Workflows (cs.workflow)

Release 15.4.1.3

CONTACT Software

Oct 29, 2018

Contents

CHAPTER 1

Service Levels

1.1 cs.workflow 15.4.1.3

1.1.1 Map Names of Inactive Users and Roles Correctly

This change introduces the view `cdbwf_resp_mapping`, which is basically a functional copy of `cdbwf_resp_browser` without the WHERE conditions.

This enables workflow tasks to map the names of their responsible roles and persons even if they have been marked organizational or deactivated after the fact and are not usable as responsibles for new tasks.

Note: To enable this fix, revert any patches for the `cdbdd_field` entries for classname `cdbwf_process_component` and field name `mapped_subject_name*`.

1.2 cs.workflow 15.4.1.2

1.2.1 Initial Render of Workflow Designer Doesn't Scale Well

Several optimizations reduce the runtime required for initially rendering the Workflow Designer of large workflows:

- Most objects are now instantiated in a single pass per class,
- initial rendering does not wait for images and responsible catalog data to be loaded.

The most important optimizations, however, will not be available per default, as they potentially disrupt customizing.

Warning: Please see Optional Performance Optimization for how to use these optimizations in your environment.

CHAPTER 2

What's New in cs.workflow 15.4.1?

2.1 New Status Names

To keep statuses consistent with other applications and classes, the following statuses have been renamed:

Object Lifecycle	Status	Old Name	New Name
cdbwf_process	30	Canceled	Failed
cdbwf_task	10	Ready	Execution
cdbwf_task	35	Canceled	Discarded
cdbwf_aggregate	10	Ready	Execution
cdbwf_aggregate	35	Canceled	Discarded

Also, Powerscript status classes have been renamed as shown in the following table. The old names are still accessible but using them will show a deprecation warning.

Warning: Please refactor any usage of these classes to get rid of all deprecation warnings. The old class names will be removed in the next release of cs.workflow.

Parent Class	Old Name	New Name
cs.workflow.processes.Process	READY	EXECUTION
cs.workflow.processes.Process	DONE	COMPLETED
cs.workflow.processes.Process	CANCELLED	FAILED
cs.workflow.processes.Process	DISMISSED	DISCARDED
cs.workflow.processes.Process	ONHOLD	FROZEN
cs.workflow.processes.Process	CHECKING	REVIEW
cs.workflow.tasks.Task	READY	EXECUTION
cs.workflow.tasks.Task	DONE	COMPLETED
cs.workflow.tasks.Task	REFUSED	REJECTED
cs.workflow.tasks.Task	CANCELLED	DISCARDED
cs.workflow.taskgroups.TaskGroup	READY	EXECUTION
cs.workflow.taskgroups.TaskGroup	DONE	COMPLETED
cs.workflow.taskgroups.TaskGroup	REFUSED	REJECTED
cs.workflow.taskgroups.TaskGroup	CANCELLED	DISCARDED
cs.workflow.taskgroups.ParallelTaskGroup	READY	EXECUTION
cs.workflow.taskgroups.SequentialTaskGroup	READY	EXECUTION
cs.workflow.taskgroups.ProcessCompletionTaskGroup	READY	EXECUTION

2.2 Workflow Designer

2.2.1 Persist changed description on blur

Changed task descriptions in the Workflow Designer are now persisted immediately after leaving the field (“on blur”) instead of requiring an explicit confirmation by clicking “ok”.

Warning: While making the description field’s behavior consistent with other fields, this change makes it easy for users to accidentally delete long texts.

CHAPTER 3

Service Levels

3.1 cs.workflow 15.4.0.13

3.1.1 Map Names of Inactive Users and Roles Correctly

This change introduces the view `cdbwf_resp_mapping`, which is basically a functional copy of `cdbwf_resp_browser` without the WHERE conditions.

This enables workflow tasks to map the names of their responsible roles and persons even if they have been marked organizational or deactivated after the fact and are not usable as responsibles for new tasks.

Note: To enable this fix, revert any patches for the `cdbdd_field` entries for classname `cdbwf_process_component` and field name `mapped_subject_name*`.

3.2 cs.workflow 15.4.0.12

3.2.1 Initial Render of Workflow Designer Doesn't Scale Well

Several optimizations reduce the runtime required for initially rendering the Workflow Designer of large workflows:

- Most objects are now instantiated in a single pass per class,
- initial rendering does not wait for images and responsible catalog data to be loaded.

