Squeeze Pull

- 01 – Übersicht
- 02 – Installation
- 03 – Konfiguration
- 04 – Betrieb und Fehlersuche

01 – Übersicht

Der Squeeze Pull-Job holt fertig gelesene Belege per Squeeze REST-API v2 nach DOCUMENTS ab und ersetzt den SOAP-Export. Diese Seite beschreibt Funktionsweise und Statuswerte für den Mappentyp Invoice. Stand 09.11.2026.

Was ist der Pull-Job

Bisher hat Squeeze fertig gelesene Belege per SOAP nach DOCUMENTS geschoben. Beim Pull holt DOCUMENTS die Belege selbst über die REST-API v2 von Squeeze ab. Je Mappentyp gibt es ein eigenes Job-Skript, das im Scheduler viertelstündlich läuft und die generische Bibliothek `Squeeze_Pull_Lib` aufruft.

Der Job fragt im Queue-Step „Export“ nur die Belege der konfigurierten Squeeze-Klasse ab. Je Beleg legt er einen Vorgang an oder aktualisiert einen vorhandenen. Er schreibt Kopffelder, Positionen (Gentable) und Anhänge und meldet den Beleg danach über das Export-Interface „Pull“ als FINISHED an Squeeze. Erst dann gilt der Vorgang als fertig und darf in den Workflow.

Die Zuordnung Beleg → Vorgang läuft über das Feld `SqueezeID`. Findet der Job darüber keinen Vorgang, sucht er zusätzlich über die externalId des Belegs (Belege aus der SOAP-Zeit).

> Die Zuordnung Beleg → Vorgang läuft ausschließlich über das Feld `SqueezeID`.

Ablauf je Beleg

1. **Vorgang suchen:** zuerst über `SqueezeID`, sonst über die externalId des Belegs. Dabei zählt nur ein aktiver Vorgang desselben Mappentyps ohne fremde `SqueezeID`.
2. **Vorgang anlegen:** Gibt es keinen Vorgang, wird er neu angelegt und **sofort** mit `SqueezeID` und `SqueezeStatus = Unfinished` gespeichert. Einen Vorgang auf `Processed` überschreibt der Job nie.
3. **Daten schreiben:** Kopffelder, Positionen und Anhänge, danach speichern. Felder, die es am Mappentyp nicht gibt, werden übersprungen (Hinweis im Log). Der Beleg wird trotzdem übernommen.
4. **Abschließen:** Bei Erfolg geht FINISHED an Squeeze, dann wird `SqueezeStatus = Processed` gesetzt, optional mit direktem Workflow-Start. Bei einem Fehler wird `SqueezeStatus = ERROR` gesetzt und kein FINISHED gemeldet. Der Beleg bleibt im Export und der nächste Lauf versucht es erneut.

Häufigkeit und Menge je Lauf

Beides ist einstellbar:

- **Ausführungshäufigkeit:** über die Einplanung des Jobs `Invoice_JOB_SqueezePull` im Scheduler.
- **Belege je Lauf:** über den Schalter `MAX_DOCS_PER_RUN` in `Squeeze_Pull_Lib`, im Paket `100`. Was ein Lauf nicht schafft, übernimmt der nächste.

Empfehlung: viertelstündlich mit 100 Belegen starten. Staut sich der Export, zuerst das Intervall verkürzen (z.B. auf 5 Minuten), statt `MAX_DOCS_PER_RUN` stark zu erhöhen. Kurze Läufe machen Fehler schneller sichtbar, und bis DOCUMENTS 6.1 gibt es keinen HTTP-Timeout für die Squeeze-Aufrufe.

> `MAX_DOCS_PER_RUN` ist kein Parameter, sondern eine Konstante in `Squeeze_Pull_Lib` und gilt damit für alle Mappentypen. Es läuft immer nur ein Lauf gleichzeitig. Ist ein Lauf beim Start der nächsten Ausführung noch nicht fertig, fällt diese Ausführung aus. Die restlichen Belege holt der nächste Lauf ab (Parallel-Sperre, siehe *Betrieb und Fehlersuche*).

