

# Scheduled services

## Scheduled services tools

- Core services
  - Flush mail queue
  - Timed status actions
  - Send notifications
  - Import emails
  - Attachment fetcher
  - Import holidays
  - Maintain Resumes
- System maintenance
  - Compress logs
  - Cleanup mail queue
  - Cleanup event log
  - User deactivation
- System information
  - Admin status email
  - License reporting
  - Website monitor
  - Send ticket to admin
  - Ressource warnings
  - Usage reporting
  - Send service usage
- Special functionality
  - OLAP Activity
  - OLAP Flowsteps

- OLAP Flow total
- OLAP fact builder
- Import from AD
- Sync Azure AD
- Kontoafstemning
- Satelite synchronize
- Satelite synch data
- Custom codeunit
- Data index builder
- File index builder
- Run BI SQL

# Core services

# Flush mail queue

The process of sending email is

1. Generate email and put in queue ("outbox")
2. Send email to recipient ("sent")

When status actions are executed or user notifications are triggered, the email queue is filled. Nothing happens before the queue is flushed, so this service is essential for communication through email.

Normally this service should run every 5 minutes. In special cases where a lot of emails are generated, some will find reassuring that it is possible to postpone the sending of system generated emails. If some error occurs manual fixes on each email is still possible.

The following services are exempt from using the queue (are sent directly)

- Welcome information
- Password resets
- Server information status
- License information status

Recommended max service interval

- Only send emails: 2 minutes
- Generate documents and send emails: 10 minutes

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.ExecuteEmailQueue`
- Default `isActive`:
- Default interval: 5

# Timed status actions

Status actions are actions bound to a certain status, that has its type set to "timed".

This service will execute the actions, and will log the results. The service ensures that each action is only executed once per item. Exception to this rule: Changing the status to another, and moving the status back, will reset the "execute only once" feature.

As the time granularity for timed actions is DAYS, suggested service interval is once or twice day (1440 or 720 minutes).

Recommended max service interval: 15 minutes

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.ExecuteStatusActions`
- Default `isActive`:
- Default interval: 60

# Send notifications

Notifications are generated from different sources

- User alerts (flags with alerts activated)
- Fieldtype user: User was selected
- Fieldtype group: Group was selected
- Status action execution

## User alerts (flags with alerts activated)

Users can create alerts by flagging items, and checking off Alert on status updates. Every time the item changes status, the corresponding users will be alerted.

## Status action execution

Status actions allows for notification of either groups or individual users. The message can also contain information from the data item in question and/or a link for the item.

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.SendUserNotifications`
- Default `isActive`:
- Default interval:

# Import emails

The service will collect incoming emails and distribute them to each recognizable record.

The solution must contain a Mailbox field, which handles display of the incoming emails. Each record has a distinct email address, and in the field it is possible to specify which users are allowed to send emails to that type of record.

Before use a catch-all POP3 email address needs to be specified.

The account details should be entered in the policies

- mailDomain
- mailServer
- mailUsername
- mailPassword

Suggested frequency is 10-15 minutes.

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: dk.p2e.blanket.services.ImportEmails
- Default isActive:
- Default interval:

Core services

# Attachment fetcher

Pick up files stored in emails on a certain IMAP account

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.email.EmailHandlerThread`
- Default `isActive`:
- Default interval:

Core services

# Import holidays

Fetches holidays one year ahead and saves them

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.LoadHolidays`
- Default `isActive`:
- Default interval:

Core services

# Maintain Resumes

Recalculates resume-values for a given list of entities

## Configuration

The list of entities to update on every run, is defined in the policy `sagIDsForMaintainResumes`.

The policy should contain a comma-separated list of SagIDs to maintain.

## Developer info

- Classname: `dk.p2e.blanket.services.MaintainResumes`
- Default `isActive`: `no`
- Default interval: 1440 minutes (daily)

# System maintenance

# Compress logs

This service will consolidate the solution access logs, removing entries that can be spared.

The entries that are left after cleanup is

- Two entries for each combination of users and records
  - First access to this record
  - Last access to this reord

Note that the number of accesses to each item is still preserved, although there is no longer kept seperate timestamps.

The service is controlled by the property: `serviceLogentriesConsolidate`

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.CleanupLoggingData`
- Default `isActive`:
- Default interval:

System maintenance

# Cleanup mail queue

The service cleans up mail queue entries older than a certain age.

Default maximum age is 1 year / 365 days.

Future releases will contain an configuration option for the age.

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.CleanupEmailQueue`
- Default `isActive`:
- Default `interval`:

System maintenance

# Cleanup event log

The service cleans up event log entries older than a certain age.

Default maximum age is 1 year / 365 days.

Future releases will contain an configuration option for the age.

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.CleanupEvents`
- Default `isActive`:
- Default `interval`:

# User deactivation

The service deactivates user accounts that has not been used for some time, in order to

- Save user licenses (inactive users are not counted)
- Guard the system against forgotten users

Default maximum inactivity is 90 days.

Future releases will contain an configuration option for the age.

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.UserDeactivateAccount`
- Default `isActive`:
- Default `interval`:

# System information

# Admin status email

This service sends application status information to the system administrator.

