z/OSMF Workflows

20.10.2021, 93. Schweizer z/OS GSE Expertenforum

Redelf Janßen
IBM Z Client Technical Specialist
redelf.janssen@de.ibm.com
+49-171-5538587

IBM

z/OSMF

Desktop style user interface
- Built-in support for data set filter and display, browse, edit etc.
- Built-in support for Unix file filter and display, browse, edit etc.
- Built-in support for JOB filter and display, browse, cancel, purge, etc

Workflow Support – guided actions
- A series of steps to accomplish a task and a tool to track each steps status
- Can involve one person or many
- Workflow authors decide on style and technical approach
- Can be: Manual instructions, Semi-automated instructions, Fully automated actions
- Consist of Jobs, Shell scripts, REXX execs, REST calls, file updates etc.
- Optionally retains a log of what has been done
- Useful for Installation, Service, Upgrade, or any configuration actions

Task focused activities
- Sysplex configuration, Workload Manager Policy setup, Network Configuration, Security Configuration validation, Sending doc into IBM, Performance and status monitoring, etc.

Usability & Skills

z/OSMF Desktop – Replaces Tab UI in z/OS 2.5

- Customer grouping of items in folders, such as data sets via PH24527 (CD)
- Search, Browse, Edit files and data sets via PH16076 (CD)
- Submit, query, browse jobs – Job Output Task via PH16076 (CD)
- Syntax highlighting, user created links, improved performance PH24527 (CD)

What is a z/OSMF Workflow?

The z/OSMF Workflow is a framework that supports **z/OS System Programmers** (Workflow provider) to define a guided flow (workflow) through steps **to accomplish a system management or configuration task**.

The z/OSMF Workflow is useful to:
- Assist people unfamiliar with how to perform a given task, or a task that they perform rarely
- Ensure that all tasks are performed in the right order and only when their dependencies have been met
- Ensure that all steps are completed
    - Even if many of the tasks have been delegated to a number of different colleagues
- Monitor and track progress toward the completion of the task
- Provide a history (audit trail) of the steps performed for a task
- Perform the same tasks on multiple systems
    - Enabling a function (e.g. zEDC)
    - Upgrading a new release of software (e.g., z/OS)

The z/OSMF Workflow also provides RESTful APIs that allow user to run workflows programmatically

What is a z/OSMF Workflow?

- **Workflow Editor task**
  A z/OSMF plug-in that provides a UI to create workflow definition files.

- ***Workflow definition file***
  An XML file that defines steps, variables and metadata. These are used to create workflow instances.

- **Workflow instance**
  A single entity generated from a workflow definition file inside of the Workflows Task.

- **Workflows task**
  A z/OSMF plugin that creates and manages workflow instances from workflow definitions.

- ***Workflows properties file***
  An optional text file used to define variable values when creating workflow instances.

What is a z/OSMF Workflow?

Workflow Step
- Contains executable code or instructions
- Steps can be assigned to different users when there is a separation of responsibility
- Conditional logic can decide if step should be executed or skipped
- Steps execution can be dependent on completion of previous steps
- Steps can be automated (when all steps can run under single user) or performed manually
- Automated steps can run in parallel when there is no dependency

Workflow step types
- The z/OSMF Workflow can execute variety of step types:
  - Instruction – Provides instructions to user about how to perform specific task
  - Invoke JCL via submitting job
  - Invoke Shell script via submitting job
  - Invoke REXX Exec via submitting job
  - Invoke REST APIs (both internal to z/OSMF and external APIs)
  - Invoke other workflows
- JCL, shell scripts or REXX exec can be coded inline or can be in z/OS dataset or files
- Output from jobs (JCL, shell scripts, REXX exec) or REST APIs are recorded in workflow instance

Why should I learn about workflow?

- Workflow helps you to persist, streamline and orchestrate z/OS tasks. You can take advantage of workflow by using others' shipped workflow or creating your own workflow definition.

