

Jobs

Wiederkehrende Ausführung von Jobs und Skripten.

- [Jobs über Benutzeroberfläche steuern](#)
- [Einrichten der Jobverwaltung](#)
- [Cleanup-Job](#)
- [Unlock-Job](#)

Jobs über Benutzeroberfläche steuern

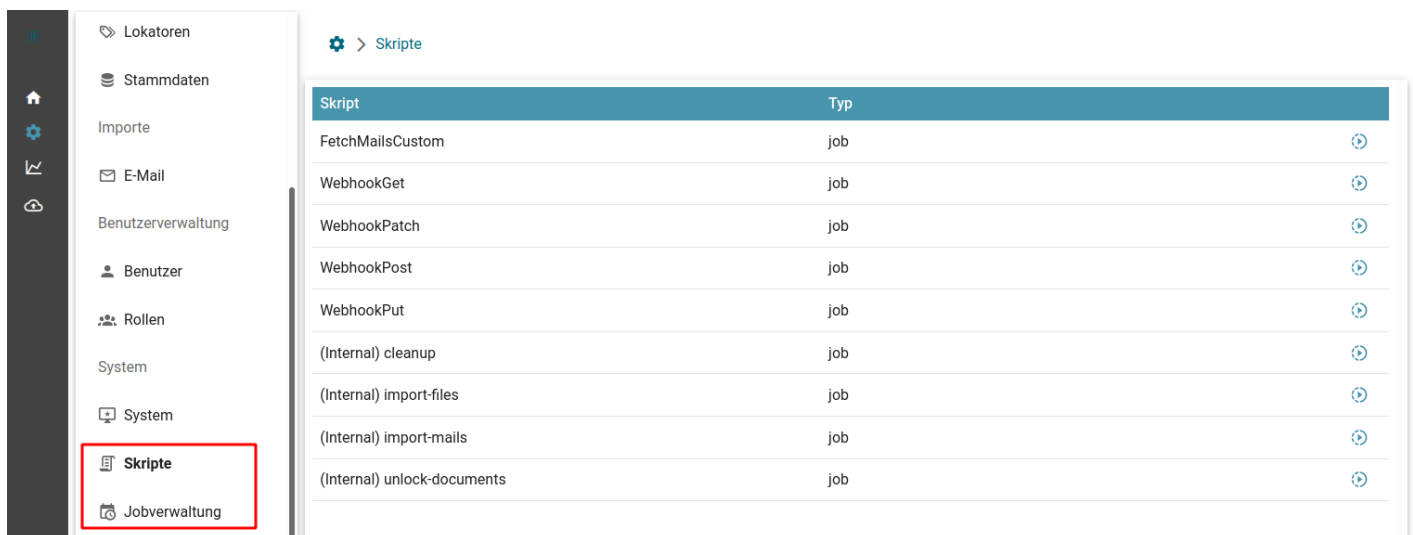
Das folgende Feature befindet sich noch in Entwicklung und steht nur einigen Testkunden bereit. Falls die dargestellten Menüpunkte nicht verfügbar sind, wurde die Funktion noch nicht freigeschalten.

Wenn sie Squeeze selbst betreiben, stellen Sie sicher, dass das Jobsystem korrekt eingerichtet ist. Dies ist hier dokumentiert: [Einrichten der Jobverwaltung SQUEEZE 2](#)

In der Administration befinden sich die folgenden zwei Menüpunkte.

Skript-Übersicht

Hier werden Skripte des aktuellen Mandanten dargestellt. Diese Skripte können Sie als Jobs wiederkehrend starten.



Skripte	
Skript	Typ
FetchMailsCustom	job
WebhookGet	job
WebhookPatch	job
WebhookPost	job
WebhookPut	job
(Internal) cleanup	job
(Internal) import-files	job
(Internal) import-mails	job
(Internal) unlock-documents	job

Job-Verwaltung

Hier können Jobs für Job-Skripte eingerichtet werden.

Dokumente

Warteschlange

Stapelklassen

Dokumentenklassen

Lokatoren

Stammdaten

Importe

E-Mail

Benutzerverwaltung

Benutzer

Rollen

System

System

Skripte

Jobverwaltung

Jobverwaltung

ID	Name	Cron Ausdruck	Skript	Aktiv
2	Webhook Get	* / 15 * / 6 * * *	WebhookGet	Ja
3	Webhook Put	* / 10 * * * *	WebhookPut	Ja
4	Webhook Post	10 10 * * *	WebhookPost	Ja
6	Mails	* / 15 * * * *	FetchMailsCustom	Ja
7	(Internal) cleanup	0 0 * * *	(Internal) cleanup	Nein
8	(Internal) import-files	* / 15 * * * *	(Internal) import-files	Nein
9	(Internal) import-mails	* / 15 * * * *	(Internal) import-mails	Nein
10	(Internal) unlock-documents	* / 5 * * * *	(Internal) unlock-documents	Nein

+ Neuer Eintrag

1 bis 8 von 8 << < 1 > >> 25

Anlage eines Jobs

Es können beliebig viele Jobs für ein Skript konfiguriert werden. Dabei stehen vorgefertigte Intervalle bereit. Sollten diese nicht ausreichen kann ein [Cron-Ausdruck](#) genutzt werden.