The recipient of this email is specified in the policy: **serviceAdminStatusEmail**

The information looks like this:

```
Status for http://<SERVER>/<APPLICATION>
```

```
Errors last hour: <X>
```

```
Errors last 24 h: <X>
```

```
Logins last hour: <X>
```

```
Logins last 24 h: <X>
```

```
Last server boot: <YYYY-MM-DD HH:MM:SS>
```

```
Last server stop: <YYYY-MM-DD HH:MM:SS>
```

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: dk.p2e.blanket.services.AdminStatusEmail
- Default isActive:
- Default interval:

# License reporting

This service sends usage information to the system administrator and another address specified in the configuration.

Each metric in reports are verified by a security hash, to ensure that numbers and dates cannot be tampered with manually.

The recipient of the licens email i specified in the policy: **serviceAdminStatusEmail**

## Configuration

For the service to work scheduled services should be running and the service needs configuration.

### Set service configuration

If you just want the license information sent to Tempus Serva license service, you can skip this step.

1. Designer > Configuration > Search for "License"
2. If **licenseReportLicenseServer** also set the recepiet email in **licenseReportingEmail**
3. If **licenseReportLicenseServer** also set the recepiet email in **serviceAdminStatusEmail**

### Enable license service

1. Designer > Modules > Service monitor
2. Click **License reporting**
3. Set Active and submit

Note the default schedule is 10080 minutes = 7 days

## Developer info

- Classname: dk.p2e.blanket.services.LicenseReporting
- Default isActive:
- Default interval:



# Website monitor

This service will poll and URL and check for a specified goodword (a string of text).

If this check fails an SMS is sent to the application administrator. Please note that this features requires an payed SMS service provider, that needs to be aquired / setup before usage of this service.

The services requires two configurations

- serviceWebsiteMonitorURL (the url to watch)
- serviceWebsiteMonitorGW (the content to find)

Account details for SMS delivery is specified in the Policy\_reference#SMS\_service configurations (8).

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: dk.p2e.blanket.services.WebsiteMonitor
- Default isActive:
- Default interval:

System information

# Send ticket to admin

Forward all unsolved support requests last 24 hours

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.AdminTicketEmail`
- Default `isActive`:
- Default `interval`:

System information

# Ressource warnings

Send emails to admin if memory or disk runs low

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.WatchRessources`
- Default `isActive`:
- Default `interval`:

System information

# Usage reporting

Report your usage metrics (ex. user count)

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.tempusserva.services.licenses.ReportService`
- Default `isActive`:
- Default interval:

System information

# Send service usage

Sends usage of different services, for billing

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `com.tsnocode.engine.services.ServiceUsageService`
- Default `isActive`:
- Default interval:

# Special functionality

Special functionality

# OLAP Activity

*Build data tables in standard OLAP cubes*

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.BuildOlapActivity`
- Default `isActive`:
- Default interval:

Special functionality

# OLAP Flowsteps

*Build data tables in standard OLAP cubes*

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: dk.p2e.blanket.services.BuildOlapFlowsteps
- Default isActive:
- Default interval:

Special functionality

# OLAP Flow total

*Build data tables in standard OLAP cubes*

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.BuildOlapFlowtotal`
- Default `isActive`:
- Default interval:

Special functionality

# OLAP fact builder

*Build data tables in standard OLAP cubes*

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.BuildOlapGeneric`
- Default `isActive`:
- Default interval:

Special functionality

# Import from AD

*Synchronize users/groups in LDAP repository*

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: dk.p2e.blanket.services.LDAPimport
- Default isActive:
- Default interval:

Special functionality

# Sync Azure AD

Synchronize Azure groups and users

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.tempusserva.services.azureusers.DirectorySynchronizer`
- Default `isActive`:
- Default interval:

Special functionality

# Kontoafstemning

Bygger afstemninger til brug for kontoafstemning (se vejledning)

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.solution.afstemning.KontoafstemningBuilder`
- Default `isActive`:
- Default interval:

Special functionality

# Satelite synchronize

*Keep a slave server model and data in sync*

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.SateliteSynchronize`
- Default `isActive`:
- Default interval:

Special functionality

# Satelite synch data

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.SateliteSynchronizeData`
- Default `isActive`:
- Default `interval`:

Special functionality

# Custom codeunit

Run custom codeunit defined in policy

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.p2e.blanket.services.RunCustomCodeunit`
- Default `isActive`:
- Default interval:

Special functionality

# Data index builder

Transfer uncommitted data to search engine

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.tempusserva.lucene.index.ServiceIndexDataStore`
- Default `isActive`:
- Default interval:

Special functionality

# File index builder

Process files in queue

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `dk.tempusserva.lucene.index.ServiceIndexFileQueue`
- Default `isActive`:
- Default interval:

Special functionality

# Run BI SQL

Runs sql for entities with config called biSqlQuery

[Describe the function the service executes]

## Configuration

[If any configuration is available, and how to]

## Developer info

- Classname: `com.tsnocode.engine.services.BiService`
- Default `isActive`:
- Default interval: