

# Additional OIC Capabilities

# Objectives

---

After completing this lesson, you should be able to:

- Define basic and advanced schedules for a scheduled orchestration
- Execute scheduled or ad hoc runs in scheduled orchestrations
- Resubmit failed scheduled orchestration run instances
- Leverage schedule parameters for multiple run instances
- Monitor integration instances and message details
- View, resubmit, or discard integration error instances
- Access diagnostic logs for troubleshooting
- Describe the OIC REST API capabilities



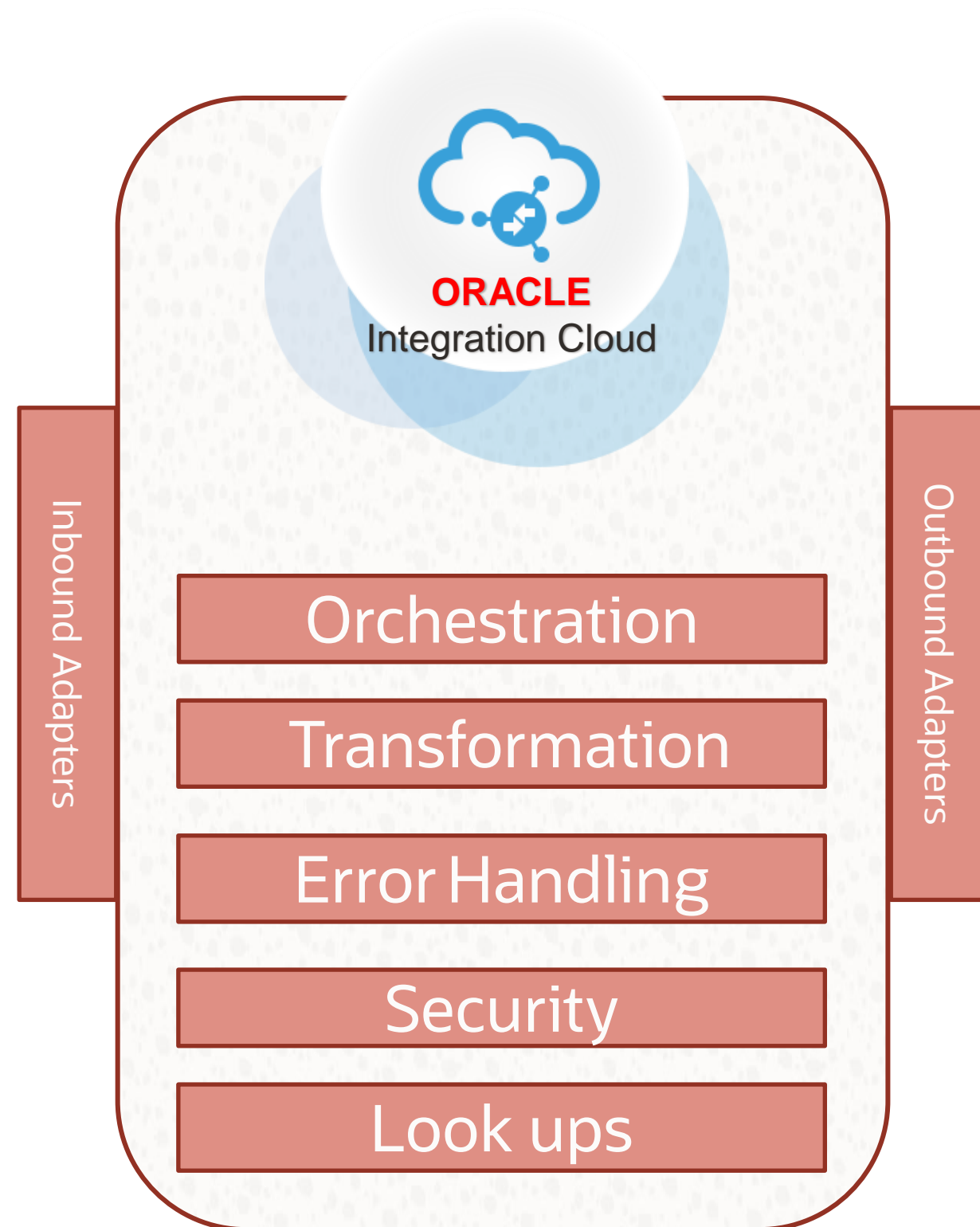


# Agenda

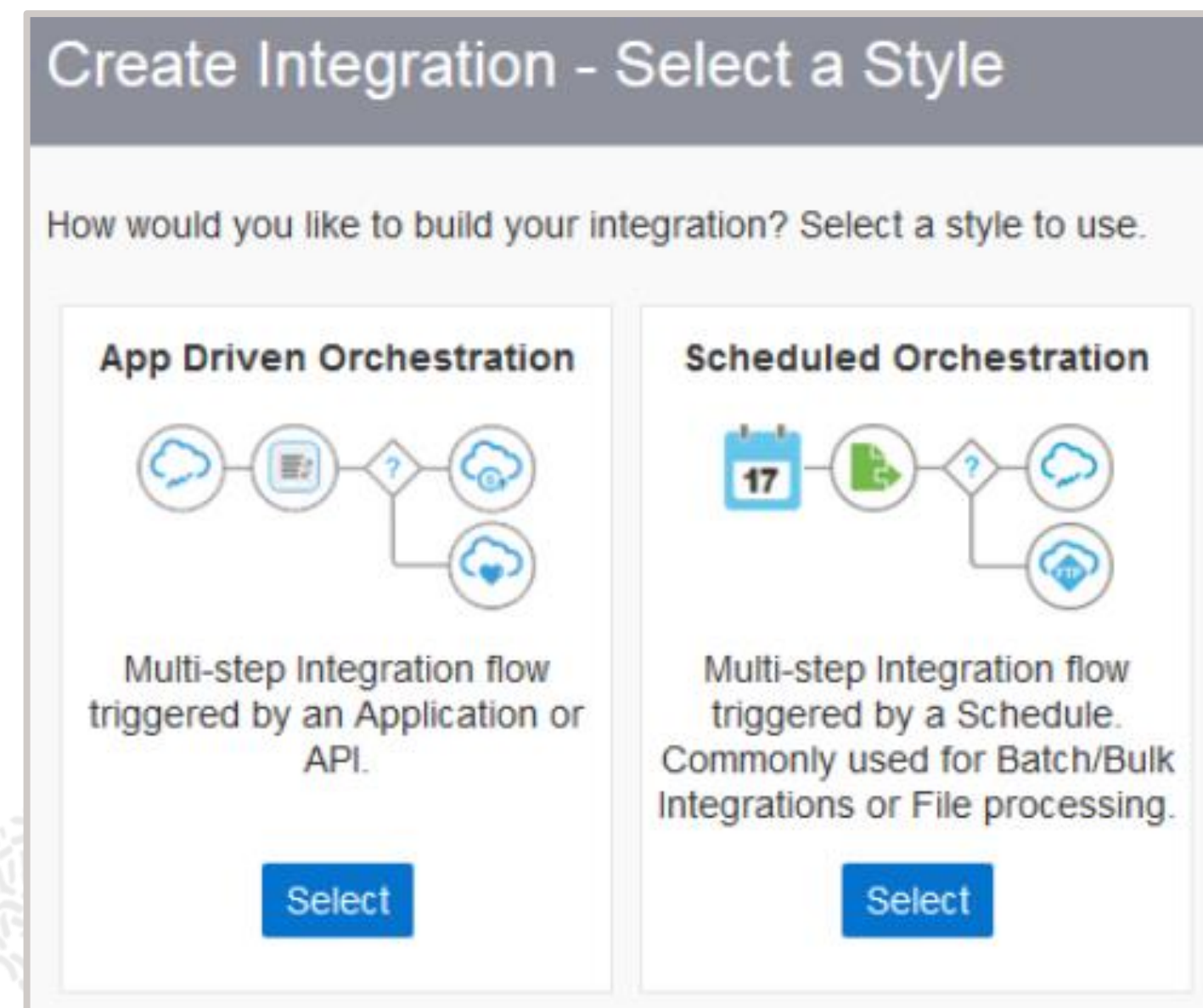
- Creating Scheduled Integrations
- Monitoring Instances and Message Tracing
- Managing Error Instances and Troubleshooting
- OIC REST API Overview



# Orchestration Styles (Review)



- **App Driven** (Interface invoked by a client application)
- **Scheduled** (Executes automatically or on-demand)

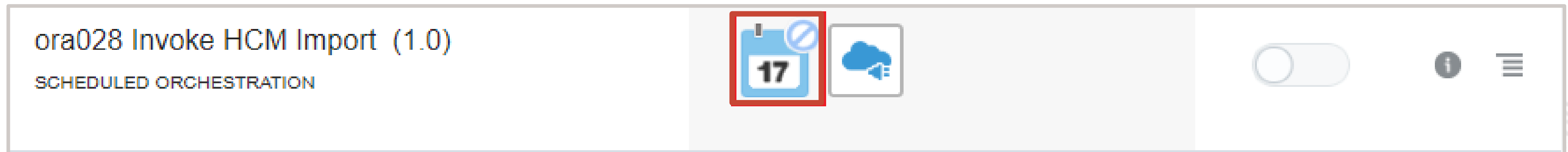
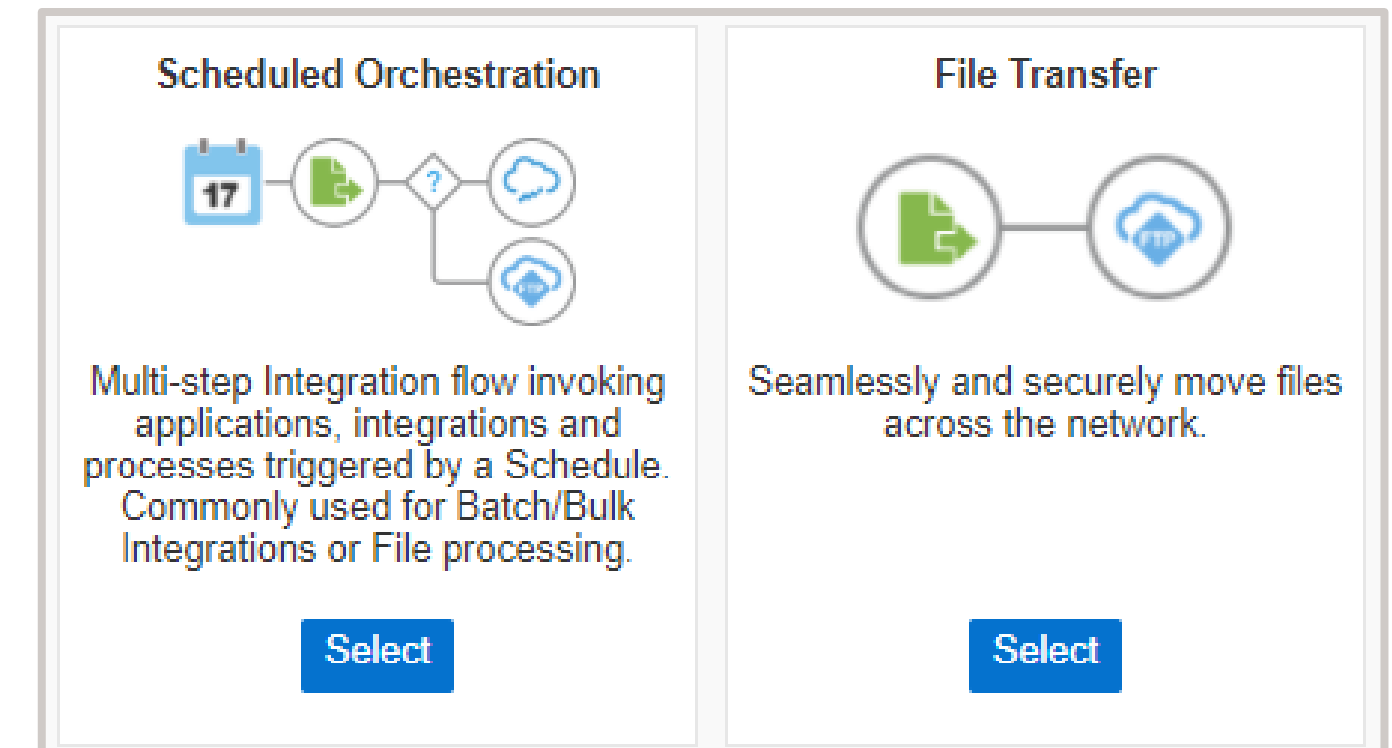


