

TYPO3 scheduler

was ist der *Scheduler*?

- Automatisierung von Aufgaben
- Nutzung vom System Befehl cron (Linux, Unix, MacOS) bzw. at (Windows)
- Verfügbar ab Version 4.3
- Voraussetzung: Zugang zu Shell



Einsatzbereiche

- Synchronisation von TYPO3 Instanzen
- Datensicherung
- Mailversand
- Datenbankoperationen (Import / Export)



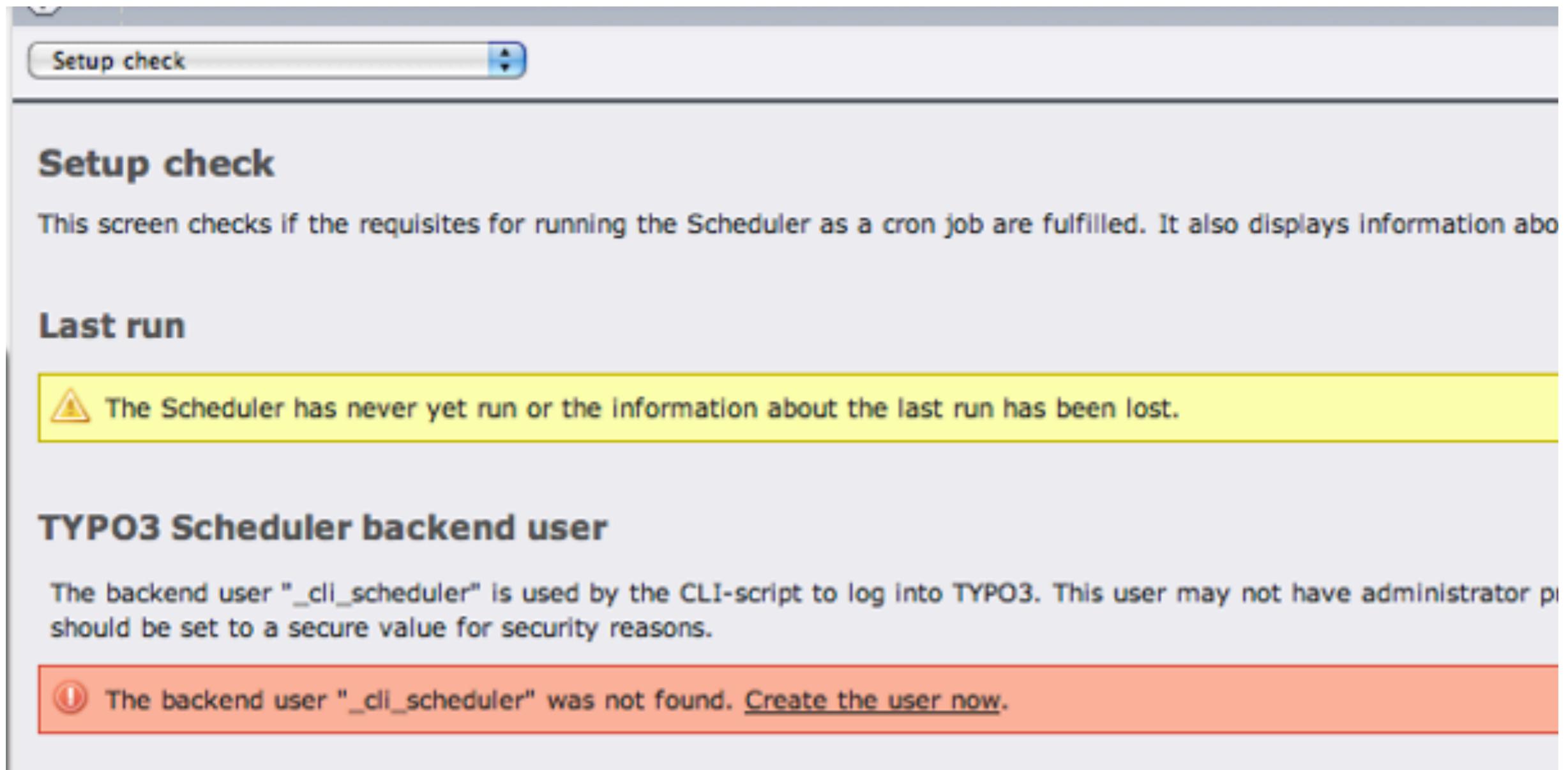
Installation



- Import von Scheduler im Installtool
- Einstellungen:
 - Maximum lifetime: Wie lange ist ein task gültig, bevor er abgebrochen wird.
 - Enable logging: Aktiviert Statusmeldungen (sinnvoll, solange Scheduler eingerichtet wird)
 - Enable Sample tasks: Zeigt die Beispieltasks an

Einrichtung im Backend

Setup check im Modul Scheduler laufen lassen



The screenshot shows the 'Setup check' page in the TYPO3 Backend Scheduler module. At the top, there is a breadcrumb trail 'Setup check'. Below it, the page title 'Setup check' is followed by a descriptive paragraph: 'This screen checks if the requisites for running the Scheduler as a cron job are fulfilled. It also displays information about...'. Under the heading 'Last run', a yellow warning box contains the message: 'The Scheduler has never yet run or the information about the last run has been lost.' Below this, the section 'TYPO3 Scheduler backend user' explains that the user '_cli_scheduler' is used by the CLI-script and may not have administrator permissions. A red error box at the bottom states: 'The backend user "_cli_scheduler" was not found. [Create the user now.](#)'

Cron Job einrichten

- Aufruf von: „crontab -e“
- neue Zeile einfügen, z.B.
`*/15 * * * * /usr/local/bin/php /home/bob/www/