SqueezeStatus am Vorgang

Wert	Bedeutung	Workflow	Nächster Lauf
`Unfinished`	wird übernommen oder Lauf abgebrochen	nein	macht mit demselben Vorgang weiter
`ERROR`	Fehler erkannt, Grund im Log	nein	versucht es erneut
`Processed`	fertig, FINISHED von Squeeze bestätigt	ja	überspringt, überschreibt nie

Die Werte `New`, `Analyze`, `Split` usw. stammen aus SOAP bzw. der Analyse. Der Pull-Job setzt sie nicht.

> **Wichtig:** Den Workflow-Start für `Unfinished` und `ERROR` sperrt der Userexit `ue_beforeStartWorkflow` in `Invoice__UserExit_CustomLib`. Fehlt er, startet `Invoice_JOB_StartWorkflow` auch unfertige Vorgänge.

02 – Installation

Diese Seite beschreibt Voraussetzungen, Einbau und Einbautest des Squeeze Pull-Jobs für den Mappentyp Invoice. Die Parameter und Schalter sind auf der Seite *Konfiguration* beschrieben.

Voraussetzungen

- DOCUMENTS 5/6 mit DEXPRO-Standard (Import-, Parameter- und Logging-Bibliothek sowie `Squeeze_FunctionLib` mit v2-Unterstützung)
- Squeeze mit REST-API v2, Zugang in der Squeeze-Konfiguration von DOCUMENTS hinterlegt
- Mappentyp mit `SqueezeID` (Text, durchsuchbar, indiziert), `SqueezeStatus` (durchsuchbar), Register für Anhänge und Gentable

> Feste Regeln: nur **ein** DOCUMENTS-System je Squeeze-Instanz, nur **ein** Job je Mappentyp und nur auf **einem** Server. SOAP und Pull nie gleichzeitig an derselben Klasse, sonst entstehen Dubletten.

Bestandteile

Kategorien: die `Squeeze_*`-Skripte in **SqueezePull**, der Job in **Invoice**, der Userexit in **Invoice_UserExit**.

Skript	Aufgabe	Anpassen?
`Invoice_JOB_SqueezePull`	Job: Job-User, SuperMode, Log, liest Parameter, ruft `runSqueezePull()`	je Mappentyp ein eigenes Job-Skript
`Squeeze_Pull_Lib`	Ablauf, gleich für alle Mappentypen	nein, nur die Schalter am Dateianfang (siehe *Konfiguration*)
`Squeeze_FunctionApiLib`	REST-Aufrufe, `SqueezeAPI` erbt von `Squeeze`	nein, aber abgleichen
`Squeeze_CustomLib`	Userexit `ue_Squeeze_ParseFieldValue`	**ja, je Kunde**
`Squeeze_ENUM_DocFile_SqueezeStatus`	Standard-Enum plus Unfinished, ERROR, Processed	nein
`Invoice__UserExit_CustomLib`	`ue_beforeStartWorkflow`: sperrt den Workflow-Start bei `Unfinished` und `ERROR`	ja, Userexit an die Kundenfassung anfügen

Unverändert bleiben die Standardskripte `Squeeze_FunctionLib`, `DEXPRO__ImportLib`, `Invoice__ImportLib`, `Invoice_JOB_StartWorkflow` und `DEXPRO_ENUM_DocFile_SqueezeStatus` (bei Kunden oft verschlüsselt).

Einbau

Sicherung

Vor dem Einspielen exportieren:

- `Invoice__UserExit_CustomLib` des Kunden
- `Squeeze_FunctionApiLib`, falls schon vorhanden

Skripte

- `Squeeze_Pull_Lib`, `Squeeze_CustomLib` und `Squeeze_ENUM_DocFile_SqueezeStatus` in die Kategorie **SqueezePull** importieren.
- `Squeeze_FunctionApiLib` in die Kategorie **SqueezePull** importieren. Gibt es schon eine Fassung, abgleichen statt überschreiben: Nur `getQueueStepEntries`, `getExportInterfaceIdByName`, `postRequest2xx` und `setExportInterfaceStatus` sind für den Pull neu bzw. erweitert.
- `Invoice_JOB_SqueezePull` in die Kategorie **Invoice** importieren.