The most important optimizations, however, will not be available per default, as they potentially disrupt customizing.

Warning: Please see Optional Performance Optimization for how to use these optimizations in your environment.

3.3 cs.workflow 15.4.0.11

3.3.1 Persist Changed Description on Blur in Workflow Designer

Changed task descriptions in the Workflow Designer are now persisted immediately after leaving the field ("on blur") instead of requiring an explicit confirmation by clicking "ok".

> **Warning:** While making the description field's behavior consistent with other fields, this change makes it easy for users to accidentally delete long texts.

3.4 cs.workflow 15.4.0.7

3.4.1 Remove Unused "cdbwf_ahwf%" Messages and Labels

The following obsolete configuration artifacts have been removed with this service level:

Labels:

- cdbwf_ahwf_00 - cdbwf_ahwf_03
- cdbwf_ahwf_05 - cdbwf_ahwf_17
- cdbwf_ahwf_19 - cdbwf_ahwf_26
- cdbwf_ahwf_28 - cdbwf_ahwf_43
- cdbwf_ahwf_copy
- cdbwf_ahwf_create
- cdbwf_ahwf_createT
- cdbwf_ahwf_info
- cdbwf_ahwf_modify
- cdbwf_ahwf_newT
- cdbwf_ahwf_new_
- cdbwf_ahwf_process

Error Messages:

- cdbwf_ahwf_m01- cdbwf_ahwf_m08

Operation Names:

- cdbwf_ahwf_copy
- cdbwf_ahwf_create
- cdbwf_ahwf_create_from_template
- cdbwf_ahwf_info
- cdbwf_ahwf_modify

3.5 cs.workflow 15.4.0.6

3.5.1 Support transporting workflow templates

This service level includes a new command-line tool to generate control statements for `cdbexp` to export a workflow structure.

Please see the docstring of `cs.workflow.updates.tools.export_process` for usage details.

3.6 cs.workflow 15.4.0.5

Warning: Please see the release notes of *15.3.0.15* (page 14) for details on a schema change.

3.7 cs.workflow 15.4.0.4

Warning: Because of changes made to the setting system of *CONTACT Tasks*, this service level requires `cs.taskmanager 15.4.0.6` or newer and CONTACT Elements 15.2 SL 20 or newer.

3.7.1 Setup for cs.web

This service level includes a lot of configuration changes to integrate with new `cs.web` features:

- *CONTACT Workflows* now requires `cs.web`.
- The following classes and operations have been activated for use with the REST API:
  - cdbwf_briefcase
  - cdbwf_form
    - CDB_ShowObject
    - cdbwf_submit_form
  - cdbwf_info_message
  - cdbwf_process
    - CDB_Create
    - CDB_Modify
    - CDB_Search
    - CDB_ShowObject
    - cdb_share_objects
    - cdbwf_cancel_workflow
    - cdbwf_dismiss_workflow
    - cdbwf_new_process_from_template
    - cdbwf_onhold_workflow
    - cdbwf_start_workflow
    - cdbwf_start_workflow_designer
  - cdbwf_process_component

  - cdbwf_protocol
    - CDB_Search
    - CDB_ShowObject
- Added an object header for processes.
- Sorted relationships with referer `cdbwf_process`.
- The Workflow Designer…
  - …now supports dragging object UI links to add objects as briefcase contents. Legacy URLs are not supported anymore. This enables users to add objects of `cs.web` applications by dragging them into a briefcase.
  - …has been updated to use the new URLs wherever possible.

Warning: As of now, only `CDB_ShowObject` can be used in the Workflow Designer. Modifying the process or using other operations of briefcase contents is currently not supported. This is a regression and will likely be possible again at some point.

Changed event_map of Process Powerscript class

The following methods of `cs.workflow.processes.Process` are now called in `pre` instead of `post_mask` for compatibility with `cs.web`:

- `setObjektArt` (`create` and `copy`)
- `check_copy_from_template` (`copy`)

Please make sure to check your code for potential incompatibilities.

3.7.2 Workflow Queue Log

For details, see the release notes of *15.3.1.13* (page 12).

3.7.3 Workflow Queue Language

For details, see the release notes of *15.3.1.13* (page 12).

3.7.4 Visibility of Info Message

Please see the release notes of *10.2.1.14* (page 17) for details on this change.

