawave	14
4.2	Imagekonvertierung mittels ImageMagick	15
4.3	JPEG-2000-Generierung mittels Open JPEG	16
4.4	Validierung von JPEG-2000-Images mittels jpylyzer	17
4.5	Volltexterkennung - OCR	24
4.6	Archivierung mittels Tessella Preservica	25
4.7	Internet Archive Download	27
4.8	ALTO Generierung aus ABBYY-xml mittels intranda AltoConvert	29
4.9	Download von mehrbändigen Werken aus dem Internet Archive mittels intranda InternetArchiveParser.jar (J-Download)	30
4.10	Import mehrbändiger Werke aus dem Internet Archive nach Goobi mittels intranda InternetArchiveParser.jar (J-Import)	31
4.11	Validierung von TIFF Images mittels JHOVE	32
4.12	Bildanalyse mittels des intranda LayoutWizzards	34
4.12.1	LayoutWizzardAnalyse	34
4.12.2	LayoutWizzardSave	34
4.12.3	Templates	36
4.12.4	Konfiguration	36
4.13	Speicherverteilung über den Storage Balancer	37
4.14	Generierung von eBooks im Epub-Format	39
5	Zusätzliche Details	41

1 Einführung

Bitte beachten Sie, dass die vorliegende Dokumentation nicht verändert oder in veränderter Form weitergegeben werden darf. Eine kommerzielle Nutzung dieser Dokumentation ist nicht gestattet.

Dieses Werk ist unter einer Creative Commons Lizenz vom Typ Namensnennung - Nicht kommerziell - Keine Bearbeitungen 4.0 International zugänglich. Um eine Kopie dieser Lizenz einzusehen, konsultieren Sie <http://creativecommons.org/licenses/by-nc-nd/4.0/> oder wenden Sie sich brieflich an Creative Commons, Postfach 1866, Mountain View, California, 94042, USA.



2 Was ist der intranda TaskManager

Der intranda TaskManager ist eine flexible Applikation zur Koordinierung verschiedener rechenintensiver Aufgaben. Der modulare Aufbau gewährleistet, dass einzelne Jobtypen individuell als Plugins zu dem TaskManager hinzugefügt und konfiguriert werden können. Dabei kann sich jedes Plugin sehr individuell verhalten.

Die Abarbeitung der jeweiligen Arbeiten findet für jeden Jobtyp als Plugin innerhalb einer eigenen Warteschlange statt, so dass die unterschiedlichen Jobs sich nicht gegenseitig beeinträchtigen. Auch können die Verarbeitungen der jeweiligen Plugins individuell auf unterschiedlichen anderen Systemen ausgelagert stattfinden oder gar geclustert werden.

Bitte beachten Sie: Abhängig von Ihrer jeweiligen Konfiguration verfügen Sie ggf. nur über einzelne der in dieser Dokumentation beschriebenen Plugins.

3 Nutzeroberfläche

Die Nutzeroberfläche des intranda TaskManagers unterteilt sich grundsätzlich in drei Bereiche, die im Folgenden detailliert beschrieben werden sollen. Es handelt sich hierbei um das Dashboard als Übersichtsseite, die Suchmaske sowie die jeweiligen Plugin-Seiten.

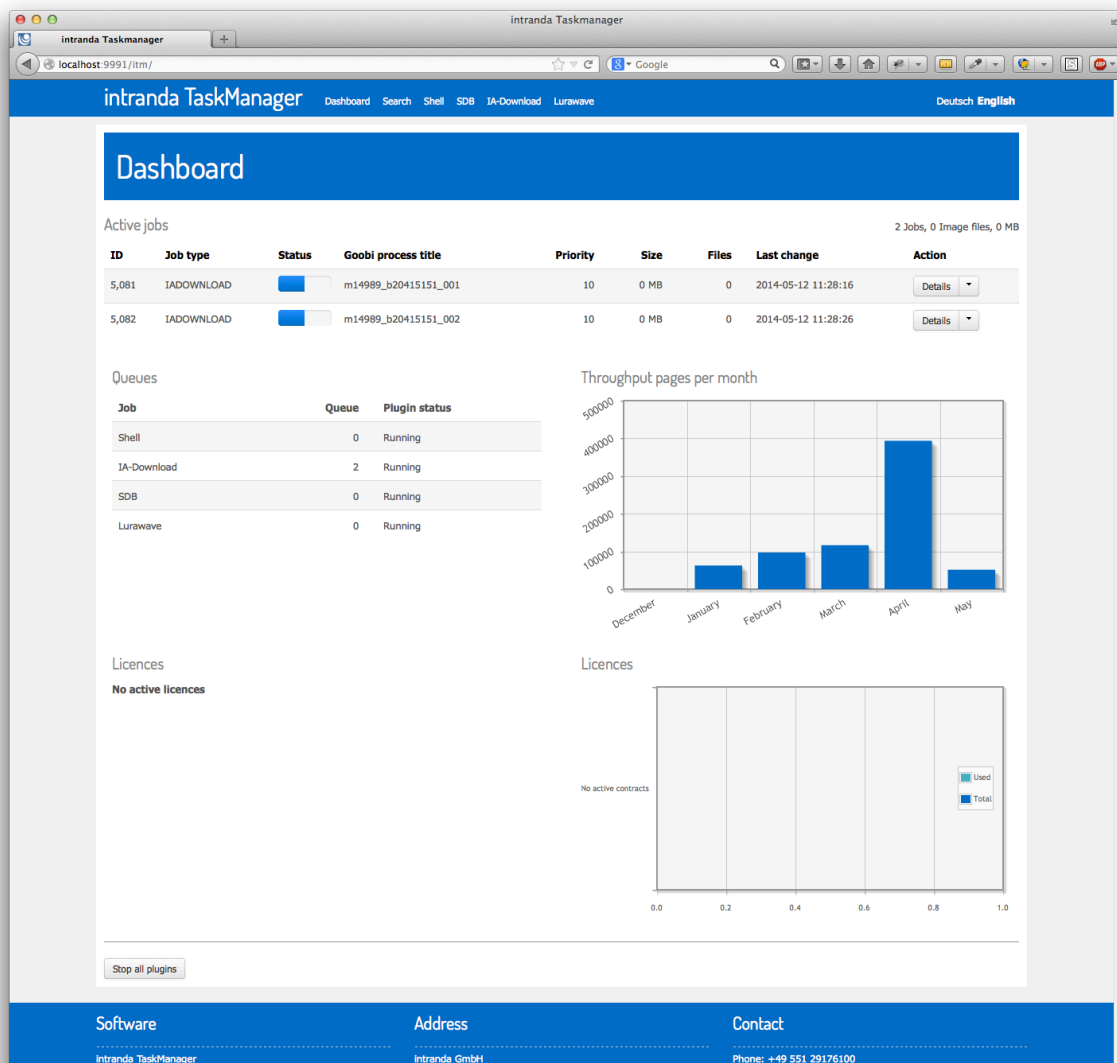
3.1 Dashboard

Öffnet man den TaskManager als Webapplikation mit einem Webbrowser, so gelangt man üblicherweise zunächst zum Dashboard. Hierbei werden bereits auf einen Blick alle entscheidenden Informationen der verwendeten Jobtypen übersichtlich dargestellt, ohne dass detailliert in die jeweiligen Warteschlangen der Plugins Einblick genommen werden muss.

Im oberen Bereich des Dashboards werden zunächst die ersten maximal fünf Jobs aufgeführt, die sich gerade in Bearbeitung befinden. Dabei werden nur diejenigen Jobs berücksichtigt, die sich in einem der folgenden Statuszustände befinden:

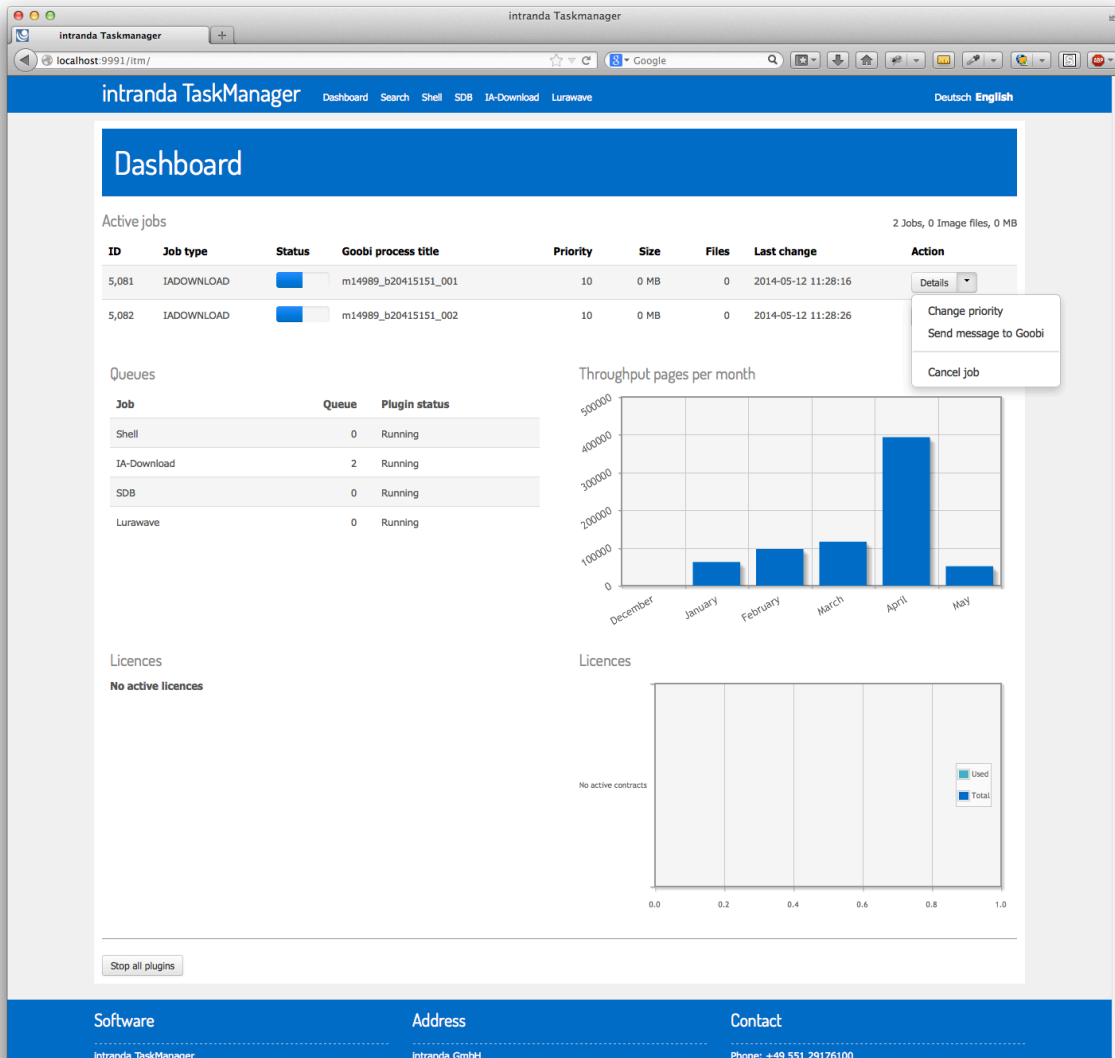
- PAUSED
- UPLOAD
- DOWNLOAD
- PROCESSING
- STARTED
- WAITINGFORUPLOAD

Oberhalb der Tabelle für die aktuell laufenden Jobs werden einige Informationen zur allen Jobs zusammengefasst aufgeführt. Dies erlaubt einen schnellen Überblick über die aktuell noch zu verarbeitenden Daten.



Bereits innerhalb des Dashboard bietet sich die Möglichkeit, zu den einzelnen aufgelisteten Jobs detaillierte Informationen einzusehen. Auch kann an dieser Stelle pro Job individuell Eingriff in einen gewählten Job genommen werden, um beispielsweise die Priorität zu ändern oder einen Job abzubereiten.

Klicken Sie zur Bearbeitung eines individuellen Jobs einfach innerhalb der Spalte **Action** auf den jeweiligen Button, um die gewünschte Änderung vorzunehmen. Eine genaue Auflistung der Eingriffsmöglichkeiten, die sich hierbei bieten, erfolgt innerhalb der detaillierten Beschreibung der Plugin-Seiten.