Den Userexit `ue_beforeStartWorkflow` an die Kundenfassung von `Invoice__UserExit_CustomLib` **anfügen**, den Rest nicht ändern.

> Enthält das Skript schon ein `ue_beforeStartWorkflow`, dann muss die Prüfung in die vorhandene Funktion eingebaut werden.

Parameter

`Squeeze_DocumentClassName` und bei Bedarf `Squeeze_StartWorkflow` anlegen, wie auf der Seite *Konfiguration* beschrieben. Angelegt werden die Parameter über die Weboberfläche.

Übersetzungen

In der Weboberfläche unter **Übersetzungen** drei Einträge anlegen.

Feld	Eintrag 1	Eintrag 2	Eintrag 3

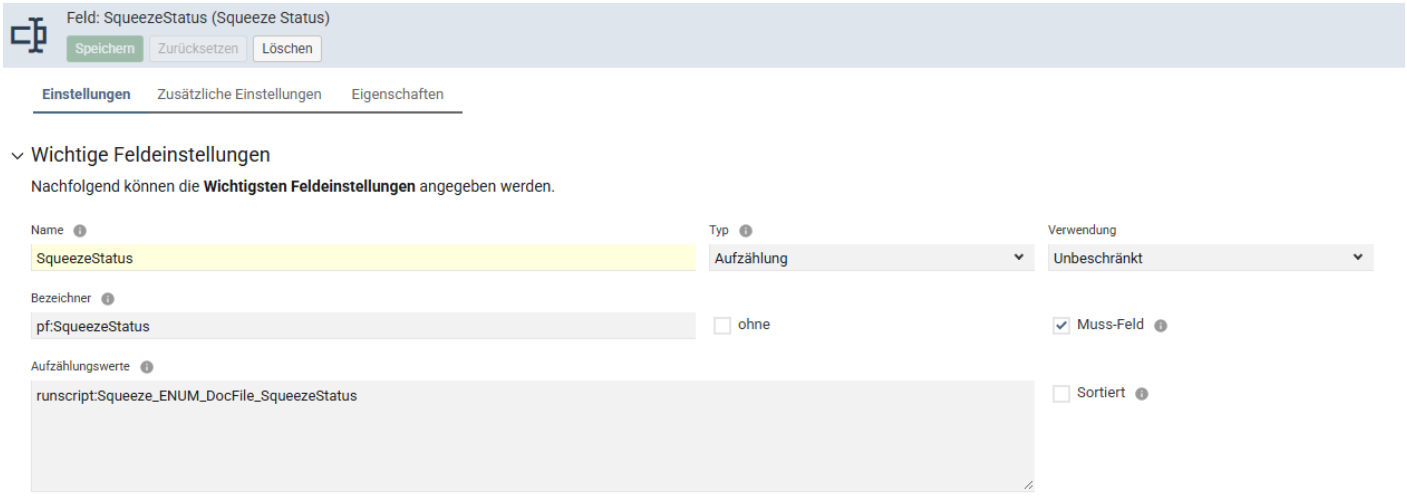
Applikation	Documents	Documents	Documents
Typ	Properties	Properties	Properties
Unterkategorie	Squeeze	Squeeze	Squeeze
Name	`SqueezeStatus_Unfinished`	`SqueezeStatus_Error`	`SqueezeStatus_Processed`
Projektspezifische Anpassung	nein	nein	nein
Deutsch	Übernahme läuft / unvollständig	Übernahme fehlgeschlagen	Übernommen
Englisch	Transfer incomplete	Transfer failed	Transferred

Danach die Properties erzeugen, sonst zeigt das Feld die neuen Werte nicht übersetzt an.