# Scheduled Integrations

Create the integration by using the **Scheduled Orchestration** or **File Transfer** style.

- *Currently, there is no difference in functionality.*

When a scheduled integration is created, a **schedule** icon is displayed for that integration on the **Integrations** page.





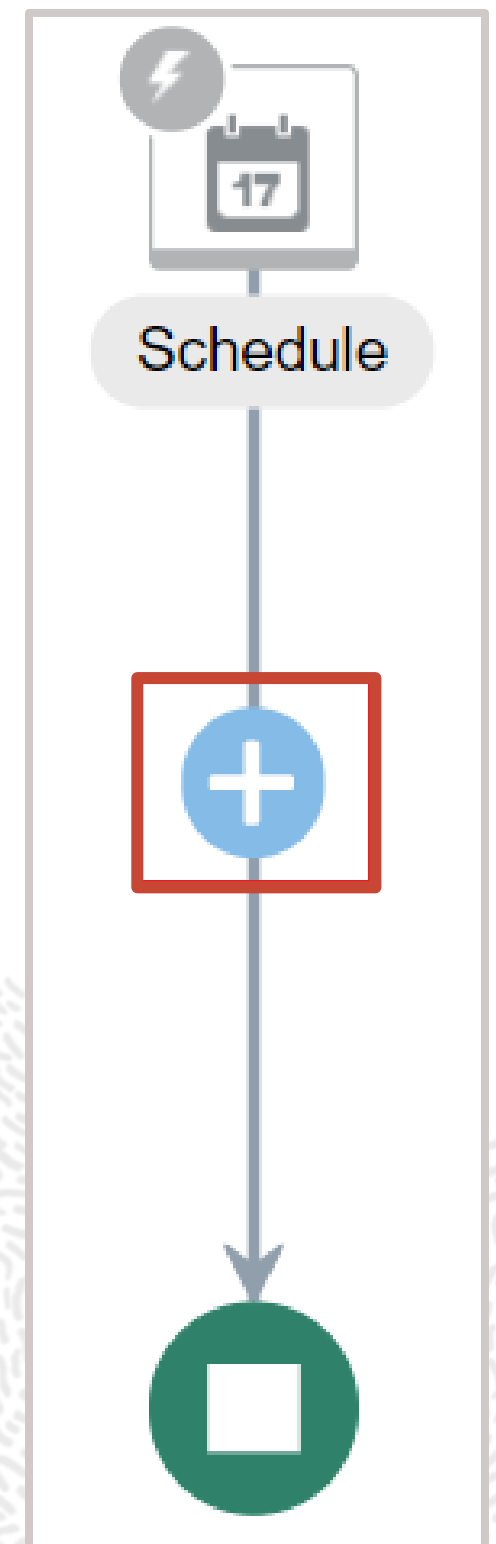
# Scheduled Orchestration Design Canvas

The design canvas is identical as when creating an App Driven Orchestration style integration.

- The same invokes and action menu is available for design.

However, unlike *App Driven Orchestrations*, the design canvas does not allow for configuring a Trigger connection.

- Instead, you are prompted to start adding invokes or actions into the flow to implement your orchestration integration use case.

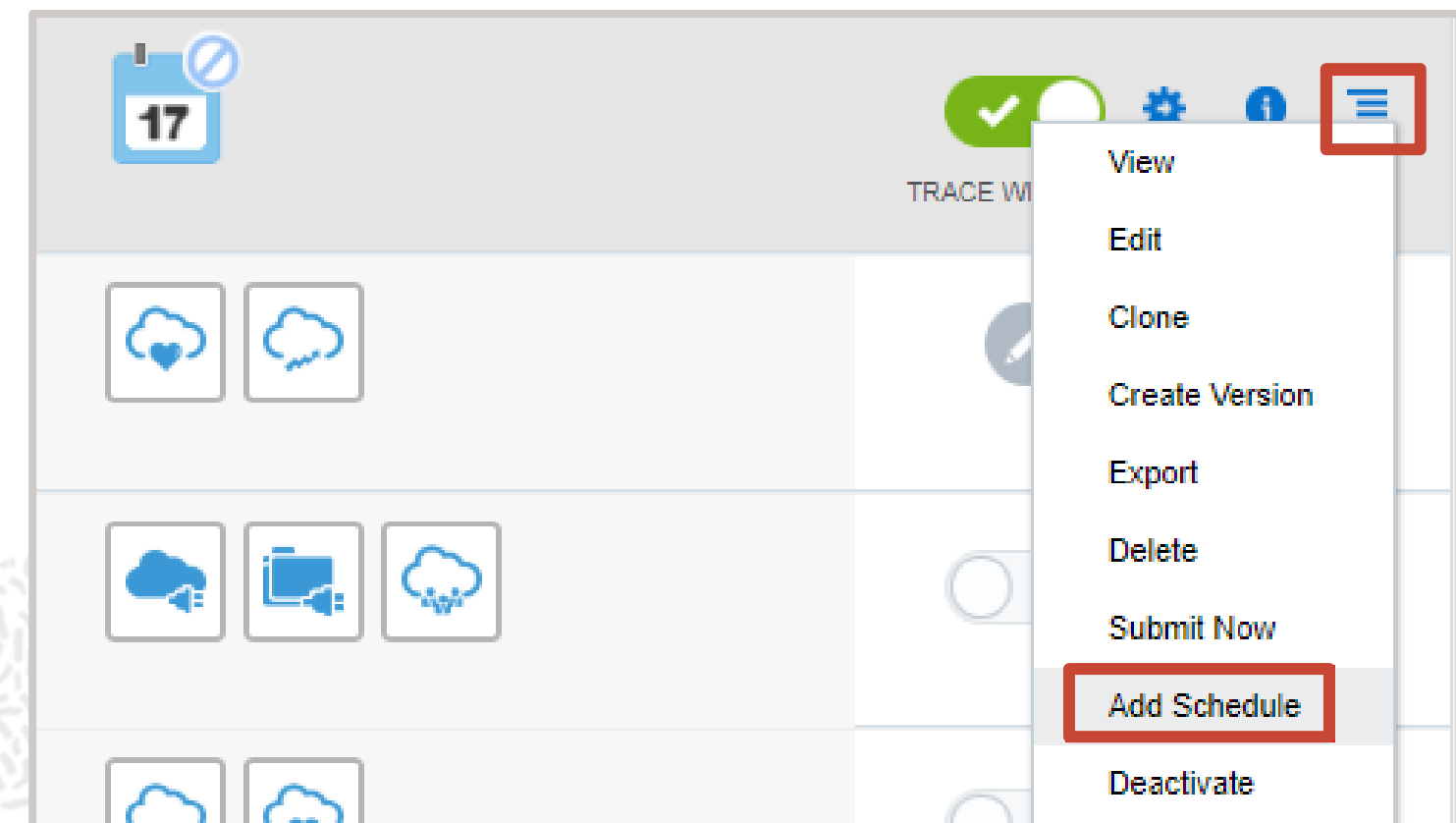
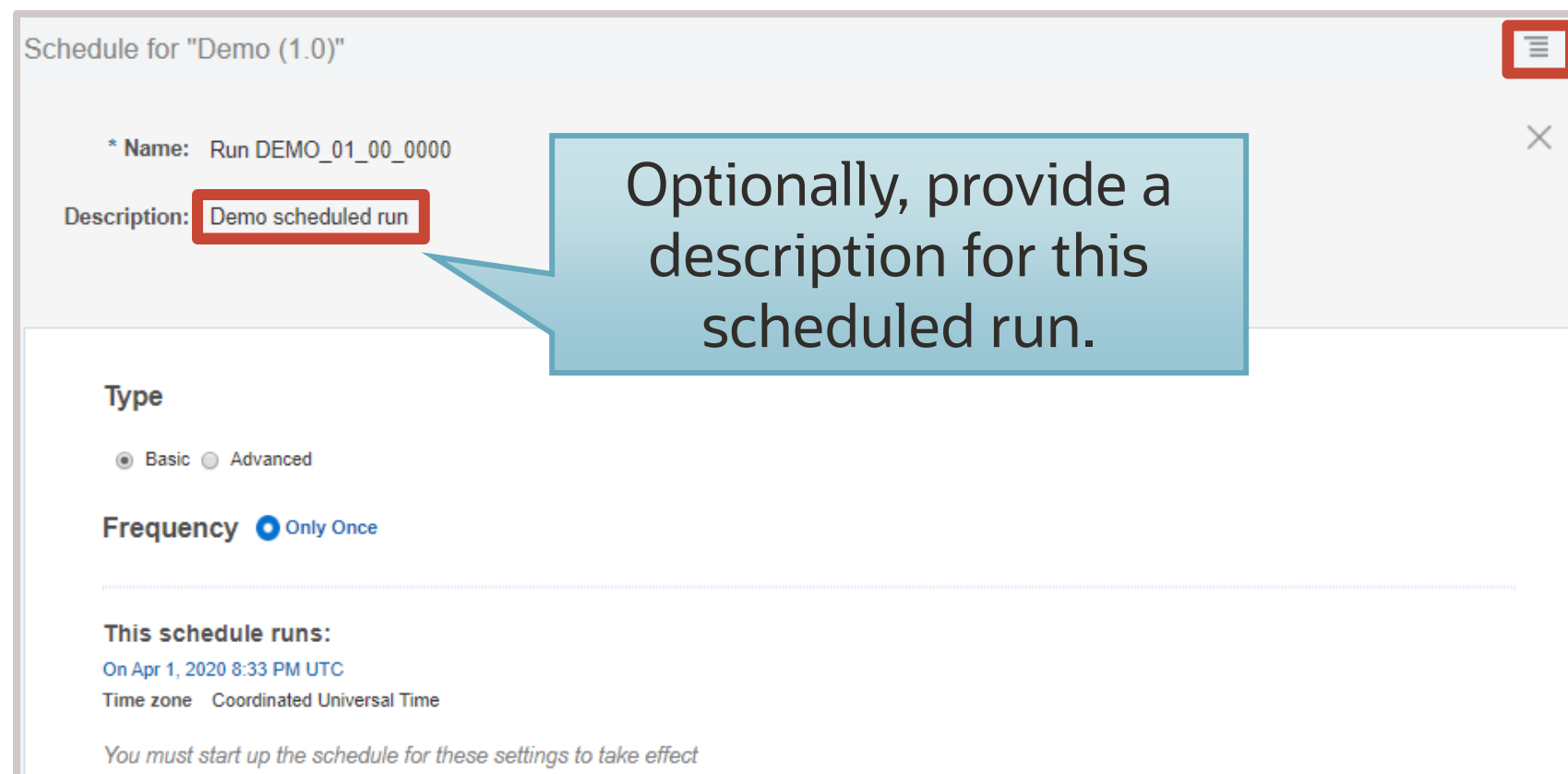
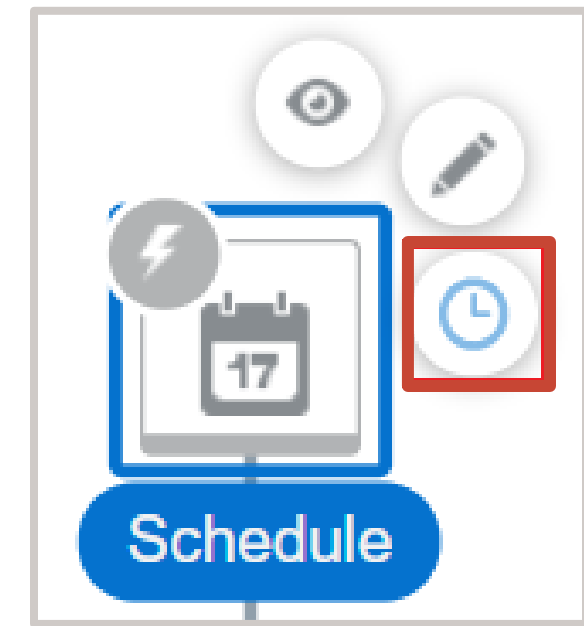


