My Enablement Title

The guide to learning faster

Abstract: Aenean sed leo lobortis, mollis sapien id, rutrum ante. Sed vitae pellentesque lorem. Donec dignissim venenatis dolor vulputate scelerisque. Curabitur dapibus mattis dolor, vel iaculis nulla tempus id.

Author: Chuck Grant

Company: My enablement organization

Contact Info: myemail@email.com

Last updated: 12/15/2021

Course Contents

Course Contents

Module 1: First Module Title

This is the introduction to this module. After the introduction, there is a module table of contents that provides quick links to all topics and exercises in this module. If the student came to this module on accident, there is a secondary link "..back to the course table of contents" that lets the student go back to the global TOC.

Objectives

- Lorem Ipsum
- Lorem Ipsum

Module Contents

Return to Course Contents

Maintenance Tasks

Maintenance Task are processes that run on a schedule in order to optimize the repository. With AEM as a Cloud Service, the need for customers to configure the operational properties of maintenance tasks is minimal. Customers can focus their resources on application-level concerns, leaving the infrastructure operations to Adobe. Learn More

TASK	OWNER	HOW TO CONFIGURE (OPTIONAL)
Datastore garbage collection	Adobe	N/A - fully Adobe owned
Version Purge	Adobe	Window owned by Adobe and OSGi configuration owned by customer
Audit Log Purge	Adobe	Window owned by Adobe and OSGi configuration owned by customer
Ad-hoc Task Purge	Customer	Maintenance Window and OSGi Configuration
Workflow Purge	Customer	Maintenance Window and OSGi Configuration
Project Purge	Customer	Maintenance Window and OSGi Configuration

Configure Tasks

OSGi confgurations allow for a maintenance task to be configured specific to the business requirements for purging content.

- Version Purge Configuration
- Audit Log Configuration

Maintenance Windows

Workflow Purge, Ad-hoc Task Purge and Project Purge Maintenance tasks to be executed during the daily, weekly, or monthly maintenance windows. These configurations should be added directly in source control.

Daily Configuration Example

```
<?xml version="1.0" encoding="UTF-8"?>
<jcr:root xmlns:sling="http://sling.apache.org/jcr/sling/1.0"
  xmlns:jcr="http://www.jcp.org/jcr/1.0"
  jcr:primaryType="sling:Folder"
  sling:configCollectionInherit="true"
  sling:configPropertyInherit="true"
  windowSchedule="daily"
  windowStartTime="03:00"
  windowEndTime="05:00"
 />
```

The Maintenance UI can be used to easily configure the windows on a local AEM instance and them syncronzied back into the AEM project to be checked into source control. Alternatively, you can add and edit the xml files directly.

Exercise 1: Create a Daily Purge Task

Scenario: Your company has started using AEM projects to help support new products being added to your website. To maintain the performance in AEM, you need to purge the unused projects that users might have abandoned or are no longer needed. Because projects are used so often, purging needs to be performed daily.

Prerequisites:

- AEM running

1. Go to Tools > Operations > Maintenance
2. Click on the **Daily Maintenance Window** card.
3. Select the **+ Add** button

4. From the dropdown, select **Project Purge** and click **Save**
5. The bottom of the Project Purge card specifies when it will run next. For testing purposes, you can manually start the task immeditately.
6. Hover over the card and click the **play** button. Notice how the purge task fails. This is because it has not been configured yet.

Congratulations! You have successfully created a daily purge task within a maintenance window.

Workflow Purge Task

Minimizing the number of workflow instances increases the performance of the workflow engine, so you can regularly purge completed or running workflow instances from the repository.

Configure **Adobe Granite Workflow Purge Configuration** to purge workflow instances according their age and status. You can also purge workflow instances of all models or of a specific model .

You can also create multiple configurations of the service to purge workflow instances that satisfy different criteria. For example, create a configuration that purges the instances of a particular workflow model when they are running for much longer than the expected time. Create another configuration that purges all completed workflows after a certain number of days to minimize the size of the repository.