Mappentyp Invoice

- Feld `SqueezeStatus` auf `runscript:Squeeze_ENUM_DocFile_SqueezeStatus` umstellen
- Register aus Wert 2 (und ggf. Wert 3) des Parameters und Gentable vorhanden
- Öffentlichen Filterordner für Invoice mit `SqueezeStatus = Unfinished` ODER `ERROR` anlegen

Das Feld `SqueezeStatus` im Manager bzw. im Webclient nach der Umstellung:

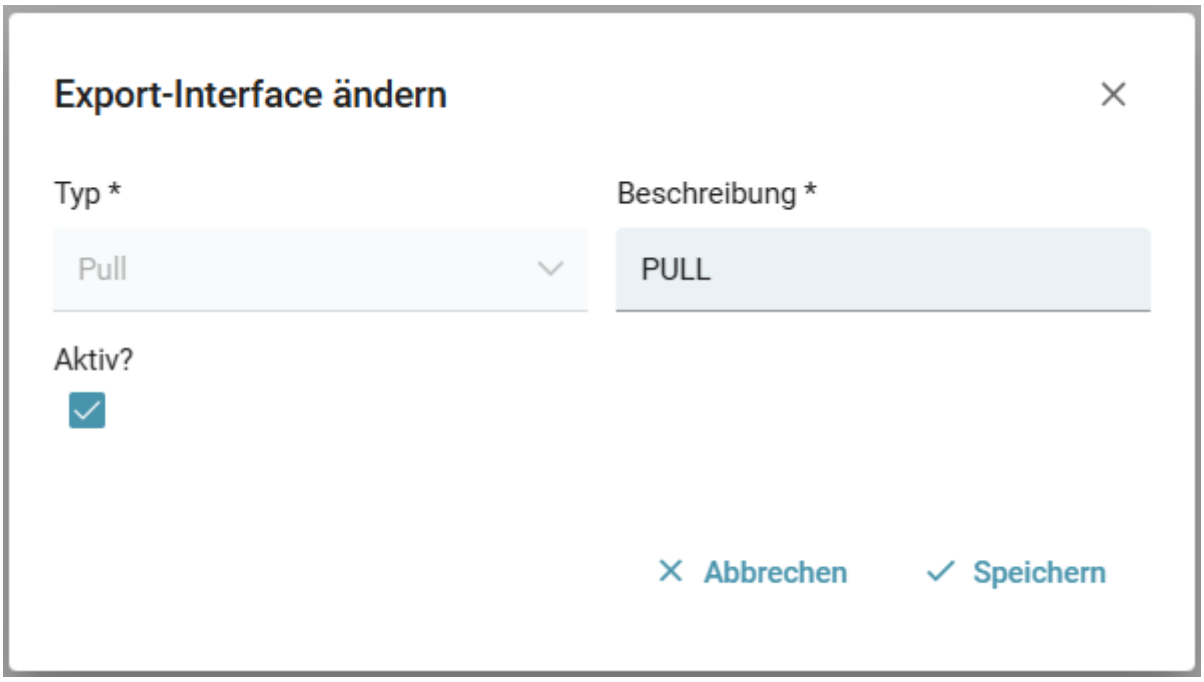


Squeeze

1. SOAP-Export der Klasse ausschalten.
2. Export-Interface „Pull“ an der Klasse anlegen und aktivieren.
3. Prüfen, dass der Queue-Step „Export“ heißt.

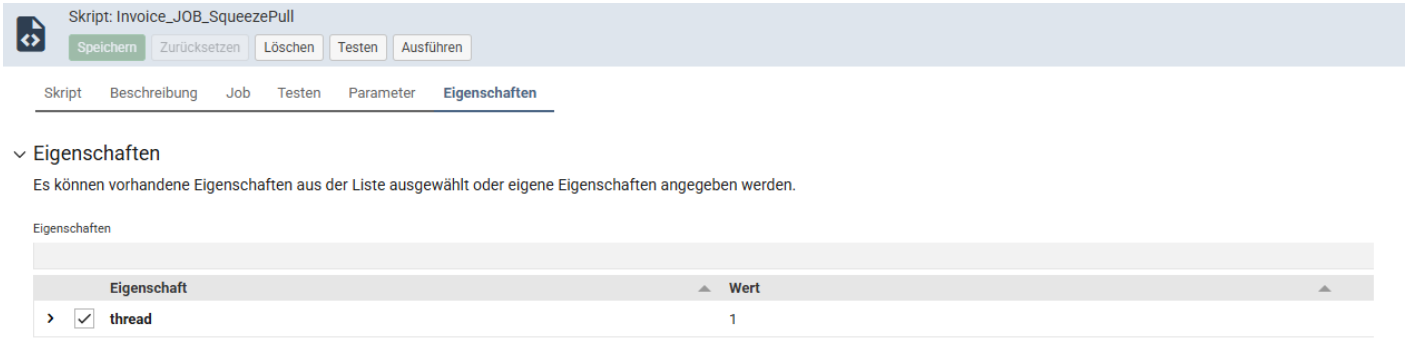
4. Mapping in der Spalte **Name (Zielsystem)** prüfen: jedes DOCUMENTS-Feld nur einmal zugeordnet (Details auf der Seite *Konfiguration*).