# Creating a Schedule

On the design canvas, click the Timer icon on the **Schedule** node.

- Alternatively, you can add a schedule on the Integrations page before or **after** the integration has been activated.

The Scheduler page is displayed:



# Defining a Basic Schedule

1. Choose the **Basic** type.
2. In the **Frequency** section:
  - Choose **Only Once** (*the default*), and then specify the start date and time
  - Or, define a recurring schedule based on Months, Weeks, Days, or Hours/Minutes
3. Define the expiration of this schedule.
  - **Never** (repeat indefinitely) is the default.

**This schedule is effective:**  
From *When schedule starts*  
Until *Never (repeat indefinitely)*  
Time zone *Coordinated Universal Time*

**This schedule is effective:**  
From *May 1, 2020 11:00 PM UTC*  
Until *May 1, 2021 11:00 PM UTC*  
Time zone *Coordinated Universal Time*

**1** Type  
☒ Basic ☐ Advanced

**2** Frequency  
Only Once  
Only Once  
Hours and Minutes  
Days  
Weeks  
Months

Frequency cannot be less than 10 minutes.

**3** Expiration  
This schedule is effective:  
From *When schedule starts*  
Until *Never (repeat indefinitely)*  
Time zone *Coordinated Universal Time*

This schedule is effective:  
From *May 1, 2020 11:00 PM UTC*  
Until *May 1, 2021 11:00 PM UTC*  
Time zone *Coordinated Universal Time*



# Defining an Advanced Schedule

- Choose the **Advanced** type to provide an iCal expression.
  - Click **Validate Expression** to verify the correct syntax.
  - The example below runs each month on the 1<sup>st</sup>, 10<sup>th</sup>, and 15<sup>th</sup> days of the month at 5:15 AM, 10:15 AM, 3:15 PM, and 8:15 PM.

```
FREQ=MONTHLY;BYMONTHDAY=1,10,15;BYHOUR=5,10,15,20;BYMINUTE=15;
```

- You can also define multiple schedule frequencies.

Type

☐ Basic ☒ Advanced

Validate Expression

FREQ=DAILY;BYHOUR=17;BYMINUTE=30,40,50;BYSECOND=0; &FREQ=DAILY;BYHOUR=18;BYMINUTE=0,10,20,30,40,50;BYSECOND=0;  
&FREQ=DAILY;BYHOUR=19;BYMINUTE=0,10,20,30;BYSECOND=0;

This schedule is effective:

From When schedule starts

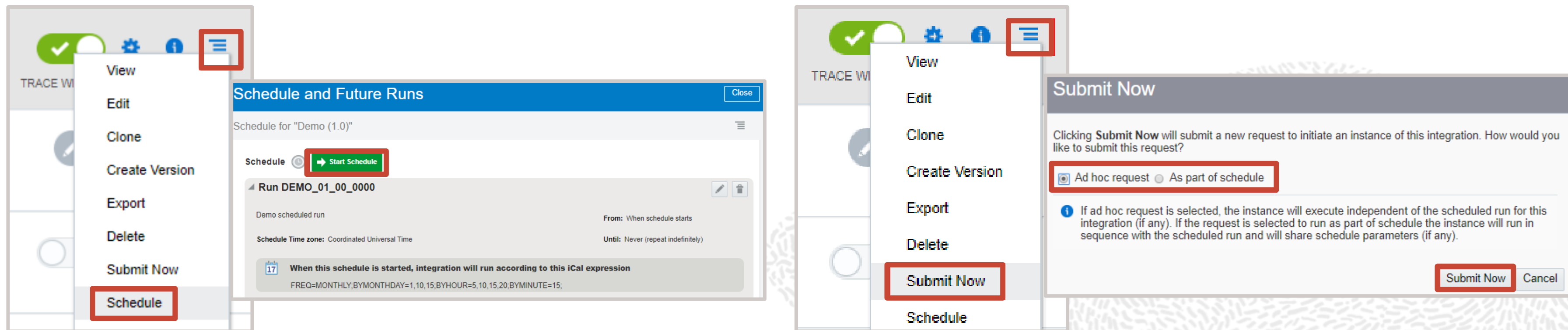
Until Never (repeat indefinitely)

Time zone Coordinated Universal Time

This example runs every day between the hours of 5:30 PM and 7:30 PM, and during these hours it executes every 10 minutes.

# Initiating an Integration Run Schedule

- If the integration is not yet active, the schedule will start automatically when the integration is activated.
- If the integration is already activated, click **Schedule** to activate the schedule.
- Another option is to click **Submit Now** to initiate an integration run immediately.
  - This option is available even if there is no schedule configured at all.
  - You can run this as an ad hoc run, or in sequence with a scheduled run.



# Managing Integration Schedules

- Started schedules can be paused and restarted.
  - A schedule can also be stopped.
- You can also view the list of future runs from this page.

Schedule and Future Runs

Close

Schedule for "Demo (1.0)"

Schedule

Stop Schedule

▶ Run DEMO\_01\_00\_0000

⏸

Future Runs

All

Scheduled

Manually Submitted

Next 2 Month

▼

Apr 1, 2020 10:15:10 PM UTC

Manually Submitted Run	RUNNING	Started Apr 1, 2020 10:15:03 PM UTC
Auto Scheduled Run	WAIT	Started Apr 10, 2020 5:15:17 AM UTC
Auto Scheduled Run	YET TO START	Starts Apr 10, 2020 10:15:17 AM UTC
Auto Scheduled Run	YET TO START	Starts Apr 10, 2020 3:15:17 PM UTC