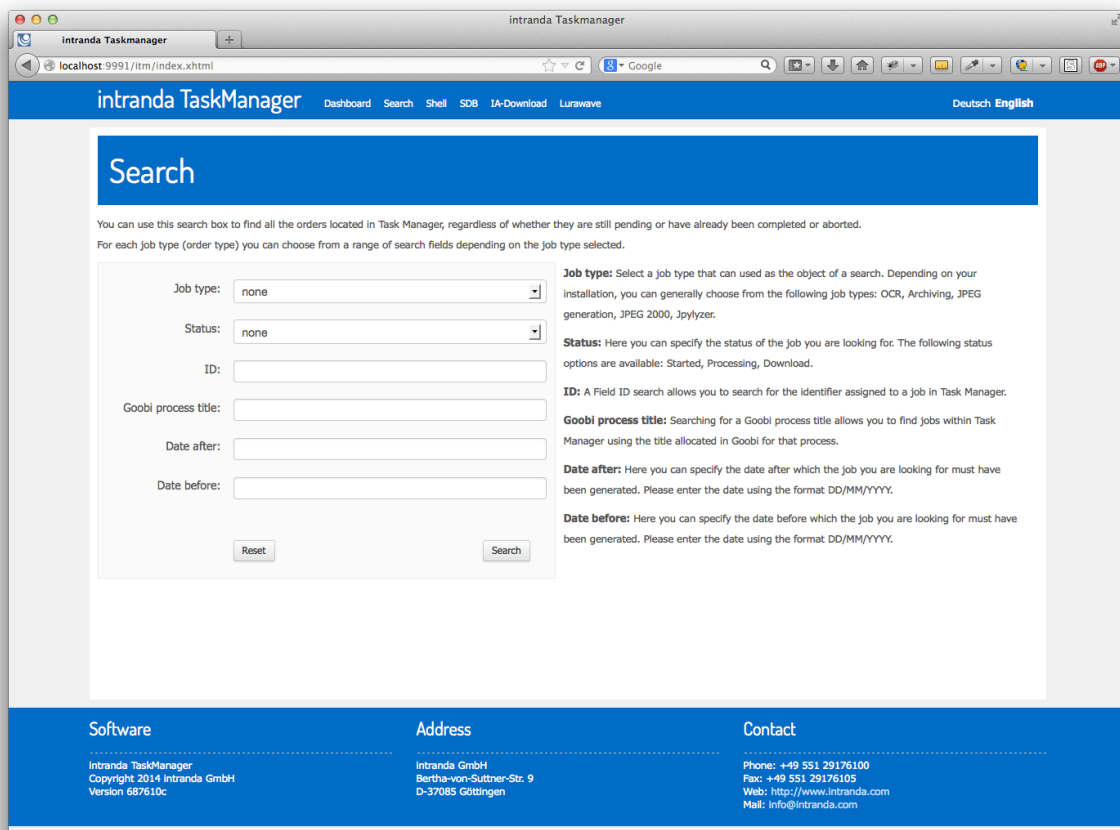
Unterhalb der Auflistung einiger aktiver Jobs wird auf dem Dashboard ebenfalls angezeigt, wie der jeweilige Status der einzelnen Plugin-Warteschlangen ist. Sollten einzelne Plugins z.B. für Wartungszwecke individuell gestoppt worden sein, lässt sich dies hier übersichtlich nachvollziehen.

Für abrechnungsrelevante Inhalte oder auch für rein statistische Zwecke wird auf dem Dashboard ebenfalls ein Überblick über die Durchführung von Jobs in der Vergangenheit angeboten. Hierzu wird einerseits die Anzahl an abgeschlossenen Jobs pro Monat aufgelistet. Andererseits wird im Falle von OCR-Durchführung sowohl tabellarisch als auch in Form eines Diagramms aufgeführt, welche Anzahl an Seiten des bestehenden Seitenkontingents für die OCR-Durchführung bereits verbraucht wurde oder noch zur Verfügung steht. Die unmittelbare Interaktion mit dem intranda Lizenzserver garantiert hierbei eine A4-Seiten-genaue Anzeige des verbrauchten Kontingents und gewährleistet somit eine maximale Transparenz des Fortschritts.

Als administrative Funktion steht auf dem Dashboard im unteren Bereich außerdem ein Button zum Stoppen aller Plugins bereit. Wird dieser angeklickt, werden alle Jobs für sämtliche aktiven Plugins pausiert, sobald dies möglich ist. Beachten Sie dabei unbedingt, dass dies im Falle einiger Plugins mehrere Minuten in Anspruch nehmen kann. Erst wenn für alle aufgeführten Plugins der Status als gestoppt angezeigt wird, sollte der TaskManager (z.B. für Wartungszwecke) heruntergefahren werden.

3.2 Suche

Für den Fall, dass ein bestimmter Job gesucht und auf seine Verarbeitung hin überprüft werden soll, bietet der TaskManager eine detaillierte Suche an. Diese erlaubt, dass granular nach individuellen Feldern gesucht werden kann. So ist es z.B. möglich, mehrere Suchparameter – wie beispielsweise den Status, ID und Titel – miteinander zu kombinieren, um das gewünschte Trefferset zu erhalten.



Search

You can use this search box to find all the orders located in Task Manager, regardless of whether they are still pending or have already been completed or aborted.
For each job type (order type) you can choose from a range of search fields depending on the job type selected.

Job type:

Status:

ID:

Goobi process title:

Date after:

Date before:

Job type: Select a job type that can be used as the object of a search. Depending on your installation, you can generally choose from the following job types: OCR, Archiving, JPEG generation, JPEG 2000, Jpylyzer.

Status: Here you can specify the status of the job you are looking for. The following status options are available: Started, Processing, Download.

ID: A Field ID search allows you to search for the identifier assigned to a job in Task Manager.

Goobi process title: Searching for a Goobi process title allows you to find jobs within Task Manager using the title allocated in Goobi for that process.

Date after: Here you can specify the date after which the job you are looking for must have been generated. Please enter the date using the format DD/MM/YYYY.

Date before: Here you can specify the date before which the job you are looking for must have been generated. Please enter the date using the format DD/MM/YYYY.

Software
Intran|da TaskManager
Copyright 2014 Intran|da GmbH
Version 687610c

Address
Intran|da GmbH
Bertha-von-Suttner-Str. 9
D-57085 Göttingen

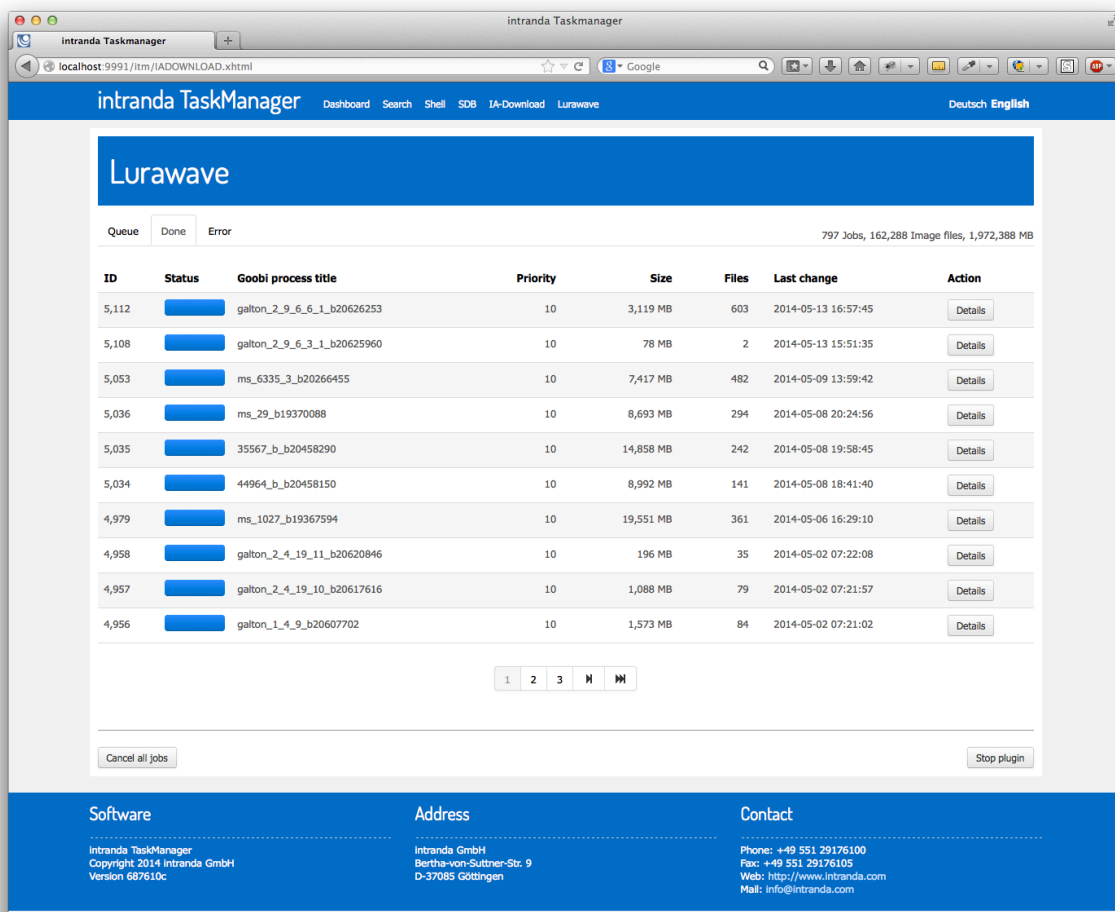
Contact
Phone: +49 551 29176100
Fax: +49 551 29176105
Web: <http://www.intran|da.com>
Mail: info@intran|da.com

Zur einfacheren Bedienung verfügt die Suche unmittelbar in der Nutzeroberfläche über eine Hilfe, die während der Eingabe von Suchparametern innerhalb des Formulars Unterstützung bietet.

3.3 Plugin-Seiten

Für jedes installierte Plugin innerhalb des TaskManagers existiert jeweils ein eigenständiger Menüpunkt in der Menüleiste. Wählt man hier das gewünschte Plugin aus, so bietet sich die Möglichkeit alle existierenden Jobs dieses Plugintyps getrennt nach ihrem Status aufzulisten. Dabei wird danach unterschieden, ob ein Job sich noch innerhalb der Warteschlange befindet, bereits erfolgreich abgeschlossen wurde oder mit einem Fehler abgebrochen ist.

Im rechten Bereich oberhalb der Tabelle werden einige statistische Informationen über alle Jobs der aktuell ausgewählten Registerkarte eingeblendet. Darin lässt sich z.B. ablesen, welche Anzahl an Dateien und welche Datenmengen vorliegen.



The screenshot shows the intranda TaskManager interface for the Lurawave plugin. The top navigation bar includes links for Dashboard, Search, Shell, SDB, IA-Download, and Lurawave. The main content area features a table of jobs with the following columns: ID, Status, Goobi process title, Priority, Size, Files, Last change, and Action. The table lists 10 jobs, all with a status of 'Queue' (indicated by a blue bar). Below the table, there are buttons for 'Cancel all jobs' and 'Stop plugin'. The footer contains contact information for intranda GmbH.

ID	Status	Goobi process title	Priority	Size	Files	Last change	Action
5,112	Queue	galton_2_9_6_6_1_b20626253	10	3,119 MB	603	2014-05-13 16:57:45	Details
5,108	Queue	galton_2_9_6_3_1_b20625960	10	78 MB	2	2014-05-13 15:51:35	Details
5,053	Queue	ms_6335_3_b20266455	10	7,417 MB	482	2014-05-09 13:59:42	Details
5,036	Queue	ms_29_b19370088	10	8,693 MB	294	2014-05-08 20:24:56	Details
5,035	Queue	35567_b_b20458290	10	14,858 MB	242	2014-05-08 19:58:45	Details
5,034	Queue	44964_b_b20458150	10	8,992 MB	141	2014-05-08 18:41:40	Details
4,979	Queue	ms_1027_b19367594	10	19,551 MB	361	2014-05-06 16:29:10	Details
4,958	Queue	galton_2_4_19_11_b20620846	10	196 MB	35	2014-05-02 07:22:08	Details
4,957	Queue	galton_2_4_19_10_b20617616	10	1,088 MB	79	2014-05-02 07:21:57	Details
4,956	Queue	galton_1_4_9_b20607702	10	1,573 MB	84	2014-05-02 07:21:02	Details

Unterhalb der jeweiligen Tabelle stehen außerdem pro Plugin zwei zusätzliche Buttons zur Verfügung, um das aktuelle Plugin zu pausieren oder alle noch laufenden Jobs abzuberechnen. Während der Abbruch eines Jobs endgültig ist und die weitere Verarbeitung des Jobs unmittelbar abbricht, führt das Stoppen eines Plugins dazu, dass die weitere Bearbeitung nur temporär unterbrochen wird. Ein erneutes Starten des Plugins führt anschließend dazu, dass die Bearbeitung des Jobs fortgeführt wird.

