

The image features the Hybris Software logo, which consists of a stylized 'h' inside a circle, followed by the words 'hybris software' in a bold, sans-serif font. Below this, the text 'An SAP Company' is written in a smaller, sans-serif font. The background is a dark blue gradient with a low-poly, geometric pattern of various shades of blue and white, creating a modern and abstract look.

(h) hybris software
An SAP Company

hybris Developer Training Part I - Core Platform

Workflows

Overview

Data Model

Working with workflows

Automated steps

- The hybris **workflow** extension enables modelling and processing workflows which are usually **backoffice** related user interaction
- A workflow can represent:
 - Complex processes with multiple tasks and users
 - Creating a new product in the catalog
 - Handling customer complaints
 - Quote negotiation (B2B)
 - Single and easy tasks which can be assigned to a backoffice user
 - Changing the image of a product

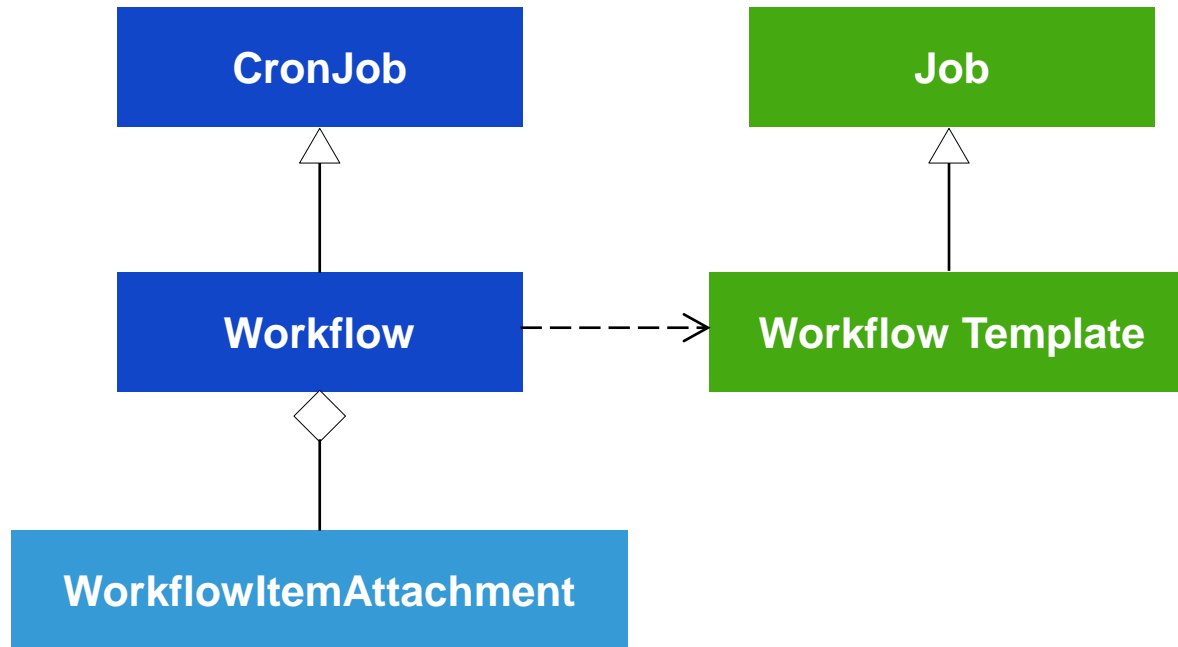
- ➔ Design human- and system-based workflows
- ➔ Start a workflow via hybris CronJobs
- ➔ Allocate users and user groups to certain tasks
- ➔ Comment on the task
- ➔ Add task-relevant attachments
- ➔ Get real-time notification via e-mail
- ➔ Track a workflow
- ➔ Cockpit & Process Engine Integration

