

Taskscheduler

Portal: *beheerportaal-Nieuw*, Kolom *Dieper Beheer*. Tegel *Taskscheduler*.

Screenidentifiers:

- MDLC_getTbTaskSchedulerList.xml
- MDDC_getTbTaskSchedulerDetail.xml

Beschrijving

Indien op de applicatieserver van OpenWave een cronjob is geïnstalleerd die om de 5 minuten (default instelling) het script `runtaskscheduler.sh` start, dan roept die cronjob via dat script de OpenWave API `runScheduledTasks` aan.

De installatie van de cronjob met script (waarin naam en wachtwoord van robot account in `tbmedewerkerstabel`) kan alleen door de ICT van Rem Automatisering worden gedaan.

De API `runScheduledTasks` kijkt in de tabel `tbtaskscheduler` of er een actie uitgevoerd dient te worden. Zo ja, dan wordt die actie gestart (als callable = runnable = een taak zonder userinterface) en sluit `runScheduledTasks` zichzelf af.

Zonder cronjob kan de API `runScheduledTasks` ook gestart worden vanuit de wizardknop onderaan het lijstscherf van de taskscheduler.

De `runScheduledTasks` kan dus maar één actie (callable) tegelijk starten. Dat betekent ook wat voor de definitie en planning van de taken in de tabel `tbtaskscheduler`:

- **ID** (dnkey). Primary key.
- **Taak** (dvtaskcodering). Unieke naam/codering van de taak.
- **Uit te voeren action** (dvtaskaction). De action (een callable = een OpenWave API zonder userinterface) die uitgevoerd moet worden. Deze callable worden vooralsnog door Rem gedefinieerd. Een voorbeeld is de callable `importSWFOpenActieverzoeken` die in de landelijke samenwerkingsruimte polt naar acties waarvoor een zaak in OpenWave moet worden aangemaakt.
- **Aan** (dltaskenabed). T of F, default F. Indien T dan mag de taak uitgevoerd worden.
- **Na uitvoer ophogen met minuten** (dninterval). Aantal minuten dat bij de `ddexecutionstarttime` wordt opgeteld na uitvoering van de taak. Moet minimaal 5 minuten zijn i.v.m. de cronjob. Indien eens per dag dan 1440 minuten. Indien eens per week dan 10080 minuten.
- **Geplande datum/tijd** (ddexecutionstarttime). Eerstvolgende datum/tijd dat taak gestart dient te worden. Kan handmatig worden ingevoerd. Bij start van de taak wordt door `runScheduledTasks` de `dninterval` hierbij opgeteld. Indien bij het sluiten van de callable deze datum/tijd kleiner is dan de systeemtijd, dan hoort de callable deze net zolang op met het interval totdat de geplande datum/tijd weer in de toekomst ligt.
- **Datum/tijd laatste uitvoering** (ddexecutionstoptime). Laatste datum/tijd dat taak is uitgevoerd. Wordt door de callable gezet.
- **Status laatste uitvoering** (dvstatus). De status van de taak: 0 = goed afgerond, 1 = gestart / bezig, 2 = afgerond, maar mislukt. De status 1 wordt gezet door de `runScheduledTasks`

wanneer de callable wordt aangeroepen. De statussen 0 en 2 worden door de callable gezet. Status 2 is eigenlijk alleen maar te interpreteren door de operationslog na te zien. De taak kan bijvoorbeeld bestaan uit het ophalen van alle gerelateerde dso-verzoeken, waarvan één verzoek een foutmelding bij het DSO oplevert, terwijl alle andere verzoeken keurig worden verwerkt. In dat geval krijgt de taak toch de status 2.

- **Toelichting** (dvtoelichting). Vrije toelichting op taak.

Het is aan te raden de *Geplande datum/tijd* in de regels uit tbtaskscheduler onderling minimaal 5 minuten te laten verschillen.

Onder de tegel *Versie-informatie* (Servicecentrum portaal) is nu een regel opgenomen met item Taskscheduler die aangeeft wanneer de taskscheduler voor het laatst is uitgevoerd en dus aangeroepen door de Cronjob.

