

Überblick

Der BWScheduler stellt integrierte Funktionen bereit, die über einen Namen mit vorangestelltem  ausgewählt werden. Damit können beispielsweise NoLock-Dateien verwaltet, Dateien überwacht, Windows-Dienste gesteuert oder Datenimporte gestartet werden.

Diese Dokumentation richtet sich an Anwender und Administratoren. Vor dem Einsatz von Funktionen, die Dateien, Prozesse, Dienste oder Benutzersitzungen beenden, sollte die Aufgabe zuerst in einer Testumgebung geprüft werden.

Allgemeine Einrichtung

Für die nachfolgend beschriebenen Funktionen wird bei der Aufgabe grundsätzlich Folgendes eingestellt:

Einstellung	Empfohlener Wert
Aktiv	aktiviert
Aufgabenart	Funktionsaufruf
App	gewünschte  -Funktion
Wochentag	gewünschte Ausführungstage
Ausführungsintervall	meistens Täglich
Ausführungsbedingung	keine Einschränkung, sofern keine besondere Prüfung benötigt wird
Modale Gruppe	passend zum betroffenen System oder Arbeitsablauf

Einmalige tägliche Ausführung

Für eine einmalige Ausführung zu einer festen Uhrzeit:

- **Intervall:** 
- **Ausführungszeit:** gewünschte Startzeit
- **Zeit von:** Beginn des erlaubten Zeitraums
- **Zeit bis:** Ende des erlaubten Zeitraums

Beispiel:

Einstellung	Wert
Zeit von	05:30
Ausführungszeit	05:31
Zeit bis	06:00
Intervall	0

Die Ausführungszeit muss innerhalb des erlaubten Zeitraums liegen. Für einen Zeitraum innerhalb desselben Tages muss **Zeit bis** später als **Zeit von** sein.

Wiederkehrende Ausführung

Für eine wiederkehrende Ausführung:

- **Intervall:** Abstand in Minuten, beispielsweise
- **Ausführungszeit:** leer lassen
- **Zeit von / Zeit bis:** Zeitraum, in dem die Wiederholungen erlaubt sind

AppParameter und AppParameter2

Je nach Funktion werden zusätzliche Angaben benötigt:

- **AppParameter:** meistens Dateipfad, Ordnerpfad oder eine Funktionsoption
- **AppParameter2:** zusätzliche Angabe, beispielsweise Dateialter oder Zielaufgabe
- **Dienstname:** Name eines Windows-Dienstes oder Prozesses

Nicht benötigte Felder bleiben leer.

Kurzreferenz

Funktion	Zweck	AppParameter	AppParameter2	Dienstname
\$CREATENOLOCKINI	NoLock.INI erstellen	leer	leer	leer
\$DELETENOLOCKINI	Vom Scheduler erstellte NoLock.INI entfernen	leer	leer	leer
\$DELETEALLNOLLOCKINI	Alle gefundenen NoLock.INI entfernen	leer	leer	leer
\$AUSLOGGEN	NoLock.INI vorübergehend setzen	leer	leer	leer

