



# How to Build an SAP Workflow on SAP S/4HANA in 40 Minutes

Martin Maguth, MaxECM Solutions and  
Sue Keohan, Lincoln Labs

Session ID: ASUG82847

# About the Speakers

## Speaker Name

- Martin Maguth
- Managing Partner at MaxECM Solutions LLC and long-time ASUG Volunteer
- Fun fact: Workflow Hacker

## Speaker Name

- Susan Keohan
- Senior Workflow Developer at Lincoln Labs and long-time ASUG Mentor
- Fun fact: Workflow Authority

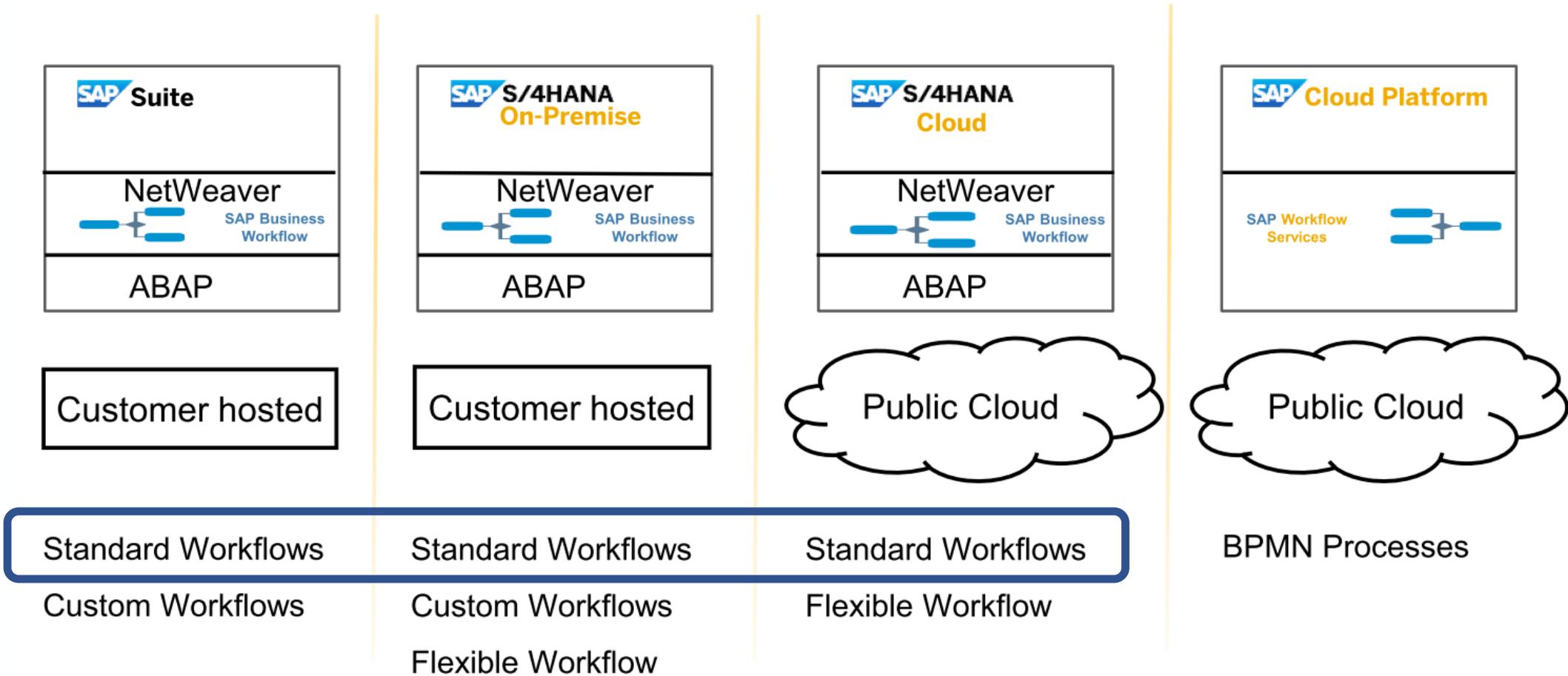
# Key Outcomes/Objectives

1. Key components of SAP Workflow engine on SAP S/4HANA
2. How to build a real workflow process
3. Key tools for SAP Workflow administration