11

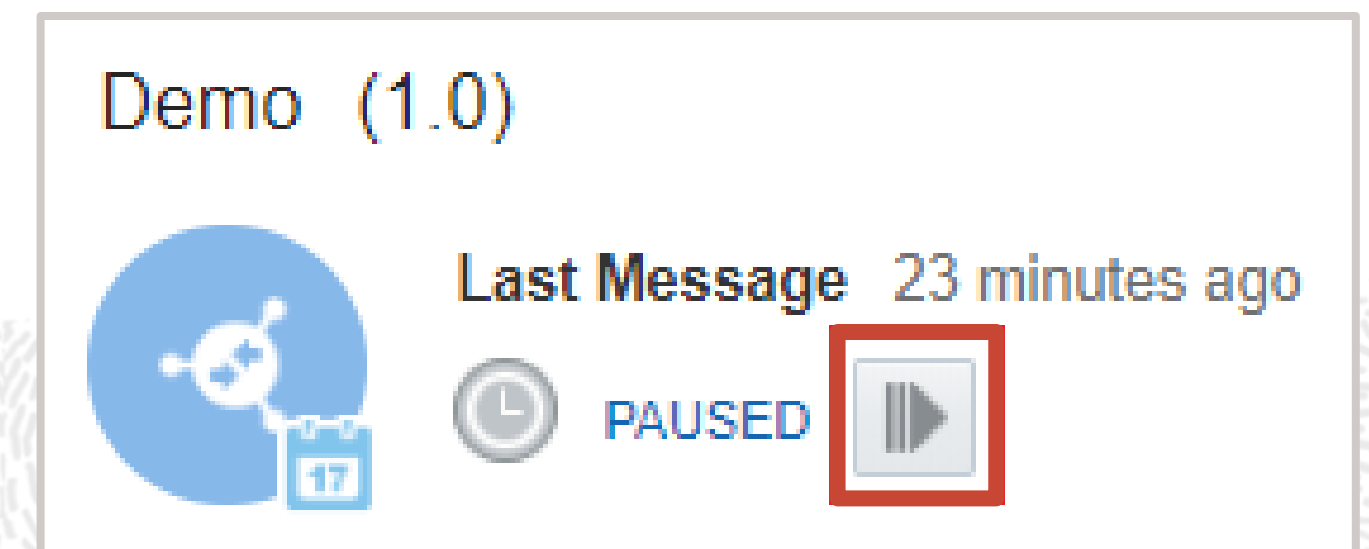
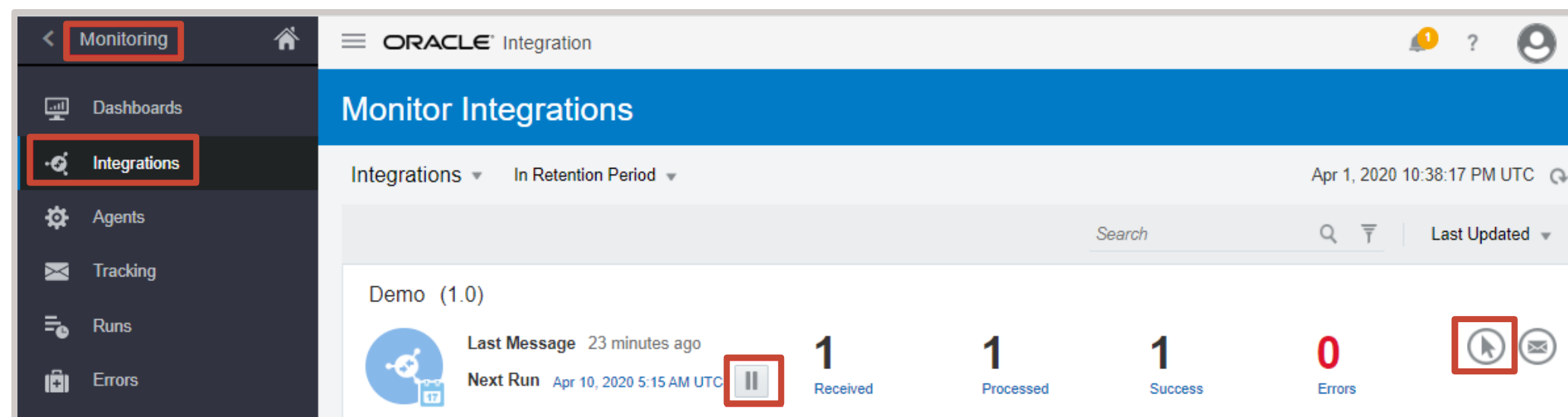
Confidential - Oracle Restricted



# Monitoring Integration Runs

Navigate to the **Monitoring** console and select **Integrations**. Locate the integration on the **Monitor Integrations** page.

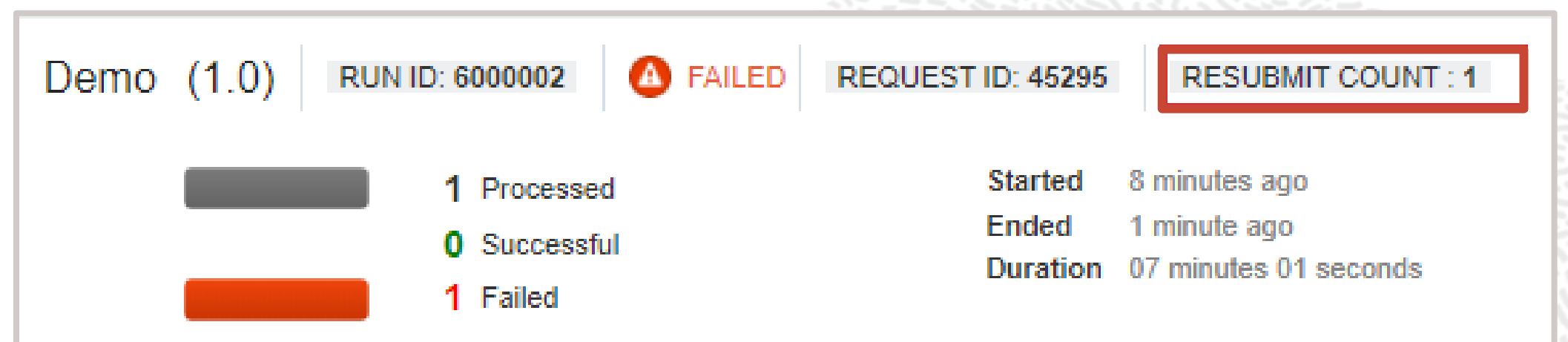
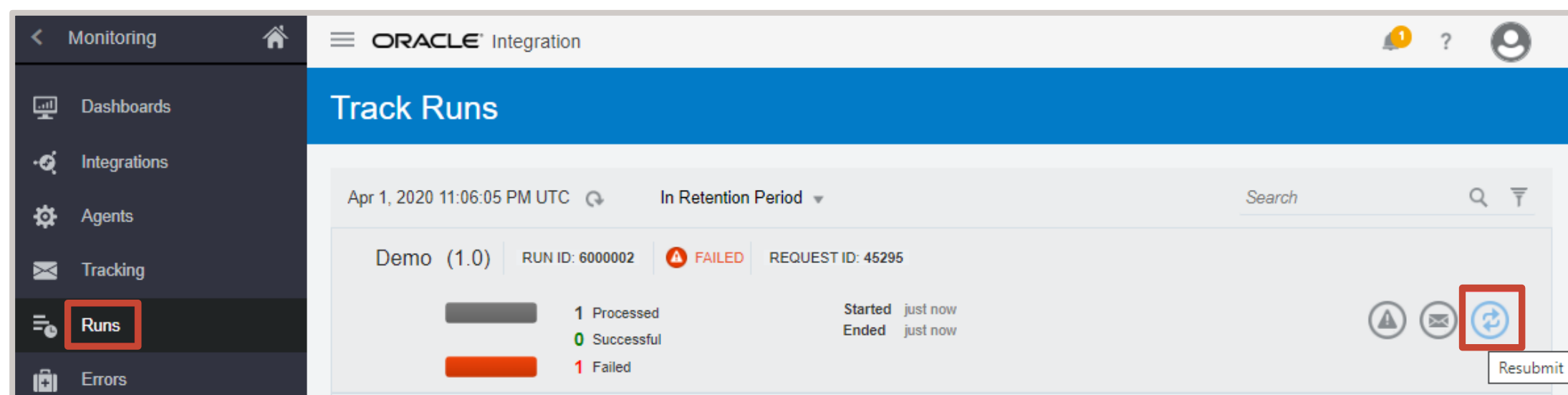
- Review information as to all completed run instances and the time for the next run.
- Click the **Submit Now** icon to start another manual run.
- Click the **Pause** icon to pause the schedule.
- If paused, click the **Play** icon to restart the schedule.



# Resubmitting Failed Runs

Navigate to the **Monitoring** console and select **Runs**. Locate the integration run on the Track Runs page.

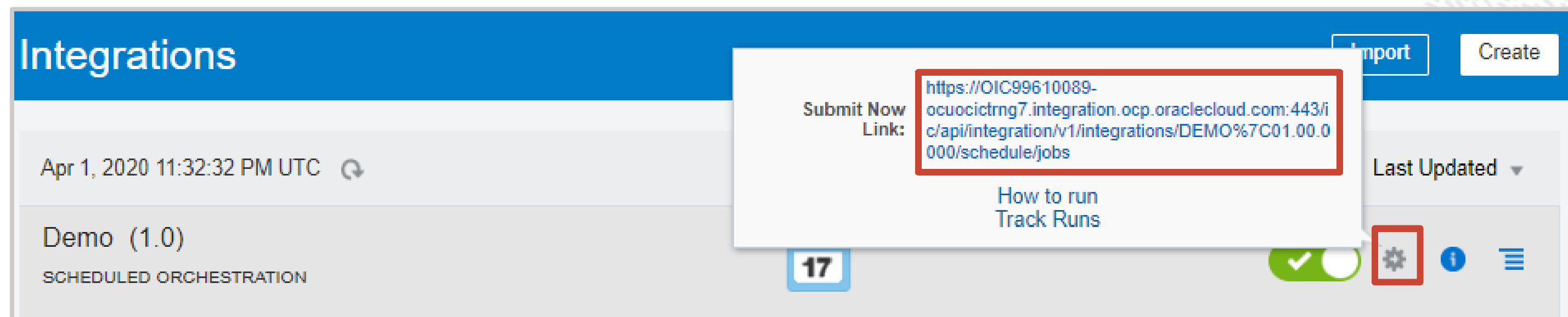
- Click the **Resubmit** icon to execute that scheduled integration instance.
- If the resubmission fails, the run status is updated with a resubmit count.



# Initiating Runs Using the Integration's REST API

Leverage this feature for use in client applications that could be designed to invoke a scheduled run based on dynamic business use cases.

1. Locate the scheduled integration on the **Integrations** page.
2. Click the **Gear** icon on the far right.
3. Copy the URL from **Submit Now Link** and provide it to the client application.
  - The application will also need OIC credentials with the *ServiceInvoker* role.
4. Develop the application logic to start the run as required.