De callable die aangeroepen wordt kijkt eerst in tinitialisatie onder de *Sectie: Operations en Item: {de naam van de callable}* (dus bijv. importSWFOpenActieverzoeken) of de betreffende taak al draait (dezelfde taak kan mogelijk ook met de hand worden gestart). Zo ja, dan wordt de callable niet uitgevoerd.

De callable doet verder verslag in een aangemaakte kaart in tboptionslog. Daar is dus uiteindelijk het resultaat van de actie terug te lezen.

Waarschuwing

Indien tijdens de operatie iets fundamenteels verkeerd gaat (stroom eraf bijv.) dan kan de operatie niet worden afgemaakt. De taak blijft op *ben bezig* staan en ook *Getal1* van de instelling (zie hieronder bij de opsomming van de callables) die ervoor zorgt dat een operatie niet twee keer tegelijkertijd kan worden gestart, blijft dan de waarde 1 houden.

In dat geval dient deze waarde handmatig op null gezet te worden. OpenWave doet dit echter automatisch nadat het aantal uren opgegeven in *Getal1* van de instelling *Sectie: Taskscheduler Item: AantalBenBezigHersteluren* verstreken is (defaultwaarde van deze instelling is 12). Indien expliciet de waarde 0 is opgegeven dan doet OpenWave niks automatisch vrijgeven. Zie verder stroomschema hieronder.

Welke callables zijn in te stellen

- **importSWFOpenActieverzoeken** (zie voorwaarde en noodzakelijke instellingen bij [Samenwerkingsfunctionaliteit](#)).
Wanneer de actie start wordt *Getal1* van de instelling *Sectie: Operations Item: ImportSWFOpenActieverzoeken* op 1 gezet. Indien klaar wordt *Getal1* op null gezet.
- **exportReportContainer** met als parameter een dvcode uit de tabel tbexportcontainer (tegel *Container exportrapportages* onder kolom *Werkbeheer* van beheerportaal *Inrichtingenbeheer*) dus bijvoorbeeld `exportReportContainer(inspectieview)` zie: [Export Report Container](#).
Wanneer de actie start wordt *Getal1* van de instelling *Sectie: Operations en Item: ExportReportContainer* op 1 gezet. Indien klaar wordt *Getal1* op null gezet.
- **exportREV** (zie [Register Externe Veiligheid](#)) bij kopje *Export Naar REV*
Wanneer de actie start wordt *Getal1* van de instelling *Sectie: Operations en Item: ExportREV* op 1 gezet. Indien klaar wordt *Getal1* op null gezet.
- **importDSOGerelateerdeZaken** (zie [DSO Gerelateerde Zaken](#))

Wanneer de actie start wordt *Geta1* van de instelling *Sectie: Operations Item: importDSOGerelateerdeZaken* op 1 gezet. Indien klaar wordt *Geta1* op null gezet.

- **importDSOGemisteVerzoeken** (zie [DSO Gemiste Verzoeken](#))

Wanneer de actie start wordt *Geta1* van de instelling *Sectie: Operations Item: importDSOGemisteVerzoeken* op 1 gezet. Indien klaar wordt *Geta1* op null gezet.

- **SynchroniseerOpenSWFRuimtes** (zie [Synchroniseer Open SWF ruimtes](#))

Wanneer de actie start wordt *Geta1* van de instelling *Sectie: Operations Item: synchroniseerOpenSWFRuimtes* op 1 gezet. Indien klaar wordt *Geta1* op null gezet.