- Several z/OS key solutions are now using workflow to streamline and automate z/OS tasks. E.g.,
- z/OS upgrade workflow has replaced z/OS Upgrade Guide book
- zCX (z/OS Container Extension) is using workflow to provision zCX image
- Many software on z/OS platform are now using Workflow for their configuration (invoked by z/OSMF Software Management task)
- Cloud Provisioning and Management for z/OS is using workflows to provision/deprovision CICS, MQ, Db2, WLP, IMS, z/OS Connect

Value Proposition

Reduce the complexity of their organization's z/OS management processes

Zach – IBM Z System Programmer

+

Attract and retain new system programmers without overwhelming them with underlying process and platform complexities
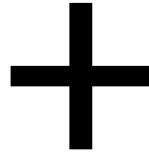
Christina – Early tenure System Programmer

Sample Workflow definition Files

- The Workflow definition is stored in XML format. z/OSMF provides several sample workflow files which could be used as a start.

Remote Site: /usr/lpp/zosmf/samples/				
Filename △	Filesize	Date	Time	Permissions
workflow_sample_automation.xml	8203	06/22/20...	05:54	-rw-r--r--
workflow_sample_automation_property.txt	153	06/22/20...	05:54	-rw-r--r--
workflow_sample_basic.xml	3748	06/22/20...	05:54	-rw-r--r--
workflow_sample_bundle0.txt	731	06/22/20...	05:54	-rw-r--r--
workflow_sample_calledwfBasic.xml	2430	06/22/20...	05:54	-rw-r--r--
workflow_sample_calledwfMD5.xml	3268	06/22/20...	05:54	-rw-r--r--
workflow_sample_calledwfVarMapping1.xml	3760	06/22/20...	05:54	-rw-r--r--
workflow_sample_calledwfVarMapping2.xml	2783	06/22/20...	05:54	-rw-r--r--
workflow_sample_condition.xml	13562	06/22/20...	05:54	-rw-r--r--
workflow_sample_file_template0.xml	740	06/22/20...	05:54	-rw-r--r--
workflow_sample_fragment0.xml	2530	06/22/20...	05:54	-rw-r--r--
workflow_sample_fragment1.xml	2296	06/22/20...	05:54	-rw-r--r--
workflow_sample_include_external.xml	5264	06/22/20	05:54	-rw-r--r--

Edit Your First Workflow – Flat file text editor

```xml
▼<workflow>
  ▼<workflowInfo>
      <workflowID>DB2V11CMMigrationWorkflow</workflowID>
      <workflowDescription>Standard Procedure to migrate DB2 V10 to V11 CM</workflowDescription>
      <workflowVersion>0.5</workflowVersion>
      <vendor>IBM</vendor>
    ▼<Configuration>
        <productID>5740XYR00</productID>
        <productName>DB2 for z/OS</productName>
        <productVersion>B10</productVersion>
      </Configuration>
    </workflowInfo>
  ▶<variable name="B" scope="instance">...</variable>
  ▼<variable name="DSNSAMP" scope="instance">
      <label>DSNSAMP</label>
      <abstract>SAMPLE LIBRARY - prefix.SDSNSAMP</abstract>
    ▼<description>
        The value of the SAMPLE LIBRARY - prefix.SDSNSAMP variable specifies the name of the JCL library w
        installation jobs.
      </description>
      <category>DB2</category>
    ▼<string>
        <maxLength>40</maxLength>
      </string>
    </variable>
  ▼<variable name="INSICPRE" scope="instance">
      <label>INSICPRE</label>
      <abstract>INSTALL IC PREFIX</abstract>
    ▼<description>
        The INSTALL IC PREFIX field specifies the prefix for image copy data sets that are created by DB2
      </description>
      <category>DB2</category>
    ▼<string>
```

Edit Your First Workflow – Workflow Editor UI

- z/OSMF "Workflow editor" allows user to create or edit workflow definition from GUI

Workflow Demo

Edit Your First Workflow – Workflow Editor UI

- Define metadata for your workflow

Edit Your First Workflow – Workflow Editor UI

- Create step for your workflow (1/2)

Workflow Editor

File Path: /u/zoscloud-beta/zosmf/samples/workflow_sample_automation.xml

| Metadata | Steps | Variables | Input Properties |

Actions ▼   Create Step   Search

Copy
Create Step   ▶   Create a Step Above...
Move Step   ▶   Create a Step Below...   omatable step
Delete   Create a New Substep...
Expand
Collapse   tomatedStep   Non-automated Step
Configure Columns...   EmptyJCL   Submit an empty JCL job
Clear Search
Expand All
Collapse All