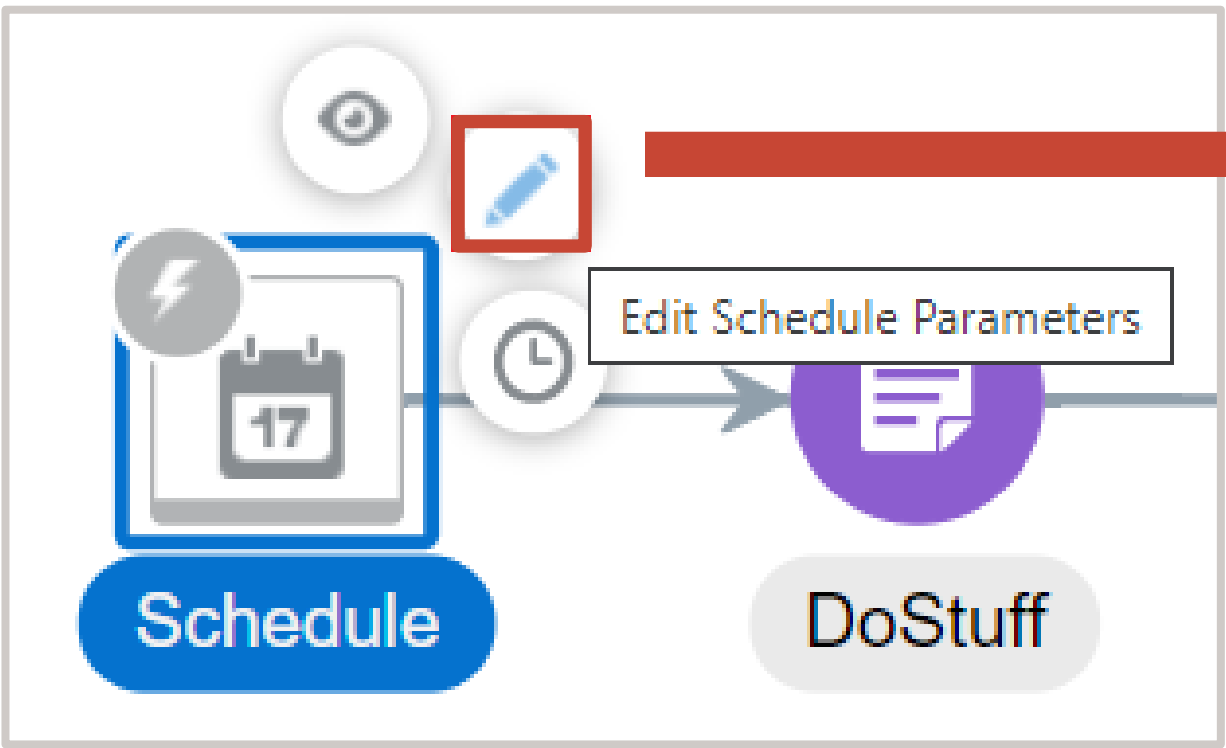
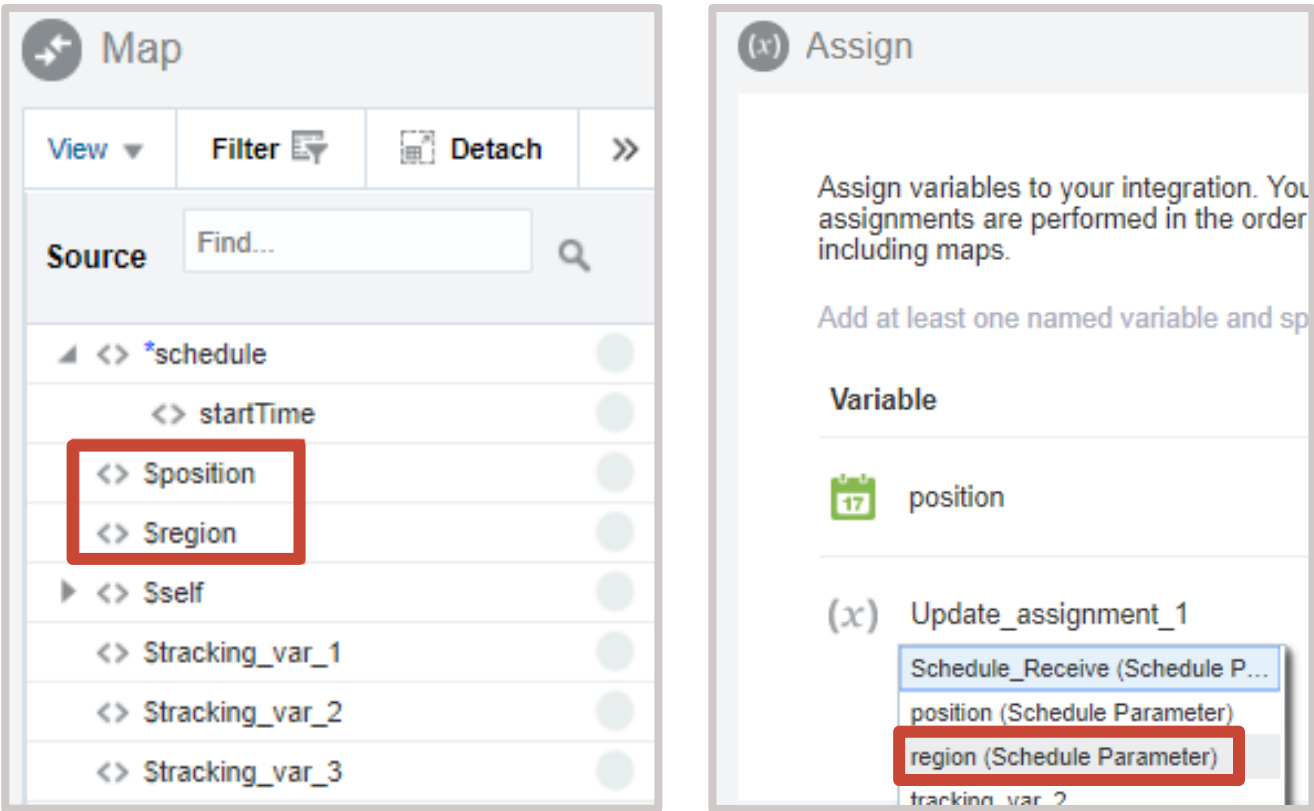




# Using Schedule Parameters

Create and update scalar parameters for scheduled orchestrations.

- Parameter values are available within the **Source** section of mappers and expression editors.
- Updates to parameter values in **Assign** actions persist on to the next run instance of the integration.



### Schedule Parameters Demo (1.0)

Schedule Parameters

Scheduled parameters are available across all scheduled runs of an integration and can be used to facilitate processing of data from one run to the next. For example, when performing batch processing a schedule parameter can be used to track the current position of batched data between runs.

Add at least one named variable. (Maximum 5 variables can be added.)

Parameter Name	Description	Value
position	Last record processed in file	20
region	The current region for FTP downstream updates	North

Create up to five parameters.



# Overriding Schedule Parameters

When you start a schedule (or submit a manual ad hoc request), you will be prompted to accept the current values or choose to provide a new value manually prior to execution.

Submit Now : Configure Schedule Parameters

Configure values for Schedule Parameters. **New Value** replaces the currently stored value and will be used for the new run. **Current Value** indicates the currently stored value.

Parameter Name	Default Value	Current Value	New Value
position	20	40	Enter new value
region	"North"	"North"	Enter new value

Submit

Cancel

View

Edit

Clone

Create Version

Export

Delete

Submit Now

Schedule

Scheduled Run : Configure Schedule Parameters

Configure values for Schedule Parameters. **New Value** replaces the currently stored value and will be used for the new run. **Current Value** indicates the currently stored value.

Parameter Name	Default Value	Current Value	New Value
position	20	40	1
region	"North"	"South"	"West"

Start Schedule

Cancel

Schedule and Future Runs

Schedule for "Demo (1.0)"

Schedule

Start Schedule

Run DEMO\_01\_00\_0000



# Agenda

- Creating Scheduled Integrations
- Monitoring Instances and Message Tracing
- Managing Error Instances and Troubleshooting
- OIC REST API Overview



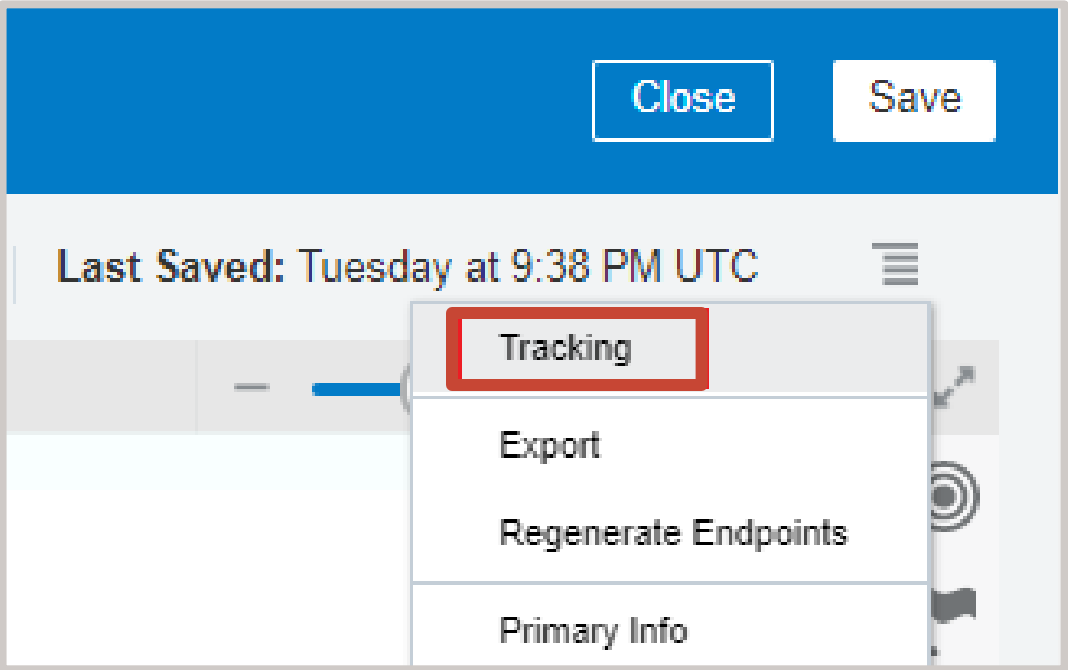


# Business Identifiers (Review)

You must configure at least one Tracking Field to complete the integration’s configuration.

Define up to three tracking fields per integration.

Scheduled orchestrations can use the startTime and scheduled parameters.



App Driven orchestrations can use any inbound source data field.

**Business Identifiers For Tracking**

View Filter Detach

Source Find...

Business identifiers enable runtime tracking on messages. Specify up to three tracking fields. A primary identifier is required. It enables you to track fields across integration flows and is always available.

Additional business identifier fields are optional. At runtime, they are available for tracking only when this integration flow is selected.

Primary	Tracking Field	Tracking Name	Tracking Variable	Help Text	
✓	startTime	start Time	tracking_var_1	How to track it?	
	SFTPDownloadDir	FTPDownloadDir	tracking_var_2	How to track it?	
	SFileToDownload	FileToDownload	tracking_var_3	How to track it?	

**Business Identifiers For Tracking**

View Filter Detach

Source Find...

Business identifiers enable runtime tracking on messages. Specify up to three tracking fields. A primary identifier is required. It enables you to track fields across integration flows and is always available.

Additional business identifier fields are optional. At runtime, they are available for tracking only when this integration flow is selected.

Primary	Tracking Field	Tracking Name	Tracking Variable	Help Text	
✓	SrNumber	Sr Number	tracking_var_1	How to track it?	
	CreatedBy	Created By	tracking_var_2	How to track it?	
	Drag a trigger field here	tracking_var_3	tracking_var_3	How to track it?	



# Filtering and Tracking Business Identifiers

Click **Tracking** in the Monitoring portal to view any integration instance.