3.3.1 Aktive Jobs innerhalb der Warteschlange

Innerhalb der Registerkarte `Warteschlange` werden alle aktiven Jobs des gewählten Plugins aufgelistet. Als aktive Jobs werden hierbei folgende Statuszustände bewertet:

- PAUSED
- UPLOAD
- DOWNLOAD
- PROCESSING
- STARTED
- WAITINGFORUPLOAD

3.3.2 Abgeschlossene Jobs

Innerhalb der Registerkarte `Abgeschlossen` werden all diejenigen Jobs des gewählten Plugins aufgeführt, die bereits erfolgreich abgeschlossen wurden. Dies beinhaltet demnach nur diejenigen Jobs mit folgendem Status:

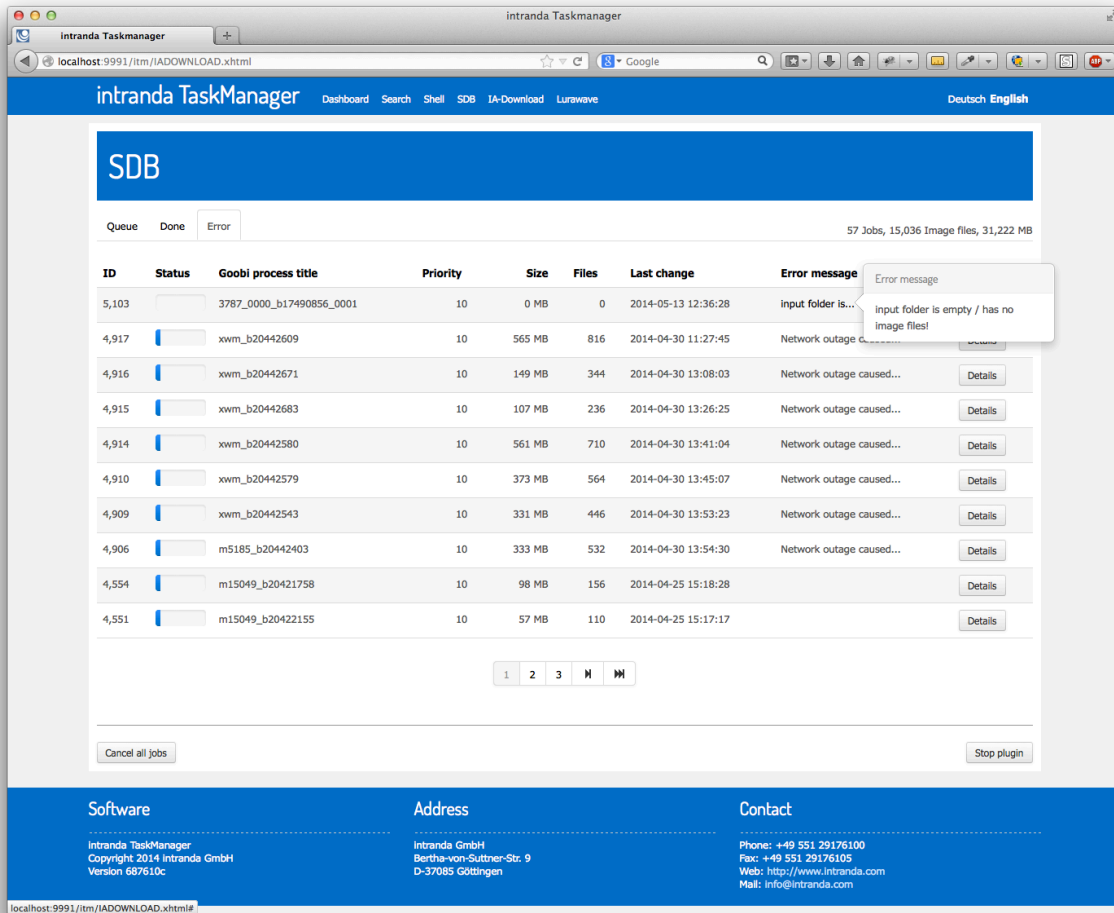
- DONE

3.3.3 Fehlerhafte Jobs

Innerhalb der Registerkarte `Fehler` werden alle Jobs des gewählten Plugins aufgeführt, die während der Verarbeitung zu einem Fehler führten. Dies beinhaltet entsprechend die Jobs mit folgenden Statuszuständen:

- ERROR
- CANCELLED

Im Unterschied zu den anderen Registerkarten kann hier pro Job individuell bereits in der Tabelle eingesehen werden, welcher Fehler bei der Jobdurchführung aufgetreten ist. Ein Klick auf die abgekürzt dargestellte Fehlermeldung öffnet ein Popup, das die vollständige Fehlermeldung anzeigt.



intran|da TaskManager Dashboard Search Shell SDB IA-Download Lurawave Deutsch English

SDB

Queue Done Error 57 Jobs, 15,036 Image files, 31,222 MB

ID	Status	Goobi process title	Priority	Size	Files	Last change	Error message
5,103		3787_0000_b17490856_0001	10	0 MB	0	2014-05-13 12:36:28	input folder is...
4,917		xwm_b20442609	10	565 MB	816	2014-04-30 11:27:45	Network outage caused...
4,916		xwm_b20442671	10	149 MB	344	2014-04-30 13:08:03	Network outage caused...
4,915		xwm_b20442683	10	107 MB	236	2014-04-30 13:26:25	Network outage caused...
4,914		xwm_b20442580	10	561 MB	710	2014-04-30 13:41:04	Network outage caused...
4,910		xwm_b20442579	10	373 MB	564	2014-04-30 13:45:07	Network outage caused...
4,909		xwm_b20442543	10	331 MB	446	2014-04-30 13:53:23	Network outage caused...
4,906		m5185_b20442403	10	333 MB	532	2014-04-30 13:54:30	Network outage caused...
4,554		m15049_b20421758	10	98 MB	156	2014-04-25 15:18:28	
4,551		m15049_b20422155	10	57 MB	110	2014-04-25 15:17:17	

Cancel all jobs Stop plugin

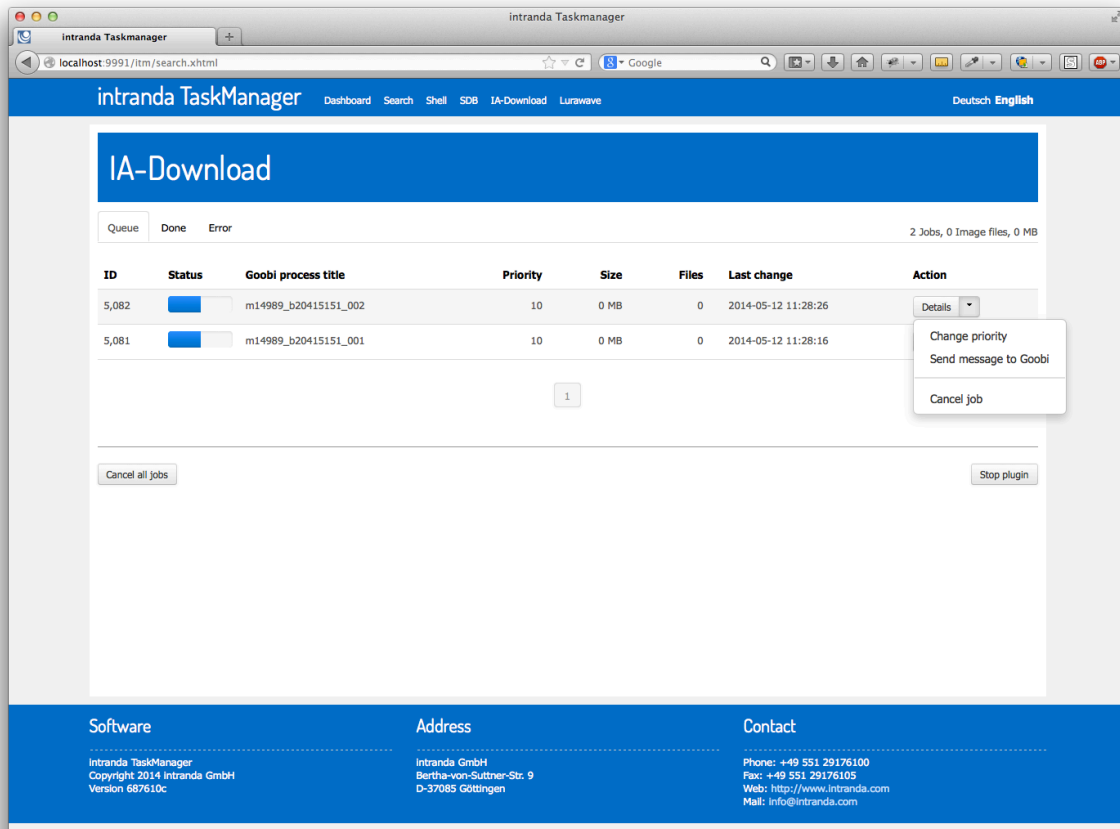
Software
Intran|da TaskManager
Copyright 2014 Intran|da GmbH
Version 687610c

Address
Intran|da GmbH
Bertha-von-Suttner-Str. 9
D-37085 Göttingen

Contact
Phone: +49 551 29176100
Fax: +49 551 29176105
Web: http://www.intran|da.com
Mail: info@intran|da.com

3.3.4 Job-Aktionen

Abhängig von der angezeigten Registerkarte für die Jobs in ihrem jeweiligen Statuszustand, besteht die Möglichkeit unterschiedliche Aktionen pro Job auszuführen. Im Folgenden werden all diese Aktionen aufgelistet und erläutert. Je nach Jobtyp kann dabei die Auswahl der zur Verfügung stehenden Aktionen unterschiedlich sein.



The screenshot shows the intranda TaskManager web interface. The main section is titled "IA-Download" and displays a table of jobs. The table has columns for ID, Status, Goobi process title, Priority, Size, Files, Last change, and Action. Two jobs are listed: ID 5,082 and ID 5,081, both with a status of "Queue" and a priority of 10. A context menu is open over the "Action" column for job 5,081, showing options: "Change priority", "Send message to Goobi", and "Cancel job".

ID	Status	Goobi process title	Priority	Size	Files	Last change	Action
5,082	Queue	m14989_b20415151_002	10	0 MB	0	2014-05-12 11:28:26	Details
5,081	Queue	m14989_b20415151_001	10	0 MB	0	2014-05-12 11:28:16	Change priority Send message to Goobi Cancel job

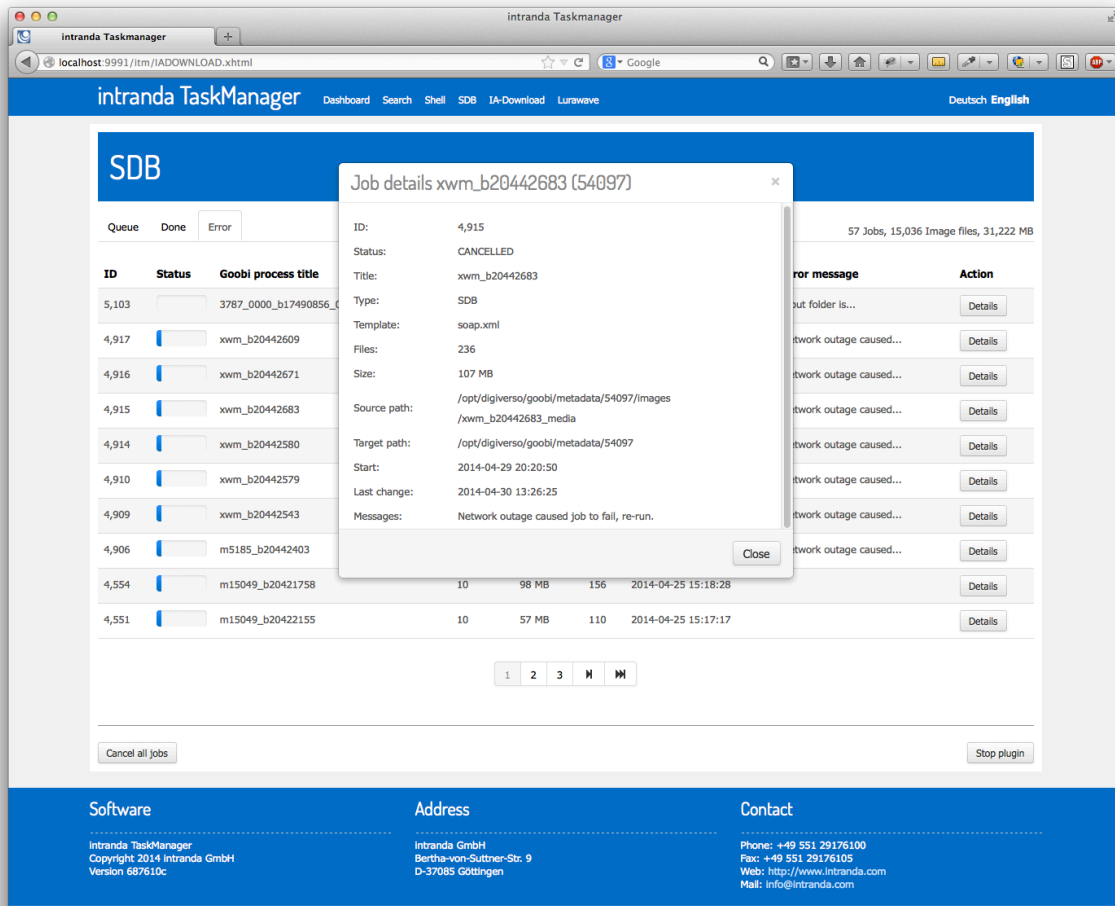
Buttons at the bottom: "Cancel all jobs", "Stop plugin".

