

Teedy File and Document Processing

- [Searching and Tags](#)
- [Manually fix broken document relations in database](#)
- [Icons in document titles](#)
- [Fix Preview Bug](#)
- [Grafana Monitoring / Statistics](#)
- [Check index integrity / recompute quota](#)
- [Importer for Windows](#)
- [Optical Character Recognition \(OCR\) and Scanning](#)

Searching and Tags

Tags

- Tags can be nested. For example, the "Insurance" tag can be created and, for example, the "Signal Iduna" and "Ammerländer" tags below the tag. These are child elements. If you search for "Ammerländer", you will only find documents that are tagged with Ammerländer. If you search for "insurance", you will find documents that are tagged with "insurance", "Ammerländer" or "Signal Iduna" at the same time.
- **Unfortunately, tags can be created twice! Attention!**

Search operators

Operator	values	Explanation
by:	String	The creator of the document
tag:	String	document with given tag
!tag:	String	document without given tag
before:	date (allowed formats: yyyy or yyyy-MM or yyyy-MM-dd)	created before date
ubefore:	date (allowed formats: yyyy or yyyy-MM or yyyy-MM-dd)	edited before date
after:	date (allowed formats: yyyy or yyyy-MM or yyyy-MM-dd)	created after date
uafter:	date (allowed formats: yyyy or yyyy-MM or yyyy-MM-dd)	edited after date
at:	date (allowed formats: yyyy or yyyy-MM or yyyy-MM-dd)	created at date
uat:	date (allowed formats: yyyy or yyyy-MM or yyyy-MM-dd)	edited at date
lang:	"eng", "fra", "ita", "deu", "spa", "por", "pol", "rus", "ukr", "ara", "hin", "chi_sim", "chi_tra", "jpn", "tha", "kor", "nld", "tur", "heb"	language

Operator	values	Explanation
mime:		<div>does not work yet!</div> <ol style="list-style-type: none"> 1. image/jpeg 2. application/zip 3. application/pdf 4. image/png 5. text/csv 6. text/plain 7. application/vnd.openxmlformats-officedocument.presentationml.presentation 8. application/vnd.openxmlformats-officedocument.wordprocessingml.document 9. application/octet-stream
shared:	yes, no	
workflow:	"me", String	
full:	String	Use OCR full-text search (files must have been processed with Tesseract!) - full search is default since Teedy 1.9
simple:	String	Performs simple search instead full search (ignores OCR)
*		Wildcard only possible at the end of the search input string. Not allowed before or in a word
		<p>Pipe operator. Use this to filter things like "or". Example</p> <ul style="list-style-type: none"> • green duck <ul style="list-style-type: none"> ◦ find docs which have green or duck in title

Operator	values	Explanation
"<string>"		<p>phrases can be put into quotes. This will return a more exact result. For example:</p> <ul style="list-style-type: none"> • "a green duck" <ul style="list-style-type: none"> ◦ rreturns docs with the exact title "a green duck" • a green duck <ul style="list-style-type: none"> ◦ returns docs which contain a, green or duck

The operators ?, NOT, AND, OR are not possible - they do nothing. [Lucene Core Dokumentation](#) → Most operators unfortunately don't work in Teedy.

All other things which cannot be expressed by the given search parameters can be scripted by SQL queries for H2 or PSQL database instead. You will need to have according access to do this.

You can find a lot of useful SQL statement for filtering out your DMS in our Grafana Dashboard → [Grafana Monitoring / Statistics](#)

Example scheme for document title for things like invociecs

```
<YYYY>\<MM> - <creditor> <type> <document number> [#<index>]
```

Search Filter - Combination from words, tags and other operators

Example: find documents which are tagged by invoice and company and which have "01" and "2018" in their title

```
tag:company tag:invoice 2018 01
```

Manually fix broken document relations in database

Find documents which have relations to other documents which already were deleted

Sometimes documents link to other documents but the links are invalid because the linked document is not available anymore. We can manually reset those links only because there is no working mechanism yet.

 Dokument nicht gefunden

Get all relations with their id's

```
SELECT
    R.rel_id_c,
    F.doc_title_c "From",
    T.doc_title_c "To"
FROM
    t_relation AS R
JOIN t_document AS T ON R.rel_iddocfrom_c = T.doc_id_c
JOIN t_document AS F ON R.rel_iddocto_c = F.doc_id_c
WHERE
    R.rel_deletedate_d IS NULL AND
    T.doc_deletedate_d IS NOT NULL AND
    F.doc_deletedate_d IS NULL

UNION

SELECT
    R.rel_id_c,
    T.doc_title_c "From",
    F.doc_title_c "To"
FROM
    t_relation AS R
```

```
JOIN t_document AS T ON R.rel_iddocfrom_c = T.doc_id_c
JOIN t_document AS F ON R.rel_iddocto_c = F.doc_id_c
WHERE
    R.rel_deletedate_d IS NULL AND
    T.doc_deletedate_d IS NULL AND
    F.doc_deletedate_d IS NOT NULL

ORDER BY "From"

;
```

Now we just set the delete flag of the relation to a date not NULL so it will unshow in Teedy visually. That will remove invalid links.

```
update t_relation SET rel_deletedate_d = NOW() WHERE rel_id_c IN ('your relation id 1', 'your
relation id 2', .. , 'your relation id n');
```

Icons in document titles

Inside Teedy we can use funny icons, if we know the correct Unicode number.