Wir empfehlen **nicht** ein Skript zu häufig oder mehrfach als Job zu konfigurieren. Handelt es sich bspw. bei einem Job um ein Stammdaten-Sync-Job, so kann es passieren, dass diese gleichzeitig ausgeführt werden.

Wir werden dieses System erweitern und entsprechend absichern, aktuell muss dies manuell beachtet werden.

Job anlegen



Name *

Mein Test-Skript

Skript *

WebhookGet



- ☐ Täglich um 3 Uhr nachts
- ☒ Alle 15 Minuten
- ☐ Jeden Sonntag um 12 Uhr Mittags
- ☐ Benutzerdefiniert

Cron Ausdruck *

*/15 * * * *

Aktiv



× Abbrechen

✓ Speichern

Einrichten der Jobverwaltung

Falls Sie Squeeze nicht selbst betreiben, ist dieser Artikel nicht relevant. Vermutlich möchten Sie nur wissen, wie Jobs in der Benutzeroberfläche zu verwalten sind:

[Jobs über Benutzeroberfläche steuern](#)

Einleitung

Ab Squeeze 2 sollen Jobs primär über die UI konfiguriert werden. Somit ist nur noch die Konfiguration eines einzelnen Jobs im Betriebssystems notwendig, der Rest geschieht dann über die UI.

In Squeeze 1 musste noch pro auszuführendem Job in Squeeze ein Job im Betriebssystem geplant werden. Das System ist in der Theorie weiterhin nutzbar, erzeugt allerdings administrativen Mehraufwand.

Job im Betriebssystem einrichten (ab Squeeze 2.5)

Für die Gewährleistung der zeitgesteuerten Ausführungen muss auf dem Host ein Job hinterlegt werden der das PHP-Job-Script `squeezer jobs: schedule --all` immer zur vollen Minute ausführt.

Dieses Skript ist verantwortlich für die eigentliche Job-Ausführung und führt je nach Parameter die Jobs aller Mandanten oder nur der angegebenen Mandanten aus.

Für weitere Informationen können alle Optionen mittels `--help` flag gelistet werden (siehe auch [Einführung zur Squeeze 2.5 CLI](#)):

```
~/Developer/SQUEEZE git:(feature/7040-php-8-docker) (0.182s)
./squeezer jobs:schedule --help
Description:
  Schedules internal jobs for tenants

Usage:
  jobs:schedule [options] [--] [<tenants>...]

Arguments:
  tenants      Argument to pass schedule jobs for specific tenants

Options:
  -a, --all          Flag to schedule jobs for all tenants
  -h, --help          Display help for the given command. When no command is given display help for the list command
  -q, --quiet         Do not output any message
  -V, --version       Display this application version
  --ansi|--no-ansi   Force (or disable --no-ansi) ANSI output
  -n, --no-interaction Do not ask any interactive question
  -v|vv|vvv, --verbose Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug
```

Beispiel Linux

Auf Linux ist besonders darauf zu achten, dass Jobs mittels des `www-data` Users ausgeführt werden.

Falls SQUEEZE auf Linux betrieben wird, ist ein Cronjob (bspw. mittels `sudo crontab -u www-data -e`) einzurichten.

```
* /1 * * * * php /var/www/html/squeeze/squeezer jobs:schedule --all
```

Job im Betriebssystem einrichten (vor Squeeze 2.5)

Für die Gewährleistung der zeitgesteuerten Ausführungen muss auf dem Host ein Job hinterlegt werden der das PHP-Jobs-Script `cli/jobs.php schedule` immer zu vollen Minute ausführt.

Dieses Skript ist verantwortlich für die eigentliche Job-Ausführung und führt Jobs aller Mandanten aus.

Beispiel Linux

Auf Linux ist besonders darauf zu achten, dass Jobs mittels des `www-data` Users ausgeführt werden.

