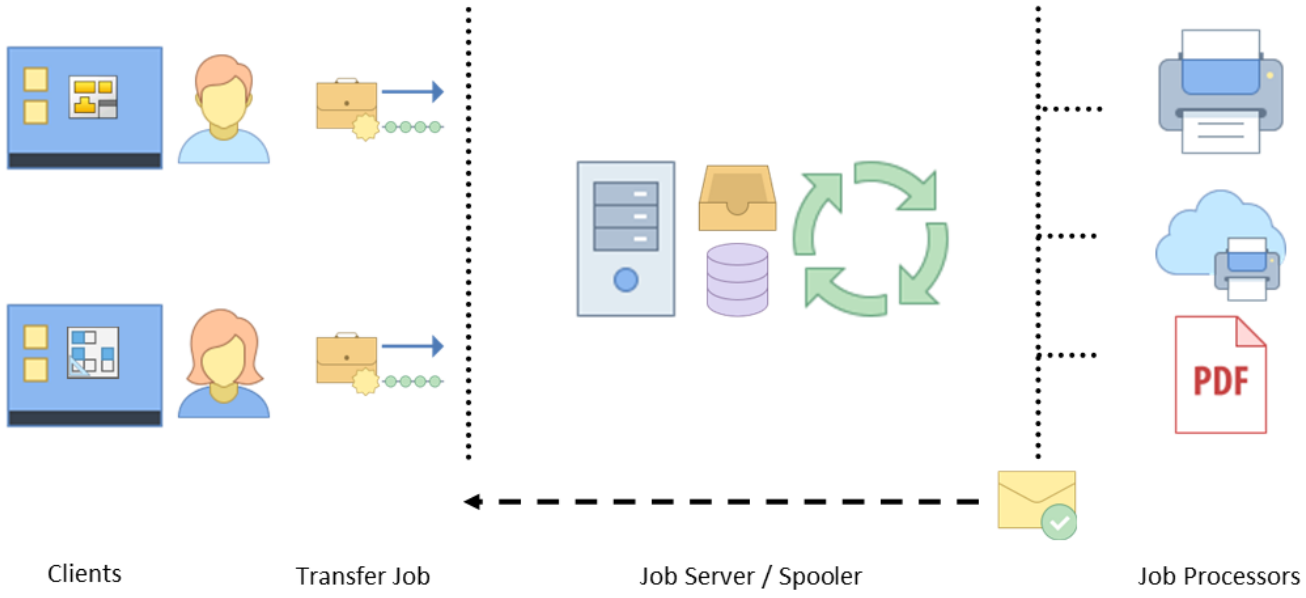
Jobserver

Der speedy-Jobserver erledigt zeitaufwendige Druckaufträge oder erstellt Vorschaubilder für speedWEB an einer zentralen Stelle.
Weiterhin kann der Jobserver das erstellen der Neutralformate bei der Freigabe oder anderer Statuswechsel übernehmen.



Um einen Jobserver einzurichten wird der Spooler-Mechanismus genutzt. Jobs können nur abgearbeitet werden wenn die entsprechende Konfiguration erstellt wurde und die entsprechenden Module vorhanden sind. Es wird eine Transferdatei genutzt um die Jobs an den Jobserver zu übergeben. Diese Datei hat die Endung *.jsjob. Wird ein Jobserver verwendet werden die Jobs auf dem Server ausgeführt.
Dies bedeutet:

- Bei Druckjobs muss das verwendete CAD System vorhanden sein.
- Bei Erstellung von Neutralformaten müssen die entsprechenden CAD Systeme oder die mit dem Originaldateityp verknüpften Anwendungen vorhanden sein.
- Entsprechende Leistung (CPU, Arbeitsspeicher, Grafik, ...) muss auf dem Server vorhanden sein.

Momentan verfügbare Jobs:

- jsPlot: Stapeldruck auf dem Jobserver
- jsRendition: Bildgenerierung/Vorschaugenerierung auf dem Jobserver
- jsNeutralformat: Neutralformatgenerierung bei Statuswechsel auf dem Jobserver
- weitere Jobs auf Anfrage bzw. in Vorbereitung

Ablauf des Jobserver

Der Jobserver überwacht den Posteingang. In den meisten Fällen ein entsprechendes freigegebenes Verzeichnis.
Sendet ein Client eine Jobdatei (*.jsjob) in den Posteingang des Jobservers wird diese eingelesen und

in der Warteschleife (js_jobs in der Datenbank) eingetragen.
Die Warteschleife wird abgearbeitet in dem zu jedem Job der passende Processor gestartet wird, wenn der Job an der Reihe ist.