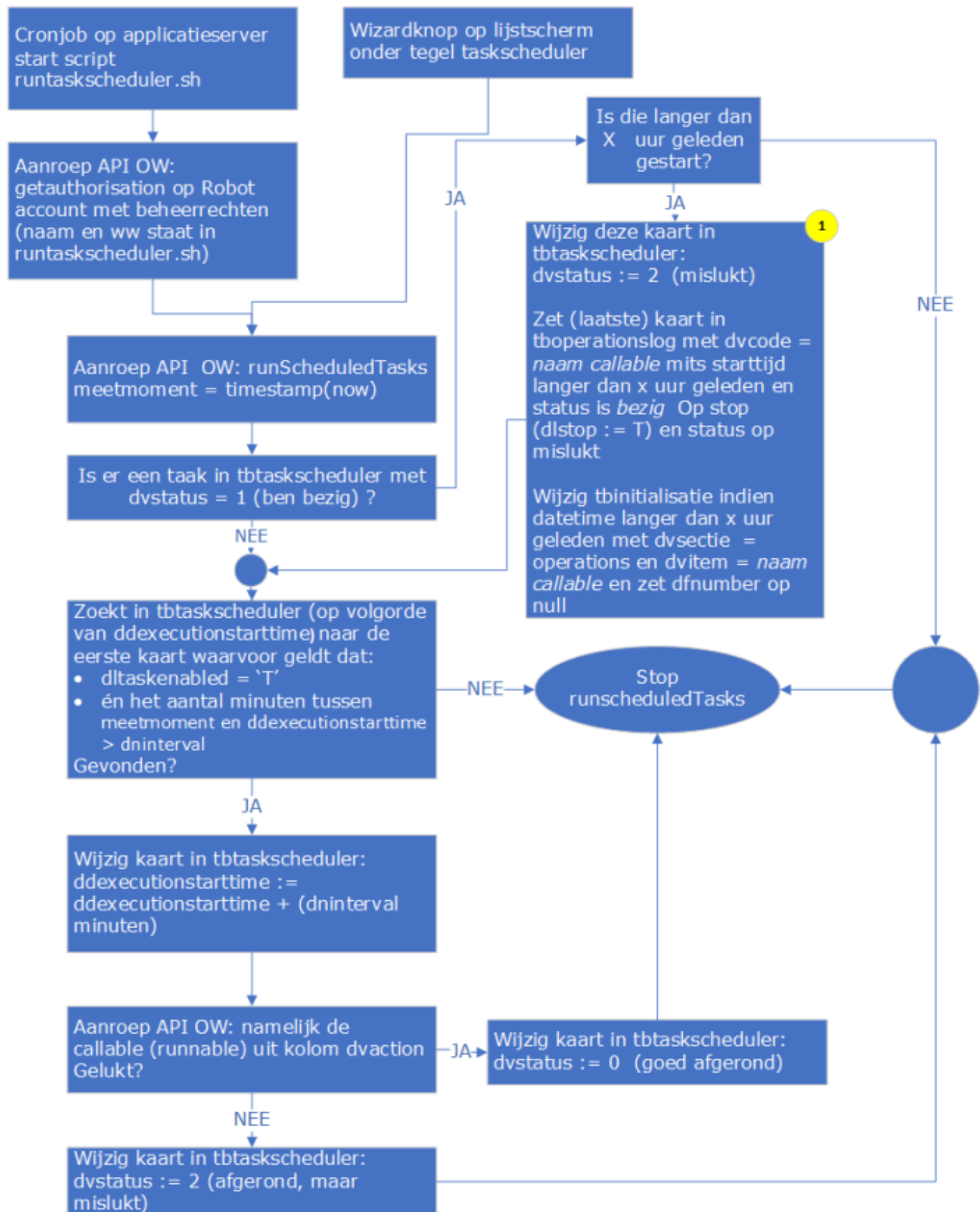
- **exportInrichtingenWFS** (zie [Data Op Kaart / Export Inrichtingen als WFS](#))

Wanneer de actie start wordt *Geta1* van de instelling *Sectie: Operations en Item: exportInrichtingenWFS* op 1 gezet. Indien klaar wordt *Geta1* op null gezet.

- **importmaandKadasterBAG** (zie [Automatisch inlezen BAG- mutaties](#))

Wanneer de actie start wordt *Geta1* van de instelling *Sectie: Operations en Item: Inlezenbag* op 1 gezet. Indien klaar wordt *Geta1* op null gezet.