Overview

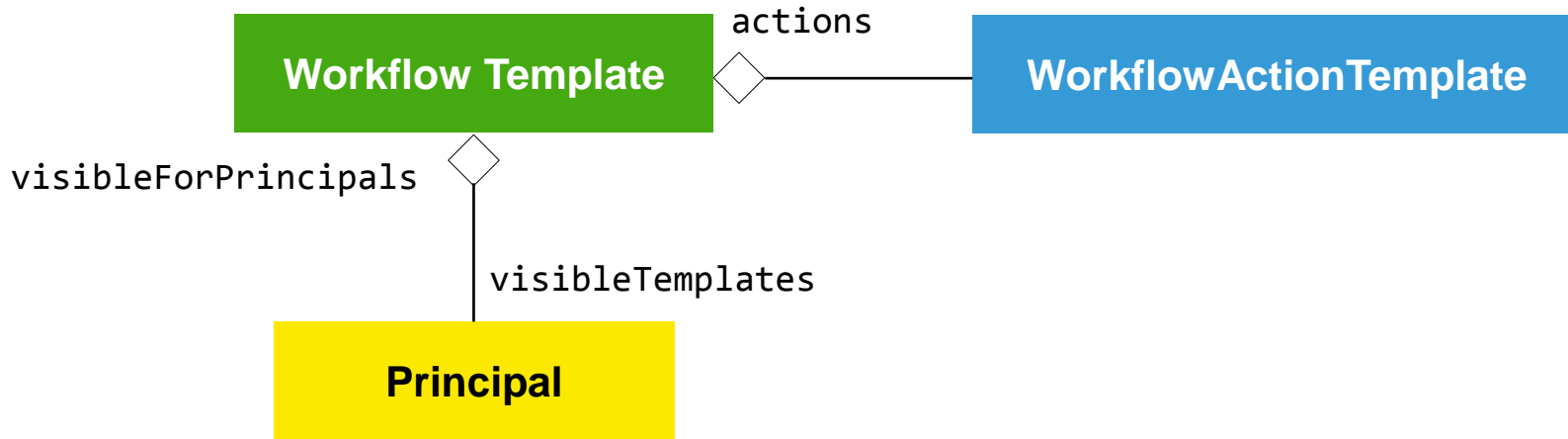
Data Model

Working with workflows

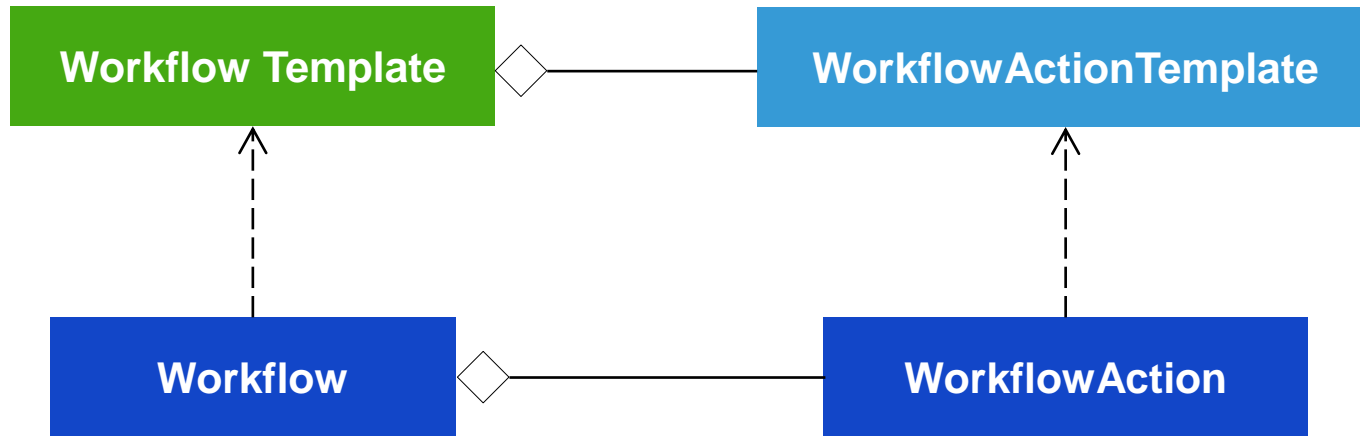
Automated steps



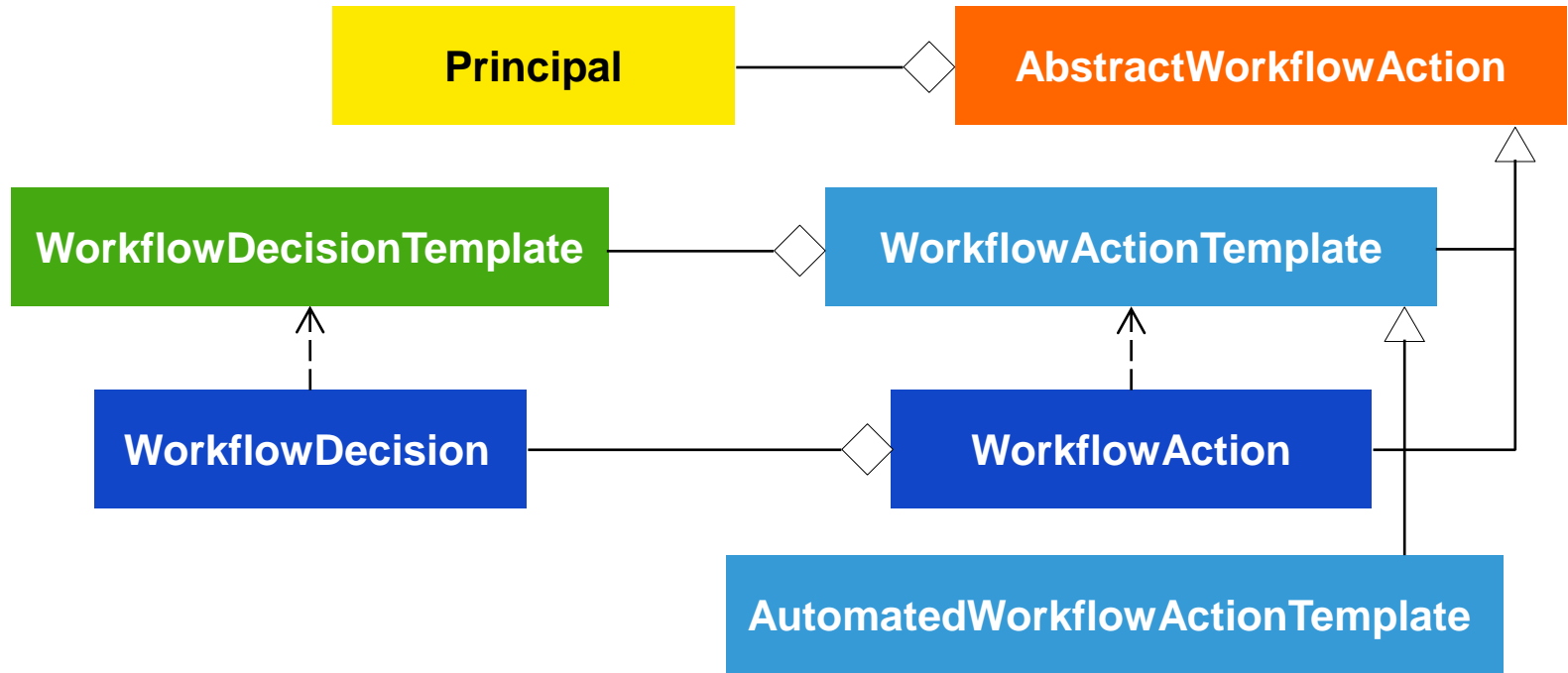
- ➔ Technically workflows are CronJobs
- ➔ Generated out of a workflow template
- ➔ Might have attached items



- ➔ Template for a real workflow
- ➔ Defines workflow structure through WorkflowActionTemplates



- ➔ Concrete task the assigned principal has to do
 - ➔ Principals should work on the task and
 - ➔ Finish the action by choosing a decision
- ➔ Gets activated through incoming decisions (AND, OR)



- ➔ Action definition in a workflow template
- ➔ Flow of WorkflowActionTemplates determined by WorkflowDecisionTemplates
- ➔ Each WorkflowActionTemplate has an assigned principal