Footer information:

- Software:** intranda TaskManager, Copyright 2014 intranda GmbH, Version 687610c
- Address:** intranda GmbH, Bertha-von-Suttner-Str. 9, D-37085 Göttingen
- Contact:** Phone: +49 551 29176100, Fax: +49 551 29176105, Web: http://www.intranda.com, Mail: info@intranda.com

3.3.4.1 Aktion: Details

Die Aktion `Details` öffnet ein Popup, in dem sämtliche Informationen zu dem gewählten Job aufgeführt werden. Neben verschiedenen Identifiern, Verzeichnisangaben, Priorität, Dateigrößen und ähnlichem kann hier auch ein Einblick in die ID gegeben werden, die der Job beispielsweise innerhalb von Goobi besitzt.



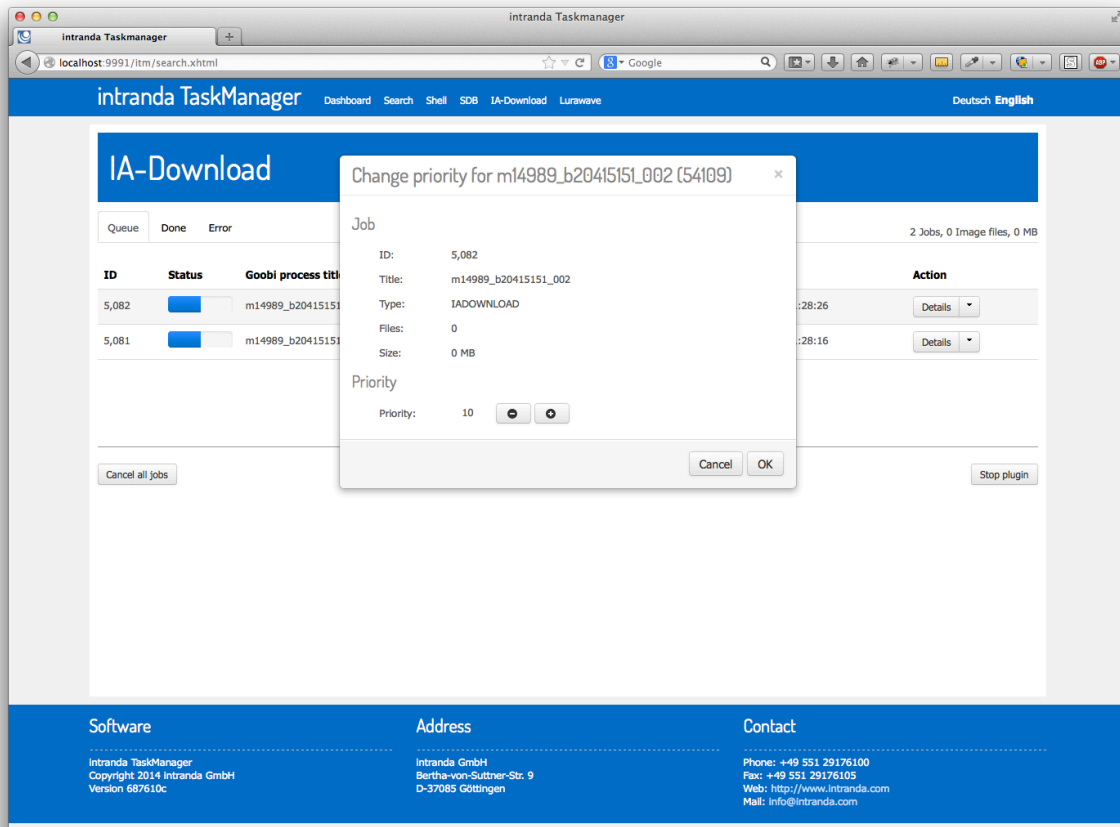
The screenshot shows the intranda TaskManager web interface. A modal window titled "Job details xwm_b20442683 (54097)" is open, displaying the following information:

- ID: 4,915
- Status: CANCELLED
- Title: xwm_b20442683
- Type: SDB
- Template: soap.xml
- Files: 236
- Size: 107 MB
- Source path: /opt/digiverso/goobi/metadata/54097/images/xwm_b20442683_media
- Target path: /opt/digiverso/goobi/metadata/54097
- Start: 2014-04-29 20:20:50
- Last change: 2014-04-30 13:26:25
- Messages: Network outage caused job to fail, re-run.

The background interface shows a table of jobs with columns for Queue, Done, Error, ID, Status, and Goobi process title. The job with ID 4,915 is highlighted. The footer of the interface contains contact information for intranda GmbH.

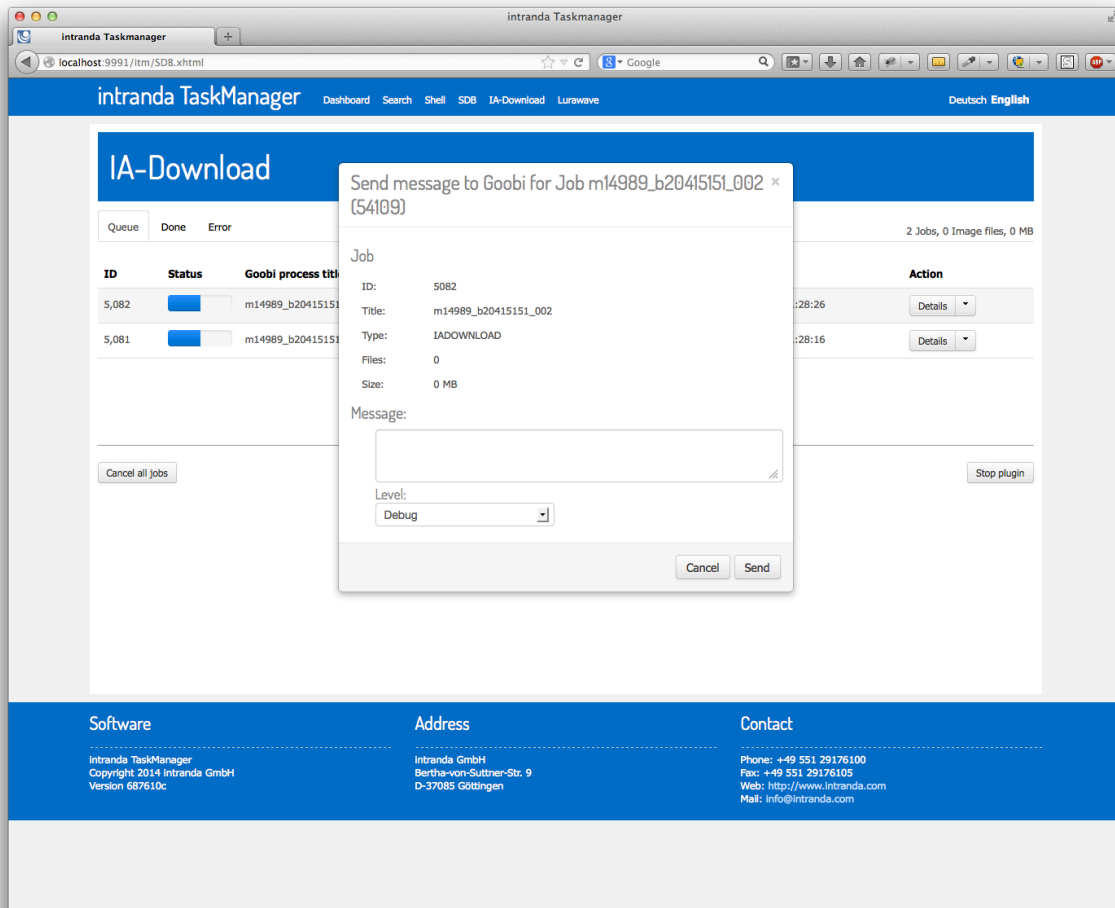
3.3.4.2 Aktion: Priorität ändern

Die Aktion `Priorität ändern` öffnet ein Popup, das die Änderung der Verarbeitungspriorität gestattet. Mittels der angezeigten Plus- und Minus-Buttons kann die Standardpriorität einfach geändert werden und nimmt somit unmittelbar Einfluss auf die weitere Verarbeitungsreihenfolge der Jobs innerhalb der Warteschlange.



3.3.4.3 Aktion: Nachricht an Goobi senden

Die Aktion `Nachricht an Goobi senden` öffnet ein Popup, das das Versenden einer Nachricht an Goobi erlaubt. Die somit versandte Nachricht erscheint abhängig vom gewählten Nachrichtentyp direkt innerhalb des Vorgangslogs und ist somit für alle Benutzer von Goobi sichtbar, die mit dem jeweiligen Vorgang innerhalb des Workflows arbeiten.



3.3.4.4 Aktion: Job abbrechen

Die Aktion `Job abbrechen` öffnet ein Popup, das das Abbrechen des gewählten Jobs erlaubt. Hierbei ist zu beachten, dass Jobs, die gerade in Bearbeitung sind ggf. zunächst interne Verarbeitungen abwarten müssen (z.B. gestartete Kopiervorgänge), bevor der gesamte Abbruch des Jobs vollzogen werden kann.

4 Job-Plugins

Im Folgenden werden alle bestehenden TaskManager-Plugins detailliert beschrieben. Eine Erweiterung um zusätzliche Jobtypen ist aufgrund der Plugin-Architektur jederzeit möglich und kann durch den intranda TaskManager ebenfalls administriert werden.

4.1 JPEG-2000-Generierung mittels Lurawave

Dieses Plugin übergibt Daten zur JPEG-2000-Konvertierung an einen Lurawave Service. Aus Goobi wird ein Kommando wie das folgende aufgerufen:

```
/usr/bin/java -jar /opt/digiverso/taskmanager/bin/TaskClient.jar -itm http://localhost:8080/itm/service -d {imagepath} -s {origpath} -e -i {stepid} -T {processtitle} -t LURAWAVE -n config_rate10.xml -gid {processid}
```

Die einzelnen Parameter im Überblick:

Parameter	Mögliche Goobi Variable	Bedeutung
-itm	http://localhost:8080/itm/service	die URL zur Schnittstelle des intranda TaskManagers
-d, --destination	{imagepath}	Pfad zum images Verzeichnis des Vorgangs
-s, --source	{origpath}	Pfad zum master Verzeichnis des Vorgangs
-e --returnError	-	Wenn angegeben, beendet sich der TaskClient mit einem Fehlercode, um das automatische Fortschreiten im Workflow zu unterbinden
-i	{stepid}	Die ID des Workflowschritts, der den Aufruf startet
-T, --title	{processtitle}	Der Vorgangstitel in Goobi, für den der Aufruf gestartet wird
-t, --jobtype	LURAWAVE	Der Typ des Jobs
-n, --templatename	config_rate10.xml	Name der zu verwendenden Konfigurationsdatei
-p	0 – 10	Priorität zur Verarbeitung dieses Jobs
-gid	{processid}	ID des Goobi-Vorgangs

Nachdem ein Lurawave Job erzeugt wurde, wird ein Ordner im Input-Ordner des Lurawave-Services erzeugt, der lokal (z.B. unter /mnt/lurawave/) eingemounted ist. Dieser Ordner heißt `_temp_(interne ID)_{processtitle}`. In diesen Ordner wird die Konfigurationsdatei sowie alle TIFF-Bilder kopiert. Danach wird der Prefix `_temp_` des Ordernamens entfernt. Dies ist ein Zeichen für Lurawave, mit der Konvertierung zu beginnen.

Der intranda Task Manager überwacht nun den Lurawave-Output Ordner auf das Vorhandensein eines Ordners desselben Namens. Wird der Ordner gefunden, werden die erzeugten JP2 Bilder während der Download-Phase in den entsprechenden Unterordner des `destination`-Ordners verschoben.

4.2 Imagekonvertierung mittels ImageMagick

Die Zusammenarbeit mit dem intranda TaskManager erlaubt Goobi, verschiedene rechenaufwändige Aufgaben auf externe Maschinen auszulagern. Ein im Rahmen von Digitalisierungsprojekten typisches Szenario ist hierfür die Generierung von Imagederivaten. So lassen sich mittels des TaskManager beispielsweise JPG-Images auf einem separaten Server erzeugen und automatisch zurück zu Goobi transferieren, ohne dass der Goobi-Server Rechenkapazität hierfür bereitstellen muss.