Save   Save As...   Cancel

Create a New Step

◉ Leaf Step   ○ Parent Step

* Name: * Weight (1-1000):
Expstep   20

* Title: Skills:
This is an example step   zOS JCL skill

* Description: * Step Type:
Demonstrate how to create a new step   Inline Template ▼

☐ Optional Step
☐ Auto-Enable
☐ Allow manual marking of step failure

OK   Cancel   Help

Edit Your First Workflow – Workflow Editor UI

- Create step for your workflow (2/2)

Edit Your First Workflow – Workflow Editor UI

- Save your workflow definition
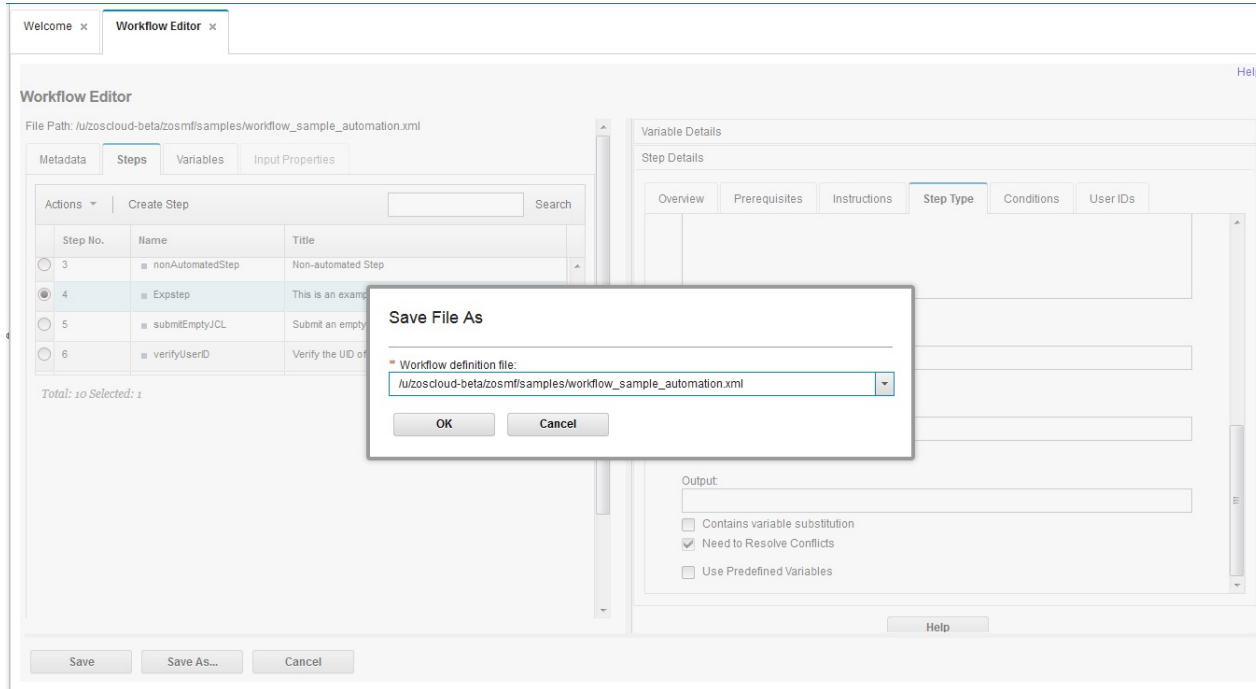
Quick Overview of Workflow UI

- Create workflow instance

Quick Overview of Workflow UI

- A typical workflow instance

Collaborating with various personas

Collaborating with various personas

Note: Notification can be sent through email

Collaborating with various personas

z/OSMF Workflow Flow Control

z/OSMF Workflow Automation (1)

z/OSMF Workflow Automation (2)

z/OSMF Workflow Operation Traces (1)

z/OSMF Workflow Operation Traces (2)

Edit Your Workflow instance

Workflow service – API list

- Workflow RESTful service allows user to create, start and manage workflow in z/OS through programmatic way instead of having to operate in z/OSMF UI.