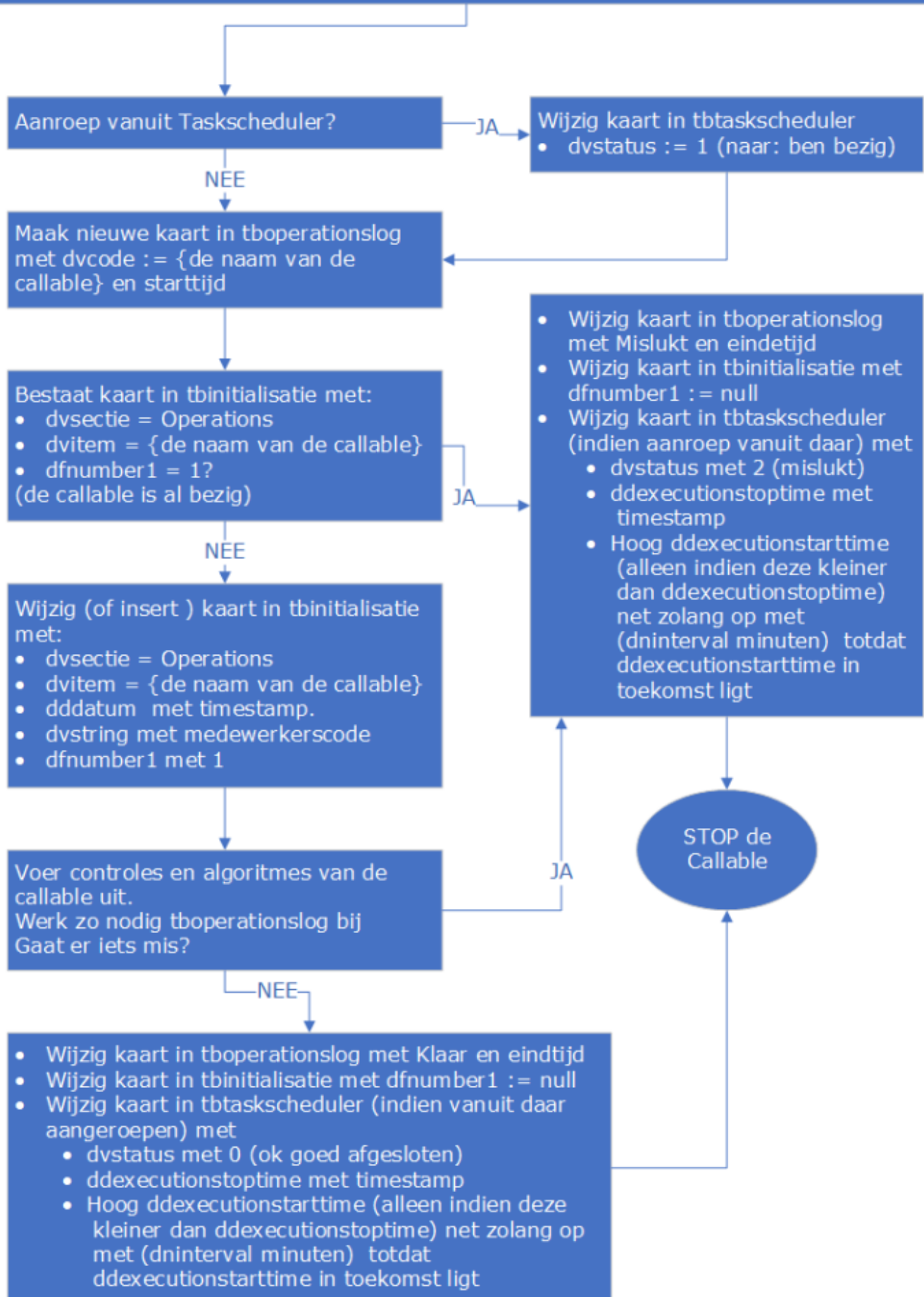
Stroomschema runScheduledTasks



ad 1. De waarde van *Geta1* van instelling *Sectie: Taskscheduler Item: AantalBenBezigHersteluren*. Indien niet aanwezig dan wordt 12 als defaultwaarde genomen. Indien 0 dan wordt de taak NIET vrijgegeven.

Stroomschema geldig voor alle Callables

Callable (= runnable) wordt aangeroepen door runScheduledtask: taak in tbtaskscheduler) OF vanuit andere actionaanroep



From:
<https://acc-doc.open-wave.nl/> - Documentatie

Permanent link:
https://acc-doc.open-wave.nl/openwave/1.31/applicatiebeheer/instellen_inrichten/taskscheduler?rev=1752759147

Last update: **2026/07/15 11:07**