Monitoring

Dashboards

Integrations

Agents

Tracking

Runs

Errors

ORACLE Integration

Track Instances

Apr 2, 2020 4:59:17 PM UTC

Last 1 Hour

	start Time: 2020-04-02T16:58:22.479+00:00 Demo   1.0 Run ID: 6400010 COMPLETED	Instance ID: 6400010 position 40 region North	Received just now Completed just now Duration 357 milliseconds
	start Time: 2020-04-02T16:57:52.861+00:00 Demo   1.0 Run ID: 6400009 COMPLETED	Instance ID: 6400009 position 1 region North	Received 1 minute ago Completed 1 minute ago Duration 371 milliseconds
	Party Id: 3 ora028 RN Incidents For OSC   1.1.0 COMPLETED	Instance ID: 6400008	Received 3 minutes ago Completed 3 minutes ago Duration 01 seconds
	Party Id: 11 ora028 RN Incidents For OSC   1.1.0 COMPLETED	Instance ID: 6000005	Received 13 minutes ago Completed 13 minutes ago Duration 01 seconds

Search

Filter icon

Received

Completed

Duration

Received

Completed

Failed

Aborted

Message

Integration

Run ID

Instance Id

Asserter Instance

Message State

13 minutes ago

Specify a time period

Click the primary business identifier to view the integration instance.

Search for a tracking field value

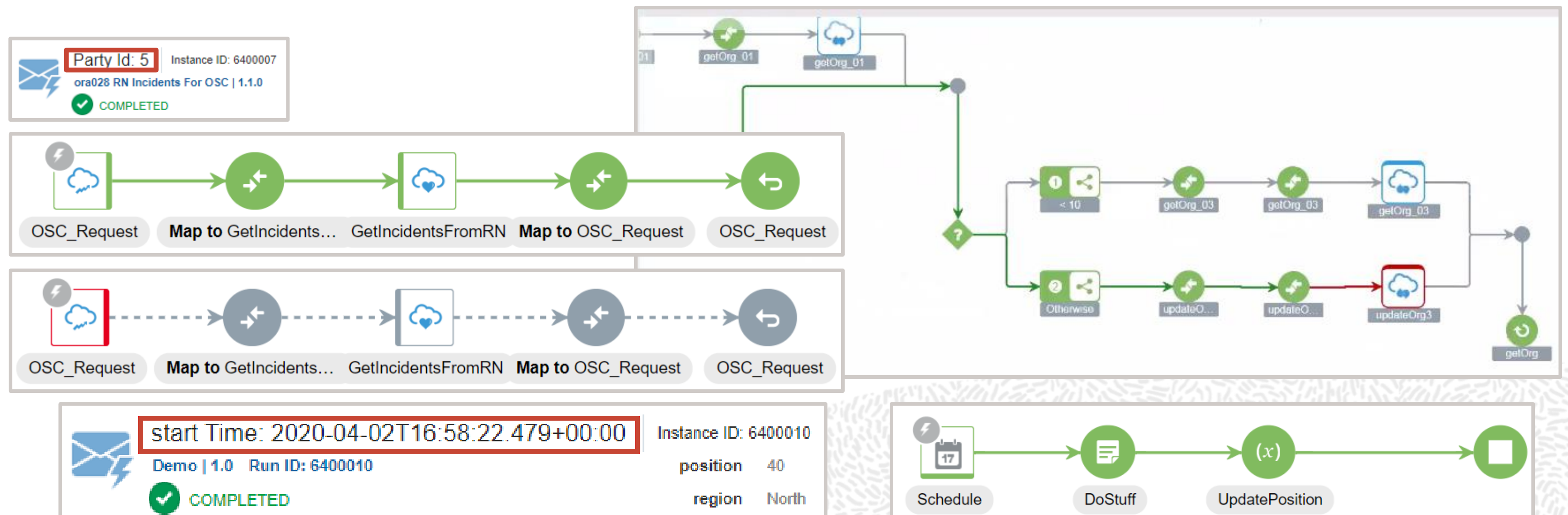
Filter options to limit display



# Viewing the Integration Instance

Click the business identifier to view the message flow.

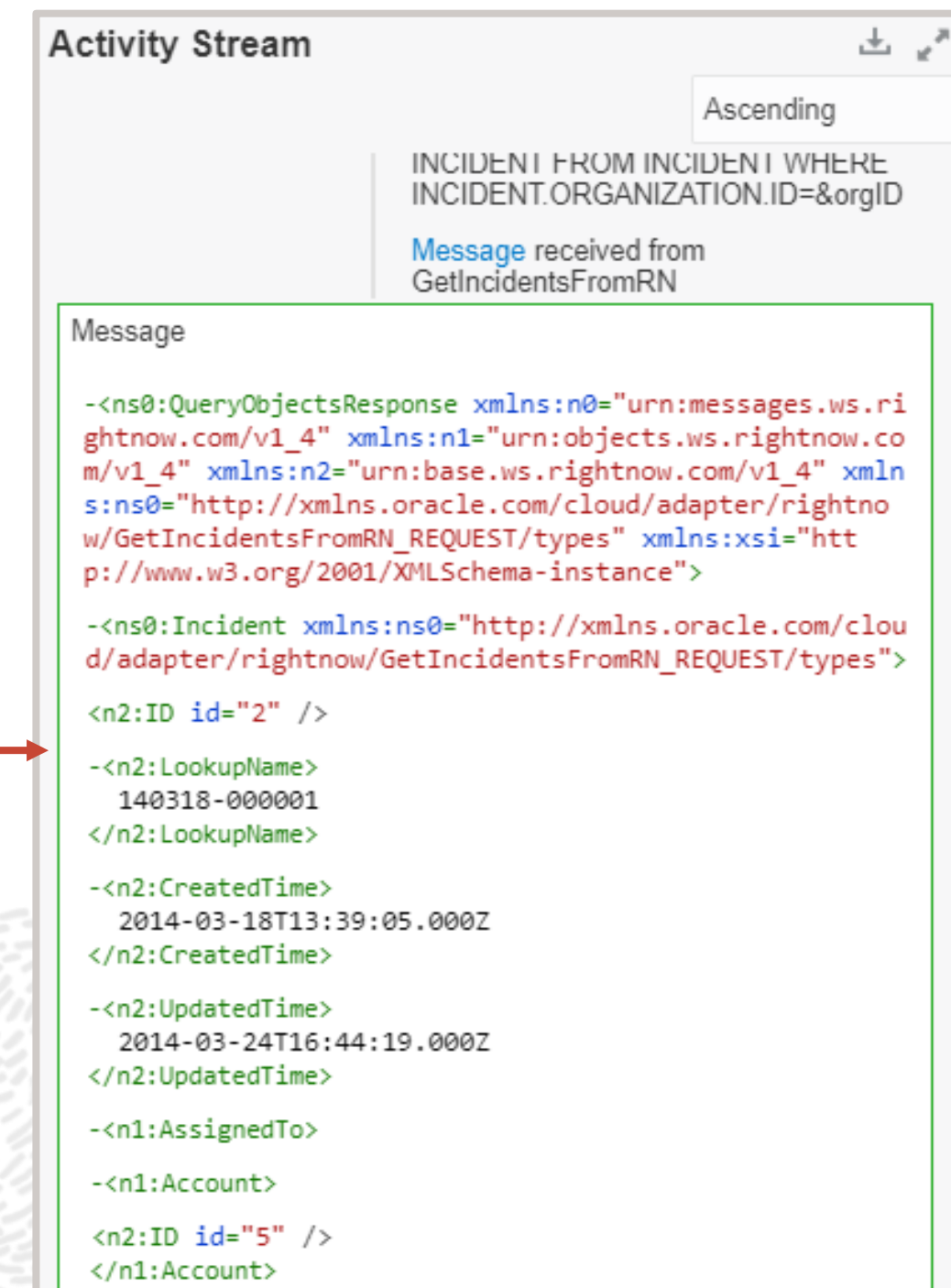
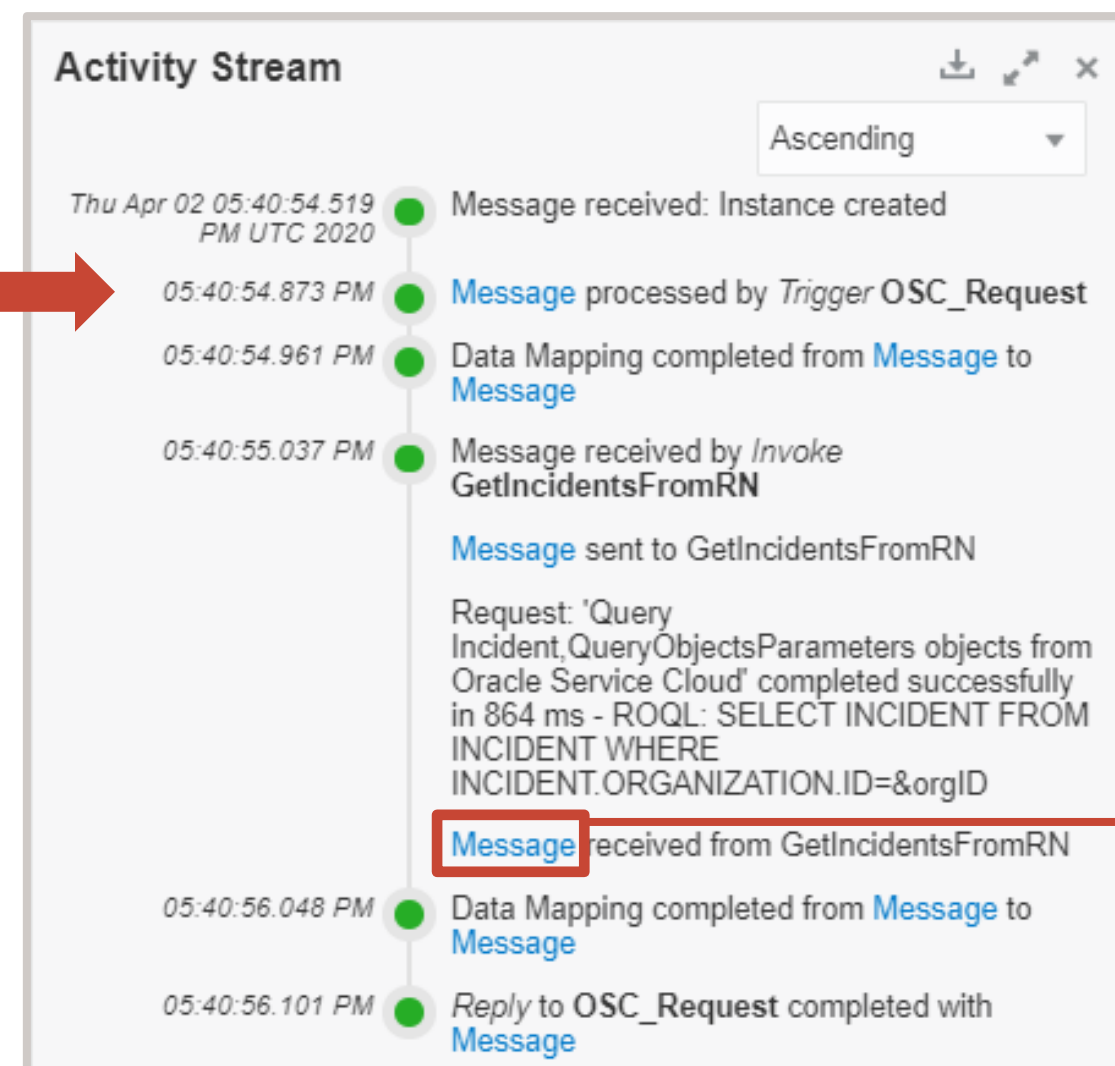
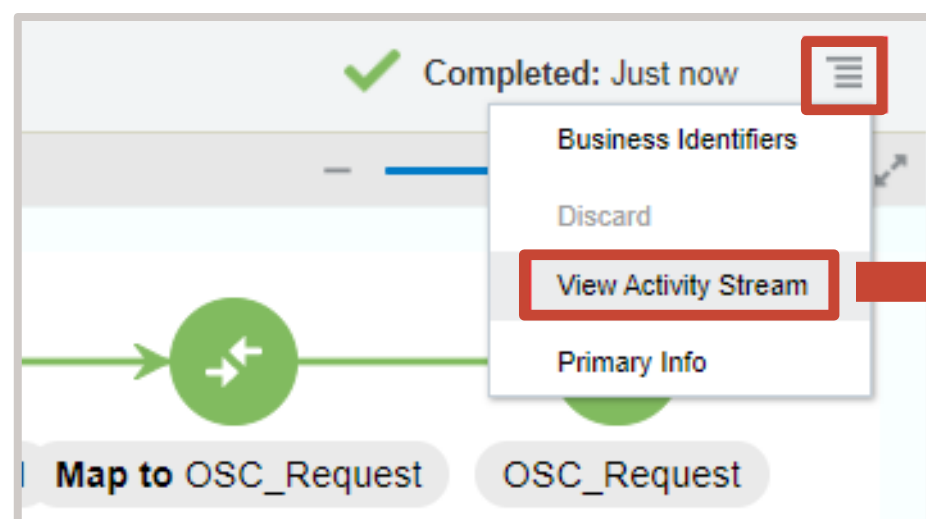
- A successful message flow is indicated with green arrows all the way to the end.
- Message flow errors will be indicated in red at the error location.