Funktion	Zweck	AppParameter	AppParameter2	Dienstname
\$BÜROWARE - BEENDEN	ERP-Suite-Prozesse und zugehörige Dienste beenden	optional: Prozesslistendatei	leer	leer
\$WEBWARE - BEENDEN	Webware-Prozesse und zugehörige Dienste beenden	optional: Prozesslistendatei	leer	leer
\$PROZESSE - BEENDEN	Mehrere definierte Prozesse oder Dienste beenden	optional: Prozesslistendatei	leer	leer
\$PROZESSE - STARTEN	Mehrere definierte Prozesse oder Dienste starten	optional: Prozesslistendatei	leer	leer
\$CREATEFILE	Datei erstellen oder ersetzen	vollständiger Dateipfad	leer	leer
\$DELETEFILE	Einzelne Datei löschen	vollständiger Dateipfad	leer	leer
\$DELETEFILES	Mehrere ältere Dateien löschen	Pfad mit Dateimaske	Alter in Tagen	leer
\$DATEIÜBERWACHUNG	Dateiänderung überwachen und Aufgabe einplanen	vollständiger Dateipfad	RUN(n)	leer
\$DTAIMPORT	DTA-Dateien importieren	Importordner	Bediener und optional Mandant	leer
\$NETZWERKLAUFWERK- VERBINDEN	Hinterlegte Netzlaufwerke verbinden	leer	leer	leer
\$STARTSERVICE	Windows-Dienst starten	leer	leer	Dienstname
\$STOPSERVICE	Windows-Dienst stoppen	leer	leer	Dienstname
\$RESTARTSERVICE	Windows-Dienst neu starten	leer	leer	Dienstname
\$KILLPROCESS	Einzelnen Prozess sofort beenden	leer	leer	Prozessname
\$RDS - LOGOFF	Remotedesktop-Sitzungen abmelden	Option, z. B. /all	leer	leer
\$SOFTWAREAUDIT	ERP-Suite Software-Audit durchführen	leer	leer	leer

NoLock- und Wartungsfunktionen

\$CREATENOLOCKINI

Erstellt im eingestellten ERP-Suite-Programmpfad eine `NoLock.ini`. Dadurch werden ERP-Suite und zugehörige Dienste in den vorgesehenen Wartungszustand versetzt. Solange die NoLock-Datei vorhanden ist, werden normale Scheduler-Aufgaben angehalten.

Empfohlene Einstellungen:

Einstellung	Wert
Aufgabenart	Funktionsaufruf
App	<code>\$CREATENOLOCKINI</code>
AppParameter	leer
Intervall	0
Zeit von	05:30
Ausführungszeit	05:31
Zeit bis	06:00
Modale Gruppe	keine (Global)

Voraussetzungen:

- Der globale ERP-Suite-Pfad muss richtig eingestellt und erreichbar sein.
- Das Benutzerkonto des BWSchedulers benötigt Schreibrechte im ERP-Suite-Ordner.
- Es sollte eine passende Aufgabe zum späteren Entfernen der NoLock-Datei vorhanden sein.

\$DELETENOLOCKINI

Entfernt NoLock-Dateien, die vom BWScheduler selbst erstellt wurden. Fremd oder manuell angelegte NoLock-Dateien bleiben zum Schutz erhalten.

Empfohlene Einstellungen:

Einstellung	Wert
Aufgabenart	Funktionsaufruf
App	<code>\$DELETENOLOCKINI</code>
AppParameter	leer
Intervall	0
Zeit von	05:30
Ausführungszeit	05:32 oder gewünschter Freigabezeitpunkt

Einstellung	Wert
Zeit bis	06:00 oder passend zum Wartungsfenster
Modale Gruppe	keine (Global)

Wird die NoLock-Datei für längere Wartungsarbeiten benötigt, muss die Löschaufgabe entsprechend später eingeplant werden.

\$DELETEALLNOLOCKINI

Entfernt alle vom BWScheduler gefundenen NoLock-Dateien – unabhängig davon, wodurch sie angelegt wurden.

⚠ **Achtung:** Diese Funktion kann auch eine bewusst manuell gesetzte Wartungssperre entfernen. Sie sollte nur durch berechtigte Administratoren und nur für eindeutig definierte Wartungsabläufe verwendet werden.

Empfohlene Einstellungen:

- **Aufgabenart:** Funktionsaufruf
- **App:** \$DELETEALLNOLOCKINI
- **AppParameter / AppParameter2:** leer
- **Modale Gruppe:** keine (Global)
- **Zeitpunkt:** nur innerhalb eines kontrollierten Wartungsfensters

\$AUSLOGGEN

Erstellt eine NoLock-Datei, wartet ungefähr 30 Sekunden und entfernt sie anschließend wieder. Die Funktion ist dafür vorgesehen, laufende ERP-Suite-Module zum geordneten Beenden beziehungsweise Abmelden aufzufordern.