✓

Symbols

✓

21.05.26 von MarioVoigt

Teilen



https://en.wikipedia.org/wiki/Miscellaneous_Symbols

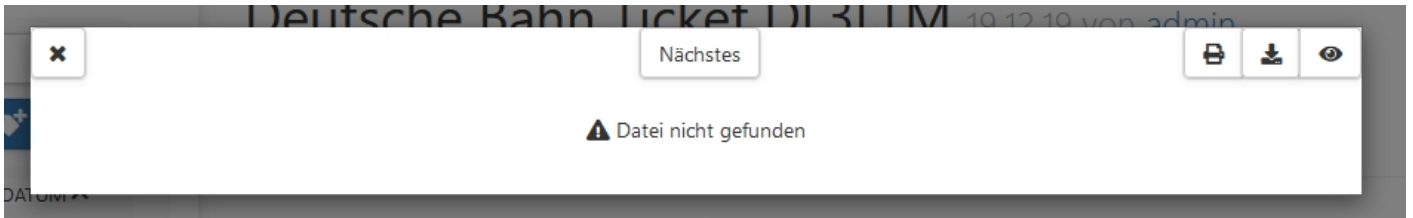
Copy + Paste. Just select the one's you need and paste them into Teedy. It works for title and description



https://en.wikipedia.org/wiki/Miscellaneous_Symbols

Fix Preview Bug

In case the file preview is erroneous/empty but the file can be processed and it can be downloaded by URL like https://dms.yourdomain.de/api/file/:FILE_ID/data:



Root Cause: unkown. Seems to happen after migration from H2 to PostgreSQL

Fixing proposal

Remove the `_thumb` and `_web` files

and let Teedy create new ones by running a (complete) re-processing

```
23:24:35 ✓ :/var/docs/storage# ll | grep 80bf2588-5f9f-4489-8462-08bcbfe8d676
-rw-r--r-- 1 jetty jetty 829634 Dec 19 15:15 80bf2588-5f9f-4489-8462-08bcbfe8d676
-rw-r--r-- 1 jetty jetty 28882 Jan 12 14:01 80bf2588-5f9f-4489-8462-08bcbfe8d676_thumb
-rw-r--r-- 1 jetty jetty 496768 Jan 12 14:01 80bf2588-5f9f-4489-8462-08bcbfe8d676_web
```

```
cd /var/docs/storage
ll | grep "<YOUR_FILE_ID>"
mv <YOUR_FILE_ID>_thumb <YOUR_FILE_ID>_thumb.bak
mv <YOUR_FILE_ID>_web <YOUR_FILE_ID>_web.bak

#or just move all stuff to some sub directory if you plan to re-process the complete file
system:
mkdir thumb_web_bak
mv *_thumb *_web thumb_web_bak/

#restart your instance to let Teedy recognize that changes due to caching
sudo systemctl restart jetty9.service
```

Reprocess documents

See [Teedy API Scripts / database queries](#) for reprocessing of everything.

Grafana Monitoring / Statistics

Description

A Grafana monitoring dashboard for Teedy (Sismics Docs) statistics. Helpful to have a look over security things and the effort you put in your instance. Please check if this is okay for your own use - regarding privacy protection of the mates working together on the same instance. Sorry the language for that dashboard is german but you can translate it easily using tools like [deepl.com](https://www.deepl.com).

Dokumente

Dateien

Benutzer

Auth Tokens

18

Kommentare

49

Audit Einträge

Tags

164

Shares

0

Gelöschte Dokumente

Gelöschte Dateien

Benutzer ohne 2FA

Dateien in Schnellablage

11

Gelöschte Kommentare

4

Erster Audit Eintrag

Gelöschte Tags

6

Datenbankgröße

57 MB

OCR indexierte Dateien

Dateien

Dateien nach Dateityp

Anzahl	Dateityp
	*.pdf
	*.jpeg
	*.csv
	unzuordenbar
	*.docx
	*.zip
	image/gif
	*.png

Dateien pro Jahr (nach Dokumentendatum)

Anzahl	Jahr
	-
	2026
	2020
	2019
	2018
	2017
	2016
	2015
	2014

Dateien pro Jahr (nach Erstelldatum)

Anzahl	Jahr
	2020
	2019

Dateien pro Benutzer in Schnellablage

Anzahl	Benutzer
--------	----------

Dokumente

Dokumente pro Jahr und Monat

Anzahl	Monat	Jahr
	Mai	2026
	Februar	2020
	Januar	2020
	Dezember	2019
	November	2019
	Oktober	2019
	September	2019
	August	2019
	Juli	2019
	Juni	2019
	Mai	2019
	April	2019

Dokumente pro Jahr

Anzahl	Jahr
	2026
	2020
	2019
	2018
	2017
	2016
	2015
	2014
	2013
	2012
	2011
	2010

Dokumente nach Tag-Häufigkeit

Anzahl	Tag
1504	
1251	
527	
524	
357	
331	
313	
274	
157	
145	
130	
108	Quittung

Dokumente nach Sprache

Anzahl	Sprache
	Deutsch
	Englisch

Dokumente nach Benutzer

Anzahl	Benutzer
--------	----------

Dokumente und Dateien pro Jahr und Monat

Audit

Audit Log

Zeitstempel	Benutzer	Kategorie	Typ	Nachricht
08.01.2019 22:48:17		Benutzer	Aktualisiert	
08.01.2019 22:48:17		Benutzer	Aktualisiert	
08.01.2019 22:48:59		Gruppe	Aktualisiert	
08.01.2019 23:28:48		Workflow Model	Aktualisiert	
08.01.2019 23:29:51		Benutzer	Erstellt	
08.01.2019 23:43:18		Berechtigung	Erstellt	
08.01.2019 23:43:18		Berechtigung	Erstellt	
08.01.2019 23:43:18		Dokument	Erstellt	
08.01.2019 23:43:18		Datei	Erstellt	
08.01.2019 23:45:01		Tag	Erstellt	
08.01.2019 23:45:01		Berechtigung	Erstellt	
08.01.2019 23:45:01		Berechtigung	Erstellt	
08.01.2019 23:45:12		Berechtigung	Erstellt	
08.01.2019 23:45:12		Berechtigung	Erstellt	
08.01.2019 23:45:12		Tag	Erstellt	