- Workflow RESTful service provides below operations (APIs):

Operation	Method
Create a workflow	POST
Get the properties of a workflow	GET
Start a workflow	PUT
Lists the workflows for a system or sysplex	GET
Delete a workflow	DELETE
Cancel workflow	PUT
Retrieve a workflow definition	GET
List the archived workflows for a system	GET
Get the properties of an archived workflow	GET
Delete an archived workflow	DELETE

z/OSMF Workflow Best Practices

Prepare workflow using workflow editor

– User won't require knowledge of XML and workflow XML schema

– Workflow editor performs syntax checks, so workflow can be run without any syntax errors

– Intuitive online help that guides user to prepare workflow

– Provides various samples in tool-box that can be imported into the workflow and modified as needed

If you are planning to use workflow editor, don't create workflow fragment files using XML editors

– Workflow editor does not preserve the fragments

If workflow needs to be run multiple times, do not specify scope="system".

– <workflowID scope="system">test</workflowID>

z/OSMF Workflow Best Practices

Workflow editor supports workflow and file templates from PDS (or Sequential) dataset as well as zFS file system files

Workflow editor makes it easy to copy or move steps

Property file can be prepared in workflow editor along with defining workflow

Workflow editor provides toolbox with samples. You can import steps from the toolbox and customize based on your need

Workflow editor also provides 'export' functions to share steps across multiple workflows

Workflow editor also provides plain text editor to view/modify workflow xml definition

– Simplifies fixing XML issues

– Provides capability to validate XML definition

z/OSMF Workflow Best Practices

What is currently not covered by workflow editor?

– Workflow editor does not validate that variables used in workflows are defined

- Create workflow variables before using them in step.

– Workflow editor does not validate that property value is defined for variable identified as "required" for the step

- Set the property value before identifying variable as 'required' for the step.

– Workflow editor can not validate JCL, shell script or REXX syntax error

z/OSMF Workflow Best Practices

Use System variables when same variable (with same value) is used across multiple workflows e.g. SYSTEM_HOST_IP_ADDR

System variables can be defined using zOSMF Settings -> Systems user interface task(or using REST APIs)

These variables can be accessed in any workflow without defining them in a workflow

– Workflow running on a specific system, can access variable value associated with that specific system.

– ${_sys-varName} (e.g. ${_sys-SYSTEM_HOST_IP_ADDR}) convention can be used in JCL, REXX or shell script to reference a system variable

System variables can be exported and imported to create same variables across multiple systems in the sysplex

z/OSMF Workflow Best Practices

Use setVariable to dynamically set variable value during workflow execution

– *<setVariable name="${instance-INSDSPRE}" scope="instance" substitution="true">*
  ${instance-INSDSPRE}.${_workflow-softwareServiceInstanceName}</setVariable>

This is a simpler way then creating output file to set variable value.

setVariable sets any WF variable at the completion of some step (could be instruction step).

– Saves 3 steps that you would need otherwise to set variable value

z/OSMF Workflow Best Practices

Variable value can use substitution in property file

_IZU_VARIABLE_SUBSTITUTION_ON &
_IZU_VARIABLE_SUBSTITUTION_OFF tags in property file

For example,

_IZU_VARIABLE_SUBSTITUTION_ON

PATH_ROOT=/usr
PATH_LIB=${PATH_ROOT}/lpp/zosmf ###-> /usr/lpp/zosmf

_IZU_VARIABLE_SUBSTITUTION_OFF

z/OSMF Workflow Best Practices

Variable value can be set dynamically by JCL, shell scripts via SYSOUT DD

Usage: A workflow author who wishes to use the sysoutDD feature will need to set the "sysoutDD" to true as part of a step's <output> element:

*<output needResolveConflicts="true" **sysoutDD="true"** substitution="false">stepName.ddName2</output>*

Example script:

```
<template>

    <inlineTemplate substitution="false">

        host=$(hostname)

        echo "var = $host"</inlineTemplate>

    <submitAs maxRc="0">shell-JCL</submitAs>

    <output needResolveConflicts="false" sysoutDD="true"
        substitution="false">STDOUT</output>

</template>
```

Usability & Skills

IBM z/OS Management Services Catalog – Planned new plugin

z/OS Management Services Catalog in z/OSMF plans to leverage the power of z/OSMF workflows to enable system programmers to run services that help complete z/OS management tasks faster and with fewer errors.