Das Export-Interface „Pull“ an der Klasse:



Jobs

`Invoice_JOB_SqueezePull` mit der Eigenschaft `thread=1` anlegen und erst nach dem Test viertelstündlich einplanen.



`Invoice_JOB_StartWorkflow` nicht ändern. Er wird nur gebraucht, wenn der Parameter `Squeeze_StartWorkflow` nicht auf `true` steht.

> Vor dem ersten Start von `Invoice_JOB_StartWorkflow`, auch im Einbautest Schritt 2, prüfen, ob es alte Vorgänge mit leerer WorkflowID gibt. Sie würden ebenfalls gestartet werden.

Einbautest

Der Test läuft mit einem einzelnen Beleg in zwei Durchgängen: zuerst ohne Rückmeldung an Squeeze, dann mit. Vorher in `Squeeze_Pull_Lib` `TEST_KEEP_IN_EXPORT = true` und `MAX_DOCS_PER_RUN = 1` setzen.

#	Aktion	Erwartung
0	Invoice-Vorgang öffnen	`SqueezeStatus` bietet zusätzlich „Übernahme läuft / unvollständig“, „Übernahme fehlgeschlagen“ und „Übernommen“ an.
1	`Invoice_JOB_SqueezePull` von Hand starten	Der Vorgang wird angelegt und steht auf „Unfinished“, keine FEHLT-Zeile im Log. Der Beleg bleibt in Squeeze im Export, weil der Testmodus kein FINISHED meldet.
2	`Invoice_JOB_StartWorkflow` von Hand starten	Prüft die Workflow-Sperre: Der Vorgang aus Schritt 1 bekommt keinen Workflow, im Log steht `ue_beforeStartWorkflow() -> false`.
3	In `Squeeze_Pull_Lib` `TEST_KEEP_IN_EXPORT = false` setzen, `Invoice_JOB_SqueezePull` erneut starten	Jetzt die echte Übernahme: FINISHED im Log, Vorgang auf „Processed“, Beleg in Squeeze aus dem Export verschwunden.
4	Workflow prüfen	Mit `Squeeze_StartWorkflow = true`: im Pull-Log „Workflow … gestartet“. Sonst `Invoice_JOB_StartWorkflow` starten, der Vorgang bekommt jetzt seinen Workflow.

Nach dem Test in `Squeeze_Pull_Lib` `MAX_DOCS_PER_RUN` wieder auf `100` setzen (`TEST_KEEP_IN_EXPORT` steht seit Schritt 3 auf `false`). Dann `Invoice_JOB_SqueezePull` viertelstündlich einplanen und, falls der Workflow nicht direkt im Pull-Lauf startet, auch `Invoice_JOB_StartWorkflow`.

03 – Konfiguration

Diese Seite beschreibt die Parameter, das Mapping in Squeeze und die Schalter in `Squeeze_Pull_Lib`.

Parameter

Parameter werden in der Weboberfläche auf der Outbar **Administration Invoice** im Ordner **Parameter** angelegt und gepflegt, nicht per SQL. Beide Parameter haben die Vorlage **Invoice** und gelten global. Als Job-User nutzt der Job den vorhandenen Parameter `JobScriptUser`, sonst `cJobUser`.