Falls SQUEEZE auf Linux betrieben wird, ist ein cronjob (bspw. mittels `sudo crontab -u www-data -e`) einzurichten.

```
* /1 * * * * php /var/www/html/squeeze/cli/jobs.php schedule
```

Beispiel Windows

In der Windows Aufgabenplanung (Task Scheduler) muss ein Task definiert werden, welcher als Trigger die folgenden Einstellungen aufweist:

- Nach einem Zeitplan
- **Täglich**
- Wiederholen jede **1 Minute** für die Dauer von **1 Tag**
Die 1 Minute kann man manuell eintragen

Neuer Trigger

Aufgabe starten: Nach einem Zeitplan

Einstellungen

☐ Einmal

☒ **Täglich**

☐ Wöchentlich

☐ Monatlich

Start: 11.10.2023 00:00:00 ☐ Zeitzonenübergreifende Synch.

Wiederholung alle: 1 Tage

Erweiterte Einstellungen

☐ Aufgabe verzögern für maximal: 1 Stunde

☒ **Wiederholen jede: 1 Minute für die Dauer von: 1 Tag**

☐ Alle ausgeführten Aufgaben am Ende der Wiederholungsdauer beenden

☐ Aufgabe beenden nach: 3 Tage

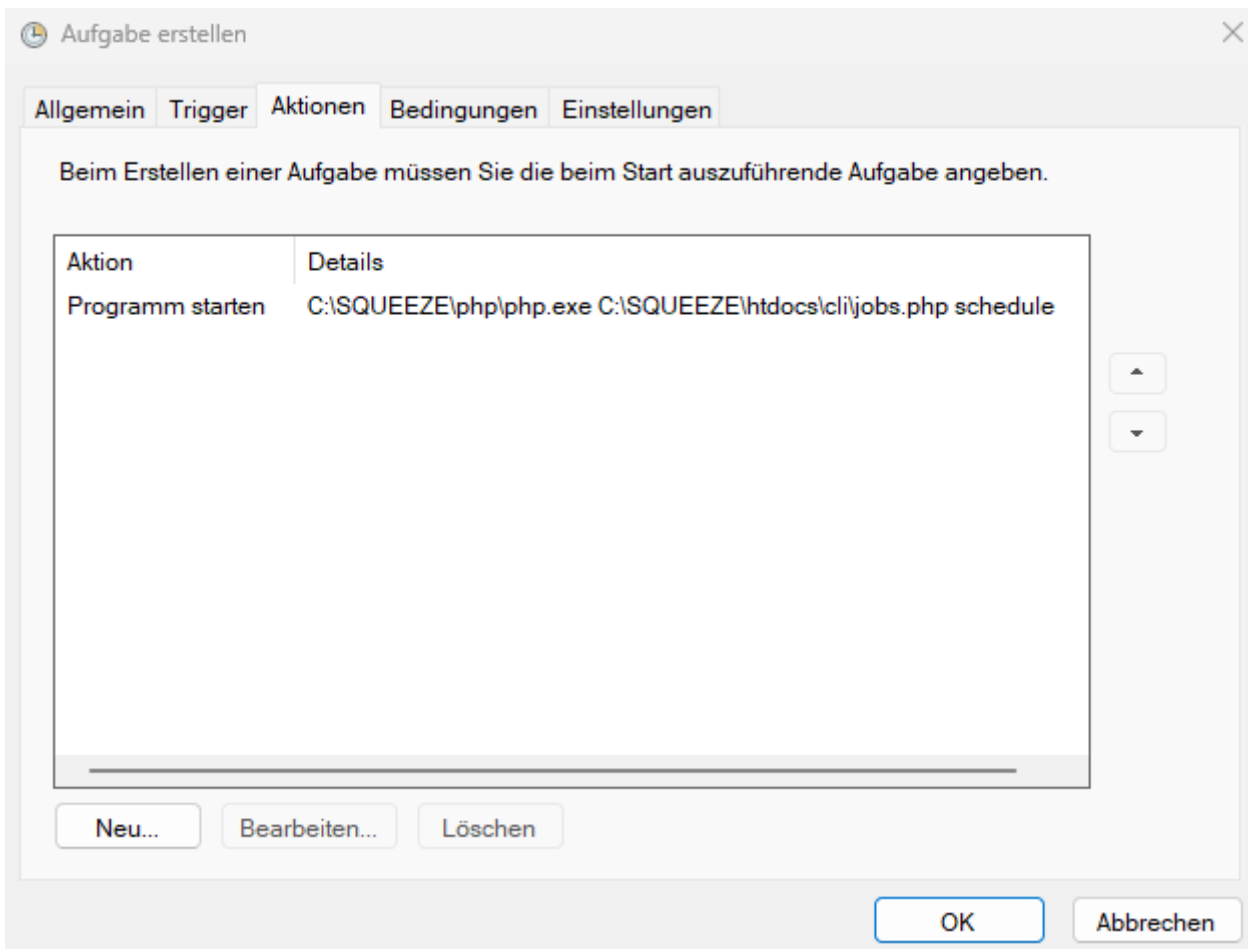
☐ Ablaufen: 11.10.2024 14:02:19 ☐ Zeitzonenübergreifende Synch.