Der Aufruf der JPG-Generierung mittels ImageMagick auf einem separaten Server mittels TaskManager muss aus einem Workflowschritt folgendermaßen erfolgen:

```
/usr/bin/java -jar /opt/digiverso/itm/bin/TaskClient.jar -itm http://localhost:8080/itm/service -e -i {stepid} -T {processtitle} -t SHELL -n iii-create_jpeg -s {origpath} -d {processpath} -gid {processid}
```

Die innerhalb dieses Aufrufs zu übergebenen Parameter haben folgende Bedeutungen:

Parameter	Mögliche Goobi Variable	Bedeutung
-itm	http://localhost:8080/itm/service	URL zur Schnittstelle des intranda TaskManagers
-e --returnError	-	Wenn angegeben, beendet sich der TaskClient mit einem Fehlercode, um das automatische Fortschreiten im Workflow zu unterbinden
-i	{stepid}	Die ID des Arbeitsschrittes, der den Aufruf startet
-T, --title	{processtitle}	Der Vorgangstitel in Goobi, für den der Aufruf gestartet wird
-t, --jobtype	SHELL	Der Typ des Jobs
-n, -- templatename	iii-create_jpeg	Name der zuvor erzeugten Konfigurationsdatei
-s, --source	{origpath}	Pfad zum Master-Verzeichnis des Vorgangs
-d, --destination	{processpath}	Pfad zum Hauptverzeichnis des Vorgangs
-p	0 – 10	Priorität zur Verarbeitung dieses Jobs
-gid	{processid}	ID des Goobi-Vorgangs

Nach erfolgter Generierung des Imagederivats werden die erzeugten Dateien durch den TaskManager vollautomatisch in den zugehörigen Goobi-Vorgangsordner verschoben. Darüber hinaus erfolgt neben dem Eintrag über den erfolgreichen Abschluss der Arbeiten in das Vorgangslog auch der Abschluss des automatischen Workflowschritts via WebAPI.

4.3 JPEG-2000-Generierung mittels Open JPEG

Die Zusammenarbeit mit dem intranda TaskManager erlaubt Goobi, die aufwändige JPEG-2000-Generierung auf externe Maschinen auszulagern.

Der Aufruf der JPEG-2000-Generierung mittels Open JPEG mittels TaskManager muss aus einem Workflowschritt folgendermaßen erfolgen:

```
/usr/bin/java -jar /opt/digiverso/itm/bin/TaskClient.jar -itm http://localhost:8080/itm/service -d {processpath} -s {origpath} -e -i {stepid} -T {processtitle} -t JP2 -n template -gid {processid}
```

Die innerhalb dieses Aufrufs zu übergebenen Parameter haben folgende Bedeutungen:

Parameter	Mögliche Goobi Variable	Bedeutung
-itm	http://localhost:8080/itm/service	URL zur Schnittstelle des intranda TaskManagers
-e --returnError	-	Wenn angegeben, beendet sich der TaskClient mit einem Fehlercode, um das automatische Fortschreiten im Workflow zu unterbinden
-i	{stepid}	Die ID des Arbeitsschrittes, der den Aufruf startet
-T, --title	{processtitle}	Der Vorgangstitel in Goobi, für den der Aufruf gestartet wird
-t, --jobtype	JP2	Der Typ des Jobs
-n, -- templatename	template	Name der zuvor erzeugten Konfigurationsdatei
-s, --source	{origpath}	Pfad zum Master-Verzeichnis des Vorgangs
-d, --destination	{processpath}	Pfad zum Hauptverzeichnis des Vorgangs
-p	0 – 10	Priorität zur Verarbeitung dieses Jobs
-gid	{processid}	ID des Goobi-Vorgangs

Nach erfolgter Generierung der JPEG 2000 Dateien werden die erzeugten Dateien durch den TaskManager vollautomatisch in den zugehörigen Goobi-Vorgangsordner verschoben. Darüber hinaus erfolgt neben dem Eintrag über den erfolgreichen Abschluss der Arbeiten in das Vorgangslog auch der Abschluss des automatischen Workflowschritts via WebAPI.

4.4 Validierung von JPEG-2000-Images mittels jpylyzer

Seit der Version 1.9.2 unterstützt Goobi im Rahmen der Workflowverarbeitung die Verwendung von Validierungsplugins. Intranda hat für den Zweck der Validierung von JPEG-2000-Images ein solches Validierungsplugin entwickelt, das vollautomatisch durch Goobi aufgerufen wird, sobald ein Nutzer einen Arbeitsschritt für das Einspielen von JPEG-2000-Images abschließen möchte.

Neben der Verwendung dieses Validierungsplugins auf dem Goobi-Server besteht ebenso die Möglichkeit, JPEG-2000-Images auf einem eigenständigen Server validieren zu lassen. Das Zusammenspiel mit dem TaskManager erlaubt eine solche ausgelagerte Validierung. Hierfür wird zunächst der TaskClient mit den folgenden Parametern aufgerufen:

```
/usr/bin/java -jar /opt/digiverso/itm/bin/TaskClient.jar -itm http://localhost:8080/itm/service -e -i {stepid} -T {processtitle} -t SHELL -n jp2validate -s {tifpath} -d {imagepath} -gid {processid}
```

Die innerhalb des TaskClient-Aufrufs zu übergebenen Parameter haben folgende Bedeutungen:

Parameter	Mögliche Goobi Variable	Bedeutung
-itm	http://localhost:8080/itm/service	URL zur Schnittstelle des intranda TaskManagers
-e --returnError	-	Wenn angegeben, beendet sich der TaskClient mit einem Fehlercode, um das automatische Fortschreiten im Workflow zu unterbinden
-i	{stepid}	Die ID des Arbeitsschrittes, der den Aufruf startet
-T, --title	{processtitle}	Der Vorgangstitel in Goobi, für den der Aufruf gestartet wird
-t, --jobtype	SHELL	Der Typ des Jobs
-n, -- templatename	jp2validate	Name der zuvor erzeugten Konfigurationsdatei
-s, --source	{tifpath}	Pfad zum media-Verzeichnis des Vorgangs
-d, --destination	{imagepath}	Pfad zum image-root-Verzeichnis des Vorgangs
-p	0 – 10	Priorität zur Verarbeitung dieses Jobs
-gid	{processid}	ID des Goobi-Vorgangs

Nach der erfolgten Validierung erzeugt jpylyzer eine ausführliche Ergebnisdatei für jedes einzelne validierte Image mit Details über den Erfolg der Validierung. Eventuell aufgetretene Validierungsprobleme werden innerhalb dieser Ergebnisdateien aufgeführt.

```
<?xml version='1.0' encoding='US-ASCII'?>
<jpylyzer xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:ns0="adobe:ns:meta/"
xmlns:ns3="http://ns.adobe.com/exif/1.0/" xmlns:ns4="http://ns.adobe.com/photoshop/1.0/"
xmlns:ns5="http://ns.adobe.com/tiff/1.0/" xmlns:ns6="http://ns.adobe.com/xap/1.0/"
xmlns:ns7="http://ns.adobe.com/xap/1.0/mm/"
xmlns:ns8="http://ns.adobe.com/xap/1.0/sType/ResourceRef#" xmlns:rdf="http://www.w3.org/1999/02/22-
rdf-syntax-ns#">
<toolInfo>
```

```

<toolName>jpylyzer.py</toolName>
<toolVersion>1.6.0</toolVersion>
</toolInfo>
<fileInfo>
  <fileName>balloon_trunc1.jp2</fileName>

<filePath>/opt/digiverso/goobi/metadata/2282/images/1331281544521_b18022029_tif/balloon_trunc1.jp
2</filePath>
  <fileSizeInBytes>670264</fileSizeInBytes>
  <fileLastModified>Tue Jul 24 03:54:44 2012</fileLastModified>
</fileInfo>
<isValidJP2>False</isValidJP2>
<tests>
  <contiguousCodestreamBox>
    <tileParts>
      <tilePart>
        <foundNextTilePartOrEOC>False</foundNextTilePartOrEOC>
      </tilePart>
    </tileParts>
    <foundEOCMarker>False</foundEOCMarker>
  </contiguousCodestreamBox>
</tests>
<properties>
  <signatureBox />
  <fileTypeBox>
    <br>jp2</br>
    <minV>0</minV>
    <cL>jp2</cL>
  </fileTypeBox>
  <jp2HeaderBox>
    <imageHeaderBox>
      <height>3701</height>
      <width>2717</width>
      <nC>3</nC>
      <bPCSign>unsigned</bPCSign>
      <bPCDepth>8</bPCDepth>
      <c>jpeg2000</c>
      <unkC>yes</unkC>
      <iPR>no</iPR>
    </imageHeaderBox>
    <colourSpecificationBox>
      <meth>Enumerated</meth>
      <prec>0</prec>
      <approx>0</approx>
      <enumCS>sRGB</enumCS>
    </colourSpecificationBox>
  </jp2HeaderBox>
  <uuidInfoBox>
    <uuidListBox>
      <nU>2</nU>
      <uuid>6a706a70-6a70-6a70-6a70-6a706a706a70</uuid>
      <uuid>61626162-6162-6162-6162-616261626162</uuid>
    </uuidListBox>
    <urlBox>
      <version>0</version>
      <loc>http://www.openplanetsfoundation.org/</loc>
    </urlBox>
  </uuidInfoBox>

```

```

<xmlBox>
<ns0:xmpmeta ns0:xmptk="Image::ExifTool 8.29">
  <rdf:RDF>
    <rdf:Description rdf:about="">
      <dc:format>image/jpeg</dc:format>
    </rdf:Description>
    <rdf:Description rdf:about="">
      <ns3:ColorSpace>65535</ns3:ColorSpace>

<ns3:NativeDigest>256,257,258,259,262,274,277,284,530,531,282,283,296,301,318,319,529,532,306,270,27
1,272,305,315,33432;7EF15F60B74B2599BAEDB6749C30991A</ns3:NativeDigest>
    <ns3:PixelXDimension>2717</ns3:PixelXDimension>
    <ns3:PixelYDimension>3701</ns3:PixelYDimension>
  </rdf:Description>
  <rdf:Description rdf:about="">
    <ns4:ColorMode>3</ns4:ColorMode>
    <ns4:History />
  </rdf:Description>
  <rdf:Description rdf:about="">
    <ns5:BitsPerSample>
      <rdf:Seq>
        <rdf:li>8</rdf:li>
      </rdf:Seq>
    </ns5:BitsPerSample>
    <ns5:Compression>1</ns5:Compression>
    <ns5:ImageLength>3701</ns5:ImageLength>
    <ns5:ImageWidth>2717</ns5:ImageWidth>

<ns5:NativeDigest>256,257,258,259,262,274,277,284,530,531,282,283,296,301,318,319,529,532,306,270,27
1,272,305,315,33432;7EF15F60B74B2599BAEDB6749C30991A</ns5:NativeDigest>
    <ns5:Orientation>1</ns5:Orientation>
    <ns5:PhotometricInterpretation>2</ns5:PhotometricInterpretation>
    <ns5:PlanarConfiguration>1</ns5:PlanarConfiguration>
    <ns5:ResolutionUnit>2</ns5:ResolutionUnit>
    <ns5:SamplesPerPixel>4</ns5:SamplesPerPixel>
    <ns5:Software>Adobe Photoshop CS3 Windows</ns5:Software>
    <ns5:XResolution>72/1</ns5:XResolution>
    <ns5:YCbCrSubSampling>1 1</ns5:YCbCrSubSampling>
    <ns5:YResolution>72/1</ns5:YResolution>
  </rdf:Description>
  <rdf:Description rdf:about="">
    <ns6:CreateDate>2008-07-19T16:14:14-07:00</ns6:CreateDate>
    <ns6:CreatorTool>Adobe Photoshop CS3 Windows</ns6:CreatorTool>
    <ns6:MetadataDate>2008-07-19T16:14:14-07:00</ns6:MetadataDate>
    <ns6:ModifyDate>2008-07-19T16:14:14</ns6:ModifyDate>
  </rdf:Description>
  <rdf:Description rdf:about="">
    <ns7:DerivedFrom rdf:parseType="Resource">
      <ns8:instanceID>uuid:AC48AD726754DD11BA6DEACED58C77FA</ns8:instanceID>
    </ns7:DerivedFrom>
    <ns7:DocumentID>uuid:6200E56DE155DD118C3CED023B237FE5</ns7:DocumentID>
    <ns7:InstanceID>uuid:6300E56DE155DD118C3CED023B237FE5</ns7:InstanceID>
  </rdf:Description>
</rdf:RDF>
</ns0:xmpmeta>
</xmlBox>
<contiguousCodestreamBox>
  <size>