# Viewing Instance Details

- From the Actions menu, click **View Activity Stream**.
- To view a specific payload, click the corresponding **Message** link.



# Agenda

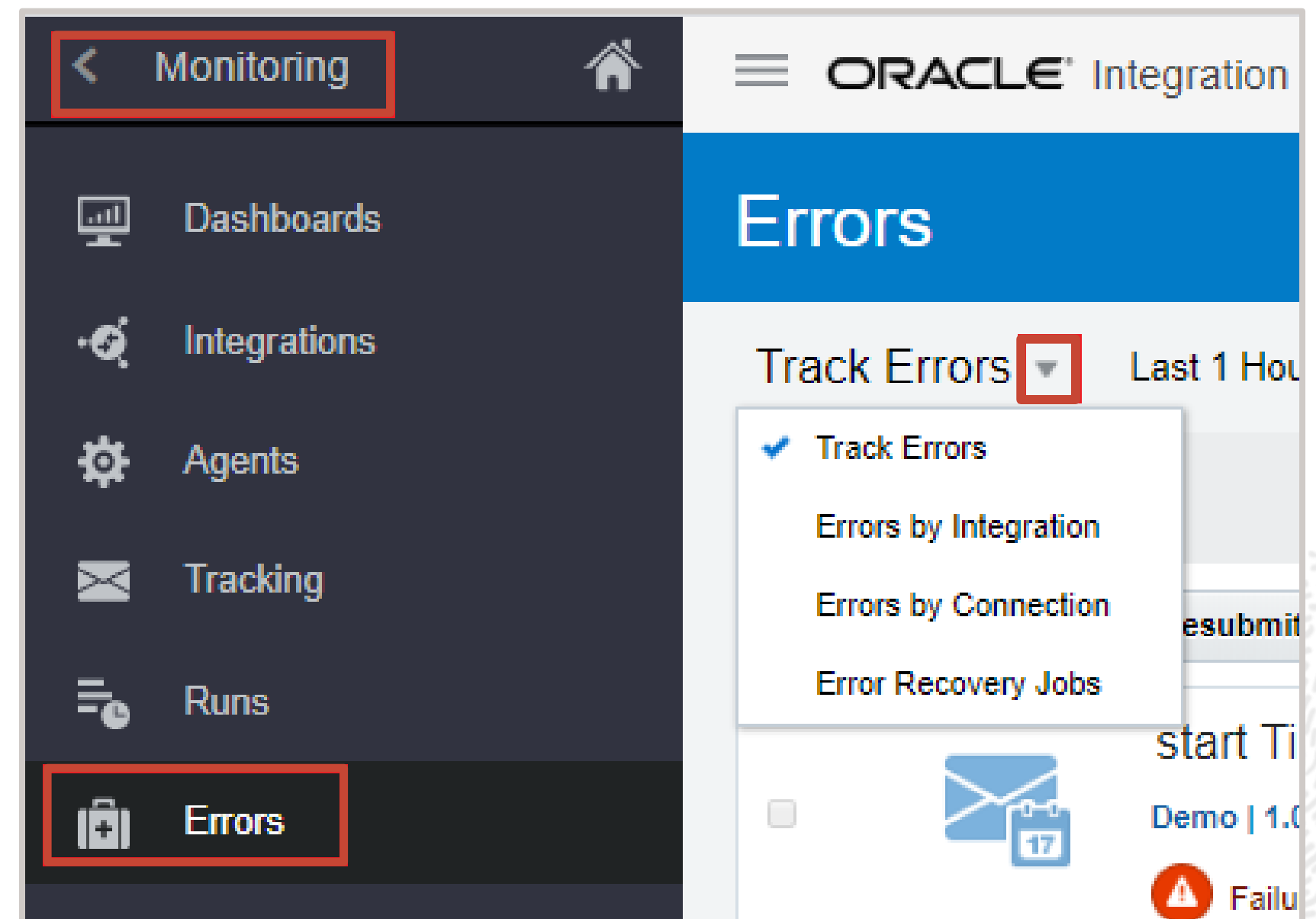
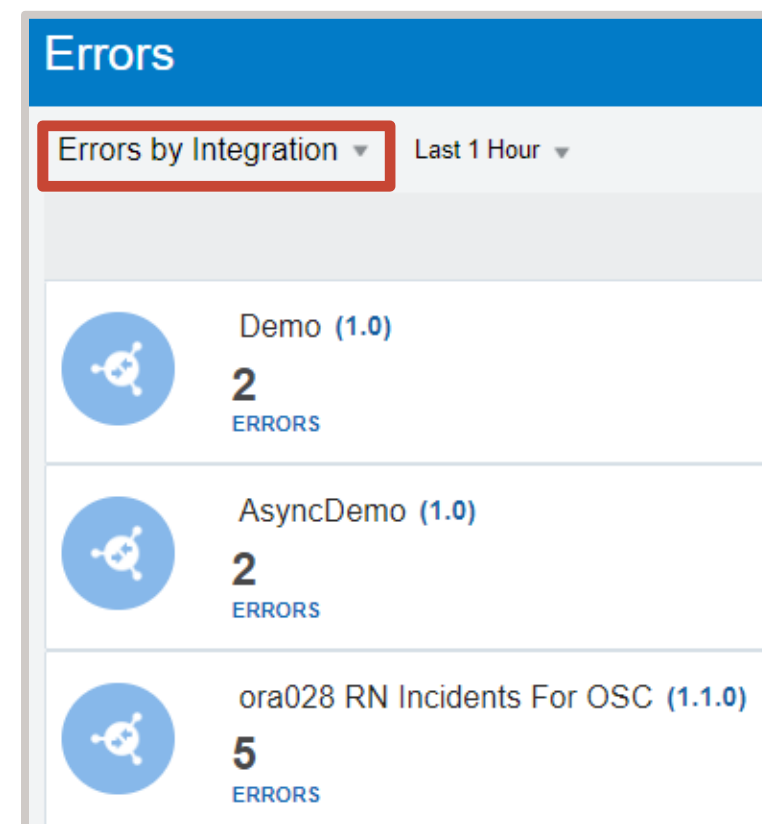
- Creating Scheduled Integrations
- Monitoring Instances and Message Tracing
- Managing Error Instances and Troubleshooting
- OIC REST API Overview



# Monitoring and Managing Errors

You can locate and manage errors from the **Errors** page using the following filters:

- All Integration Instances
- Errors by Integration
- Errors by Connection
- Error Recovery Jobs





# Viewing Errors

To view a specific error message, you have two options:

1. Locate the failed instance on the **Track Instances** page, and then view it in the Activity Stream.
2. Locate the instance on the **Errors** page, and then click the **View Error** icon.

The image displays three screenshots from the Oracle Service Cloud interface, illustrating two methods to view error messages.

**Track Instances Page:** The top-left screenshot shows the 'Track Instances' page. A failed instance is highlighted with a red box and labeled '1'. The instance details include 'Party Id: 3', 'Instance ID: 6000007', and 'ora028 RN Incidents For OSC | 1.1.0'. The status is 'FAILED'.

**Activity Stream:** The top-middle screenshot shows the 'Activity Stream' for the selected instance. A red box highlights the error message, labeled '2'. The error message is: 'Error processing message in invoke GetIncidentsFromRN'. The error details are: 'Error message is - Error sending bytes: Invocation failed due to following error n1:exceptionCode:INVALID\_REQUEST n1:exceptionMessage:near "&": syntax error This is most likely a defect. Please raise a support ticket. :Application Error'.

**Errors Page:** The top-right screenshot shows the 'Errors' page. A failed instance is highlighted with a red box and labeled '2'. The instance details include 'Party Id: 3', 'Instance ID: 6000007', and 'ora028 RN Incidents For OSC | 1.1.0'. The status is 'Failed at RightNow Demo | 1.0 | Oracle Service Cloud (RightNow)...'. The error message is: 'Error sending bytes: Invocation failed due to following error n1:exceptionCode:INVALID\_REQUEST'. The detail error message is: 'Error sending bytes: Invocation failed due to following error n1:exceptionCode:INVALID\_REQUEST n1:exceptionMessage:near "&": syntax error This is most likely a defect. Please raise a support ticket. :Application Error'.