Empfohlene Einstellungen:

- **App:** \$AUSLOGGEN
- **AppParameter / AppParameter2:** leer
- **Intervall:** 0
- **Modale Gruppe:** keine (Global)
- Ausreichend Abstand zu nachfolgenden Aufgaben einplanen

Beispiel: Ausführung um 23:00 innerhalb eines erlaubten Zeitraums von 19:30 bis 23:55.

ERP-Suite- und Prozesssteuerung

\$BÜROWARE - BEENDEN

Versetzt die ERP-Suite zunächst in den Wartungszustand, wartet ungefähr 30 Sekunden und beendet danach die vorgesehenen ERP-Suite-Prozesse und Dienste. Nach einer weiteren Kontrolle werden noch vorhandene Prozesse erneut beendet. Anschließend wird die vom Scheduler erstellte NoLock-Datei entfernt.

Empfohlene Einstellungen:

- **App:** \$BÜROWARE - BEENDEN
- **AppParameter:** Pfad zur vorgesehenen ERP-Suite-Prozesslistendatei oder leer für die Standardliste
- **Intervall:** 0
- **Modale Gruppe:** keine (Global)
- **Zeitfenster:** außerhalb der normalen Arbeitszeit

“ **Achtung:** Offene ERP-Suite-Sitzungen können beendet werden. Stimmen Sie die Ausführung mit den Anwendern ab und prüfen Sie die verwendete Prozessliste sorgfältig.

\$WEBWARE - BEENDEN

Entspricht dem Ablauf von \$BÜROWARE - BEENDEN, verwendet jedoch die für Webware vorgesehenen Prozesse und Dienste.

Empfohlene Einstellungen:

- **App:** \$WEBWARE - BEENDEN
- **AppParameter:** Pfad zur Webware-Prozesslistendatei oder leer für die Standardliste
- **Intervall:** 0
- **Modale Gruppe:** keine (Global)
- **Zeitfenster:** außerhalb der normalen Nutzung

\$PROZESSE - BEENDEN

Beendet die Einträge einer Prozesslistendatei. Nach einer kurzen Wartezeit wird erneut geprüft und ein zweiter Beendigungsversuch durchgeführt.

Empfohlene Einstellungen:

- **App:** \$PROZESSE-BEENDEN
- **AppParameter:** vollständiger Pfad zur Prozesslistendatei; leer verwendet die allgemeine Standardliste
- **Intervall:** 0
- **Modale Gruppe:** eigene Farbgruppe für den betroffenen Arbeitsablauf oder Global

⚠ **Achtung:** Prüfen Sie jeden Eintrag der Prozessliste. Eine falsche Liste kann nicht vorgesehene Anwendungen oder Dienste beenden.

\$PROZESSE - STARTEN

Startet die in einer Prozesslistendatei hinterlegten Prozesse und Dienste.

Empfohlene Einstellungen:

- **App:** \$PROZESSE-STARTEN
- **AppParameter:** vollständiger Pfad zur Prozesslistendatei; leer verwendet die allgemeine Standardliste
- **Netzwerklaufwerk verbinden:** aktivieren, wenn Programme oder Dateien über ein Laufwerk im Netzwerk erreicht werden
- **Modale Gruppe:** dieselbe Gruppe wie die zugehörige Beendigungs- oder Wartungsaufgabe

Planen Sie den Start erst nach Abschluss aller Wartungs- und Sicherungsarbeiten ein.

\$KILLPROCESS

Beendet alle laufenden Prozesse mit dem angegebenen Prozessnamen sofort.

Empfohlene Einstellungen:

Einstellung	Wert
App	\$KILLPROCESS
Dienstname	Prozessname ohne .exe , beispielsweise bwin32
AppParameter	leer
Intervall	0
Modale Gruppe	passend zum betroffenen System

Achtung: Diese Funktion ermöglicht kein geordnetes Speichern oder Beenden. Sie sollte nur verwendet werden, wenn ein normales Beenden nicht möglich oder ausdrücklich unerwünscht ist.