Download

- <https://gitea.fablabchemnitz.de/vmario/teedy-statistics/src/branch/master>
- <https://grafana.com/grafana/dashboards/11556>

Check index integrity / recompute quota

Sometimes the used space is wrong and may look like this:

-427MB (-1,7%) verwendet von 25.000MB

1814 Dokumente gefunden

See also: <https://github.com/sismics/docs/issues/345>

Checks on file system

```
cd /var/docs/

find ./ -type f -name '*' -exec du -ch {} + | grep total$
find ./ -type f -name '*_web' -exec du -ch {} + | grep total$
find ./ -type f -name '*_thumb' -exec du -ch {} + | grep total$

cd /var/docs/storage/
ll |grep -v "thumb\|web" |wc -l #Anzahl der Dateien, die weder _web noch _thumb sind
ll | grep "web" | wc -l      #Anzahl _web Dateien (sollte idealerweise mit _thumb deckend
sein)
ll | grep "thumb" | wc -l   #Anzahl _thumb Dateien (sollte idealerweise mit _web deckend
sein)
```

Checks in Database / Recalculate Quota

Get a list of all files from database which should exist/not exist on filesystem

```
/*Get all files which should be existent on HDD*/
SELECT
fil_id_c AS "FileID",
fil_iduser_c AS "Besitzer"
FROM t_file
WHERE fil_deletedate_d IS NULL
;

/*Get all files which should be deteled from HDD*/
SELECT
fil_id_c AS "FileID",
```

```
fil_iduser_c AS "Besitzer"
FROM t_file
WHERE fil_deletedate_d IS NOT NULL
;
```

Get the filesystem storage list

The following commands generate file output and send them by mail to you. If your application server and your database are on the same server you just can create some bash script to make some complete scripting solution automating psql.

```
cd /var/docs/

#get recent storage list as CSV
ll | grep -v "thumb\|web" | awk 'NR > 3 {print "\"",$5,"\"";\"",$9,"\""}' | sed
's/[[:blank:]]//g' | mail -s "Teedy FileSystem" your@mail.address

#or get it as SQL statement
ll | grep -v "thumb\|web" | awk 'NR > 3 {print
"UPDATE#TMP_QUOTA_CHECK#SET#fil_size=\x27",$5,"\x27#WHERE#fil_id_c=\x27",$9,"\x27;"}' | sed
's/[[:blank:]]//g' | sed 's/#/ /g' | mail -s "Teedy FileSystem" your@mail.address
```

Create a temporary SQL table to calculate quota

```
CREATE TABLE TMP_QUOTA_CHECK(
    fil_id_c character varying(36),
    fil_iduser_c character varying(36),
    fil_size integer
);
```

Pre-Fill the table with existing data

```
INSERT INTO TMP_QUOTA_CHECK SELECT
fil_id_c,
fil_iduser_c
FROM t_file
WHERE fil_deletedate_d IS NULL
;
```

Insert file size data from upper generated SQL UPDATE awk statements (bash), then perform some check and calculate total quota sizes

```
SELECT * FROM tmp_quota_check WHERE fil_size IS NOT NULL;
SELECT * FROM tmp_quota_check WHERE fil_size IS NULL; --if your filesystem is consistent this
must be empty! If not please check if Teedy failed to delete files in the past

SELECT
    CONCAT('UPDATE t_user SET use_storagecurrent_n='',SUM(fil_size),' ' WHERE use_id_c
   ='',fil_iduser_c,'');')
FROM TMP_QUOTA_CHECK
GROUP BY fil_iduser_c
;
```

Insert the new values into existing target table using the generated output statements from above

Drop temporary table

```
DROP TABLE TMP_QUOTA_CHECK;
```

Removed commit

<https://github.com/sismics/docs/commit/d0335b6b161058250ec8cc44eeb2357f96176f54#diff-54e77110186f974f07aeb29c2673496fL690>

Importer for Windows

- The Bulk file importer tool is based on NodeJS
- Documentation also available under

<https://github.com/sismics/docs/tree/master/docs-importer>

Download the importer

See [Downloads](#) for recent compiled setups. The importer can be also downloaded at Github. The most recent version to find is

<https://github.com/sismics/docs/releases/download/v1.5/docs-importer-win.exe> (which is an old one). We can also build ourselves. See below.

Building the importer

Requirements

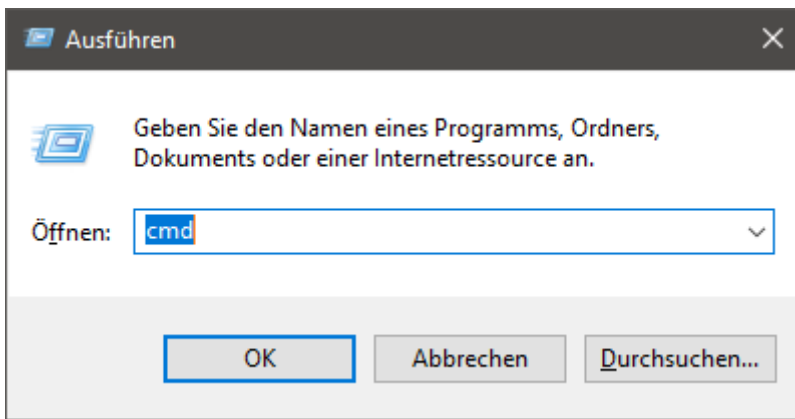
- NodeJS v10.18.0 (64 Bit) → <https://nodejs.org/dist/v10.18.0/node-v10.18.0-win-x64.zip> - newer version will fail!
- Git → <https://git-scm.com/download/win>