Exercise 2: Configure a Purge Task

Scenario: In AEM, runtime workflow models are stored in `/var/workflows/models/`. In this exercise you will configure a workflow purge task for the Adobe provide Publish Example workflow to purge completed workflows that are older than 30 days.

Prerequisites:

- AEM running
- Eclipse with a Maven Project imported

Task 1: Create OSGi Config

1. In AEM, navigate to **Tools > Operations > Web Console**
2. In the configuration console find the **Adobe Granite Workflow Purge Configuration** factory

> ***Hint***: *Use the browser find feature (crtl+f) to quickly find the configuration*

> ***Note***: *a factory configuration means you can have multiple instances of the configuration*

3. Click on the the configuration factory and observe the different options of the configuration:

OPTION NAME	DESCRIPTION
scheduledpurge.name	Name that shows up in the logs
scheduledpurge.workflowStatus	COMPLETED/RUNNING
scheduledpurge.modelIds	List of models to purge
scheduledpurge.daysold	Age of models to purge

> ***Note***: *The Factory Persistance Identity (PID) will also be needed to create the json config file.*

4. Keep this browser window open to observe the configuration file against these configurations.
5. Open your project in Eclipse.
6. Since configurations are considered immutable content, navigate to **devops.ui.apps > src/main/content/jcr_root > apps > training**

7. Right click on **config.author** and select **New > File**

8. Enter **com.adobe.granite.workflow.purge.Scheduler~training.cfg.json** as the file name and click **Finish.**
9. The file will open in the editor. In the **Exercise Files**, navigate to **training-files/Maintenanc**e and copy the contents of **com.adobe.granite.workflow.purge.Scheduler~training.cfg.json**

> ***Note**: This purge task has **daysold** set to 0 to show the purge task successfully running in the next task.*

10. Verify the file contents verify against the configuration in the web console in the browser. **Save** your changes in the Eclipse editor.

Task 2: Install and verify the purge configuration

1. In the Project Explorer, right-click **devops-training** and select **Run As > Maven Build**. The build starts.
2. Verify that the build completed successfully
3. In AEM, navigate to **Tools > Operations > Web Console**

4. In the configuration console find the **Adobe Granite Workflow Purge Configuration** factory. There should now be an instance of the factory called **com.adobe.granite.workflow.purge.Scheduler~training**. Open it.
5. Notice the configurations you added to the JSON file are now properly configured.
6. To observe the purge task successfully running, you will need to complete a Publish Example workflow. Navigate to the **Sites Console**
7. Select **WKND Site > Create > Workflow**
    a. Workflow model = Publish Example
    b. Workflow Title = Publish WKND
    c. Select **Next**
    d. Select **Create**
8. The workflow has successfully started. To complete the workflow, Go to the **Inbox** in the top right corner (bell icon) and click **View all**.
9. Select the work item, **Validate Content** and click **Complete**.
10. Click **Ok** in the popup to complete the work item. The page publishes and the workflow is complete.
11. Verify the Completed workflow by going to **Tools > Workflow > Archive**. Observe the completed workflow.
12. You can manually kick off maintenance tasks outside the normal maintenance window for testing purposes. Navigate to **Tools > Operations > Maintenance**
13. Click on **Weekly Maintenance Window**. On the **Workflow Purge** card, click the **play** button.
14. You will see the card go from yellow to green when the task is complete. To verify the purge successfully occurred, navigate back to **Tools > Workflow > Archive** and notice the archived Publish Example workflow instance no longer exists.

Congratulations! You have successfully configured a purge task and tested it successfully.

Exercise 3: Create a Monthy Window

Scenario: Your company has started using ad-hoc tasks and you want to make sure older tasks are properly removed from the system. This is not a widely used feature and purging old tasks is not top priority. You want to create a monthly maintenance window to purge any tasks out of compliance with the out of the box ad-hoc task purge configuration. This configuration purges completed tasks older than 30 days and active tasks older than 90 days. See `com.adobe.granite.taskmanagement.impl.purge.TaskPurgeMaintenanceTask` in the web console.