- Experienced z/OS system programmers are planned to be able to create a catalog of customized services, each written with unique institutional knowledge, protocols, and processes.
- These services can then be run by less experienced colleagues.
- IBM plans to provide an initial set of services to help z/OS system programmers of all skill levels get started, demonstrate accepted practices, and simplify information sharing.
  - Such as grow a ZFS.

Capabilities:
- Planned are a powerful graphical interface for creating new services, editing IBM-provided services, and updating existing services
- Step-by-step guidance for completing z/OS management tasks
- History of all services performed on a system is planned

You can learn more and watch for future developments on the release at the z/OS Management Services Catalog content page.

Ansible drives Workflow – What's Ansible

Red Hat Ansible Automation Platform

Flexibility

Bring disparate IT into a coherent whole using a market leading open solution backed with enterprise support

Interact directly with z/OS resources or integrate with existing platform tools

Consistency

Integrate z/OS into an enterprise automation strategy in a consistent way

Centralize management of your IT infrastructure

Simplicity

The certified collections codify much of the z/OS specific knowledge and complexity

Developer or system programmer can focus on their tasks and be more productive

Ansible drives Workflow – z/OSMF Ansible collection

- **IBM z/OSMF Ansible collection "ibm_zosmf", intends to provide simple and consistent experience for Ansible users to drive z/OSMF REST APIs for z/OS operations and automation.**

  - **Part of Red Hat Ansible Certified Content for IBM Z.**

  - **Available in both Ansible Galaxy and Red Hat Ansible Automation Hub:**

    https://cloud.redhat.com/ansible/automation-hub/repo/published/ibm/ibm_zosmf

    https://galaxy.ansible.com/ibm/ibm_zosmf

  - **Documentation:**
    https://ibm.github.io/z_ansible_collections_doc/ibm_zosmf/docs/ansible_content.html

- **What's available today via z/OSMF Ansible collection "ibm_zosmf"**

"ibm_zosmf" collection drives z/OSMF REST APIs start with:
- **Workflow operations**
Drive a z/OSMF workflow to complete, Delete a workflow instance, Query workflow status, etc.
- **Provision and Manage z/OS software instances** via Cloud Provisioning and Management for z/OS
Provision or deprovision a z/OS middleware/software instance, start or stop the software instance, etc.

Ansible drives Workflow – Use case

- **Workflow extends z/OS tasks Ansible can drive**
  - **Semi-automated tasks**
  - **Tasks that require to keep data on z/OS platform**

- **Ansible extends flexibility of Workflow and CP&M**
  - **Scheduled Workflow or CP&M operations through Ansible Tower**
  - **Ansible can orchestrate Workflow or CP&M services with tasks on other platforms**
  - **Ansible provides consistent front-end to manage automation cross platforms**

How can you help modernize z/OS?

z/OS System Programmers have historically built their own home grown processes to perform common system management tasks

Organizations are wanting to modernize z/OS platform and reduce the complexity of their organization's z/OS management processes
– Additional skills are needed to create z/OSMF Workflows

More experienced z/OS System Programmers need to share best practices and build workflow patterns
– Workflows can be fully automated and driven through modern tools like z/OSMF UI or REST APIs

Z Open Repository Of Workflows (zorow)

Provides a repository for z/OS systems programmers and product vendors to contribute and share z/OSMF workflows
- All workflows made available under an Apache 2 license
- IBM and Rabobank have made initial contributions of z/OSMF workflows to this community

Community is led by both vendors along with customers, and open to anyone to participate.
- Vendor-neutral open source governance established with the guidance of the Open Mainframe Project.

How to get involved in Zorow

Sign up to the Zorow email lists

User List:
https://lists.openmainframeproject.org/g/zorow-user/
Developer List:
https://lists.openmainframeproject.org/g/zorow-dev/

Find other users and developers on Slack

https://slack.openmainframeproject.org
Channel: #zorow

View the code and contribute workflows

https://github.com/openmainframeproject/zorow
Contribution Guidelines at
https://github.com/openmainframeproject/zorow/blob/master/CONTRIBUTING.md

Thank You!

THANK YOU!

IBM