Check your `%PATH%` variable. This should contain the following executables

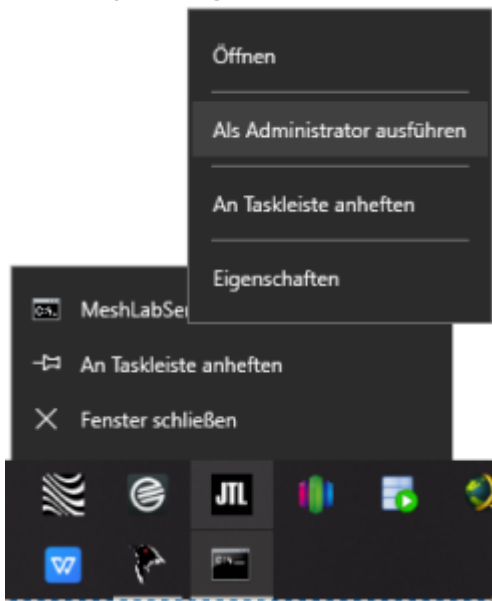
- nodejs.exe
- git.exe

Open elevated CMD Shell

1. press CTRL + R to open "Run"
2. Enter "cmd"
3. Click ok



4. Start elevated shell from current cmd. This will open up a new cmd shell with admin privileges



Clone Repository and run build

```
cd C:\
git clone https://github.com/sismics/docs.git
cd docs\docs-importer
npm install
npm install -g pkg
pkg .
```

```

C:\>cd docs
C:\docs>cd docs-importer
C:\docs\docs-importer>npm install
audited 350 packages in 2.259s

4 packages are looking for funding
  run `npm fund` for details

found 9 vulnerabilities (3 moderate, 6 high)
  run `npm audit fix` to fix them, or `npm audit` for details

C:\docs\docs-importer>npm install -g pkg
C:\node-v10.18.0-win-x64\pkg -> C:\node-v10.18.0-win-x64\node_modules\pkg\lib-es5\bin.js
+ pkg@4.4.2
added 138 packages from 199 contributors in 11.163s

C:\docs\docs-importer>pkg .
> pkg@4.4.2
> Targets not specified. Assuming:
  node10-linux-x64, node10-macos-x64, node10-win-x64
> Fetching base Node.js binaries to PKG_CACHE_PATH
  fetched-v10.17.0-linux-x64 [=====] 100%
  fetched-v10.17.0-win-x64   [=====] 100%
  fetched-v10.17.0-macos-x64 [=====] 100%

C:\docs\docs-importer>_

```

Check the built output

Lokaler Datenträger (C:) > docs > docs-importer >

Name	Änderungsdatum	Typ	Größe	Erstelldatum
node_modules	09.01.2020 22:13	Dateiordner		09.01.2020 22:11
main.js	09.01.2020 22:10	JavaScriptdatei	8 KB	09.01.2020 22:10
package.json	09.01.2020 22:13	JSON-Datei	1 KB	09.01.2020 22:13
package-lock.json	09.01.2020 22:13	JSON-Datei	61 KB	09.01.2020 22:13
README.md	09.01.2020 22:10	MD-Datei	1 KB	09.01.2020 22:10
SismicsDocs.ico	09.01.2020 22:10	GIMP 2.10.12	84 KB	09.01.2020 22:10
teedy-importer-linux	09.01.2020 22:19	Datei	59.323 KB	09.01.2020 22:19
teedy-importer-macos	09.01.2020 22:19	Datei	60.314 KB	09.01.2020 22:19
teedy-importer-win.exe	09.01.2020 22:19	Anwendung	55.711 KB	09.01.2020 22:19