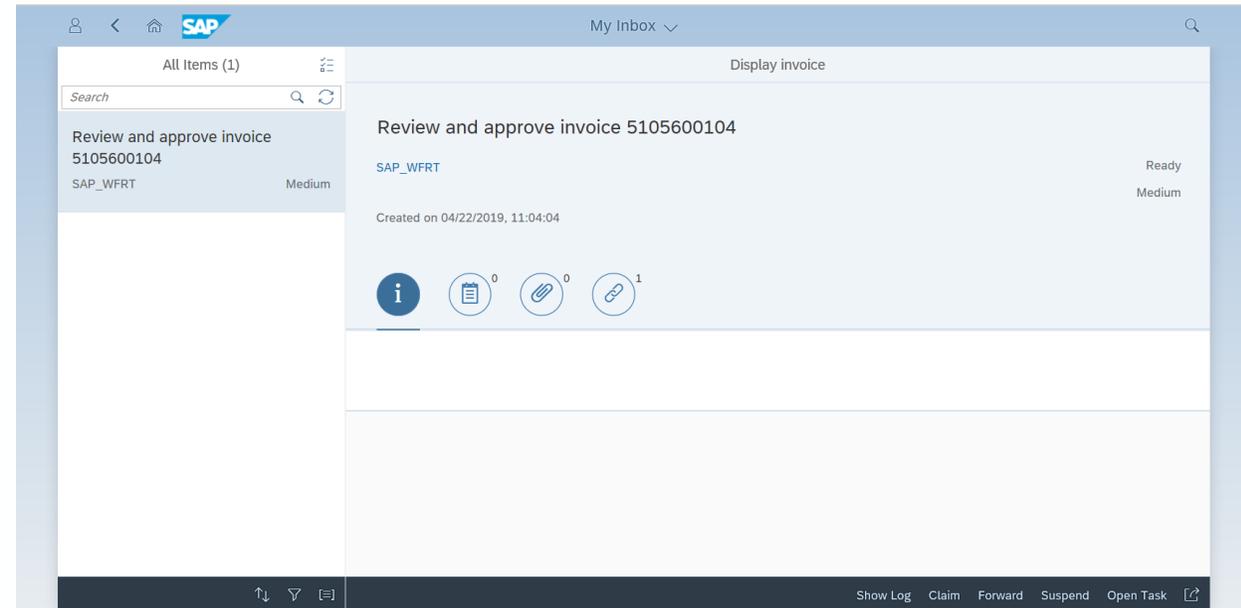
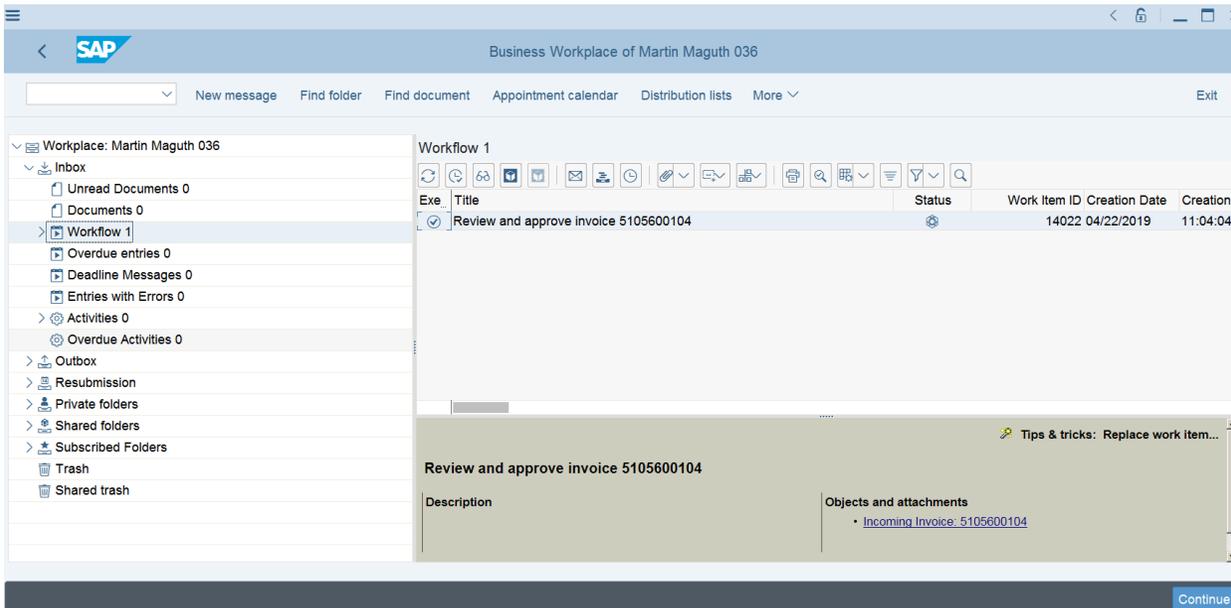
# Agenda