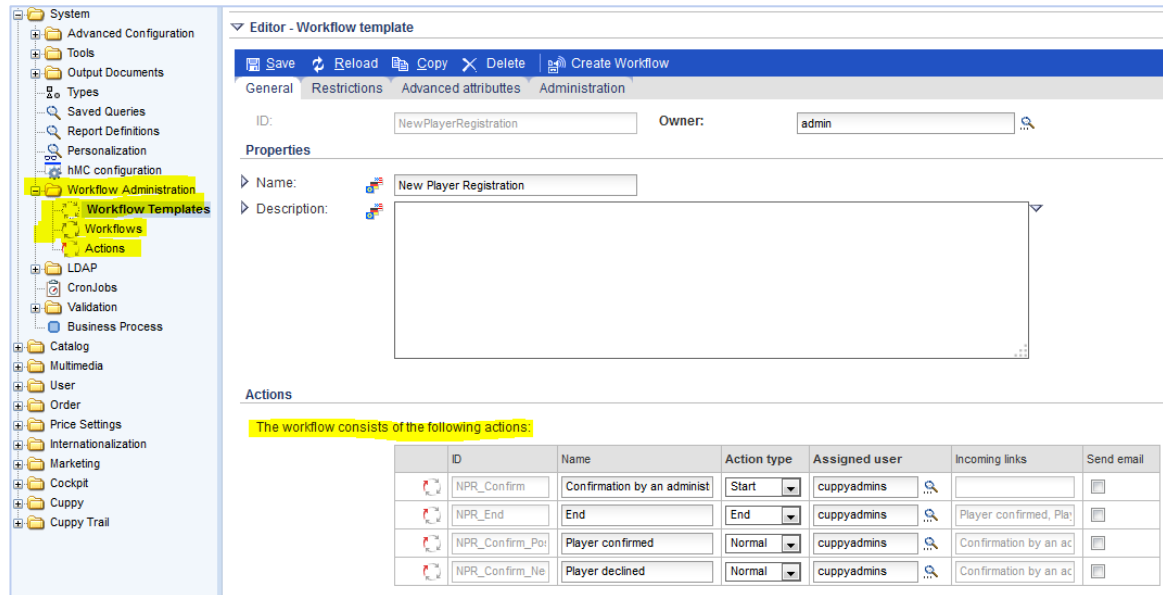
Overview

Data Model

Working with workflows

Automated steps

- ➔ Through API
 - ➔ Create WorkflowTemplateModel, ... (Service Layer)
 - ➔ ImpEx
- ➔ hybris Management Console (hMC)



Editor - Workflow template

Save Reload Copy Delete Create Workflow

General Restrictions Advanced attributes Administration

ID: NewPlayerRegistration Owner: admin

Properties

Name: New Player Registration

Description:

Actions

The workflow consists of the following actions:

ID	Name	Action type	Assigned user	Incoming links	Send email
NPR_Confirm	Confirmation by an administ	Start	cuppyadmins		
NPR_End	End	End	cuppyadmins	Player confirmed, Pla	
NPR_Confirm_Po	Player confirmed	Normal	cuppyadmins	Confirmation by an ac	
NPR_Confirm_Ne	Player declined	Normal	cuppyadmins	Confirmation by an ac	

- ➔ Through API
 - ➔ WorkflowService: creating, monitoring, ...
 - ➔ WorkflowProcessingService: starting, terminating, ...
- ➔ Activation Script
 - ➔ Evaluated each time an items is created, saved, removed
 - ➔ Access to *item*, *type*, *initial values* and / or *current values*
- ➔ Manually through UI (cockpit) if user is in `visibleForPrincipals`

Displaying Workflows



- ➔ In Product Cockpit
- ➔ Workflows for the current user

The screenshot displays the Hybris Product Cockpit interface. On the left, a sidebar menu includes sections for 'Menu', 'Workflows', 'Quickcollection', 'Catalog', 'Overview', 'Lead changes', 'Duplicated products', 'A48 family', 'History', and 'Workflow'. The 'Workflows' section is expanded, showing a list of workflows. The main area displays a workflow diagram titled 'Promotions My Ad hoc Workflow'. The diagram shows a 'Start' node leading to a 'Decision' node, which branches into 'Approval' and 'No Approval' paths. The 'Approval' path leads to a 'Task' node, which then leads to a 'End' node. To the right of the diagram is a table with the following columns: 'Start', 'Current step', 'Assigned user', 'Since', 'Thumbnail', 'Item Type', 'ID', and 'Name'. The table contains several rows of data, including workflow instances for 'Workflow Product Manager Demo' and 'Workflow Product Manager Demo 1'.

Start	Current step	Assigned user	Since	Thumbnail	Item Type	ID	Name
2013-06-03 02:20:06		Workflow Product Manager Demo	2013-06-03 02:20:06		Product	1014790	PowerShot A480
2013-06-03 02:20:06		Workflow Product Manager Demo	2013-06-03 02:20:06		Product	1014794	PowerShot A480
2013-06-03 02:20:06		Workflow Product Manager Demo	2013-06-03 02:20:06		Product	1014793	PowerShot A480
2013-06-03 02:20:06		Workflow Product Manager Demo	2013-06-03 02:20:06		Product	4073007	SDHC Card
2013-06-03 02:20:06		Workflow Product Manager Demo	2013-06-03 02:20:06		Product	4073010	SDHC Card
2013-06-03 02:20:06		Workflow Product Manager Demo	2013-06-03 02:20:06		Product	4080190	SDHC Card

Overview
Data Model
Working with workflows
Automated steps

- ➔ Use automated steps if no manual interaction is needed
 - ➔ Any service layer business logic
- ➔ Implementation steps
 1. Implement interface `AutomatedWorkflowTemplateJob`
 2. Create a bean
 3. Any implementing class gets added automatically to `AutomatedWorkflowTemplateRegistry` during server start-up

- ➔ Imagine you want to create a workflow for assigning a Car to each newly created Employee. What do steps you would have to do
 - ➔ The assignment task should be done by admin
 - ➔ Admin can also decide to let the Car being assigned automatically

- ➔ <https://wiki.hybris.com/display/release5/workflow+Extension+-+Technical+Guide>
- ➔ <https://wiki.hybris.com/pages/viewpage.action?pageId=141793777>

(x)