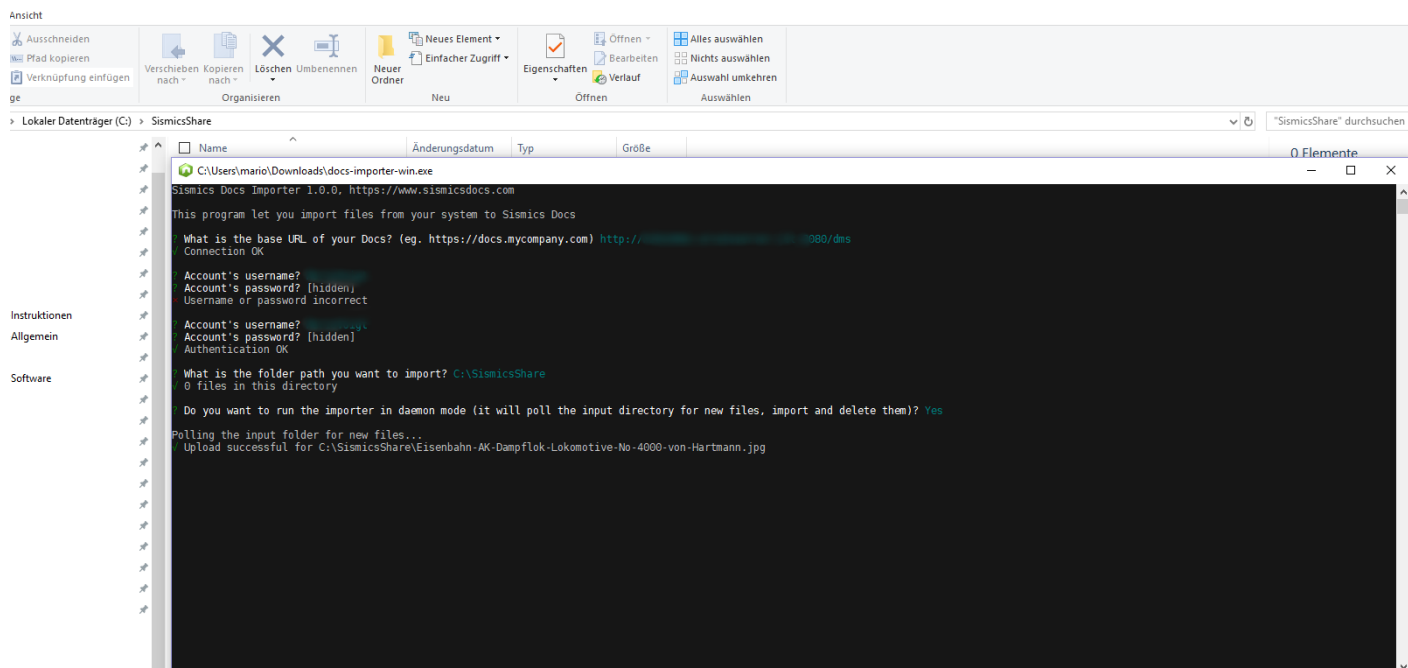
Configure to use the importer

Create new share upload directory

A special upload folder should be created, e.g. `C:\TeedyShare` - from this folder the documents will be uploaded and **cut later**.

Start docs-importer-win.exe and configure it

Please put the docs-importer-win.exe to some fixed place where it should stay, like in `C:\Teedy\docs-importer-win.exe`



Note that the screenshot contains some older directory name.

After entering the connection data this information will be persisted in

%userprofile%\config\preferences\com.sismics.docs.importer.pref

Start as daemon and test upload

The program can be started with the switch `-d`. It queries the specified folder every 30 seconds and uploads any existing documents to the DMS. The files are then deleted locally.

```
Eingabeaufforderung - docs-importer-win.exe -d
Microsoft Windows [Version 10.0.17134.472]
(c) 2018 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\mario>cd ..
C:\Users>cd ..
C:\>cd Sismics
C:\Sismics>docs-importer-win.exe -d
Sismics Docs Importer 1.0.0, https://www.sismicsdocs.com

This program let you import files from your system to Sismics Docs

Starting in quiet mode with the following configuration:
Base URL: http://.../dms
Username:
Password:
Daemon mode: true

Polling the input folder for new files...
```

Install as Windows Service

Create file `C:\Teedy\teedy-service.ps1`

```
Start-Process -WindowStyle hidden -FilePath C:\Teedy\docs-importer-win.exe -ArgumentList "-d"
```

Create a new task in task scheduler

Sorry for german screenshots. And please replace "SismicsDocs" with "Teedy" everywhere.

The screenshot shows the Windows Task Scheduler interface. The top pane displays a list of tasks, including 'SismicsDocs'. The bottom pane shows the 'Eigenschaften von SismicsDocs (Lokaler Computer)' dialog box.

Eigenschaften von SismicsDocs (Lokaler Computer)

Allgemein | Trigger | Aktionen | Bedingungen | Einstellungen | Verlauf (deaktiviert)

Name: SismicsDocs

Speicherort: \

Autor: DESKTOP-Q30SI7Lmario

Beschreibung:

Sicherheitsoptionen

Beim Ausführen der Aufgaben folgendes Benutzerkonto verwenden:
mario Benutzer oder Gruppe ändern...

☒ Nur ausführen, wenn der Benutzer angemeldet ist

☐ Unabhängig von der Benutzeranmeldung ausführen

☐ Kennwort nicht speichern. Die Aufgabe greift nur auf lokale Computerressourcen zu.

☐ Mit höchsten Privilegien ausführen

☐ Ausgeblendet Konfigurieren für: Windows 10

OK **Abbrechen**

Eigenschaften von SismicsDocs (Lokaler Computer)

Allgemein | Trigger | Aktionen | Bedingungen | Einstellungen | Verlauf (deaktiviert)

Beim Erstellen einer Aufgabe können Sie die Bedingungen angeben, die die Aufgabe auslösen.

Trigger	Details	Status
Beim Start	Beim Systemstart	Aktiviert

Neu... **Bearbeiten...** **Löschen**

OK **Abbrechen**

Eigenschaften von SismicsDocs (Lokaler Computer) X

Allgemein Trigger Aktionen **Bedingungen** Einstellungen Verlauf (deaktiviert)

Geben Sie die Bedingungen und Trigger an, die bestimmen, ob die Aufgabe ausgeführt werden soll. Die Aufgabe wird nicht ausgeführt, wenn eine der hier angegebenen Bedingungen nicht erfüllt ist.

Leerlauf

☐ Aufgabe nur starten, falls Computer im Leerlauf ist für: 10 Minuten

Auf Leerlauf warten für: 1 Stunde

☒ Beenden, falls Computer aus dem Leerlauf reaktiviert wird

☐ Neustart bei längerem Leerlauf

Energie

☐ Aufgabe nur starten, falls Computer im Netzbetrieb ausgeführt wird

☒ Beenden, wenn Computer in den Akkubetrieb wechselt

☐ Computer zum Ausführen der Aufgabe reaktivieren

Netzwerk

☐ Nur starten, wenn folgende Netzwerkverbindung verfügbar ist:

Alle Verbindungen

OK Abbrechen

Eigenschaften von SismicsDocs (Lokaler Computer) X

Allgemein Trigger **Aktionen** Bedingungen Einstellungen Verlauf (deaktiviert)

Beim Erstellen einer Aufgabe müssen Sie die beim Start auszuführende Aufgabe angeben.