3.8 cs.workflow 15.4.0.1

3.8.1 Dedicated Service User

For details, see the release notes of *15.3.0.13* (page 15). The user account has to be marked as a system user (attribute `is_system_account`).

Warning: As a result of this change, cs.workflow 15.4.0.1 requires CONTACT Elements 15.2 SL 15 or newer.

3.8.2 Flag for Processes Completing Successful

For details, see the release notes of *15.3.0.13* (page 15).

3.8.3 URL Generation in Notification E-Mails

For details, see the release notes of *15.3.1.11* (page 12).

3.8.4 Briefcase Mode in Tables

The Python-based table column `cs.workflow.briefcases.IOTypeColumn` has been added, requiring CONTACT Elements 15.2 SL 5 or newer.

CHAPTER 4

What’s New in cs.workflow 15.3.2?

4.1 Forms

cs.workflow 15.3.2 supports attaching template-based forms to your tasks for easy entry of structured data. For further information, see the “Forms” chapter in the user and administration manuals.

CHAPTER 5

cs.workflow 15.3.1 Release Notes

Contents:

5.1 What's New in cs.workflow 15.3.1?

5.1.1 New Plugin for CONTACT Tasks 15.3.0

cs.workflow 15.3.1 contains a new plugin for the new release 15.3.0 of CONTACT Tasks (cs.taskmanager). For details please read the release notes of cs.taskmanager 15.3.0.

5.1.2 New Enum Interface

Starting with cs.workflow 15.3.1 SL 3, a new enum interface is used. This effectively breaks all access to the `key` and `index` attributes of `cs.workflow.briefcases.IOType`. Please make sure to change all accesses to `key` to `name`, and `index` to `value`.

5.1.3 Explicit Module Assignment for System Tasks and Parameters

Starting with cs.workflow 15.3.1 SL 9, objects of the classes `cdbwf_system_task_definition` and `cdbwf_parameter_definition` are explicitely assigned a `cdb_module_id`.

Please make sure to assign modules to your custom system task and parameter definitions.

5.1.4 Briefcase Mode in Tables

Starting with cs.workflow 15.3.1.10, the Python-based table column `cs.workflow.briefcases.IOTypeColumn` is added, requiring at least platform `15.1.15`.

5.2 Service Levels

5.2.1 cs.workflow 15.3.1.19

Map Names of Inactive Users and Roles Correctly

This change introduces the view `cdbwf_resp_mapping`, which is basically a functional copy of `cdbwf_resp_browser` without the WHERE conditions.

This enables workflow tasks to map the names of their responsible roles and persons even if they have been marked organizational or deactivated after the fact and are not usable as responsibles for new tasks.

Note: To enable this fix, revert any patches for the `cdbdd_field` entries for classname `cdbwf_process_component` and field name `mapped_subject_name*`.

5.2.2 cs.workflow 15.3.1.18

Persist Changed Description on Blur in Workflow Designer

Changed task descriptions in the Workflow Designer are now persisted immediately after leaving the field ("on blur") instead of requiring an explicit confirmation by clicking "ok".

Warning: While making the description field's behavior consistent with other fields, this change makes it easy for users to accidentally delete long texts.

5.2.3 cs.workflow 15.3.1.16

Support transporting workflow templates

This service level includes a new command-line tool to generate control statements for `cdbexp` to export a workflow structure.

Please see the docstring of `cs.workflow.updates.tools.export_process` for usage details.

5.2.4 cs.workflow 15.3.1.15

Warning: Please see the release notes of *15.3.0.15* (page 14) for details on a schema change.

5.2.5 cs.workflow 15.3.1.14

Visibility of Info Message

Please see the release notes of *10.2.1.14* (page 17) for details on this change.

5.2.6 cs.workflow 15.3.1.13

Workflow Queue Log

Starting with this service level, the queue executing system tasks and the subprocesses it spawns use a dedicated configuration file, enabling separate logging for system tasks. Enable logging by setting `CADDOK_DEBUG` in a new configuration file `$CADDOK_BASE/etc/wfqueue.conf`.

Workflow Queue Language

To control the language the workflow queue generates messages in (for example, when a system task fails), you can now explicitely set the language in the service arguments.

Warning: Starting with this service level, the default language will be english. If you want to change the queue's language back to german, you can add the string `--language de` to the service's current arguments. Make sure to separate arguments with at least one space (for example: `--svcuser bhmrmnn --language tr`).