1. Introduction
2. Workflow architecture
3. Container concept
4. Event concept
5. Workflow administration

# Introduction – OnPremise Vs. Cloud Platform



# Introduction – Fiori integration



- Work items can be accessed via SAPGUI and FIORI out of the box
- Note: Workflows created on-premise with SAPGUI development are available for configuration in 'Manage workflows' app in FIORI, but not vice versa

# Agenda

1. Introduction
2. Workflow architecture
  - Object types
  - Single step tasks
  - Workflow templates
  - Responsibilities
3. Container concept
4. Event concept
5. Workflow administration

# Object types

- Represent a “real-world” object that exists in the business environment
- Stored in SAP Object Repository (Transaction SWO1)
- Object type components are
  - Interfaces (where does the object apply)
  - Key (how is it identified)
  - Attributes (how is it described)
  - Methods (what can be done with it)
  - Events (what can happen to it)

# Object types

Object type BUS2081 Incoming Invoice

- Interfaces
  - IFSAP SAP standard interface
  - IFIDOCIN IDoc inbound processing
  - IFARCH21 SAP ArchiveLink interface
  - IFGOSXSERV Funct. Extension Gener. Object Services
  - IFNAST output control
- Key fields
  - IncomingInvoice.InvoiceDocNumber Invoice Document Number
  - IncomingInvoice.FiscalYear Fiscal Year
- Attributes
  - IncomingInvoice.ObjectType Object type
  - IncomingInvoice.MessageTypesInbound Possible IDoc message types/basic IDoc types for inbound processing
  - IncomingInvoice.CompanyCode Company Code
  - IncomingInvoice.InvoicingParty Invoicing Party
  - IncomingInvoice.Currency Currency Key
  - IncomingInvoice.Status Status
  - IncomingInvoice.ReleaseAgent Virtual: Processor for Release for Posting
  - IncomingInvoice.UserName User Name
- Methods
- Events

- Example:  
Object BUS2081  
(Incoming invoice)
- Key consists of document number, and fiscal year
- Many standard methods and events to process invoice in workflow

# Single step task

- Represents a specific step in a workflow process
- Access with transaction PFTC
- Key components of single step tasks are
  - Object type (what is worked on)
  - Method (what is done with the object)
  - Possible users (who may execute the task in general)
  - Work item text (task headline that appears in inbox)
  - Descriptions (texts for inbox and deadline notifications)
  - Triggering/terminating event (when is task started/complete)

# Single step task

The screenshot shows the SAP configuration interface for a standard task. The task is identified as '20000704 MMIV Display' with the name 'Display invoice document' and package 'MRM'. The application component is 'MM-IV'. A blue box highlights the navigation tabs: 'Basic data', 'Description', 'Container', 'Triggering events', 'Terminating events', 'Default rules', and 'SAPphone'. The 'Basic data' tab is active, showing fields for 'Name' (MMIV Display, Display invoice document), 'Release status' (Released), 'Work Item Text' (Check invoice blocked for price reasons & \_WI\_OBJECT\_ID.INVOICEDOCNUMBER& ...), 'Object method' (BOR Object Type, BUS2081 Incoming Invoice, Method: DISPLAY), and 'Execution' options (Background processing, Executable with SAPforms, Confirm end of processing).

Standard Task: Display

Display <-> Change Check Where-used list Other standard task... More ▾

Standard task: 20000704 MMIV Display  
Name: Display invoice document  
Package: MRM Appic. Component: MM-IV

Basic data Description Container Triggering events Terminating events Default rules SAPphone

Name  
Abbr.: MMIV Display  
Name: Display invoice document  
Release status: Released ▾

Work Item Text  
Work item text: Check invoice blocked for price reasons & \_WI\_OBJECT\_ID.INVOICEDOCNUMBER& ...

Object method  
Object Category: BOR Object Type ▾  
Object Type: BUS2081 Incoming Invoice  
Method: DISPLAY Display  
 Synchronous object method  
 Object method with dialog

Execution  
 Background processing  Executable with SAPforms  
 Confirm end of processing

- Example: TS20000704 (Display invoice)
- Work item text (?)
- Object method reference: Use BOR object or ABAP Class
- Define task container
- Assign events and default rules
- Determine execution type

# Workflow template

- Represents a chain of single step tasks and events
- Access with transaction SWDD (Workflow Builder)/SWDD\_SCENARIO (New Workflow Builder)
- Multi step task components are
  - Control
    - Single step task
    - Dialog vs. background
  - Responsibility
    - Selected agents at runtime
  - Deadlines
    - Earliest/latest requested start/end
    - Notifications

# Workflow template

The screenshot displays the SAP Workflow Builder interface for a workflow template named 'LeaOfAbseHTM'. The main window shows a graphical model of the workflow, which is a vertical sequence of steps with a decision point and a loop. The steps are:

- Workflow starts
- Approval for LOA (with HTML)
- Request of Leave Of Absence is approved? (Decision)
- Notification of rejection (with HTML) (if 'No')
- Find employee (get pers.officer)
- Entry for Leave Of Absence
- Leave of absence mail to employee (if 'Yes')
- Leave of absence mail to manager (if 'Yes')
- 2 From 2 (Merge)
- Resubmitting? (Event)
- Workflow completed

The left sidebar shows the 'Steps' list for this workflow, including steps like '000006 Resubmitting?', '000251 Approval for LOA (with HTML)', and '000486 Leave of absence mail to manager'. Below the steps list is a 'Step Types That Can Be Inserted' menu with options like Activity, Web activity, Send Mail, Form, User Decision, etc.

At the bottom, the 'Result of Syntax Check' table is empty:

Step Number	Message
-------------	---------

- Example: WS1000062 (Leave of absence with HTML)
- Shown in Workflow Builder

# Responsibilities

- Represent the possible and the selected user(s) for the execution of a task
- Can be assigned statically or dynamically
- Generally defined based on the HR organizational structure
- Users can be excluded

# Responsibilities

SAP Standard task: Maintain Agent Assignment

Create agent assignment... Delete agent assignment Attributes... Expand All Collapse all Org. assignment Update Index Column configuration More

Name	ID	General or Background Task	Task Version
Display invoice document	TS 20000704	General Task	

- General task assignment for single step as possible users

Agents

Rule 01000003 Personnel officer employee

Role Binding (Exists)

Rule

Organizational Unit

Job

Position Task Complete

Work center Confirm End of Processing

User

Expression

Workflow initiator (expression)

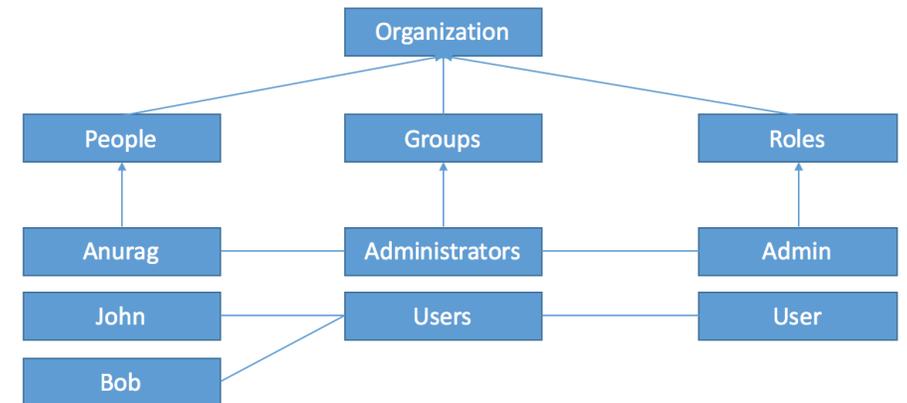
Superior of workflow initiator (rule)

- Rule 1000003 assigned for agent determination at runtime in workflow template definition
- Many other options possible ...