 typo3/cli_dispatch.phpsh scheduler`
- in diesem Beispiel wird das Skript alle 15min aufgerufen
- Beim editieren von Systemtabellen (/etc/crontab and /etc/cron.d/*), muss noch die Gruppe hingefügt werden
- **Beispiel:** `*/15 * * * * www /usr/local/bin/php /home/bob/
 www/typo3/cli_dispatch.phpsh scheduler`



Architektur

- Zwei Klassen notwendig:
 - myClass extends tx_scheduler_Task
 - myClassFields implements tx_scheduler_AdditionalFieldProvider (optional)
- Der einfachste Weg: Kopieren der Beispiele

Autoload

Die verwendeten Klassen müssen bekannt gemacht werden:

Dazu wird die Funktion Autoload genutzt.

`ext_autoload.php`

```
return array(
    'tx_saupdatemailer_scheduler' => t3lib_extMgm::extPath('saupdatemailer', 'classes/
class.tx_saupdatemailer_scheduler.php'),
    'tx_saupdatemailer_schedulermailqueue' => t3lib_extMgm::extPath('saupdatemailer',
'classes/class.tx_saupdatemailer_schedulermailqueue.php'),
);
```

Klassen bekannt machen

Eintrag in `TYPO3_CONF_VARS` ergänzen

```
$GLOBALS['TYPO3_CONF_VARS']['SC_OPTIONS']['scheduler']['tasks']['tx_scheduler_TestTask'] =
    'extension'           => $_EXTKEY,
    'title'               => 'LLL:EXT:' . $_EXTKEY . '/locallang.xml:testTask.name',
    'description'        => 'LLL:EXT:' . $_EXTKEY . '/locallang.xml:testTask.description',
    'additionalFields'   => 'tx_scheduler_TestTask_AdditionalFieldProvider'
```



Beispiel

Probleme / Verbesserungen



- Achtung auf verschiedene IP Adressen
 - Webserver selbst
 - aufrufender Klient
- AdditionalFieldProvider: Keine Nutzung des TCE Rendering möglich

Vielen Dank!

Stefan Busemann

stefan.busemann@in2form.com