Note: The service will immediately quit if the supplied language is not an active GUI language (see `Multilingual capability > Configuration` in the `Administration and Configuration Manual`).

5.2.7 cs.workflow 15.3.1.11

Dedicated Service User

For details, see the release notes of *15.3.0.13* (page 15).

Warning: As a result of this change, cs.workflow 15.3.1.11 requires CONTACT Elements 15.1 SL 24 or newer.

Flag for Processes Completing Successful

For details, see the release notes of *15.3.0.13* (page 15).

URL Generation in Notification E-Mails

Notification E-Mails directly link to the tasks in *CONTACT Tasks*, which is opened in a web browser by default.

If your users are mainly using the legacy PC Client, you can set the environment variable :envvar:`CADDOK_PREFER_LEGACY_URLS` to `"True"` to generate old-style operation URLs that will open in the PC Client (if registered to handle `cdb://` URLs).

System Task "Status Change" Can Now Unlock Objects

For details, see `Configuration of Standard System Task "Status Change"` in the administrator's manual.

CHAPTER 6

cs.workflow 15.3.0 Release Notes

Contents:

6.1 What's New in cs.workflow 15.3.0?

6.1.1 Additional Access Checks

Starting with cs.workflow 15.3.0.13, additional access checks are performed, which may impact your running workflows. Please make sure to test thoroughly.

For details, see Accessing Folder Contents.

6.1.2 Internationalization

CONTACT Elements 15 and its applications contain many newly translated and updated terms. The following paragraphs list all affected elements in this application, while details concerning the actual conversion (called "internationalization" for simplicity) can be found in each product's release notes.

Translated Relationship Access Profiles

The following Relationship Access Profiles have been renamed:

`Prozesskomponenten` Renamed to `Workflow components`

Please rename any usages in custom relationships accordingly.

Translated Catalogs

The following catalogs previously not supporting multiple languages are updated as detailed in the product release notes:

- **Responsible (`cdbwf_resp_browser`), used in**
  - Workflow (`cdbwf_process`)
  - Component (`cdbwf_process_component`)

  - Information (cdbwf_info_message)

Note: The attribute subject_name was removed from the class Component (cdbwf_process_component). If the attribute was used in your customizing, its usages have to be adapted accordingly.

Internationalized Catalogs

These classes had their name attributes internationalized:

- Workflow Category (cdbwf_process_categ)
- System Task Definition (cdbwf_system_task_definition) (to ml_name)

Their usages in the following classes have also been internationalized:

- Workflow (cdbwf_process)
- System Task (cdbwf_system_task)
- Parameter Definition (cdbwf_parameter_definition)

Translated Status Names

Status names have been internationalized for the classes:

- Workflow (cdbwf_process)
- Component (cdbwf_process_component)

6.1.3 Miscellaneous

The following configuration objects were removed, as they are not used anymore:

- The catalogs cdbwf_states_1, cdbwf_states_n, and cdbwf_objart
- The tables cdbwf_objart, and cdb_wflow_browse

6.2 Service Levels

6.2.1 cs.workflow 15.3.0.15

Schema Change

The attribute cdb_extension_class has been moved from class cdbwf_interactive_task to cdbwf_process_component to silence some unnecessary error messages while copying processes.

Warning: This change makes the view cdbwf_interactive_task_v obsolete. Please change usage of the view in your customizing accordingly.

Visibility of Info Message

Please see the release notes of *10.2.1.14* (page 17) for details on this change.

6.2.2 cs.workflow 15.3.0.13

Dedicated Service User

The arguments of service `cs.workflow.services.WFServer` have been changed. They should not contain a stringified dict anymore, but valid command line arguments. Existing service configurations will be updated automatically.

The service now requires the argument `--svcuser` containing a valid user ID. This user is a proxy for the service's license checks.

The service will _not_ start if no valid svcuser is given.

> **Warning:** The service user is shipped with cs.workflow, but will most likely not be created due to the update mode of the class `cdb_person`. Make sure to revert the `DELETED` patch containing the user `cs.workflow.svcuser` or use a custom `--svcuser` for the `WFServer`.

> **Warning:** As a result of this change, cs.workflow 15.3.0.13 requires CONTACT Elements 15 SL 58 or newer.