# Rules

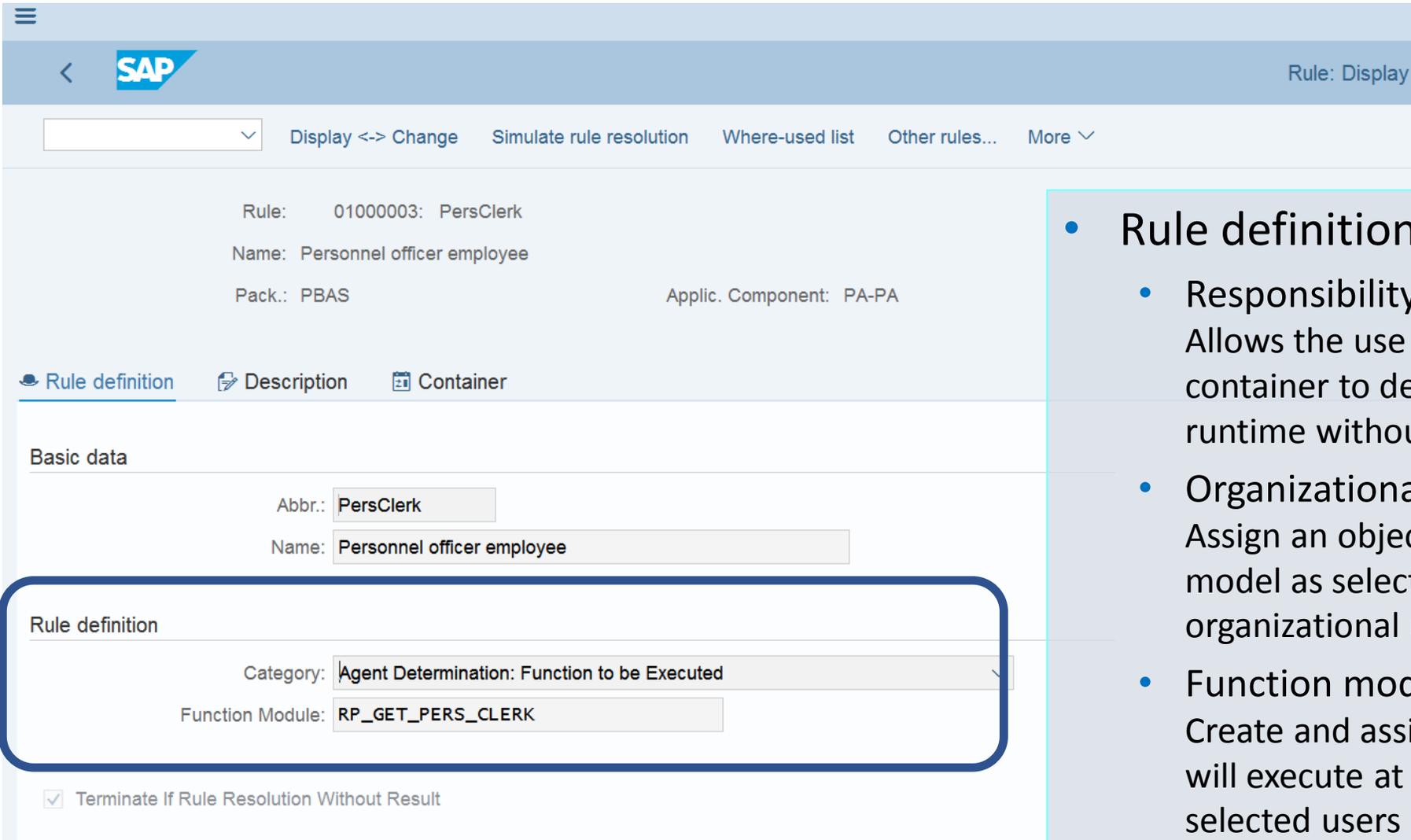
Rules allow for the dynamic agent determination to a task in a workflow at runtime.

Rules are created and maintained with transaction PFAC.



[This Photo](#) by Unknown Author is licensed under [CC BY-SA](#)

# Rules



The screenshot shows the SAP Rule definition interface. At the top, there is a navigation bar with the SAP logo and a 'Rule: Display' button. Below this is a toolbar with options: 'Display <-> Change', 'Simulate rule resolution', 'Where-used list', 'Other rules...', and 'More'. The main content area displays the following information:

- Rule: 01000003: PersClerk
- Name: Personnel officer employee
- Pack.: PBAS
- Applic. Component: PA-PA

Below this information are three tabs: 'Rule definition' (selected), 'Description', and 'Container'. The 'Rule definition' tab is active and shows the following fields:

- Abbr.: PersClerk
- Name: Personnel officer employee
- Category: Agent Determination: Function to be Executed
- Function Module: RP\_GET\_PERS\_CLERK

At the bottom, there is a checkbox labeled 'Terminate If Rule Resolution Without Result' which is checked.

- Rule definition categories:

- Responsibility

- Allows the use of elements in the rule container to determine agents at runtime without programming