Im Fall von Druckjobs ist die Stapeldruck-Einstellung im Job enthalten. Der Druckjob wird wie der Stapeldruck lokal nur auf dem Server ausgeführt. Die entsprechenden Anwendungen müssen auf dem Server zur Verfügung stehen.

jsPlot

Dieser Job kann dafür verwendet werden die Aufgaben und die Druckaufträge des Stapeldrucks auf den Server zu verlagern. Dies hat den Vorteil dass Zeitaufwendige Generierungen keine Zeit auf den Clients beanspruchen. Es muss zwingend Notwendig das benutzte CAD-Programm oder Office Programm (je nach Druckregeln) auf dem Server installiert und lizenziert sein, da sonst kein Druck möglich ist.
Der Jobserver arbeitet die Jobaufträge nach einander ab. Dies bedeutet dass die Ausführung und Generierung unter bestimmten Umständen länger dauern kann.

Einstellungen

Name	Beschreibung
plot.jobserver.available	Gibt an ob der Jobserver installiert und verfügbar ist. Standardwert=0
plot.jobserver.path	Pfad zu den Jobdateien.
plot.jobserver.timeslot	Zeitangabe wann der Jobserver Jobs ausführen darf. Beispiel: 12:00-13:00;19:00-23:00;
plot.jobserver.maxprio	Prioritätsschwelle über der die Druckjobs trotz Timeslot trotzdem ausgeführt werden.
plot.jobserver.sendmsg	Gibt an ob beim beenden des Jobs eine Nachricht gesendet werden soll.
plot.jobserver.receiver	Gibt die zu empfangenden User oder Gruppe an an den die Nachricht gesendet werden soll.

js_types

types_name	types_parallelcount	types_maxtime	types_killable	types_maxrestart	types_settinghead	types_exepath	types_flag
plot	1	50000	1	1	plot	c:\DevProjects\speedy\Dev\bin64\jsPlot.exe	0

jsRendition

Dieser Job ist dafür gedacht Vorschaubilder für das Web Frontend zu generieren. Diese Generierung erfolgt im Hintergrund damit die Clients damit nicht belastet werden. Es muss zwingend Notwendig das benutzte CAD-Programm auf dem Server installiert und lizenziert sein, da sonst keine Generierung der Vorschaubilder möglich ist.
Der Jobserver arbeitet die Jobaufträge nach einander ab. Dies bedeutet dass die Ausführung und Generierung unter bestimmten Umständen länger dauern kann.

Einstellungen

Name	Beschreibung
rendition.jobserver.available	Gibt an ob der Jobserver installiert und verfügbar ist. Standardwert=0
rendition.jobserver.path	Pfad zu den Jobdateien.
rendition.jobserver.timeslot	Zeitangabe wann der Jobserver Jobs ausführen darf. Beispiel: 12:00-13:00;19:00-23:00;
rendition.jobserver.maxprio	Prioritätsschwelle über der die Druckjobs trotz Timeslot trotzdem ausgeführt werden.
rendition.jobserver.sendmsg	Gibt an ob beim beenden des Jobs eine Nachricht gesendet werden soll.
rendition.jobserver.receiver	Gibt die zu empfangenden User oder Gruppe an an den die Nachricht gesendet werden soll.

js_types

types_name	types_parallelcount	types_maxtime	types_killable	types_maxrestart	types_settinghead	types_exepath	types_flag
rendition	1	50000	1	1	rendition	c:\DevProjects\speedy\Dev\bin64\jsRendition.exe	0

jsNeutralformat

Dieser Job kann dafür verwendet werden Neutralformate die bei einem Statuswechsel in speedyPDM generiert werden auf den Server zu verlagern. Dies hat den Vorteil dass zeitaufwendige Generierungen keine Zeit auf den Clients beanspruchen. Es muss Zwingend Notwendig das benutzte CAD-Programm auf dem Server installiert und lizenziert sein, da sonst keine Neutralformat-Generierung möglich ist.
Der Jobserver arbeitet die Jobaufträge nach einander ab. Dies bedeutet dass die Ausführung und Generierung unter bestimmten Umständen länger dauern kann.