```

```

<lsiz>47</lsiz>
<rsiz>ISO/IEC 15444-1</rsiz>
<xsiz>2717</xsiz>
<ysiz>3701</ysiz>
<xOsiz>0</xOsiz>
<yOsiz>0</yOsiz>
<xTsiz>1024</xTsiz>
<yTsiz>1024</yTsiz>
<xTOsiz>0</xTOsiz>
<yTOsiz>0</yTOsiz>
<numberOfTiles>12</numberOfTiles>
<csiz>3</csiz>
<ssizSign>unsigned</ssizSign>
<ssizDepth>8</ssizDepth>
<xRsiz>1</xRsiz>
<yRsiz>1</yRsiz>
<ssizSign>unsigned</ssizSign>
<ssizDepth>8</ssizDepth>
<xRsiz>1</xRsiz>
<yRsiz>1</yRsiz>
<ssizSign>unsigned</ssizSign>
<ssizDepth>8</ssizDepth>
<xRsiz>1</xRsiz>
<yRsiz>1</yRsiz>
</siz>
<cod>
<lcod>18</lcod>
<precincts>yes</precincts>
<sop>yes</sop>
<eph>yes</eph>
<order>RPCL</order>
<layers>6</layers>
<multipleComponentTransformation>yes</multipleComponentTransformation>
<levels>5</levels>
<codeBlockWidth>64</codeBlockWidth>
<codeBlockHeight>64</codeBlockHeight>
<codingBypass>no</codingBypass>
<resetOnBoundaries>no</resetOnBoundaries>
<termOnEachPass>no</termOnEachPass>
<vertCausalContext>no</vertCausalContext>
<predTermination>no</predTermination>
<segmentationSymbols>yes</segmentationSymbols>
<transformation>9-7 irreversible</transformation>
<precinctSizeX>128</precinctSizeX>
<precinctSizeY>128</precinctSizeY>
<precinctSizeX>128</precinctSizeX>
<precinctSizeY>128</precinctSizeY>
<precinctSizeX>128</precinctSizeX>
<precinctSizeY>128</precinctSizeY>
<precinctSizeX>128</precinctSizeX>
<precinctSizeY>128</precinctSizeY>
<precinctSizeX>256</precinctSizeX>
<precinctSizeY>256</precinctSizeY>
<precinctSizeX>256</precinctSizeX>
<precinctSizeY>256</precinctSizeY>
</cod>
<qcd>
<lqcd>35</lqcd>

```

```

<qStyle>scalar expounded</qStyle>
<guardBits>2</guardBits>
<mu>1816</mu>
<epsilon>13</epsilon>
<mu>1770</mu>
<epsilon>13</epsilon>
<mu>1770</mu>
<epsilon>13</epsilon>
<mu>1724</mu>
<epsilon>13</epsilon>
<mu>1792</mu>
<epsilon>12</epsilon>
</qcd>
<com>
  <lcom>17</lcom>
  <rcom>ISO/IEC 8859-15 (Latin)</rcom>
  <comment>Jpylyzer demo</comment>
</com>
<tileParts>
  <tilePart>
    <sot>
      <lsot>10</lsot>
      <isot>0</isot>
      <psot>67161</psot>
      <tpsot>0</tpsot>
      <tnsot>1</tnsot>
    </sot>
  </tilePart>
  <tilePart>
    <sot>
      <lsot>10</lsot>
      <isot>1</isot>
      <psot>99064</psot>
      <tpsot>0</tpsot>
      <tnsot>1</tnsot>
    </sot>
  </tilePart>
  <tilePart>
    <sot>
      <lsot>10</lsot>
      <isot>2</isot>
      <psot>36130</psot>
      <tpsot>0</tpsot>
      <tnsot>1</tnsot>
    </sot>
  </tilePart>
  <tilePart>
    <sot>
      <lsot>10</lsot>
      <isot>3</isot>
      <psot>56048</psot>
      <tpsot>0</tpsot>
      <tnsot>1</tnsot>
    </sot>
  </tilePart>
  <tilePart>
    <sot>
      <lsot>10</lsot>

```

```

<isot>4</isot>
<psot>140022</psot>
<tpsot>0</tpsot>
<tnsot>1</tnsot>
</sot>
</tilePart>
<tilePart>
<sot>
<lsot>10</lsot>
<isot>5</isot>
<psot>24008</psot>
<tpsot>0</tpsot>
<tnsot>1</tnsot>
</sot>
</tilePart>
<tilePart>
<sot>
<lsot>10</lsot>
<isot>6</isot>
<psot>46691</psot>
<tpsot>0</tpsot>
<tnsot>1</tnsot>
</sot>
</tilePart>
<tilePart>
<sot>
<lsot>10</lsot>
<isot>7</isot>
<psot>62671</psot>
<tpsot>0</tpsot>
<tnsot>1</tnsot>
</sot>
</tilePart>
<tilePart>
<sot>
<lsot>10</lsot>
<isot>8</isot>
<psot>26306</psot>
<tpsot>0</tpsot>
<tnsot>1</tnsot>
</sot>
</tilePart>
<tilePart>
<sot>
<lsot>10</lsot>
<isot>9</isot>
<psot>45614</psot>
<tpsot>0</tpsot>
<tnsot>1</tnsot>
</sot>
</tilePart>
<tilePart>
<sot>
<lsot>10</lsot>
<isot>10</isot>
<psot>38428</psot>
<tpsot>0</tpsot>
<tnsot>1</tnsot>

```

```

</sot>
</tilePart>
<tilePart>
  <sot>
    <lsot>10</lsot>
    <isot>11</isot>
    <psot>25064</psot>
    <tpsot>0</tpsot>
    <tnsot>1</tnsot>
  </sot>
</tilePart>
</tileParts>
</contiguousCodestreamBox>
<compressionRatio>45.01</compressionRatio>
</properties>
</jpylyzer>

```

Nach Abschluss der Validierung analysiert der TaskManager sämtliche dieser erzeugten Ergebnisdateien und aktualisiert vollautomatisch das Vorgangslog des Goobi-Vorgangs mit den Informationen über die Validierung.

Im Falle einer erfolgreichen Validierung aller Images eines Vorgangs schließt der TaskManager den zugehörigen Arbeitsschritt in Goobi mittels Aufruf der WebApi ab.

Unabhängig von dem Ergebnis werden die Ergebnisdateien jedes jpylyzer-Durchlaufs innerhalb des Ordners `validation` im Goobi-Vorgangsordner gespeichert. Ein beispielhafter Pfad für die Speicherung von jpylyzer-Ergebnisdateien ist z.B. dieser:

```
/opt/digiverso/goobi/metadata/2282/validation/2012-20-05_11-20-01_jpylyzer/
```

4.5 Volltexterkennung - OCR

Der TaskClient Aufruf für einen OCR-Job ist den Aufrufen für andere Jobtypen sehr ähnlich:

```
/usr/bin/java -jar /opt/digiverso/itm/bin/TaskClient.jar -itm http://localhost:8080/itm/service -s {tifpath} -d {processpath} -e -gid {processid} -i {stepid} -T {processtitle} -f {process.Schrifttyp} -n template.xml -l ${metas.DocLanguage} -st intranda-abbyy
```

Parameter	Mögliche Goobi Variable	Bedeutung
-itm	http://localhost:8080/itm/service	URL zur Schnittstelle des intranda TaskManagers
-s, --source	{tifpath}	Pfad zum media Verzeichnis des Vorgangs
-d, --destination	{processpath}	Pfad zum Hauptverzeichnis des Vorgangs
-e --returnError	-	Wenn angegeben, beendet sich der TaskClient mit einem Fehlercode, um das automatische Fortschreiten im Workflow zu unterbinden
-i	{stepid}	Die ID des Arbeitsschrittes, der den Aufruf startet
-T, --title	{processtitle}	Der Vorgangstitel in Goobi, für den der Aufruf gestartet wird
-t, --jobtype	OCR	Der Typ des Jobs
-n, -- templatename	template.xml	Name der zuvor erzeugten Konfigurationsdatei
-f, --fontType	bspw. {product.Schrifttyp} oder {process.Schrifttyp}	Schrifttyp, unterstützt werden Fraktur und Antiqua
-l, --language	\${metas.DocLanguage}	Sprache des zu erkennenden Textes
-st	intranda-abbyy	OCR Servertyp
-p	0 – 10	Priorität zur Verarbeitung dieses Jobs
-gid	{processid}	ID des Goobi-Vorgangs

Wenn ein neuer OCR Job im intranda TaskManager eingegangen ist, werden je nach Anzahl der zu erkennenden Bilder verschiedene Tickets erzeugt. Typische Ticketgrößen sind maximal 500 Images bei maximal 10 GB Speichergröße der Bilder. Jedes Ticket besteht aus einer Liste der Bilddateien sowie aus einer Angabe, welche OCR Ausgabeformate erzeugt werden sollen.

Die erzeugten Tickets werden einzeln abgearbeitet. Der intranda TaskManager lädt das Ticket und die dazugehörigen Bilder in den OCR Input Ordner. Dieser Ordner kann entweder ein lokaler Ordner, ein gemounteter Ordner oder ein WebDav Ordner sein. Die OCR Software überwacht diesen Ordner und fängt nach vollständiger Übertragung der Daten mit der Texterkennung an.

Der intranda TaskManager überwacht nun den Error-Ordner auf Fehler sowie den Control-Ordner der OCR Software auf Ergebnisberichte. Ist eine entsprechende Control-Datei eingegangen, werden im Output-Ordner alle zu diesem Ticket gehörenden Daten zusammengestellt und heruntergeladen. Sie werden im Unterordner `ocr` des jeweiligen Vorgangsordners in einzelne Unterordner anhand ihrer Datei-Endung gespeichert.

4.6 Archivierung mittels Tessella Preservica

Der Preservica ingest besteht aus drei Teilen. Die beiden ersten Teile des Preservica ingest übernimmt ein Programm namens `SoapGenerator`. Dieses Programm kopiert im ersten Schritt die Daten, die ingestet werden sollen an die entsprechenden Ordner, damit Preservica diese lesen kann. In einem zweiten Schritt wird eine Konfigurationsdatei `soap.xml` anhand von Datenbank-Informationen über den Prozess erzeugt. In dieser Konfigurationsdatei stehen Informationen, die zur Generierung des SOAP Calls notwendig sind. Hier werden die für Preservica notwendigen Accountdaten und die notwendigen Parameter wie `securityTag`, `CollectionName` und `schemaName` hinterlegt.

```
/usr/bin/java -jar /opt/digiverso/goobi/scripts/SoapGenerator.jar -sd {tifpath} -fn {processtitle} -mp {importpath} -p {processid} -cf /opt/digiverso/goobi/scripts/soapGenerator.properties
```

Parameter	Mögliche Goobi Variable	Bedeutung
-sd, --sourceDirectory	{tifpath}	Ausgangsmaterial, das ingestet werden soll
-fn, --foldername	{processtitle}	Name, wie die Ordner während des SDB ingests heißen sollen, notwendig zur eindeutigen Zuordnung
-mp, --metadataPath	{importpath}	Pfad, an den die AMD Metadaten zurückgespielt werden sollen
-p, --processId	{processid}	ProzessID, wird zur Datenbank-Suche benötigt
-cf, --configFile	/opt/digiverso/goobi/scripts/soapGenerator.properties	Pfad zur Konfigurationsdatei des SoapGenerators, enthält Pfadangaben, Datenbank-Zugangsdaten etc.

Nachdem der SoapGenerator seine Arbeit getan hat, wird der TaskClient mit folgenden Parametern aufgerufen:

```
/usr/bin/java -jar /opt/digiverso/taskmanager/bin/TaskClient.jar -itm http://localhost:8080/itm/service -d {processpath} -s {tifpath} -e -i {stepid} -T {processtitle} -t SDB -n soap.xml -gid {processid}
```

Parameter	Mögliche Goobi Variable	Bedeutung
-itm	http://localhost:8080/itm/service	URL zur Schnittstelle des intranda TaskManagers
-d, --destination	{processpath}	Pfad zum Hauptverzeichnis des Vorgangs
-s, --source	{tifpath}	Pfad zum media Verzeichnis des Vorgangs
-e --returnError	-	Wenn angegeben, beendet sich der TaskClient mit einem Fehlercode, um das automatische Fortschreiten im Workflow zu unterbinden
-i	{stepid}	Die ID des Workflowschritts, der den Aufruf startet
-T, --title	{processtitle}	Der Vorgangstitel in Goobi, für den der Aufruf gestartet wird

-t, --jobtype	SDB	Der Typ des Jobs
-n, -- templatename	soap.xml	Name der zuvor erzeugten Konfigurationsdatei
-p	0 – 10	Priorität zur Verarbeitung dieses Jobs
-gid	{processid}	ID des Goobi-Vorgangs

Dieser Aufruf erzeugt einen Preservica Job im intranda TaskManager. Dieser Typ unterscheidet sich von den anderen Jobtypen. Der Upload besteht aus einem SOAP Call an das Preservica. Der SOAP Call wird aus den Werten in der Vorgang-spezifischen Konfigurationsdatei `soap.xml` gebildet.

Der intranda TaskManager überwacht anschließend den AMD-Output Ordner auf das Vorhandensein einer XML-Datei. Wird diese Datei gefunden, wird sie in den entsprechenden Vorgangsordner verschoben.