Windows-Dienste steuern

\$STARTSERVICE

Startet den unter **Dienstname** angegebenen Windows-Dienst. Läuft der Dienst bereits, bleibt er unverändert.

\$STOPSERVICE

Stoppt den unter **Dienstname** angegebenen Windows-Dienst. Ist der Dienst bereits beendet, ist keine weitere Aktion erforderlich.

\$RESTARTSERVICE

Stoppt den angegebenen Windows-Dienst und startet ihn anschließend erneut. Der BWScheduler wartet auf das Beenden des Dienstes und versucht danach den Start.

Empfohlene Einstellungen für alle drei Dienstfunktionen:

Einstellung	Wert
Aufgabenart	Funktionsaufruf
App	<code>\$STARTSERVICE</code> , <code>\$STOPSERVICE</code> oder <code>\$RESTARTSERVICE</code>
Dienstname	Anzeigename oder Dienstname des gewünschten Windows-Dienstes
AppParameter / AppParameter2	leer
Intervall	normalerweise 0
Modale Gruppe	eine gemeinsame Farbe für zusammengehörige Dienste

Voraussetzungen:

- Der angegebene Dienst muss auf demselben Windows-System vorhanden sein.
- Das Benutzerkonto des BWSchedulers benötigt die Berechtigung zum Steuern des Dienstes.

- Bei abhängigen Diensten muss eine sinnvolle Reihenfolge und ausreichend Zeitabstand eingeplant werden.

Dateifunktionen

\$CREATEFILE

Erstellt eine Datei am angegebenen Ort. Eine bereits vorhandene Datei wird ersetzt. Die Funktion eignet sich insbesondere zum Erzeugen von Steuer- oder Triggerdateien.

Empfohlene Einstellungen:

- **App:** \$CREATEFILE
- **AppParameter:** vollständiger Dateipfad, beispielsweise C:\ERP\Export\run.ini
- **AppParameter2:** leer
- **Netzwerklaufwerk verbinden:** aktivieren, wenn ein hinterlegtes Netzlaufwerk verwendet wird

Der Zielordner muss bereits vorhanden sein und das verwendete Konto benötigt Schreibrechte.

\$DELETEFILE

Löscht genau die unter **AppParameter** angegebene Datei.

Empfohlene Einstellungen:

- **App:** \$DELETEFILE
- **AppParameter:** vollständiger Dateipfad
- **AppParameter2:** leer
- **Modale Gruppe:** dieselbe Gruppe wie eine zugehörige Erstellungs- oder Verarbeitungsaufgabe

⚠ **Achtung:** Kontrollieren Sie den vollständigen Pfad vor der Aktivierung. Die gelöschte Datei wird nicht in den Windows-Papierkorb verschoben.

\$DELETEFILES

Löscht mehrere Dateien eines bestimmten Musters, sobald sie das angegebene Alter erreicht haben.

Beispiel:

Einstellung	Wert
App	\$DELETEFILES
AppParameter	C:\ERP\Log*.log
AppParameter2	30

In diesem Beispiel werden passende Protokolldateien berücksichtigt, die älter als 30 Tage sind.

Hinweise:

- Verwenden Sie im AppParameter einen vollständigen Ordnerpfad mit Dateimaske.
- Prüfen Sie die Dateimaske besonders sorgfältig.
- Der Ordner muss erreichbar sein.
- Verwenden Sie möglichst ein eigenes, klar begrenztes Archiv- oder Protokollverzeichnis.

Dateiüberwachung

\$DATEIÜBERWACHUNG

Überwacht eine bestimmte Datei auf Änderungen. Wird eine Änderung erkannt, plant der BWScheduler eine andere Aufgabe zur Ausführung ein.