Flag for Processes Completing Successful

Previously, if a process is completing successfully so far was calculated in powerscript. To make the mechanism more predictable, it is replaced by the persistent attribute `cdbwf_process.completing_ok`:

- The attribute will contain a `1` for processes in status 0 (NEW) or 10 (READY). The object lifecycle `cdbwf_process` takes care of this.
- Cancelling or dismissing a workflow sets the attribute to `0` before entering the completion task.

The attribute replaces `cs.workflow.schemacomponents.SchemaComponent._processCompletingSuccessfully`, which is removed completely. It will be used in the object rule `wf-designer: process completing successfully` instead.

The attribute will be initialized by an update script as follows: - 1 for processes neither in status 30 (CANCELLED) nor 40 (DISMISSED) - 0 for processes in status 30 (CANCELLED) and 40 (DISMISSED)

CHAPTER 7

cs.workflow 10.2.1 Release Notes

cs.workflow 10.2.1 adds the following functionality to *cs.workflow*:

7.1 Frozen Workflows

Using the new operation “Freeze Workflow”, you can pause Workflows indefinitely (for details, see manual). The Workflow’s status becomes “frozen”, and a corresponding flag (“parent Workflow is frozen yes/no”) is added to its Tasks.

7.2 Object Life Cycle for Workflow Templates

Workflow templates now have their own Object Lifecycle (for details, see manual).

Note: cs.workflow 10.2.1 will run the update script “cs.workflow.updates.v10_2_1_0”, which will set the status of all Workflow Templates that are not “dismissed” to “approved”.

7.3 Limiting Users’ Choice of Templates

Depending on a Workflow Template’s Tasks, it may not be suitable for certain objects. For example, a template including a status change can only be used with objects that have an Object Lifecycle.

You can now assign Object Rules to Workflow Templates to enforce these limitations (for details, see manual).

7.4 Updated Relationship Profiles

Profiles for Relationships between cs.workflow objects have been updated. Your contact at CONTACT Software will assist you in updating your custom modifications.

7.5 Customizing Tasks

Tasks can now more easily be customized to fit your needs. Please ask your contact at CONTACT Software for further details.

7.6 Service Level 11

Starting with Service Level 11, the Access Control Domain “cdbwf: All Tasks” grants the Common Role “public” the access right “accept”.

7.7 Service Level 14

This service level changes the visibility of information messages of canceled and failed workflows. These are no longer filtered by the object rule in *CONTACT Tasks*.

To emulate the old behavior, follow these steps:

- Set the default setting `cs.workflow`, `cancel_info_on_failure` to `1` (or create it in the first place).
- Information messages that are not part of a workflow’s completion task is deactivated when the workflow is canceled or fails (`cdbwf_info_message.is_active` is set to `0`).

Existing workflows can be migrated using the optional update script `cs/workflow/updates/tools/deactivate_obsolete_info_messages.py`.

CHAPTER 8

cs.workflow 10.2.0 Release Notes

Contents:

8.1 Introduction

cs.workflow 10.2.0 is a comprehensive overhaul.

8.1.1 New Components and Functionality

- Constraints
- Briefcases
- Information Messages
- System Tasks
- Workflow Designer

8.1.2 Removed Components and Functionality

- Ad Hoc Workflows (including the Ad Hoc Workflow Designer)
- Information Tasks
- Manual Task Groups
- Object Roles and Object Rules

The Workflow Engine itself has been updated, so any customizing in Powerscript has to be reviewed, and possibly be adapted.

8.2 Additions

The following features have been added to cs.workflow in version 10.2.0:

8.2.1 System Tasks

With System Tasks, you can define Workflow Tasks that require no user interaction. They can perform any automatable task, and even run custom code.

8.2.2 Workflow Service

A CIM Database service (Message Queue) that runs System Tasks.

8.2.3 Information Messages

Messages are a new class of objects replacing Information Tasks. They are automatically generated whenever a System Task of type “Information” is ready. Messages are displayed by the Task Manager just like Information Tasks were.

8.2.4 Constraints

Constraints bringt together Workflow Tasks and Object Rules. Whenever a Task would be ready, all Object Rules referenced by its Constraints are checked, and potentially cancel the Task (see cs.workflow manual).

8.2.5 Briefcases

Briefcases are folder objects, and can contain objects to be used, modified, or created inside a Workflow. They also help Tasks communicate with each other. Briefcases replace the previously used Object Roles.