Aktion	Details
Programm starten	powershell C:\Sismics\sismics-service.ps1

Neu... Bearbeiten... Löschen

OK Abbrechen

Eigenschaften von SismicsDocs (Lokaler Computer) X

Allgemein Trigger Aktionen Bedingungen **Einstellungen** Verlauf (deaktiviert)

Geben Sie weitere Einstellungen für das Verhalten der Aufgabe an.

☒ Ausführung der Aufgabe bei Bedarf zulassen

☐ Aufgabe so schnell wie möglich nach einem verpassten Start ausführen

☐ Falls Aufgabe scheitert, neu starten alle: 1 Minute

Neustartversuche bis maximal: 3 Mal

☐ Aufgabe beenden, falls Ausführung länger als: 3 Tage

☒ Beenden der aktiven Aufgabe erzwingen, falls sie auf Aufforderung nicht beendet wird

☐ Falls keine weitere Ausführung geplant ist, Aufgabe löschen nach: 30 Tage

Folgende Regel anwenden, falls die Aufgabe bereits ausgeführt wird:

Keine neue Instanz starten

OK Abbrechen

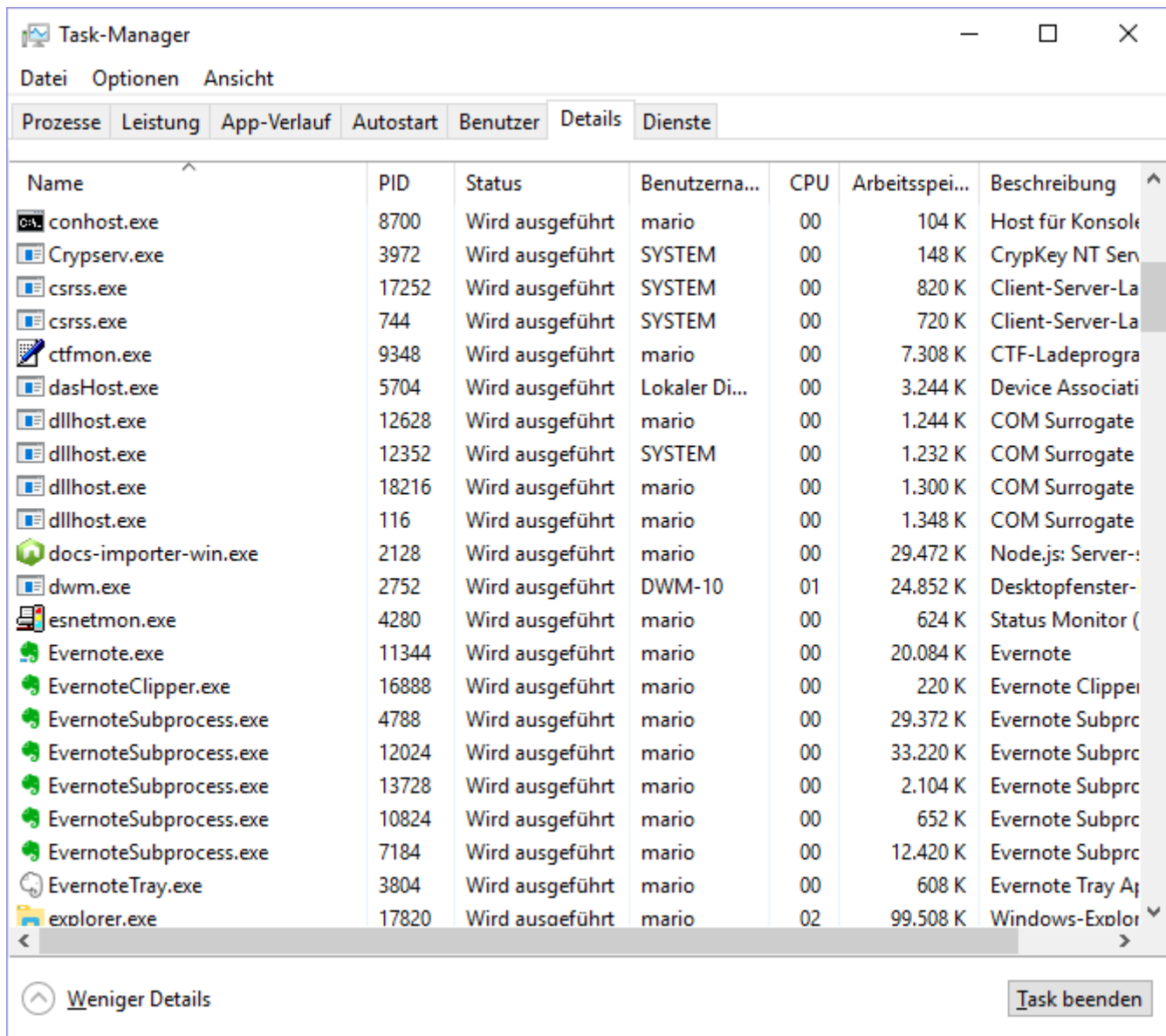
Eigenschaften von SismicsDocs (Lokaler Computer) X

Allgemein Trigger Aktionen Bedingungen Einstellungen **Verlauf (deaktiviert)**

Anzahl von Ereignissen: 0

Ebene	Datum ...	Ereig...	Aufgabenkate...	Vorgangscode	Korrelations...
-------	-----------	----------	-----------------	--------------	-----------------

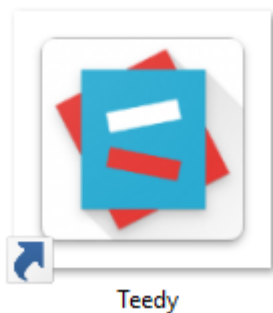
Check if service is running. Look for `docs-importer-win.exe`