Prerequisites:

- AEM running

1. In AEM, Navigate to **Tools > General > CRXDE Lite**
2. In the JCR tree on the left, navigate to **/libs/settings/granite/operations/maintenance**
3. Copy the **granite_weekly** node and paste it under **/conf/global/settings/granite/operations/maintenance**
4. Save.
5. Under **/conf/global/settings/granite/operations/maintenance** delete the 3 nodes under **granite_weekly** and **Save**.
6. Rename **granite_weekly** to **granite_monthly** and **Save**
7. Select **granite_monthly** and in the properties pane on the bottom right, update the **windowSchedule** to **monthy** and **Save**.
8. At this point we can use the Maintenance UI to complete the window. In AEM, navigate to **Tools > Operations > Maintenance**
9. Hover over **Monthly Maintenance Window** and click the **Configure gears.**
10. Change the start and end days to **Friday** and **Save**
11. Click on the **Monthly Maintenance Window** to open it.
12. Click the **Add** icon.
13. Select **Purge of ad-hoc tasks** and **Save**.

Congratulations! You have successfully added the ad-hoc purge task to a monthly maintenance window.

Module 2: Second Module Title

This is the introduction to this module. After the introduction, there is a module table of contents that provides quick links to all topics and exercises in this module. If the student came to this module on accident, there is a secondary link "..back to the course table of contents" that lets the student go back to the global TOC.

Objectives

- Lorem Ipsum
- Lorem Ipsum

Module Contents

Return to Course Contents

Tasks

Another Task are processes that run on a schedule in order to optimize the repository. With AEM as a Cloud Service, the need for customers to configure the operational properties of maintenance tasks is minimal. Customers can focus their resources on application-level concerns, leaving the infrastructure operations to Adobe.

TASK	OWNER	HOW TO CONFIGURE (OPTIONAL)
Datastore garbage collection	Adobe	N/A - fully Adobe owned
Version Purge	Adobe	Window owned by Adobe and OSGi configuration owned by customer
Audit Log Purge	Adobe	Window owned by Adobe and OSGi configuration owned by customer
Ad-hoc Task Purge	Customer	Maintenance Window and OSGi Configuration
Workflow Purge	Customer	Maintenance Window and OSGi Configuration
Project Purge	Customer	Maintenance Window and OSGi Configuration

Configure

OSGi confgurations allow for a maintenance task to be configured specific to the business requirements for purging content.

- Version Purge Configuration
- Audit Log Configuration

Windows

Workflow Purge, Ad-hoc Task Purge and Project Purge Maintenance tasks to be executed during the daily, weekly, or monthly maintenance windows. These configurations should be added directly in source control.

Daily Configuration Example

```
<?xml version="1.0" encoding="UTF-8"?>
<jcr:root xmlns:sling="http://sling.apache.org/jcr/sling/1.0"
  xmlns:jcr="http://www.jcp.org/jcr/1.0"
  jcr:primaryType="sling:Folder"
  sling:configCollectionInherit="true"
  sling:configPropertyInherit="true"
  windowSchedule="daily"
  windowStartTime="03:00"
  windowEndTime="05:00"
 />
```

The Maintenance UI can be used to easily configure the windows on a local AEM instance and them syncronzied back into the AEM project to be checked into source control. Alternatively, you can add and edit the xml files directly.

Exercise 1: Update AEM

Scenario: Your company has started using AEM projects to help support new products being added to your website. To maintain the performance in AEM, you need to purge the unused projects that users might have abandoned or are no longer needed. Because projects are used so often, purging needs to be performed daily.

Prerequisites:

- AEM running

1. Go to Tools > Operations > Maintenance
2. Click on the **Daily Maintenance Window** card.
3. Select the **+ Add** button

4. From the dropdown, select **Project Purge** and click **Save**
5. The bottom of the Project Purge card specifies when it will run next. For testing purposes, you can manually start the task immeditately.
6. Hover over the card and click the **play** button. Notice how the purge task fails. This is because it has not been configured yet.

Congratulations! You have successfully created a daily purge task within a maintenance window.