8.2.6 Workflow Designer

The Workflow Designer is a new eLink application. You can easily create Workflows, and also visualize running or already completed Workflows. It replaces the “Ad Hoc Workflow Designer”, and complements the usual structure view.

8.3 Changes in cs.workflow 10.2.0

8.3.1 Task Structure

Regular Tasks can now be used as top-level Tasks in Workflows. Previously, these had to be wrapped in top-level Task Groups.

8.3.2 Status Changes

In previous versions, running and stopping (e.g. canceling and dismissing) Workflows were tightly coupled to the Workflow’s status change operation. Now, they are dedicated operations, which will execute all necessary status changes. Status changes can not be executed manually on Workfows anymore.

8.4 Deprecations

8.4.1 Ad Hoc Workflows

Starting with 10.2.0, the two classes `Managed Processes` and `Ad Hoc Workflows` are combined into the single class Workflows (`cdbwf_process`).

The eLink application to create "Ad Hoc Workflows" has been replaced by the new Workflow Designer.

The operations *Ad Hoc Workflow/New* and *Ad Hoc Workflow/Create from Template* are renamed to *Workflow/New* and *Workflow/Create from Template*.

The User Exit `create_ad_hoc_workflow` is removed. Its operations are now implemented in the Powerscript class `cs.workflow.briefcases.BriefcaseContent` (see Administrator's manual).

8.4.2 Information Tasks

The Task Type "Information" is replaced by *Information Messages* (page 19).

8.4.3 Manual Task Groups

Manual Task Groups are removed. Equivalent behavior can be achieved by using nested parallel and sequential Task Groups (see *Migration* (page 21)).

8.4.4 Object Roles and Object Rules

Object Roles and Object Rules are replaced by Briefcases.

Status changes can now be used at any time in a Workflow by using the corresponding System Task (see *Migration* (page 21)).

The classes `cdbwf_p2obj`, `cdbwf_p2doc`, `cdbwf_p2mod`, `cdbwf_p2part`, `cdbwf_p2task` and all Relationships they were used in are removed.

8.5 Powerscript

Some changes in Powerscript code may be relevant to your customizing. You should review (and possibly adapt) your code having read this document.

8.5.1 Import Names

Powerscript classes have been renamed and moved across modules. Please check both your code and configuration elements for references to old names.

`cs.workflow.SchemaComponent` Renamed to `cs.workflow.schemacomponents.SchemaComponent`

`cs.workflow.Process` Renamed to `cs.workflow.processes.Process`

`cs.workflow.AdHocWorkflow` Removed (see *Deprecations* (page 20))

`cs.workflow.ObjectRole` Removed (see *Deprecations* (page 20))

`cs.workflow.ObjectRule` Removed (see *Deprecations* (page 20))

`cs.workflow.ObjectReference` Removed. A similar behavior is offered by `cs.workflow.briefcases.BriefcaseReference` (see *Powerscript References* (page 21))

cs.workflow.MasterObjectReference Removed. A similar behavior is offered by `cs.workflow.briefcases.BriefcaseReference` (see *Powerscript References* (page 21))

cs.workflow.TaskGroup Renamed to `cs.workflow.taskgroups.TaskGroup`

cs.workflow.ParallelTaskGroup Renamed to `cs.workflow.taskgroups.ParallelTaskGroup`

cs.workflow.SequentialTaskGroup Renamed to `cs.workflow.taskgroups.SequentialTaskGroup`

cs.workflow.ManualTaskGroup Removed (see *Deprecations* (page 20))

cs.workflow.Task Renamed to `cs.workflow.tasks.Task`

cs.workflow.ExecutionTask Renamed to `cs.workflow.tasks.ExecutionTask`

cs.workflow.ApprovalTask Renamed to `cs.workflow.tasks.ApprovalTask`

cs.workflow.ExaminationTask Renamed to `cs.workflow.tasks.ExaminationTask`

cs.workflow.InformationTask Removed (see *Deprecations* (page 20))

cs.workflow.Protocol Renamed to `cs.workflow.protocols.Protocol`

8.5.2 Status Changes

Previously, all behavior controlling a Workflow's progress was implemented in its status change User Exits (see *Status Changes* (page 19)). Now, they are refactored to be dedicated operations.

If you extended status change logic, you might have to migrate your code to the methods `propagate_done`, `propagate_cancel`, and `propagate_refuse` of Task Groups and Workflows.