Task-Manager						
Datei Optionen Ansicht						
Prozesse Leistung App-Verlauf Autostart Benutzer Details Dienste						
Name	PID	Status	Benutzerna...	CPU	Arbeitsspei...	Beschreibung
conhost.exe	8700	Wird ausgeführt	mario	00	104 K	Host für Konsole
Crypse.exe	3972	Wird ausgeführt	SYSTEM	00	148 K	CrypKey NT Sen
csrss.exe	17252	Wird ausgeführt	SYSTEM	00	820 K	Client-Server-La
csrss.exe	744	Wird ausgeführt	SYSTEM	00	720 K	Client-Server-La
ctfmon.exe	9348	Wird ausgeführt	mario	00	7.308 K	CTF-Ladeprogra
dasHost.exe	5704	Wird ausgeführt	Lokaler Di...	00	3.244 K	Device Associati
dllhost.exe	12628	Wird ausgeführt	mario	00	1.244 K	COM Surrogate
dllhost.exe	12352	Wird ausgeführt	SYSTEM	00	1.232 K	COM Surrogate
dllhost.exe	18216	Wird ausgeführt	mario	00	1.300 K	COM Surrogate
dllhost.exe	116	Wird ausgeführt	mario	00	1.348 K	COM Surrogate
docs-importer-win.exe	2128	Wird ausgeführt	mario	00	29.472 K	Node.js: Server-
dwm.exe	2752	Wird ausgeführt	DWM-10	01	24.852 K	Desktopfenster-
esnetmon.exe	4280	Wird ausgeführt	mario	00	624 K	Status Monitor (
Evernote.exe	11344	Wird ausgeführt	mario	00	20.084 K	Evernote
EvernoteClipper.exe	16888	Wird ausgeführt	mario	00	220 K	Evernote Clipper
EvernoteSubprocess.exe	4788	Wird ausgeführt	mario	00	29.372 K	Evernote Subprc
EvernoteSubprocess.exe	12024	Wird ausgeführt	mario	00	33.220 K	Evernote Subprc
EvernoteSubprocess.exe	13728	Wird ausgeführt	mario	00	2.104 K	Evernote Subprc
EvernoteSubprocess.exe	10824	Wird ausgeführt	mario	00	652 K	Evernote Subprc
EvernoteSubprocess.exe	7184	Wird ausgeführt	mario	00	12.420 K	Evernote Subprc
EvernoteTray.exe	3804	Wird ausgeführt	mario	00	608 K	Evernote Tray Ap
explorer.exe	17820	Wird ausgeführt	mario	02	99.508 K	Windows-Explor

Weniger Details Task beenden

Create a new Desktop Shortcut for your share directory



Optical Character Recognition (OCR) and Scanning

Handling

OCR data is stored in Teedy database table `t_file` which contains the string column `fil_content_c`. In H2 the data is stored als plaintext string. In PostgreSQL the column is filled as datatype `::text`. A normal select returns number. The unencrypted OCR text data can be accessed from the large object by using some SQL statement like

```
select
fil_name_c,
convert_from(loread(lo_open(fil_content_c::int, 131072), 999999999), 'UTF8')
from t_file WHERE fil_deletedate_d IS NULL AND fil_content_c IS NOT NULL limit 1;
```

Teedy uses a built in process runner to start the binary `tesseract` with a language parameter. This works if "`tesseract`" is contained in `$PATH` (Linux) or `%PATH%` (Windows) environment variable.

Fixing faulty `fil_content_c` data the easy way

Some quick fix for issue described in <https://github.com/sismics/docs/issues/451>

```
SELECT fil_content_c FROM t_file
WHERE  LENGTH(fil_content_c) > 6
ORDER BY fil_createdate_d DESC;

UPDATE t_file SET fil_content_c = NULL
WHERE  LENGTH(fil_content_c) > 6;
```

Converting LOB data to plain text (was required at some point from updating Teedy 1.8 to Teedy 1.9)

```
/*
Show items which start with useless linefeeds. We need to correct those because otherwise we
cannot continue with following statements (casting "fil_content_c::int" will fail and other
issues)

Result may be empty
```

```

*/
SELECT
    fil_id_c,
    fil_name_c,
    fil_content_c
FROM t_file
WHERE
    fil_content_c LIKE E'%\n'
;

/*Trim beginning linefeeds (only) away*/
UPDATE t_file SET fil_content_c = TRIM(e'\n' FROM fil_content_c)
WHERE
    fil_content_c LIKE E'%\n'
;

/*
Show faulty data which would return "invalid byte sequence for encoding "UTF8": 0x00" or
similar.
First we build some function to check for valid UTF8 bytea because sometimes we have faulty
stuff inside DB
Result may be empty
*/
CREATE FUNCTION is_valid_utf8(bytea) RETURNS boolean
    LANGUAGE plpgsql AS
$$BEGIN
    PERFORM convert_from($1, 'UTF8');
    RETURN TRUE;
EXCEPTION
    WHEN character_not_in_repertoire THEN
        RAISE WARNING '%', SQLERRM;
        RETURN FALSE;
END;$$;
SELECT
    fil_id_c,
    fil_name_c,
    lread(lo_open(fil_content_c::int, CAST( x'20000' AS integer)), 999999999) AS BYTE_DATA,
    LENGTH(lread(lo_open(fil_content_c::int, CAST(x'20000' AS integer)), 999999999)) AS LEN
FROM t_file