4.7 Internet Archive Download

Ein effizientes Harvesting des Internet Archive wird in Goobi ebenfalls durch die Zusammenarbeit mit dem intranda TaskManager möglich. Hierzu müssen zwei Voraussetzungen zunächst erfüllt sein. Das WebDavCommunicator-Plugin muss in dem Plugin-Ordner des TaskManagers vorliegen. Standardmäßig handelt es sich hierbei um folgenden Pfad:

```
/opt/digiverso/itm/plugins/WebDavCommunicator-<version>.jar
```

Das Plugin selbst wird ebenfalls in den Plugin-Ordner kopiert. Standardmäßig handelt es sich hierbei um folgenden Pfad:

```
/opt/digiverso/itm/plugins/IADownloadPlugin-<version>.jar
```

Der Aufruf des Internet-Archive-Harvestings wird innerhalb von Goobi in einem Workflowschritt folgendermaßen konfiguriert:

```
/usr/bin/java -jar /opt/digiverso/itm/bin/TaskClient.jar -itm http://localhost/itm/service -s  
http://archive.org/download/${meta.CatalogIDDigital} -d {imagepath}/source/ -n template -e -i {stepid} -T  
{processtitle} -gid {processid} -t IADOWNLOAD
```

Die innerhalb dieses Aufrufs zu übergebenen Parameter haben folgende Bedeutungen:

Parameter	Mögliche Goobi Variable	Bedeutung
-itm	http://localhost:8080/itm/service	URL zur Schnittstelle des intranda TaskManagers
-e --returnError	-	Wenn angegeben, beendet sich der TaskClient mit einem Fehlercode, um das automatische Fortschreiten im Workflow zu unterbinden
-i	{stepid}	Die ID des Arbeitsschrittes, der den Aufruf startet
-T, --title	{processtitle}	Der Vorgangstitel in Goobi, für den der Aufruf gestartet wird
-t, --jobtype	IADOWNLOAD	Der Typ des Jobs
-n, --templatename	template	Name der zuvor erzeugten Konfigurationsdatei
-s, --source	http://archive.org/download/\${meta.CatalogIDDigital}	Pfad zur URL des Werkes im Internet Archive
-d, --destination	{processpath}	Pfad zum Hauptverzeichnis des Vorgangs
-p	0 – 10	Priorität zur Verarbeitung dieses Jobs
-gid	{processid}	ID des Goobi-Vorgangs

Dem Plugin wird eine URL zu einem Band im Internet Archive übergeben. Von dort werden die folgenden Daten in das jeweils übergebene Zielverzeichnis heruntergeladen:

```
scandata.xml  
marc.xml  
abbyy.gz  
jp2.zip
```

Schlägt ein Download fehl, wird zunächst die Priorität des Jobs herabgesetzt und versucht den nächsten Job herunterzuladen. Diese fehlerhaften Versuche werden an Goobi gemeldet und dort im Vorgangslog als Warnung dargestellt.

Jeder Job hat zu Beginn seiner Laufzeit im TaskManager die Priorität 10. Je höher die Priorität, desto eher wird ein Job abgearbeitet. Ist die Priorität eines Jobs bei 0 angelangt, wird der Download noch erneute vier Mal neu versucht. Schlagen alle Versuche des Downloads fehl, wird der Job als fehlerhaft gewertet und mit einer Fehlermeldung an Goobi zurückgesendet.

4.8 ALTO Generierung aus ABBYY-xml mittels intranda AltoConvert

Die Zusammenarbeit mit dem intranda TaskManager erlaubt Goobi, die aufwändige ALTO Generierung mit der Erkennung von Lauftiteln und Seitenzahlen auf externe Maschinen auszulagern. Der Aufruf der ALTO-Generierung mittels `intranda AltoConvert` muss aus einem Workflowschritt folgendermaßen erfolgen:

```
/usr/bin/java -jar /opt/digiverso/itm/bin/TaskClient.jar -itm http://localhost:8080/itm/service -s {tifpath} -d {processpath}/ocr/{processtitle}_alto/ -e -gid {processid} -i {stepid} -T {processtitle} -t ALTOCONVERT -n altoconvert
```

Die innerhalb dieses Aufrufs zu übergebenen Parameter haben folgende Bedeutungen:

Parameter	Mögliche Goobi Variable	Bedeutung
-itm	http://localhost:8080/itm/service	URL zur Schnittstelle des intranda TaskManagers
-e --returnError	-	Wenn angegeben, beendet sich der TaskClient mit einem Fehlercode, um das automatische Fortschreiten im Workflow zu unterbinden
-i	{stepid}	Die ID des Arbeitsschrittes, der den Aufruf startet
-T, --title	{processtitle}	Der Vorgangstitel in Goobi, für den der Aufruf gestartet wird
-t, --jobtype	ALTOCONVERT	Der Typ des Jobs
-n, --templatename	altoconvert	Name der zuvor erzeugten Konfigurationsdatei
-s, --source	{tifpath}	Pfad zum tif-Verzeichnis des Vorgangs
-d, --destination	{processpath}/ocr/{processtitle}_alto /	Pfad zum zu erzeugenden ALTO-Verzeichnis im Vorgang
-p	0 – 10	Priorität zur Verarbeitung dieses Jobs
-gid	{processid}	ID des Goobi-Vorgangs

Nach der Generierung der ALTO-Dateien werden die erzeugten Dateien durch den TaskManager vollautomatisch in das an den TaskClient übergebene Zielverzeichnis verschoben. Darüber hinaus erfolgt neben dem Eintrag über den erfolgreichen Abschluss der Arbeiten in das Vorgangslog auch der Abschluss des automatischen Workflowschritts via WebAPI.

4.9 Download von mehrbändigen Werken aus dem Internet Archive mittels intranda InternetArchiveParser.jar (J-Download)

Die Zusammenarbeit mit dem intranda TaskManager erlaubt Goobi, den Download von mehrbändigen Werken vom Internet Archive in eine externe, geregelt ablaufende Warteschlange auszulagern.

Der Aufruf des Downloads muss aus einem Workflowschritt folgendermaßen erfolgen:

```
/usr/bin/java -jar /opt/digiverso/itm/bin/TaskClient.jar -itm http://localhost:8080/itm/service -s {tifpath} -d {processpath} -e -gid {processid} -i {stepid} -T {processtitle} -t JDOWNLOAD -n jdownload
```

Die innerhalb dieses Aufrufs zu übergebenen Parameter haben folgende Bedeutungen:

Parameter	Mögliche Goobi Variable	Bedeutung
-itm	http://localhost:8080/itm/service	URL zur Schnittstelle des intranda TaskManagers
-e --returnError	-	Wenn angegeben, beendet sich der TaskClient mit einem Fehlercode, um das automatische Fortschreiten im Workflow zu unterbinden
-i	{stepid}	Die ID des Arbeitsschrittes, der den Aufruf startet
-T, --title	{processtitle}	Der Vorgangstitel in Goobi, für den der Aufruf gestartet wird
-t, --jobtype	JDOWNLOAD	Der Typ des Jobs
-n, --templatename	jdownload	Name der zuvor erzeugten Konfigurationsdatei
-s, --source	{tifpath}	Pfad zum tif-Verzeichnis des Vorgangs
-d, --destination	{processpath}	Pfad zum Goobi-Prozess Verzeichnis
-p	0 – 10	Priorität zur Verarbeitung dieses Jobs
-gid	{processid}	ID des Goobi-Vorgangs

Nach dem Download der Dateien erfolgt neben dem Eintrag über den erfolgreichen Abschluss der Arbeiten in das Vorgangslog auch der Abschluss des automatischen Workflowschritts via WebAPI.

4.10 Import mehrbändiger Werke aus dem Internet Archive nach Goobi mittels intranda InternetArchiveParser.jar (J-Import)

Die Zusammenarbeit mit dem intranda TaskManager erlaubt Goobi, den Import von mehrbändigen, aus dem Internet Archive heruntergeladenen Werken in eine externe, geregelt ablaufende Warteschlange auszulagern.

Der Aufruf des Imports muss aus einem Workflowschritt folgendermaßen erfolgen:

```
/usr/bin/java -jar /opt/digiverso/itm/bin/TaskClient.jar -itm http://localhost:8080/itm/service -s {tifpath} -d {processpath} -e -gid {processid} -i {stepid} -T {processtitle} -t JIMPORT -n jimport
```

Die innerhalb dieses Aufrufs zu übergebenen Parameter haben folgende Bedeutungen:

Parameter	Mögliche Goobi Variable	Bedeutung
-itm	http://localhost:8080/itm/service	URL zur Schnittstelle des intranda TaskManagers
-e --returnError	-	Wenn angegeben, beendet sich der TaskClient mit einem Fehlercode, um das automatische Fortschreiten im Workflow zu unterbinden
-i	{stepid}	Die ID des Arbeitsschrittes, der den Aufruf startet
-T, --title	{processtitle}	Der Vorgangstitel in Goobi, für den der Aufruf gestartet wird
-t, --jobtype	JIMPORT	Der Typ des Jobs
-n, --templatename	jimport	Name der zuvor erzeugten Konfigurationsdatei
-s, --source	{tifpath}	Pfad zum tif-Verzeichnis des Vorgangs
-d, --destination	{processpath}	Pfad zum Goobi-Prozess Verzeichnis
-p	0 – 10	Priorität zur Verarbeitung dieses Jobs
-gid	{processid}	ID des Goobi-Vorgangs

Nach dem Import der Dateien erfolgt neben dem Eintrag über den erfolgreichen Abschluss der Arbeiten in das Vorgangslog auch der Abschluss des automatischen Workflowschritts via WebAPI.

4.11 Validierung von TIFF Images mittels JHOVE

Dieses Plugin übergibt Daten zur Validierung an das JSTOR/Harvard Object Validation Environment (JHOVE) Framework. Mit diesem Framework kann überprüft werden, ob die vorliegenden Bilder vollständig sind, keine Bitstream Fehler enthalten und den zu verwendenden Parametern entsprechen.

Der Aufruf des Downloads muss aus einem Workflowschritt folgendermaßen erfolgen:

```
/usr/bin/java -jar /opt/digiverso/itm/bin/TaskClient.jar -itm http://localhost:8080/itm/service -s {origpath} -d ${meta.CatalogIDDigital} -e -i {stepid} -T {processtitle} -t TIFVALIDATION -n tiff.template -gid {processid}
```

Die innerhalb des TaskClient-Aufrufs zu übergebenen Parameter haben folgende Bedeutungen:

Parameter	Mögliche Goobi Variable	Bedeutung
-itm	http://localhost:8080/itm/service	URL zur Schnittstelle des intranda TaskManagers
-e --returnError	-	Wenn angegeben, beendet sich der TaskClient mit einem Fehlercode, um das automatische Fortschreiten im Workflow zu unterbinden
-i	{stepid}	Die ID des Arbeitsschrittes, der den Aufruf startet
-T, --title	{processtitle}	Der Vorgangstitel in Goobi, für den der Aufruf gestartet wird
-t, --jobtype	TIFVALIDATION	Der Typ des Jobs
-n, --templatename	tiff.template	Name der zu verwendenden Konfigurationsdatei
-s, --source	{origpath}	Pfad zum master-Verzeichnis des Vorgangs
-d, --destination	{meta.CatalogIDDigital}	Identifizier des Vorgangs
-p	0 – 10	Priorität zur Verarbeitung dieses Jobs
-gid	{processid}	ID des Goobi-Vorgangs

Für die Validierung prüft JHOVE als erstes, ob das übergebene Bild wohl-geformt und valide ist, also ob es den Spezifikationen des Formates entspricht.

Anschließend können die Parameter Kompression, Auflösung, Farbtiefe und Farbraum überprüft werden. Dazu werden in der Konfigurationsdatei des Plugins die erlaubten Werte für jeden Parameter aufgelistet. Wenn für einen Parameter verschiedene Werte äquivalent sind, können diese wiederholt definiert werden:

```
compression_value=1
resolution_value=300
colorspace_value=2
color_depth_value=8,8,8
color_depth_value=8
```


Bei jedem Parameter wird geprüft, ob mindestens einer der konfigurierten Werte mit dem tatsächlichen Wert übereinstimmt. Nur wenn dies der Fall ist, entspricht das Image den sicherzustellenden Projekteinstellungen.

4.12 Bildanalyse mittels des intranda LayoutWizzards

Das Goobi-Plugin LayoutWizzard erlaubt eine automatische Bildanalyse von Digitalisaten zur Bildaufbereitung, zusammen mit einer Computer-unterstützten Validierung der Ergebnisse. Mit Hilfe des intranda TaskManagers kann die rechen- und zeitaufwändige automatische Analyse, sowie die abschließende Speicherung der aufbereiteten Digitalisate auf externe Maschinen ausgelagert werden.

Dieses Plugin unterteilt sich in zwei Komponenten: `LayoutWizzardAnalyse` und `LayoutWizzardSave`. Erstere übernimmt die eigentliche Bildanalyse, während letztere die aufbereiteten Bilder in einem Ausgabeordner (üblicherweise der Derivate-Ordner des Goobi-Vorganges) abspeichert. Beide Komponenten werden im Folgenden kurz separat beschrieben.

4.12.1 LayoutWizzardAnalyse

Der Aufruf der Bildanalyse mittels des LayoutWizzards muss aus einem Workflowschritt folgendermaßen erfolgen:

```
/usr/bin/java -jar /opt/digiverso/itm/bin/TaskClient.jar -itm http://localhost:8080/itm/service -t
LayoutWizzardAnalyse -n Left-To-Right,individual-pages -s {origpath} -d {tiffpath} -gid {processid} -i {stepid}
-T {processtitle} -e
```

Wenn vorhanden, übernimmt dieser Schritt Analyseparameter aus der Analysedatei im Vorgangsordner des Goobi-Vorganges:

```
imageData.ser
```

Andernfalls verwendet er die Parameter aus der in der Plugin-Konfiguration angegebenen LayoutWizzard-Konfigurationsdatei (siehe Abschnitt [Konfiguration](#)).