Workflow Purge Task

Minimizing the number of workflow instances increases the performance of the workflow engine, so you can regularly purge completed or running workflow instances from the repository.

Configure **Adobe Granite Workflow Purge Configuration** to purge workflow instances according their age and status. You can also purge workflow instances of all models or of a specific model .

You can also create multiple configurations of the service to purge workflow instances that satisfy different criteria. For example, create a configuration that purges the instances of a particular workflow model when they are running for much longer than the expected time. Create another configuration that purges all completed workflows after a certain number of days to minimize the size of the repository.

Exercise 2: My Configuration Task

Scenario: In AEM, runtime workflow models are stored in `/var/workflows/models/`. In this exercise you will configure a workflow purge task for the Adobe provide Publish Example workflow to purge completed workflows that are older than 30 days.

Prerequisites:

- AEM running
- Eclipse with a Maven Project imported

Task 1: Create OSGi Config

1. In AEM, navigate to **Tools > Operations > Web Console**
2. In the configuration console find the **Adobe Granite Workflow Purge Configuration** factory

> ***Hint***: *Use the browser find feature (crtl+f) to quickly find the configuration*

> ***Note***: *a factory configuration means you can have multiple instances of the configuration*

3. Click on the the configuration factory and observe the different options of the configuration:

OPTION NAME	DESCRIPTION
scheduledpurge.name	Name that shows up in the logs
scheduledpurge.workflowStatus	COMPLETED/RUNNING
scheduledpurge.modelIds	List of models to purge
scheduledpurge.daysold	Age of models to purge

> ***Note***: *The Factory Persistance Identity (PID) will also be needed to create the json config file.*

4. Keep this browser window open to observe the configuration file against these configurations.
5. Open your project in Eclipse.
6. Since configurations are considered immutable content, navigate to **devops.ui.apps > src/main/content/jcr_root > apps > training**

7. Right click on **config.author** and select **New > File**

8. Enter **com.adobe.granite.workflow.purge.Scheduler~training.cfg.json** as the file name and click **Finish.**
9. The file will open in the editor. In the **Exercise Files**, navigate to **training-files/Maintenanc**e and copy the contents of **com.adobe.granite.workflow.purge.Scheduler~training.cfg.json**

> ***Note***: *This purge task has **daysold** set to 0 to show the purge task successfully running in the next task.*

10. Verify the file contents verify against the configuration in the web console in the browser. **Save** your changes in the Eclipse editor.

Task 2: Install and verify the purge configuration

1. In the Project Explorer, right-click **devops-training** and select **Run As > Maven Build**. The build starts.
2. Verify that the build completed successfully
3. In AEM, navigate to **Tools > Operations > Web Console**

4. In the configuration console find the **Adobe Granite Workflow Purge Configuration** factory. There should now be an instance of the factory called **com.adobe.granite.workflow.purge.Scheduler~training**. Open it.
5. Notice the configurations you added to the JSON file are now properly configured.
6. To observe the purge task successfully running, you will need to complete a Publish Example workflow. Navigate to the **Sites Console**
7. Select **WKND Site > Create > Workflow**
    a. Workflow model = Publish Example
    b. Workflow Title = Publish WKND
    c. Select **Next**
    d. Select **Create**
8. The workflow has successfully started. To complete the workflow, Go to the **Inbox** in the top right corner (bell icon) and click **View all**.
9. Select the work item, **Validate Content** and click **Complete**.
10. Click **Ok** in the popup to complete the work item. The page publishes and the workflow is complete.
11. Verify the Completed workflow by going to **Tools > Workflow > Archive**. Observe the completed workflow.
12. You can manually kick off maintenance tasks outside the normal maintenance window for testing purposes. Navigate to **Tools > Operations > Maintenance**
13. Click on **Weekly Maintenance Window**. On the **Workflow Purge** card, click the **play** button.
14. You will see the card go from yellow to green when the task is complete. To verify the purge successfully occurred, navigate back to **Tools > Workflow > Archive** and notice the archived Publish Example workflow instance no longer exists.

Congratulations! You have successfully configured a purge task and tested it successfully.