☒ Aktiviert

OK Abbrechen

Als Aktion muss die PHP Executable aus dem SQUEEZE Installationsverzeichnis mit den CLI script `jobs.php` ausgeführt werden:

- Programm starten
- `<INSTALLATIONS_PFAD_SQUEEZE>/php/php.exe`
- Argumente: `<INSTALLATIONS_PFAD_SQUEEZE>/htdocs/cli/jobs.php schedule`
(für alle Mandanten)



Ohne Angabe von Mandanten, werden die Jobs für alle SQUEEZE Mandanten geprüft. Der Mandant kann im Aufruf angegeben werden:

```
<INSTALLATIONS_PFAD_SQUEEZE>/htdocs/cli/jobs.php schedule <tenant>
```

Hinweis bzgl. Performance

Bei einem System mit vielen Mandanten kann es vorkommen, dass die Job-Ausführung länger dauert. Das ist im Kontext von Job-Skripten besonders kritisch, da sich ggf. Jobs "überholen" oder parallel ablaufen könnten. In solchen Fällen kontaktieren Sie bitte den DEXPRO Support, damit gemeinsam evaluiert werden kann, ob eine Sonderanpassung notwendig ist.

Cleanup-Job

Diese Seite dokumentiert, wie Squeeze Daten nach Ablauf ihrer Haltefrist löscht und welche Daten nicht gelöscht werden.

Squeeze 2

Verhalten

Ein Dokument in Squeeze 2 wird nicht mehr benötigt, wenn keine weiteren Bearbeitungsschritte mehr anstehen. Üblicherweise sind das Dokumente in den Steps "Verarbeitet" (Backup) und Gelöscht (Deleted). Squeeze hält diese Dokumente noch eine gewisse Zeit vor und löscht sie danach permanent. Hierbei werden die Dateien (Bilder und PDFs), Einträge in der SQL-DB und Suchindizes (in Elasticsearch) gelöscht, welche aus dem Dokument extrahiert wurden.

Ausgenommen sind Informationen, die für Audits relevant sind.

Haltefrist

Die Haltefrist gibt an wie lange ein Dokument gespeichert bleibt, nachdem es von Squeeze selbst nicht mehr benötigt wird. Als Admin kann man die aktuellen Einstellungen im "System"-Bereich der UI einsehen.

Um die Haltefrist einzustellen, muss man der Mandantenkonfiguration (s. [Server- und Mandantenkonfiguration](#)) dieses Objekt hinzufügen:

```
"cleanupJob": {  
  "maxDocumentAge": "100",  
  "maxLogAge": "0"  
}
```

Eine `maxDocumentAge` vom Wert `0` bedeutet hierbei, dass Dokumente schnellstmöglich vom System gelöscht werden.

Auditlog

Die finale Löschung eines Dokuments wird ab Squeeze 2.3.4 im Auditlog (SQL Tabelle `queueentrychanges`) vermerkt.

Squeeze 1

In Squeeze 1 werden Cleanup-Jobs anders gepflegt als in Squeeze 2. Diese Dokumentation wird aktuell nicht gepflegt, da wir uns auf die Verbesserung und Dokumentation des Cleanup-Jobs von Squeeze 2 konzentrieren.

Unlock-Job

Falls Dokumente von Nutzern gesperrt sein sollten, welche nicht mehr am Dokument arbeiten, kann der "unlock-documents" Job helfen.

Zu finden ist der Job in der Jobverwaltung (Internal) unlock-documents. Standard mäßig läuft der Job alle 5 Minuten und entsperrt alle Dokumente, in welchen zurzeit nicht gearbeitet wird.

Dokumente

Warteschlange

Stapelklassen

Dokumentenklassen

Lokatoren

Stammdaten

Importe

E-Mail

Benutzerverwaltung

Benutzer

Rollen

System

System

Sprachen

Export & Import Konfigurationen

Skripte

Jobverwaltung

Migrationen

System Prüfung

System zurücksetzen

Jobverwaltung

Information

Die aktuelle Serverzeit ist: 11.12.24 11:14:39 UTC +01:00

ID	Name	Cron Ausdruck	Skript	Aktiv	
1	(Internal) cleanup	0 0 * * *	(Internal) cleanup	Nein	
2	(Internal) import-files	* /15 * * * *	(Internal) import-files	Nein	
3	(Internal) import-mails	* /15 * * * *	(Internal) import-mails	Nein	
4	(Internal) unlock-documents	* /5 * * * *	(Internal) unlock-documents	Nein	
5	Refresh Tokens	* /15 * * * *	(Internal) refresh-tokens	Ja	

+ Neuer Eintrag

1 bis 5 von 5 << < 1 > >> 25