Der Schritt berechnet für alle Bilder im Eingabeordner (`-s`) die Drehung der Seite, Position der Seitenkanten und der Buchfalz. Diese Daten werden anschließend in oben genannter Analysedatei für spätere Workflow-Schritte gespeichert.

Sollte bei der Analyse ein Fehler auftreten, wird der Vorgang abgebrochen und die Fehlermeldung an Goobi gesendet.

4.12.2 LayoutWizzardSave

Das Speichern der aufbereiteten Bilder mittels des LayoutWizzards muss aus einem Workflowschritt folgendermaßen erfolgen:

```
/usr/bin/java -jar /opt/digiverso/itm/bin/TaskClient.jar -itm http://localhost:8080/itm/service -t
LayoutWizzardSave -n none -s {origpath} -d {tiffpath} -gid {processid} -i {stepid} -T {processtitle} -e
```

Dieser Schritt liest die Analyseergebnisse der Analysedatei im Vorgangsordner des Goobi-Vorganges aus:

```
imageData.ser
```

Diese muss in einem vorhergehenden Analyseschritt, entweder in Goobi oder ebenfalls im TaskManager, erzeugt worden sein. Anschließend werden gerade-gerichtete und zugeschnittene Versionen der Ausgangsbilder im Ausgabeordner (-d) als unkomprimierte Tiff-Dateien gespeichert. Sollte es dabei, z.B. wegen ungültiger Analysewerte, zu einem Fehler kommen, wird der Vorgang abgebrochen und die Analysedatei gelöscht, damit in einem Korrekturschritt neue Werte erzeugt werden können. Ansonsten wird die Analysedatei für eine eventuelle Wiederholung des Schrittes beibehalten.

Die für beide Komponenten zu übergebenden Parameter haben folgende Bedeutungen:

Parameter	Mögliche Goobi Variable	Bedeutung
-itm	http://localhost:8080/itm/service	URL zur Schnittstelle des intranda TaskManagers
-e --returnError	-	Wenn angegeben, beendet sich der TaskClient mit einem Fehlercode, um das automatische Fortschreiten im Workflow zu unterbinden
-i	{stepid}	Die ID des Arbeitsschrittes, der den Aufruf startet
-T, --title	{processtitle}	Der Vorgangstitel in Goobi, für den der Aufruf gestartet wird
-t, --jobtype	LayoutWizzardAnalyse LayoutWizzardSave	Der Typ des Jobs
-n, --templatename	Left-To-Right,individual-pages	Komma-separierte Liste von Bearbeitungsparametern (siehe Abschnitt <code>Templates</code>) für LayoutWizzardAnalyse. Ohne Bedeutung für LayoutWizzardSave.
-s, --source	{origpath}	Pfad zum Verzeichnis des zu bearbeitenden Bilder
-d, --destination	{tiffpath}	Pfad zum Zielverzeichnis, in dem die bearbeiteten Bilder gespeichert werden sollen (nur für LayoutWizzardSave erforderlich)
-p	0 – 10	Priorität zur Verarbeitung dieses Jobs
-gid	{processid}	ID des Goobi-Vorgangs

4.12.3 Templates

Der Analyseschritt akzeptiert zwei Parameter als Template, die z.B. aus Vorgangseigenschaften des Goobi-Vorgangs ausgelesen werden können. Sie können dem Parameter `-t` angefügt werden. Die zur Verfügung stehenden Parameter sind in folgender Tabelle aufgeführt.

Parameter	Mögliche Werte	Bedeutung
Orientation	Left-To-Right	Das Digitalisat wurde von links nach rechts digitalisiert, also in Leserichtung der meisten europäischen Sprachen.
	Right-To-Left	Das Digitalisat wurde von rechts nach links digitalisiert, also in Leserichtung von semitischen Sprachen wie Arabisch oder Hebräisch.
PageMode	individual-pages	Das Digitalisat wurde ausschließlich in Einzelseiten gescannt.
	double-pages	Das Digitalisat wurde ausschließlich in Doppelseiten gescannt.
	double-pages-with-individual-covers	Das Digitalisat wurde in Doppelseiten gescannt, abgesehen von der jeweils ersten und letzten Seite (üblicherweise den Buchdeckeln).

Werden Parameter auf diese Weise übergeben, müssen alle möglichen Parameter als Komma-separierte Liste in der angegebenen Reihenfolge übergeben werden, also immer in der folgenden Form:

```
<Orientation>,<PageMode>
```

4.12.4 Konfiguration

Das Plugin verwendet eine Konfigurationsdatei, die im `config`-Ordner des TaskManager Plugins liegen muss, also üblicherweise unter folgendem Pfad:

```
/opt/digiverso/itm/plugins/config/
```

Der Dateiname muss dabei folgendermaßen lauten:

```
plugin_LayoutWizzard.xml
```

Ihr Inhalt besteht aus dem Pfad zur LayoutWizzard-Konfigurationsdatei als einzigem Parameter:

```
<config_plugin>
  <layout-wizzard-config-path>
    /opt/digiverso/LayoutWizzard/config/layoutwizzard_config.xml
  </layout-wizzard-config-path>
</config_plugin>
```

4.13 Speicherverteilung über den Storage Balancer

Das intranda TaskManager Storage Balancer Plugin kann dazu verwendet werden, die Daten von Goobi auf verschiedene Netzwerk-Shares zu verteilen. Das Plugin wird dazu in einem automatischen Schritt in Goobi eingebunden und ist üblicherweise am Ende eines Workflows eingeordnet. Das Storage Balancer Plugin bekommt von Goobi einen Ordner übergeben, der dann auf den kleinsten passenden Storage aus einer konfigurierbaren Liste kopiert wird.

Die Arbeitsweise des Plugins verläuft so, dass die Daten des Goobi-Vorgangs auf ein Netzlaufwerk kopiert werden, das zuvor automatisch vom Plugin aus einer konfigurierten Liste ausgewählt wurde. Bestehen die kopierten Daten einen Vergleich der Prüfsummen mit den Originaldaten, werden anschließend die Originaldaten gelöscht und ein symbolischer Link angelegt, der vom originalen Goobi-Verzeichnis auf das Verzeichnis der Kopien verweist.

Ein Beispiel-Aufruf des TaskClients lautet folgendermaßen:

```
/usr/bin/java -jar /opt/digiverso/itm/bin/TaskClient.jar -itm http://localhost:8080/itm/service -s {processpath} -d dest -e -i {stepid} -T {processtitle} -t STORAGEBALANCER -n template -gid {processid}
```

Die innerhalb des TaskClient-Aufrufs zu übergebenen Parameter haben folgende Bedeutungen:

Parameter	Mögliche Goobi Variable	Bedeutung
-itm	http://localhost:8080/itm/service	URL zur Schnittstelle des intranda TaskManagers
-e --returnError	-	Wenn angegeben, beendet sich der TaskClient mit einem Fehlercode, um das automatische Fortschreiten im Workflow zu unterbinden
-i	{stepid}	Die ID des Arbeitsschrittes, der den Aufruf startet
-T, --title	{processtitle}	Der Vorgangstitel in Goobi, für den der Aufruf gestartet wird
-t, --jobtype	STORAGEBALANCER	Der Typ des Jobs
-n, --templatename	template	Dieser Parameter wird vom Plugin nicht verwendet.
-s, --source	{processpath}	Der Pfad des Vorgangs im Dateisystem
-d, --destination	dest	Dieser Parameter wird vom Plugin nicht verwendet, da das Ziel automatisch aus einer Liste von Storages bestimmt wird.
-p	0 – 10	Priorität zur Verarbeitung dieses Jobs
-gid	{processid}	ID des Goobi-Vorgangs

Die Konfiguration der Storage-Mountpoints erfolgt in der folgenden Datei:

```
/opt/digiverso/itm/plugins/config/storagebalancerconfig.xml
```

Der Inhalt dieser Konfiguration kann beispielhaft wie folgt aufgebaut sein:

```
<?xml version="1.0" encoding="UTF-8"?>
<storages>
  <storage>
    <path>/mnt/storage1</path>
    <buffer>10G</buffer>
  </storage>
  <storage>
    <path>/mnt/storage2</path>
    <buffer>20M</buffer>
  </storage>
  <storage>
    <path>/mnt/storage3</path>
    <buffer>50G</buffer>
  </storage>
</storages>
```

Das `<buffer>` Element kann innerhalb der Konfiguration entweder auf M oder auf G enden. Es wird dann entweder in MB oder in GB umgerechnet. Der hierbei konfigurierte Wert wird im Betrieb des Storage Balancers dafür verwendet, einen Speicherbereich in der konfigurierten Höhe nicht mit Daten zu beschreiben.

Sollen weitere Speicherbereiche innerhalb der Konfiguration ergänzt werden, können diese auch im laufenden Betrieb hinzugefügt werden.

4.14 Generierung von eBooks im Epub-Format

Das EPUB-Plugin generiert auf der Basis von ABBYY-XML-Dateien bzw. auch alternativ auf Basis von ALTO-Dateien sowie den zugehörigen METS-Dateien aus Goobi eBooks im epub-Format. Dabei werden Formatierungen, Tabellen und auch Bilder aus dem ABBYY-Format in das eBook übernommen.

Ein Beispiel-Aufruf des TaskClients lautet folgendermaßen:

```
java -jar TaskClient.jar -itm http://localhost:8080/itm/service -e -d {imagepath} -gid {processid} -n
template.xml -s {processpath}/ocr/{processtitle}_xml/ -T {processtitle} -t EPUB -if {tifpath} -mf {metaFile} -
amf {processpath}/meta_anchor.xml
```

Die innerhalb des TaskClient-Aufrufs zu übergebenen Parameter haben folgende Bedeutungen:

Parameter	Mögliche Goobi Variable	Bedeutung
-itm	http://localhost:8080/itm/service	URL zur Schnittstelle des intranda TaskManagers
-e --returnError	-	Wenn angegeben, beendet sich der TaskClient mit einem Fehlercode, um das automatische Fortschreiten im Workflow zu unterbinden
-i	{stepid}	Die ID des Arbeitsschrittes, der den Aufruf startet
-T, --title	{processtitle}	Der Vorgangstitel in Goobi, für den der Aufruf gestartet wird
-t, --jobtype	EPUB	Der Typ des Jobs
-n, --templatename	template.xml	Dieser Parameter wird vom Plugin nicht verwendet.
-s, --source	{processpath}/ocr/{processtitle}_xml /	Der Pfad zum Ordner der ABBYY-XML-Dateien
-d, --destination	-d {imagepath}	Der Pfad zum Ordner, in dem das eBook gespeichert werden soll
-p	0 – 10	Priorität zur Verarbeitung dieses Jobs
-gid	{processid}	ID des Goobi-Vorgangs
-if	{tifpath}	Pfad zu den Bildern, die für die Texterkennung durch Abbyy zuvor verwendet wurden
-mf	{metaFile}	Pfad zu der METS-Datei innerhalb von Goobi
-amf	{processpath}/meta_anchor.xml	Pfad zu der METS-Datei des Anchors innerhalb von Goobi. Dieser Parameter ist nur für Periodika relevant.
-loc	de	Optionaler Parameter für die Angabe einer Sprachvariante zur Erzeugung von Bezeichnungen wie dem Titel des Inhaltsverzeichnis. Ist keine Sprache angegeben, wird als Sprache englisch (en) angenommen.

-rtl	true / false	Optionaler Parameter der festlegt, ob es sich bei dem Werk um ein Buch handelt, was von rechts nach links zu verarbeiten ist (z.B. hebräische Werke). Fehlt dieser Parameter, wird von einem Buch ausgegangen, das von links nach rechts zu verarbeiten ist.
------	--------------	--

Die Konfiguration der eBook-Generierung erfolgt unter folgendem Pfad:

```
/opt/digiverso/itm/plugins/config/epubconfig.xml
```

Beispielhaft sieht eine solche Konfigurationsdatei folgendermaßen aus:

```
<?xml version="1.0" encoding="UTF-8"?>
<epubConfig>
  <tempPath>/tmp/</tempPath>
  <jobsInProcessing max="5"/>
</epubConfig>
```


5 Zusätzliche Details

Für weitere Details stehen die Entwickler von Goobi und der Plugins jederzeit zur Verfügung. Bitte wenden Sie sich an folgende Ansprechpartner:

Ansprechpartner:

Florian Alpers
Matthias Geerdsen
Steffen Hankiewicz
Oliver Paetzel
Robert Sehr

Kontakt:

intran|da GmbH
Bertha-von-Suttner Str. 9
D – 37085 Göttingen

<http://www.intranda.com>
info@intranda.com