```



```

WHERE
    fil_content_c IS NOT NULL AND
    fil_content_c != '' AND
    LENGTH(fil_content_c) <= 6 AND
    is_valid_utf8(fil_content_c::bytea) IS FALSE
;

/*We set NULL to all items with faulty UTF-8 encoding (if there were some from previous
statement)*/
UPDATE t_file SET fil_content_c = NULL
WHERE
    fil_content_c IS NOT NULL AND
    fil_content_c != '' AND
    LENGTH(fil_content_c) <= 6 AND
    is_valid_utf8(fil_content_c::bytea) IS FALSE
;

/*
Select OCR content which is in LOB format (Large Object) and valid UTF-8
*/
SELECT
    fil_id_c,
    fil_name_c,
    fil_content_c,
    fil_content_c::bytea, /*shows "invisible" data which does not trigger NULL or ''*/
    lread(lo_open(fil_content_c::int, CAST(x'20000' AS integer)), 999999999) AS BYTE_DATA,
    /*we use the encoding we used to create the database. See setup instructions. Usually
this is "UNICODE" or "UTF8"*/
    LENGTH(lread(lo_open(fil_content_c::int, CAST(x'20000' AS integer)), 999999999)) AS LEN,
    convert_from(lread(lo_open(fil_content_c::int, CAST(x'20000' AS integer)), 999999999),
'UNICODE') as "fil_content_c"
FROM t_file
WHERE
    fil_content_c IS NOT NULL AND
    fil_content_c != '' AND
    LENGTH(fil_content_c) <= 6 AND
    is_valid_utf8(fil_content_c::bytea) IS TRUE
ORDER BY LEN ASC
;

```

```

/*Convert LOB data into plain text. First we do it for a custom selected file with fil_id_c*/
UPDATE t_file SET fil_content_c = convert_from(loread(lo_open(fil_content_c::int, CAST(
x'20000' AS integer))), 999999999), 'UNICODE')::TEXT
WHERE
    fil_id_c = '13411bb0-12fd-4e25-b483-2e2d18b344ed'
;

/*Check the conversion value*/
SELECT
    fil_id_c,
    fil_name_c,
    fil_content_c
FROM t_file
WHERE
    fil_id_c = '13411bb0-12fd-4e25-b483-2e2d18b344ed'
;

/*
Now we do mass processing for LOB to plain text
DO NOT CONTINUE WITH OTHER STATEMENTS IF THIS ONE FAILS AND CHECK THE UPPER ONES AGAIN
*/
UPDATE t_file SET fil_content_c = convert_from(loread(lo_open(fil_content_c::int, CAST(
x'20000' AS integer))), 999999999), 'UNICODE')::TEXT
WHERE
    fil_content_c IS NOT NULL AND
    fil_content_c != '' AND
    LENGTH(fil_content_c) <= 6 AND
    is_valid_utf8(fil_content_c::bytea) IS TRUE
;

/*We fix again useless linefeeds by trimming*/
UPDATE t_file SET fil_content_c = TRIM(e'\n' FROM fil_content_c)
WHERE
    fil_content_c LIKE E'%\n'
;

/*
Now that we converted all the LOB stuff we do mass processing for remaining stuff with length

```

```

lesser than 6 chars because those OCR values are just crap
WARNING: DO NOT RUN THIS BEFORE CONVERTING BECAUSE YOU WILL OVERWRITE. IF YOU DID YOU WILL
NEED TO REPROCESS ALL DOCUMENTS!
*/
UPDATE t_file SET fil_content_c = NULL
WHERE
    fil_content_c IS NOT NULL AND
    fil_content_c != '' AND
    LENGTH(fil_content_c) <= 6
;

/*Finally we check again the values visually*/
SELECT
    fil_id_c,
    fil_name_c,
    fil_content_c
FROM t_file

/*Finally re-run the indexing from background UI web interface or API to have a good search
index again*/

```

Tesseract OCR command line binary

The installation of tesseract is simple. Note that for different operating system versions there are different tesseract versions. All tesseract versions work different in their speed and quality. We figured out that tesseract 3 on Ubuntu 16 works much faster than tesseract 4 on Ubuntu 18.

<https://github.com/tesseract-ocr/tesseract/wiki>

Installation

For Linux users:

```

#install regular version
sudo apt install tesseract-ocr tesseract-ocr-deu #will install the most recent version
belonging to your OS. So older system you might get older tesseract

#install devel version. See https://launchpad.net/~alex-p/+archive/ubuntu/tesseract-ocr-devel
sudo add-apt-repository ppa:alex-p/tesseract-ocr-devel sudo apt-get update

```

```
sudo apt install tesseract-ocr tesseract-ocr-deu #add your desired languages here
```

For Windows users:

<https://github.com/tesseract-ocr/tesseract/wiki/4.0-with-LSTM#4x-for-windows>

Critical optimization

<https://github.com/tesseract-ocr/tesseract/issues/2611>

Some users said that disabling multiprocessing in tesseract fixes speed problems. Therefore some environment flag should be set using export. See also [Environment Configuration](#)

```
export OMP_THREAD_LIMIT=1
```

Scanner Apps for Smartphones

There are a LOT of scanner apps in PlayStore. Most of them have nearly same naming. The following list is only a minimalistic overview of stuff around the web. Mainly we are looking for open source applications.

- [Genius Scan](#)
- [CamScanner](#)
- [Notebloc](#)
- [OpenNoteScanner](#)
- [SwiftScan](#)

Wishes

- automatic upload to or sending by mail
- problem: what if you use multiple instances of DMS? Then you will need multiple upload locations. All known app do not deal with that feature. With app cloning the scanner app could be multiplied so each Scanner app instance has its own configuration. Then the scanner app could send to the correct inbox per DMS instance