Einstellungen

Name	Beschreibung
neutralformat.jobserver.available	Gibt an ob der Jobserver installiert und verfügbar ist. Standardwert=0
neutralformat.jobserver.path	Pfad zu den Jobdateien.
neutralformat.jobserver.timeslot	Zeitangabe wann der Jobserver Jobs ausführen darf. Beispiel: 12:00-13:00;19:00-23:00;
neutralformat.jobserver.maxprio	Prioritätsschwelle über der die Druckjobs trotz Timeslot trotzdem ausgeführt werden.
neutralformat.jobserver.sendmsg	Gibt an ob beim beenden des Jobs eine Nachricht gesendet werden soll.
neutralformat.jobserver.receiver	Gibt die zu empfangenden User oder Gruppe an an den die Nachricht gesendet werden soll.

js_types

types_name	types_parallelcount	types_maxtime	types_killable	types_maxrestart	types_settinghead	types_exepath	types_flag
neutralformat	1	50000	1	1	neutralformat	c:\speedy7\bin64\jsNeutralFormat.exe	0

Konfiguration

Um den Spooler als Jobserver zu Nutzen ist folgende Konfiguration in der dwSpool.ini zu machen:

```
[WATCHDIR]
WatchDirX = //"Pfad für die Ablage der Jobdateien"    (siehe auch
[*.jobserver.path]) //

[WatchDirX.JSJOB]
Filter=*.jsjob
Description=
Command=%1
ShowWindow = 1
```

Dies dient als Beispiel und kann ihrer Konfiguration angepasst werden.
Desweiteren werden 3 Tabellen in der Datenbank benötigt:

- js_jobs
- js_proc
- js_types

Diese 3 Tabellen nutzt der Jobserver um die Jobs abzuarbeiten.

In der **js_types** Tabelle werden die zur Verfügung stehenden Jobs definiert

Schlüssel	**Beschreibung**
types_id	ID des Typs
types_name	Name des Types. Beispiel: plot→Beschreibt den Druckjob
types_parallelcount	Beschreibt wieviele Prozesse von diesem Typ gleichzeitig gestartet werden können
types_maxtime	Maximale Zeit die ein Job brauchen darf(in Millisekunden). Nach dieser Zeit wird der Job beendet. Dies kann genutzt werden um aufgehängte Jobs neu starten zu können
types_killable	Gibt an ob nach abgelaufener maxtime der Prozess beendet werden kann oder nicht
types_maxrestart	Max Wert wie oft ein Prozess erneut gestartet werden kann
types_settinghead	Gibt den Knoten der Eigenschaften an die zu diesem Job gehören. Beispiel: plot→ dadurch werden die Einstellungen plot.jobserver eingelesen und verwendet. *.jobserver.available gibt an ob der Jobserver aktiv ist. *.jobserver.path gibt an welches Verzeichnis überwacht werden soll.
types_exepath	Vollständiger Pfad zur Job-Anwendung.
types_flag	Allgemeines Flag

Beispielkonfiguration eines Drucktyps:

types_id	types_name	types_parallelcount	types_maxtime	types_killable	types_maxrestart	types_settinghead	types_exepath	types_flag
2	plot	1	50.000	1	1	plot	[redacted]jsPlot.exe	0

Konfigurationsparameter

Name	**Beschreibung**
jobserver.database.keepAll	Definiert, ob abgearbeitete Jobs, mit entsprechender Markierung, in der Datenbank verbleiben oder aus der Warteschleife gelöscht werden.

Name	Beschreibung
jobserver.archiveDir	Archivverzeichnis.
jobserver.errorDir	Verzeichnis für fehlerhafte Druckjobs.
jobserver.timeslot	Zeitangabe wann der Jobserver Jobs ausführen darf. Beispiel: 12:00-13:00;19:00-23:00;
jobserver.maxprio	Prioritätsschwelle über der die Druckjobs trotz Timeslot trotzdem ausgeführt werden.
jobserver.message.subject	Betreff der Nachricht die automatisch gesendet wird.
jobserver.message.body	Nachricht die bei der automatisch generierten Nachricht gesendet werden soll.
jobserver.servername	Gibt den Computername vom Jobserver an.

dwSessionSvc

Über die dwSessionSvc kann die Session als Dienst ausgeführt werden. Dies kann helfen die Anmeldezeit in Speedy-Modulen drastisch zu verbessern. Hauptsächlich macht der dienst auf Servermaschinen wie z.B. der Jobserver Sinn.
Die dwSessionSvc.exe muss im bin64 Verzeichnis liegen(momentan noch nicht mit ausgeliefert, bei mmh nachfragen). Zur Installation muss eine Kommandozeile(cmd) als Administrator ausgeführt werden. Es gibt folgende 3 Übergabeparameter:

Parameter	Beschreibung
-Service	Der Service wird als Dienst im Windows registriert. Er kann dann auch unter den Windows-Diensten eingesehen und Konfiguriert werden.(Präferiert)
-RegServer	Service wird mit COM-Modul registriert.
-UnregServer	Lokaler Server wird Deregistriert. Ebenso der Service.

Am besten danach den Dienst auf Automatisch(Verzögerter Start) stellen.

From:
https://wiki.speedy-pdm.de/ - **speedyPDM - Wiki**

Permanent link:
https://wiki.speedy-pdm.de/doku.php?id=speedy:30_modules:jobserver

Last update: **2024/03/21 09:22**