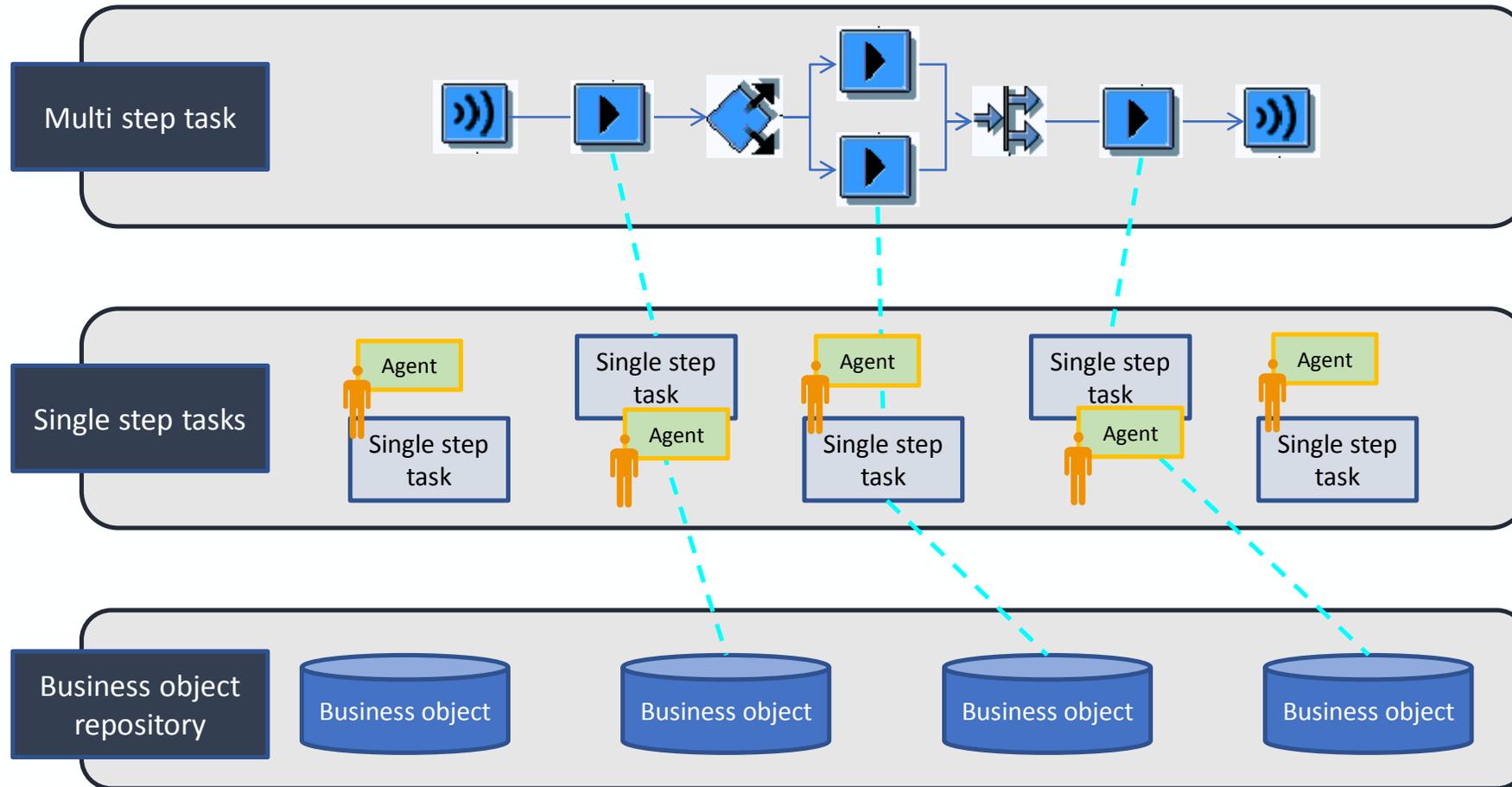
- Organizational data

- Assign an object from HR organizational model as selected user, such as organizational unit, position, or job

- Function module

- Create and assign a function module that will execute at runtime to determine selected users

# Workflow architecture



# Workflow architecture - Demo

Step 1: Explore BUS2081 object

Step 2: Create single step task

Step 3: Create workflow template

# Agenda

1. Introduction
2. Workflow architecture
3. Container concept
  - Definition
  - Container types
  - Architecture
4. Event concept
5. Workflow administration

# Container definition

**Containers** store information and allow the exchange of that information among the different components of a workflow process.