Empfohlene Einstellungen der Überwachungsaufgabe:

Einstellung	Wert
Aufgabenart	Funktionsaufruf
App	\$DATEIÜBERWACHUNG
AppParameter	vollständiger Pfad der zu überwachenden Datei
AppParameter2	RUN(n) mit der Nummer der Zielaufgabe
Intervall	0
Zeit von / Zeit bis	Zeitraum, in dem Änderungen verarbeitet werden dürfen

Beispiel:

AppParameter: C:\ERP\Import\auftrag.dat
AppParameter2: RUN(5)

Damit wird bei einer Änderung der Datei die Aufgabe 5 eingeplant.

Voraussetzungen für die Zielaufgabe:

- Die Zielaufgabe muss aktiviert sein.
- Die Zielaufgabe muss **Intervall 0** verwenden.
- Überwachungsaufgabe und Zielaufgabe dürfen nicht dieselbe Aufgabe sein.
- Unter **Mindestwartezeit bei Triggerausführung** sollte ein ausreichender Abstand eingestellt werden, damit mehrere unmittelbar aufeinanderfolgende Dateiänderungen nicht zu wiederholten Starts führen.

Bei Netzwerkpfaden muss der Pfad aus dem Ausführungskonto des BWSchedulers erreichbar sein.

DTA-Datenimport

\$DTAIMPORT

Importiert `.dta`-Dateien aus einem angegebenen Ordner. Erfolgreich verarbeitete Dateien werden in den Unterordner `Importiert` verschoben.

Empfohlene Einstellungen:

Einstellung	Wert
App	<code>\$DTAIMPORT</code>
AppParameter	vollständiger Pfad des DTA-Importordners
AppParameter2	Bedienernummer und optional Mandant, z. B. <code>000 000001</code>
Intervall	z. B. 15 Minuten
Ausführungszeit	bei Intervallausführung leer
Modale Gruppe	eigene Gruppe für den jeweiligen Importbereich

Alternativ können Bedienernummer, Mandantennummer und Importmodul zentral im Bereich **Datenimport** eingestellt werden.

Zu prüfen sind:

- Importordner ist vorhanden und beschreibbar.
- ERP-Suite-Programmpfad ist korrekt.
- Bedienernummer und Mandantennummer sind gültig.
- Gewünschtes Importmodul ist ausgewählt: WAWI, FIBU oder automatische Erkennung über den Dateinamen.
- Bei automatischer Erkennung sollte der Dateiname eindeutig `WAWI` oder `FIBU` enthalten.

Während einer aktiven NoLock-Sperre beziehungsweise Wartung wird kein DTA-Import durchgeführt.

Netzlaufwerke verbinden

\$NETZWERKLAUFWERK - VERBINDEN

Verbindet die im Bereich **Netzwerk** hinterlegten Netzlaufwerke.

Für bis zu drei Verbindungen können eingestellt werden:

- Server-Pfad, beispielsweise `\\Server\Freigabe`
- Zugewiesener Laufwerksbuchstabe
- Benutzer
- Kennwort
- Persistente Verbindung

Empfohlene Aufgabeneinstellungen:

- **App:** `$NETZWERKLAUFWERK - VERBINDEN`
- **AppParameter / AppParameter2:** leer
- **Zeitpunkt:** vor allen Aufgaben, die diese Laufwerke benötigen
- **Modale Gruppe:** passend zu den abhängigen Aufgaben

Die Einstellung **Netzwerklaufwerk verbinden** innerhalb einer einzelnen Aufgabe kann zusätzlich verwendet werden, damit die hinterlegten Verbindungen unmittelbar vor dieser Aufgabe hergestellt werden.

Remotedesktop-Sitzungen abmelden

\$RDS - LOGOFF

Meldet Remotedesktop-Benutzersitzungen entsprechend der gewählten Option ab.

