Status Actions

The actions are fired when a certain condition is meet AND the item is in the parent status

- An item enters the status
- An item leaves the status
- Time has passed while an item was in this status

Timed actions

The time specified is relative to a value of the item

- Creation date
- Last change
- Last status update
- Dynamic **date field**

Time is specified in days and can assume both negative and positive values.

Action types

Notification

This action sends an email for users or groups.

Various options for the target email exists, including

- Raw email: Static reference
- User: A specific user in the database
- Field value: The contents of another field
  - Email
  - Phone (sent by SMS)
  - CPR nr (sent by eBoks)

Different types of content can be included in the emails

- Data from the record
- Document generated using data of the record

- Links to the record (internal users)
- Interface tokens (external users)

Read about special tags and formatting her Tutorial/Status_notifications

Task create

This will dynamically create new tasks for any 'Task list' fields found in the solution.

Codeunit

This action triggers the execution of a special codeunit: Codeunit/Statusaction.

Except for timed actions you would implement this inside a Codeunit/Formevents.

Export

The template is optional: If no template is found raw XML will be generated (FTP/mailto will not work).

The following types of routing are supported for template based exports

- Local file system (optionally a mapped share)
- Remote FTP server
- Sent by email

Please note that export actions are handled in seperate threads (performance), and there is no guarantee the operation succeds (tjeck the event log)

Values from record

The target reference may contain field references in the {FIELD} format, that will be populated at runtime.

Other special tags include

- {SagID}
- {DataID}
- {Resume}
- {NanoTime}

Note that values in records are not filtered for illegal og troublesome characters (such as \ or .)

Usage examples

Routing	Syntax	Syntax example
File system	*local filesystem path*	c:\exportFolder\{GROUP}\{TITLE}.docx
FTP server	*ftp location and connection string*	ftp://username:password@acme.com/exportFolder
Send by mail	"mailto:"+[email]+":"+[subject]+":"+[filename]	mailto:boss@acme.com:Attention:your_2013_report

Shift status

A status merely changes the status to a new value.

This type of action only makes sense to use in timed actions.

Webhook

This action performs an HTTP-request with parameters from the updated item and update the item based on the result.

Datapoint		Type	Notes
url	Required	String	The url that is to be called. This supports input of parameters from the item.
method	Optional	String	The HTTP-method of the call, one of: "POST", "GET", "PUT" or "DELETE". Default: "GET".

type	Optional	String	The datatype of data returned, supported types: "json", "raw" or "XML". Default: "json".
headers	Optional	Map	A map of extra headers that should be set
params	Optional	Map	Path parameters that should be set and send.
body	Optional	String	A string that will be set and send. Not available for "GET".
update	Optional	Map	Fields that should be updated based on the data returned. If type is "raw", only the first item in this list will be updated, and it will be set to the entire response. If type is "json", attributes with keys formatted as `[FIELDNAME]` will be updated from the given key in the returned data.

In `params` and `body`, values formatted as `[FIELDNAME]`, will be filled with data from the record.

Sample

```
{
  "url": "http://localhost:3001/users/1",
  "method": "PUT",
  "type": "json",
  "headers": {
    "authorization": "Bearer XYZ",
    "Content-type": "application/json"
  },
  "params": {
    "param": "DATA"
  },
  "body": {
     "id": 1,
     "name": "[NAME]",
     "email": "john.doe@example.com"
  },
  "update": {
    "[NAME]": "name"
  }
}
```

Revision #1
Created 8 April 2025 11:19:06 by Theis Villumsen
Updated 8 April 2025 11:25:25 by Theis Villumsen