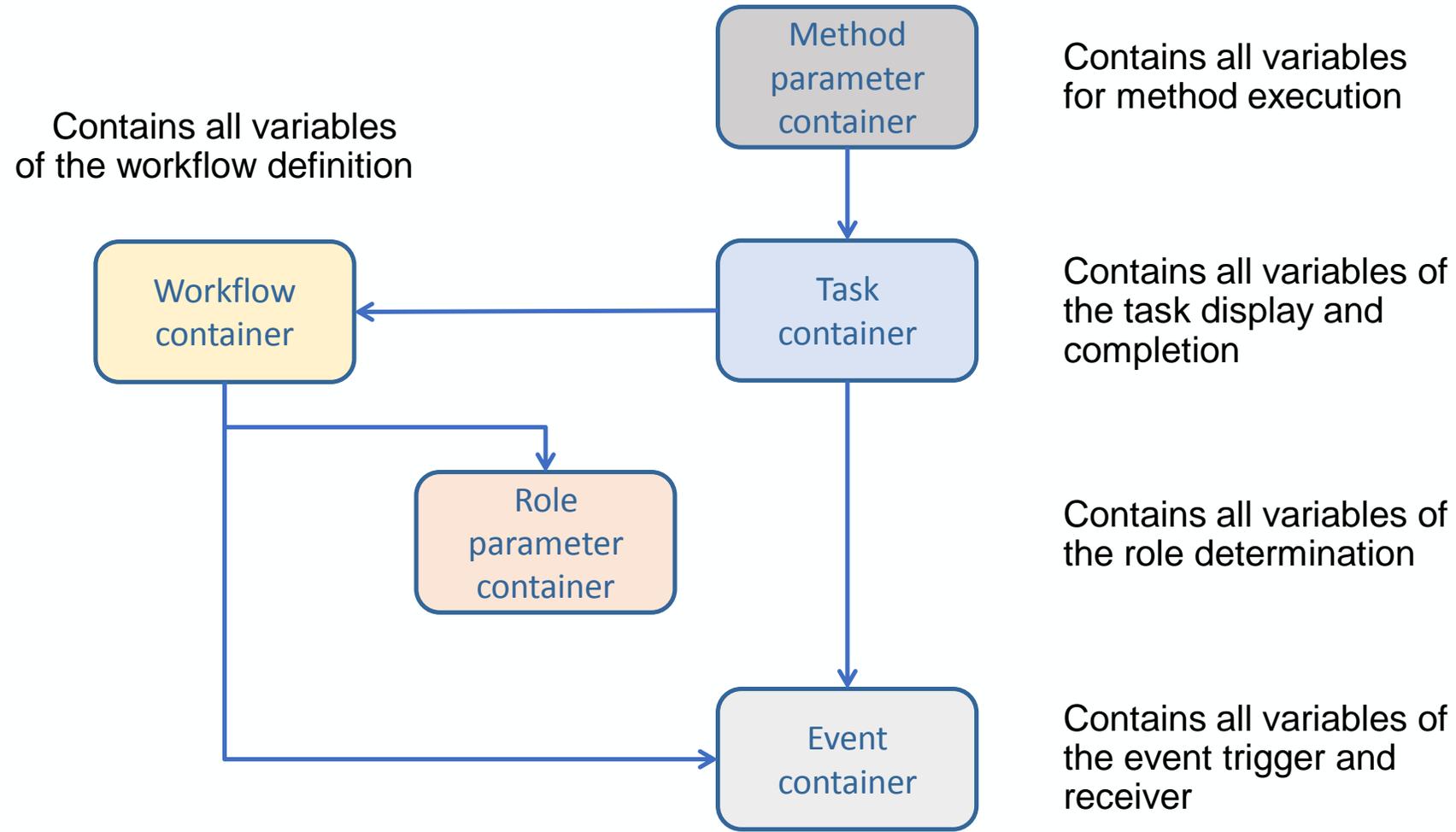
The information is stored as a data element and processed at runtime through **bindings**.

Bindings administer the flow from one container to another. Bindings are defined at time of implementation and are used at runtime.