Squeeze_DocumentClassName (Pflicht)

Bezeichnung: *Squeeze: Dokumentklasse und Register*

Feld	Wert 1	Wert 2	Wert 3
Beschreibung	Squeeze-Dokumentklasse	Register fuer Anhaenge	Register je Dateiendung (optional)
Typ	String	String	String
Standardwert	leer	leer	leer
Eintrag (Beispiel)	`Invoices`	`Documents`	leer oder `Documents2:eml,msg`

Wert 3 hat das Format `Register:Endung,Endung`. Mehrere Regeln werden mit `;` getrennt, zum Beispiel `Documents2:eml,msg;Documents3:xml`. Endungen ohne Punkt angeben, Groß- und Kleinschreibung spielt keine Rolle. Ist Wert 3 leer, landen alle Anhänge im Register aus Wert 2.

> Ist Wert 3 ungültig, holt der Job nichts ab. Fehlt ein Register aus Wert 2 oder 3 am Mappentyp, verarbeitet der Job die Klasse nicht oder bricht beim ersten Beleg ab; dieser Vorgang steht dann auf `ERROR`. Die Belege bleiben im Export-Step, der Grund steht im Log.

Squeeze_StartWorkflow (optional)

Bezeichnung: *Squeeze: Workflow direkt nach Übernahme starten*

Feld	Wert 1	Wert 2

Beschreibung	Workflow direkt starten	Workflow-Name (leer = WorkflowRules)
Typ	Boolean	String
Standardwert	`false`	leer
Eintrag (Beispiel)	`true`	leer

Einen Eintrag nur anlegen, wenn der Workflow direkt im Pull-Lauf starten soll. Ohne Eintrag startet `Invoice_JOB_StartWorkflow` den Workflow.

> Standardwert von Wert 1 auf `false` setzen. Bei `true` kann der Workflow direkt starten, obwohl kein Eintrag angelegt ist.

Mapping in Squeeze

Das Mapping steht in Squeeze an der Dokumentklasse unter **Felder** bzw. **Tabellen** in der Spalte **Name (Zielsystem)**. In der REST-API heißt sie `externalName`. Spalten ohne Eintrag werden nicht übernommen. Jedes DOCUMENTS-Feld darf nur einmal zugeordnet sein; der Job meldet doppelt zugeordnete Kopffelder beim Start.

ID	Name	Beschreibung	Typ	Standard Wert	Lokator	Lokator-Unterfeld	Name (Zielsystem)
2	Company	Mandant	Text	1000	Mandant		Principal
195	CompanyName	Mandantenname	Text		Mandanten-Name		
196	CompanyCountry	Mandantenland	Text		Mandanten-Land		
4	Creditor	Lieferant	Text		Lieferant		CreditorID
5	CreditorName	Lieferantenname	Text		Lieferantenname		CreditorName
6	CreditorCountry	Lieferantenland	Text		Lieferantenland		Creditor_Country
11	IBAN	IBAN	Text		IBAN		IBAN
12	VatId	Ust-ID	Text		UstId		Creditor_TaxID
3	DocumentType	Vorgang	Text		Vorgang		InvoiceCreditVoucher
8	DocumentReference	Belegnummer	Text		Belegnummer		InvoiceNumber
7	DocumentDate	Belegdatum	Datum		Belegdatum		InvoiceDate

Positionen nicht übernehmen

Dafür gibt es keinen Schalter. In Squeeze unter Dokumentklasse → **Tabellen** die Spalte **Name (Zielsystem)** für alle Spalten leeren. Ohne Zeilen wird die Gentable nicht geschrieben. Das Mapping wird bei jedem Lauf neu gelesen.

>

> Eine bereits befüllte Gentable eines vorhandenen Vorgangs wird dabei nicht geleert.

Schalter in Squeeze_Pull_Lib

Am Dateianfang von `Squeeze_Pull_Lib` stehen die Schalter. Im Paket sind die Betriebswerte gesetzt.

Variable	Test	Betrieb	Bedeutung
`TEST_KEEP_IN_EXPORT`	`true`	**`false`**	kein FINISHED, Beleg bleibt im Export, Vorgang bleibt Unfinished
`MAX_DOCS_PER_RUN`	`1`	`100`	höchstens so viele Belege je Lauf
`LOCK_TIMEOUT_MIN`	`10`	`10`	Laufkennzeichen gilt danach als veraltet
`USE_ACTIVITY_LIST`	`false`	`false`	erst ab DOCUMENTS 6.3
`TEMP_PATH`, `PAGE_SIZE`, `MAX_PAGES`	–	`<DEXPRO>\SqueezeDocs`, 35, 200	normalerweise nicht abzuändern

> `TEST_KEEP_IN_EXPORT` muss im Betrieb auf `false` stehen. Im Testmodus schreibt der Job einen deutlichen Hinweisblock ins Log.

04 – Betrieb und Fehlersuche

Diese Seite beschreibt Log, typische Fehlerbilder und die Parallel-Sperre des Squeeze Pull-Jobs.

Log

Das Log liegt monatlich unter:

```
<DEXPRO>\Logs\Invoice\JOBS\Invoice_JOB_SqueezePull_JJJJMM.log
```

Typische Fälle

Thema	Vorgehen
Fehler finden	Filterordner Unfinished/ERROR öffnen. Diese Vorgänge bekommen keinen Workflow, bis der Pull für sie klappt.
Feld fehlt	Log: „Hinweis: Feld 'X' gibt es am Mappentyp ... nicht“ → Name (Zielsystem) in Squeeze korrigieren oder entfernen.
Register fehlt	Konfigurationsfehler: Der Lauf für die Klasse bricht ab, die restlichen Belege bleiben im Export. Parameter prüfen.
Processed nicht speicherbar	Log: „... bitte SqueezeStatus von Hand auf 'Processed' setzen“ → von Hand setzen, der Beleg ist in Squeeze schon abgeschlossen.
Processed, aber noch im Export	Meist eine Verzögerung in Squeeze, das erledigt sich von selbst. Bleibt der Beleg liegen, überspringt der Job ihn bei jedem Lauf ohne FINISHED (Log: `uebersprungen (SqueezeStatus ist Processed)`). Dann den Beleg über „Erneut übergeben“ neu holen oder in Squeeze von Hand abschließen.
Erneut übergeben	Vorgang in DOCUMENTS löschen, in Squeeze den Export-Status „Pull“ zurücksetzen. Der nächste Lauf legt einen neuen Vorgang an.
Zahlen und Datum	Zahlen: letztes Trennzeichen = Dezimaltrenner (wie Squeeze `/formatValueAs`). Datum nur `dd.mm.yyyy`; ungültige Werte bleiben leer, der Beleg wird nicht angehalten. Achtung: `1.000` in einem Squeeze-Textfeld wird als 1 gelesen, im Log steht dazu ein Hinweis.

Parallel-Sperre

Damit nie zwei Läufe denselben Mappentyp bearbeiten, setzt der Job ein Laufkennzeichen im `propCache` (Arbeitsspeicher, nach einem Neustart leer):

```
SqueezePullLock_Invoice = <Lauf-ID>|<Zeitstempel>
```

Der Lauf erneuert den Zeitstempel vor jedem Beleg. Findet ein Lauf ein gültiges Kennzeichen, endet er sofort ohne Abholung. Wird der Zeitstempel 10 Minuten nicht erneuert, übernimmt der nächste Lauf.

> Nur im Notfall von Hand entfernen: `propCache.removeProperty("SqueezePullLock_Invoice")`. Vorher sicherstellen, dass kein Lauf aktiv ist.

Bekannte Einschränkungen

Punkt	Stand
HTTP-Timeout für Squeeze-Aufrufe	erst ab DOCUMENTS 6.1 möglich. Bis dahin schützt `thread=1` die übrigen Jobs.
`getActivityList()`	ab DOCUMENTS 6.3, eingebaut, schreibt bisher nur ins Log
Standard-Bug `Invoice_Action_StartWorkflow`	`cnt` undefiniert, wenn der Userexit `false` liefert → Skriptfehler statt Meldung; der Workflow startet trotzdem nicht
`PosQuantity` in Squeeze als Text angelegt	Der Wert wird als Zahl übernommen. Eine Menge `1.000` wird dabei als 1 gelesen (letztes Trennzeichen = Dezimaltrenner).
`Invoice_Posting_Head` wird bei Updates nicht nachgeführt	Für die Dublettenprüfung unkritisch, Gadgets und Reports zeigen ggf. einen veralteten Status.