AppParameter	Wirkung
leer	getrennte Sitzungen abmelden
<code>/all</code>	alle vorgesehenen Remotedesktop-Sitzungen abmelden
<code>/U=benutzername</code>	den angegebenen Benutzer abmelden

Beispiel:

Empfohlene Einstellungen:

- **App:** \$RDS-LOGOFF
- **Intervall:** 0
- **Modale Gruppe:** keine (Global)
- **Zeitfenster:** außerhalb der normalen Arbeitszeit

⚠ **Achtung:** Nicht gespeicherte Benutzerdaten können verloren gehen.
Informieren Sie betroffene Benutzer vor einer automatischen Abmeldung.

Bestimmte administrative Benutzer können über die Datei `rds-logoff-blacklist.txt` von der automatischen Abmeldung ausgenommen werden. Änderungen an dieser Liste sollten nur durch die zuständige Administration erfolgen.

ERP-Suite Software-Audit

\$SOFTWAREAUDIT

Erfasst die vorgesehenen ERP-Suite-Systeminformationen und führt den eingerichteten Datenabgleich durch.

Empfohlene Einstellungen:

- Verwenden Sie vorzugsweise die vorbereitete Aufgabe **ERP-Suite Software Audit**.
- Hinterlegen Sie bei der Einrichtung die korrekte Kundennummer und die vom Anbieter bereitgestellte Freigabe.
- Stellen Sie sicher, dass der ERP-Suite-Pfad korrekt ist.
- Planen Sie die Aufgabe einmal täglich außerhalb der Hauptarbeitszeit ein.
- **AppParameter / AppParameter2:** leer

Diese Funktion sollte nur verwendet werden, wenn der Software-Audit für das jeweilige Kundensystem eingerichtet und freigegeben wurde.

Empfohlene Reihenfolge für ein Wartungsfenster

Ein möglicher Ablauf ist:

1. Mit `$AUSLOGGEN` Benutzer zum Beenden der ERP-Suite-Sitzungen auffordern.
2. Mit `$CREATENOLockINI` den Wartungszustand aktivieren.
3. Mit `$BÜROWARE-BEENDEN`, `$WEBWARE-BEENDEN` oder `$PROZESSE-BEENDEN` benötigte Komponenten beenden.
4. Sicherung, Aktualisierung oder andere Wartungsaufgabe durchführen.
5. Mit `$PROZESSE-STARTEN` oder `$STARTSERVICE` benötigte Komponenten wieder starten.
6. Mit `$DELETENOLockINI` den Wartungszustand aufheben.

Verwenden Sie für den gesamten Ablauf dieselbe modale Farbgruppe oder **keine (Global)**, wenn während der Wartung keine andere modale Aufgabe laufen darf. Planen Sie zwischen den einzelnen Schritten ausreichend Zeit ein.

Sicherheits- und Prüfhinweise

Vor der Aktivierung einer neuen Funktionsaufgabe:

1. Pfade, Dateimasken, Dienstnamen und Prozessnamen kontrollieren.
2. Berechtigungen des Kontos prüfen, unter dem der BWScheduler ausgeführt wird.
3. Zeitfenster und Wochentage kontrollieren.
4. Bei Intervall 0 eine gültige Ausführungszeit innerhalb des Zeitfensters eintragen.
5. Bei Intervall größer 0 die Ausführungszeit leer lassen.
6. Eine geeignete modale Gruppe auswählen.
7. Kritische Funktionen zunächst manuell in einer Testumgebung ausführen.
8. Nach der Ausführung die Aufgabenanzeige und das Protokoll kontrollieren.

Besondere Vorsicht ist bei `$DELETEALLNOLockINI`, `$DELETEFILE`, `$DELETEFILES`, `$KILLPROCESS`, `$PROZESSE-BEENDEN`, `$BÜROWARE-BEENDEN`, `$WEBWARE-BEENDEN` und `$RDS-LOGOFF` erforderlich.

Revision #1

Created 2026-09-08 11:35:00 CEST by Martin Dauwa

Updated 2026-09-08 11:35:23 CEST by Martin Dauwa