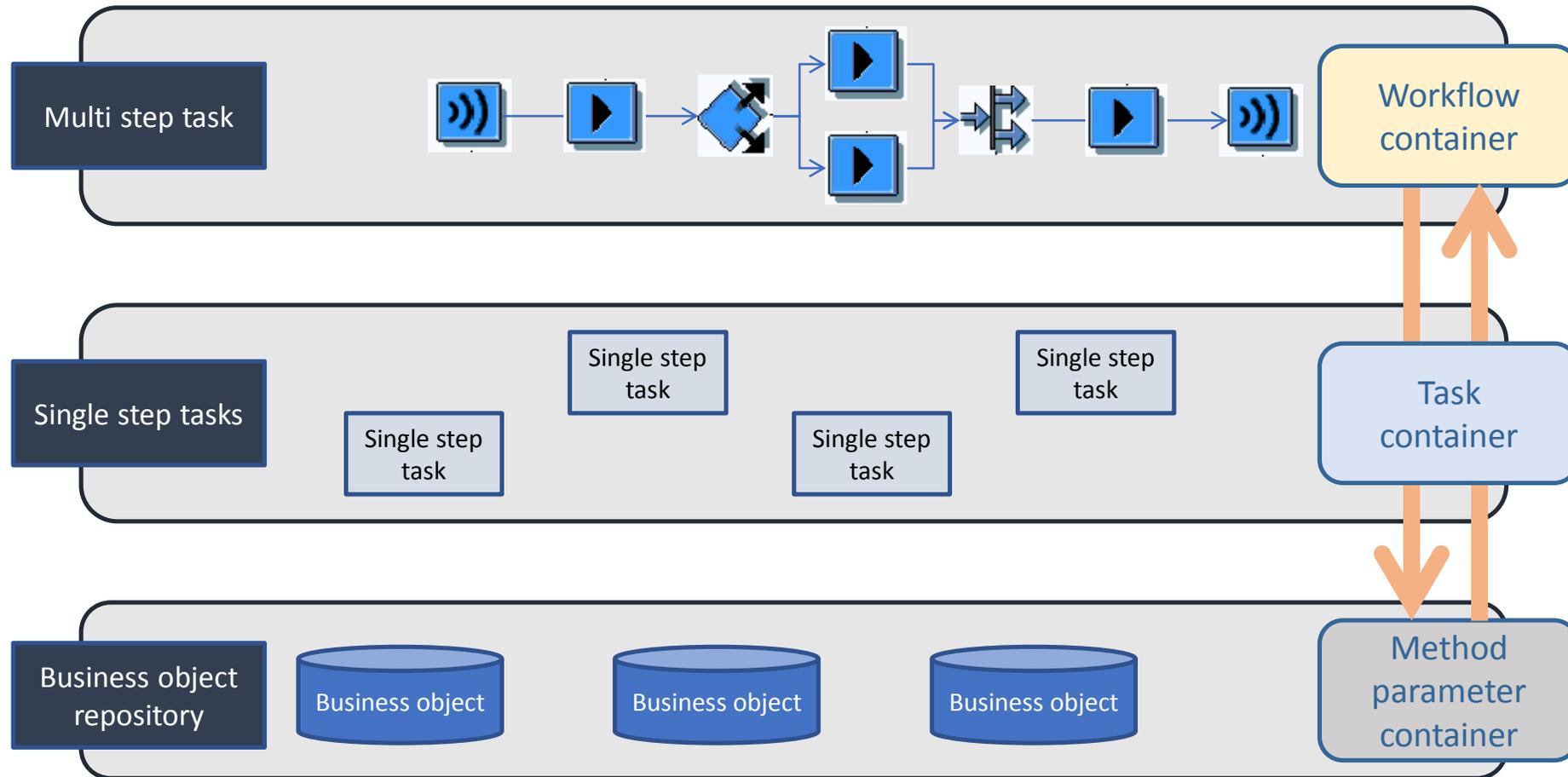


[This Photo](#) by Unknown Author is licensed under [CC BY-SA](#)

# Container types



# Workflow architecture



# Agenda

1. Introduction
2. Workflow architecture
3. Container concept
4. Event concept
  - Definition
  - Event types
  - Event creation
5. Workflow administration

# Events

An **event** is an occurrence on an object in the system.

Events are published and managed by the event manager who creates the link between creator and receiver and transfers the information between the two.



[This Photo](#) by Unknown Author is licensed under [CC BY-NC](#)

# Event types

## Triggering event

- Triggers a single step or a multi step task
- Has to be defined in task
- Has to be activated in the event link table

## Terminating event

- Terminates a single step task
- Has to be defined in task

# Event creation

- Events are published by the “Event Creator” and transferred to the “Event receiver” by the “Event manager”
- Events can be created in multiple ways
  - Program code with function module (SWE\_EVENT\_CREATE)
  - Link to change management
  - Link to status management
  - Link to message system
  - Link to HR tables (administration of info-types)
- Event has to be defined for a business object in the SAP object repository

# Container & Events - Demo

Step 1: Create workflow container element BUS2081

Step 2: Integrate with task container

Step 3: Integrate with work item description and mail

Step 4: Link event BUS2081.CREATED to workflow template

Step 5: Test workflow

# Agenda

1. Introduction
2. Workflow architecture
3. Container concept
4. Event concept
5. Workflow administration

# Workflow administration

- Access workflow administration by using transaction SWLD, then go to Administration->Workflow runtime
- Important administrative transactions
  - SWI2\_ADM1: Work items without agents
  - SWI2\_DIAG: Diagnosis of workflow errors
  - SWPR: Restart workflow after error
  - SWU3: Basic workflow customizing
  - SWEQADM: Manage event queue
  - RSWWERRE: Execute workflow error monitoring
  - SWW\_SARA: Archive work items

ASUG82919 - 'Tips and Tricks for Every Workflow Developer or Administrator for SAP Software'  
Wednesday, May 8, 2019,  
from 2:00 PM to 2:40 PM  
Room 310H

# Take the Session Survey.

We want to hear from you! Be sure to complete the session evaluation on the SAPPHIRE NOW and ASUG Annual Conference mobile app.



# Presentation Materials

Access the slides from 2019 ASUG Annual Conference here:

<http://info.asug.com/2019-ac-slides>

# Q&A

For questions after this session, contact us at [martinm@maxecm.com](mailto:martinm@maxecm.com) and [skeohan@mit.edu](mailto:skeohan@mit.edu)

# Let's Be Social.

Stay connected. Share your SAP experiences anytime, anywhere.

Join the ASUG conversation on social media: **@ASUG365 #ASUG**