8.5.3 Powerscript References

The class `cs.workflow.MasterObjectReference` was removed. If you use this class in your code, you will have to either remove your code, or refactor it to use `cs.workflow.briefcases.BriefcaseReference`. This class also contains the reference `ReferredObject`.

8.6 Migration

When updating to cs.workflow 10.2.0, you will have to migrate your data and configuration.

8.6.1 Data

Data is migrated automatically (see *Additions* (page 18) and *Deprecations* (page 20)).

The migration script will update all Workflows and Workflow Templates to the new version. You should always check the migration results. Especially Templates may be optimized manually.

The migration script works as follows:

- Information Tasks are replaced by System Tasks of type *Information*

  If the Task was already *ready* or *done*, an *Information Message* is created also.

  Tip: When `Information Messages` are created, E-Mail notifications may be sent (subject to configuration). If you want to prevent this to happen during migration, you can temporarily set the environment variable CADDOK_STOP_EMAIL_NOTIFICATION to any string (for example "TRUE"). To re-enable notifications after the migration script is finished, just remove the variable.

- Manual Task Grops are repaced by Parallel Task Groups

  Each Task in the Task Group is replaced by a new Sequential Task Group containing an Execution Task titled Activate manual task <task_name>, and the original Task.

- Object Roles are migrated
  - Each Object Role becomes a Briefcase
  - The Role's objects are assigned to the Briefcase

  Note: Custom object assignments (by deriving from ObjectReference) are not migrated. To mitigate this, you can run the script cs.workflow.updates.assign_objects:

  ```
  powerscript assign_objects.py --help
  ```

  - If needed, a *Status Change* System Task titled "<Object Role's name> + successful" is added to the Workflow's Completion Task Group.
  - If needed, a *Status Change* System Task titled "<Object Role's name> + failure" is added to the Workflow's Completion Task Group.
  - A corresponding CIM Database Object Rule is created (if not existing yet).
  - Parameters are created (filtered by the generated CIM Database Object Rules).

8.6.2 Configuration

cs.workflow 10.2.0 will automatically update your configuration.

Please check your system afterwards if you customized cs.workflow.

8.7 Service Levels

8.7.1 Changes in Service Level 8

Updated System Tasks

Each System Task now starts its own process, using the account of the person who started the Workflow. This user is written to the Workflow's new attribute started_by when started.

For Workflows that are already running, the Workflow owner will be written to started_by, if it is a person.

Workflows owned by a role will not be migrated. You will have to provide a value for started_by manually to be able to run System Tasks inside this Workflow.

Updates Status Change Tasks

Status Change Tasks now also take input from Information Briefcases.

When a status change for an object fails, it is attempted for the other objects, then the Status Change Task's status becomes canceled.

8.7.2 Changes in Service Level 15

Changed Powerscript References `Previous` and `Next`

The Powerscript References `cs.workflow.schemacomponents.SchemaComponent.Previous` and `cs.workflow.schemacomponents.SchemaComponent.Next` now return the following:

- `Previous` returns the Workflow's Task or Task Group that triggered the calling object's status change.
- `Next` returns the Workflow's Task or Task Group that was/will be triggered by the calling object.

8.7.3 Changes in Service Level 16

Default Task Titles and Owners

Whenever a Workflow starts, it sets the title of all Tasks that do not currently have one to its Task Type (Execution, Approval, or Review).

It will also check if all Tasks have an owner ("Responsible" person or Role). Adding Tasks to already running Workflows can result in a Task becoming ready that does not have an owner. In this case, the Task's "Responsible" (`subject_id`) attribute is filled with the Workflow owner when set ready.

`cdb_module_id` Added to System Task Definitions

System Task Definitions now use the attribute `cdb_module_id` to better support the Component Architecture.

Automatic Inclusion of Optional Updates

Updates classified as "optional" (for example access rights, Object Lifecycles, Object Rules) will be automatically applied when updating from cs.workflow 10.1.0 to guarantee correct functionality.

Workflow Deadlines

When starting a Workflow, its starting date will be set. If a max. duration is set, the Workflow's deadline will be set to starting date + max. duration.

8.7.4 Changes in Service Level 19

Workflow Service Added to Navigation

Jobs to be executed by the asynchronous Workflow Service can now be found in the Navigation panel at *Administration/Configuration -> Adminstration -> Workflows -> System Task Jobs*.

List of Figures

List of Tables

Index