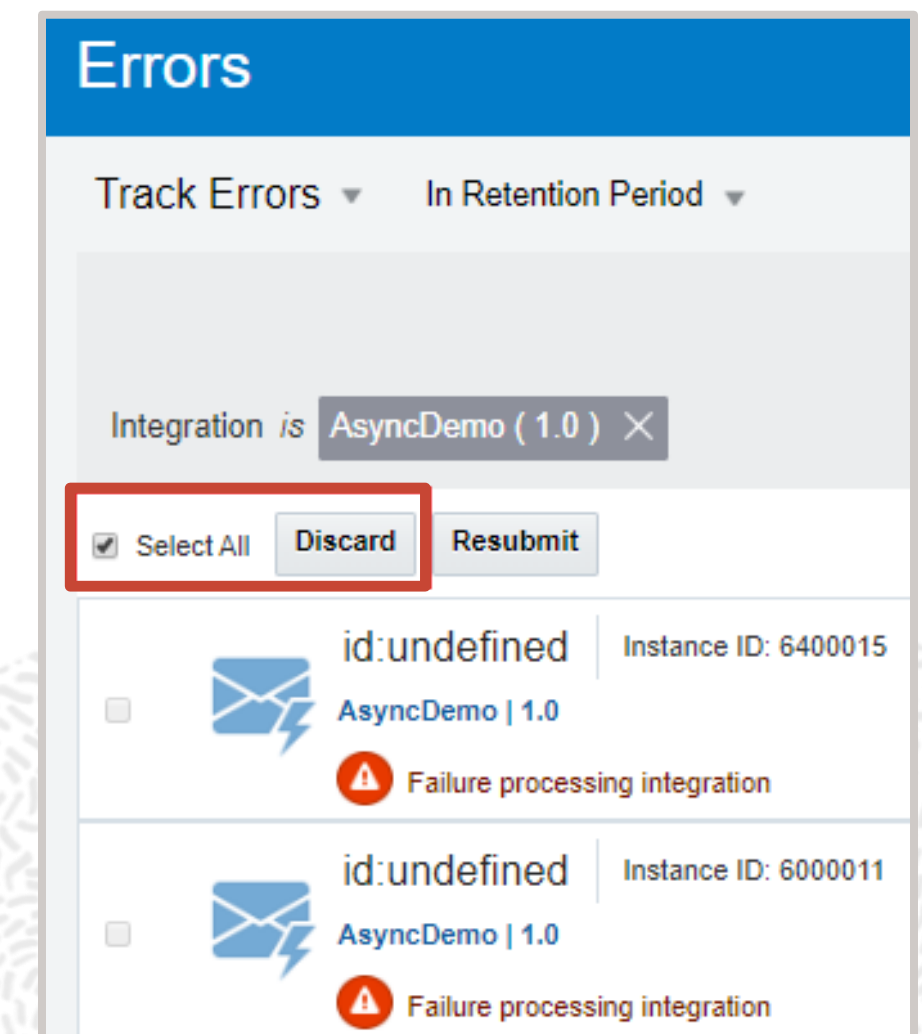
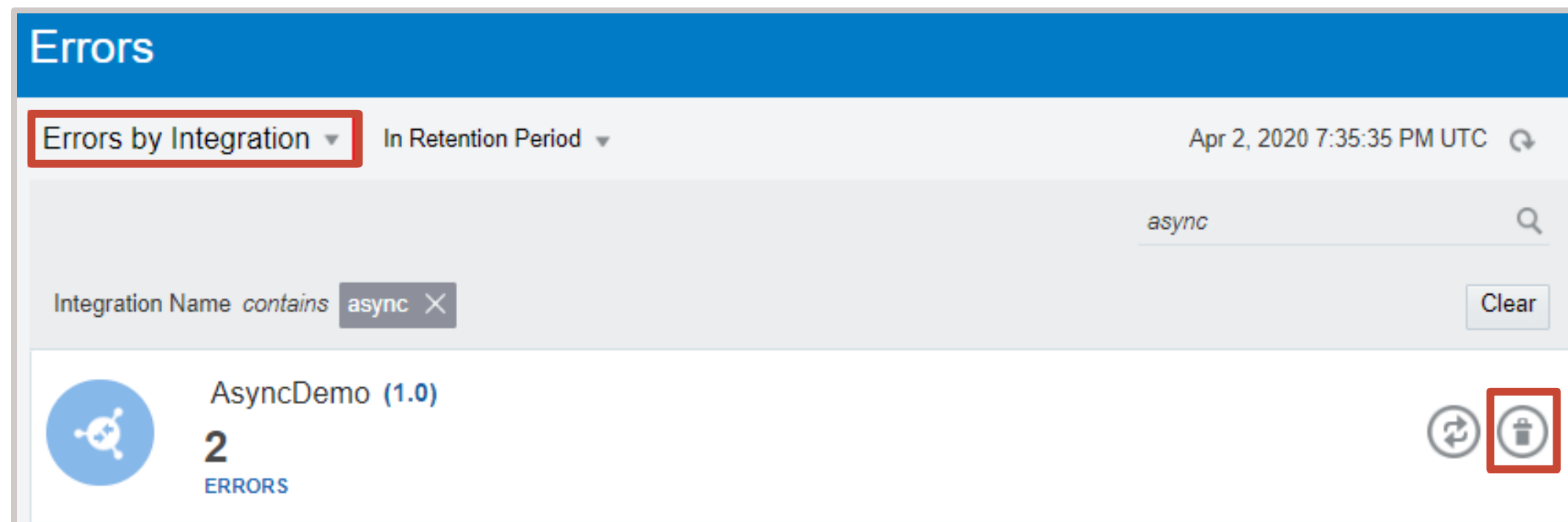
**Process Flow Diagram:** The bottom screenshot shows a process flow diagram. The steps are: 'OSC\_Request' (green box) → 'Map to GetIncidents...' (green circle) → 'GetIncidentsFromRN' (blue box with a red error icon) → 'Map to OSC\_Request' (blue circle) → 'OSC\_Request' (blue circle). The 'GetIncidentsFromRN' step is highlighted with a red box.



# Discarding Errors

After viewing or resolving error information, you can discard one or more errors.

- Discard all specific integration errors, or select one or more specific error instances.
- Discarded errors are removed from the **Errors** page but can still be seen on the **Tracking** page in the discarded state.



# Resubmitting Failed Instances

For recoverable errors, if the external issue can be resolved, you can choose to manually resubmit a single instance or execute a bulk resubmission.

- Resubmitted error instances that are successful will be removed from the error list.

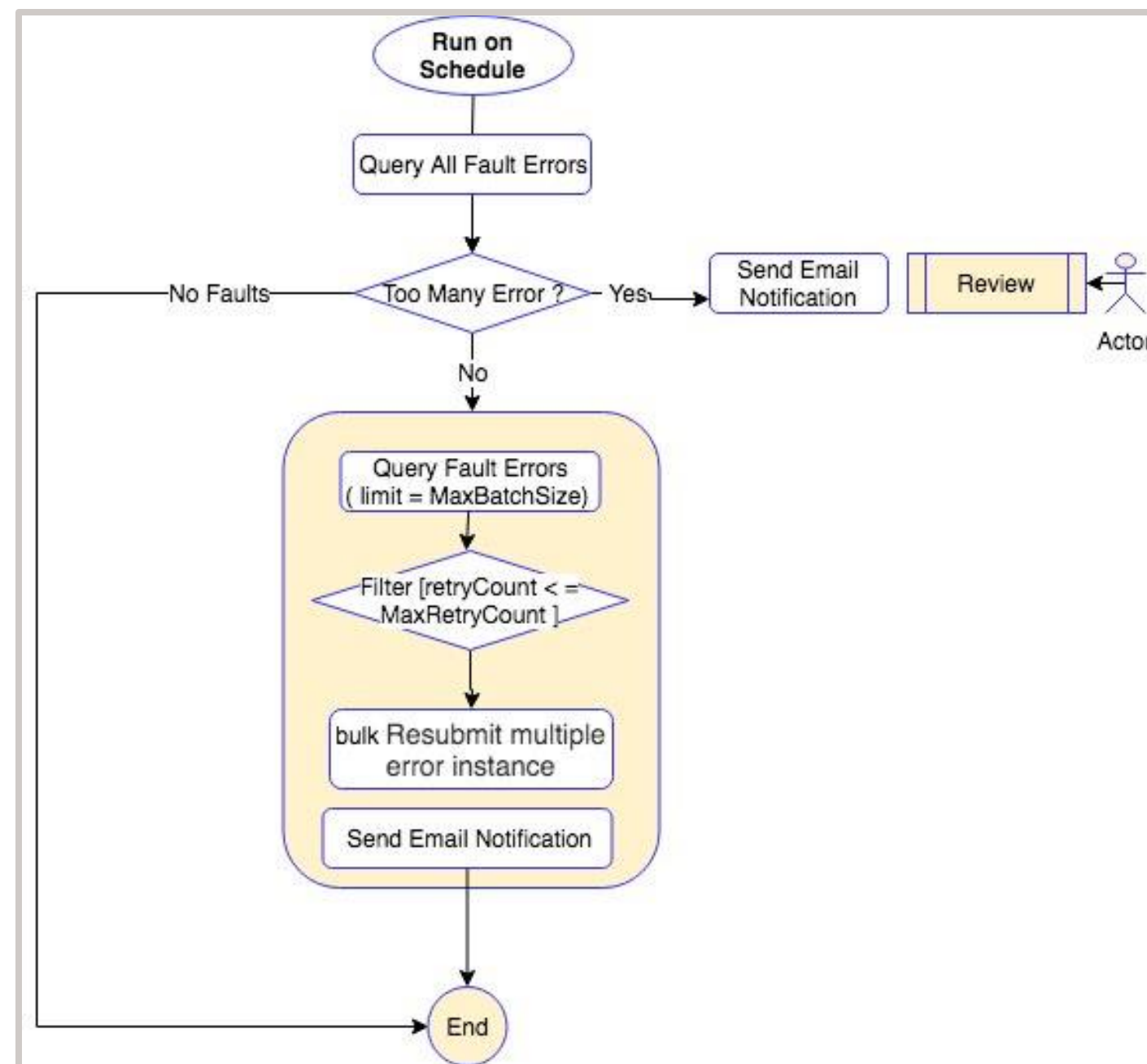
The screenshot shows the 'Errors' dashboard with a blue header. Below the header, there are filters for 'Errors by Integration' and 'In Retention Period'. A search bar contains the text 'async'. Below the search bar, there is a filter for 'Integration Name contains async'. The main content area shows a card for 'AsyncDemo (1.0)' with '2 ERRORS'. To the right of the card, there are two icons: a circular arrow (resubmit) and a trash can (delete). A confirmation dialog is open in the foreground, asking 'Are you sure you want to resubmit all the errors for the selected integration?'. The dialog has a green checkmark icon and two buttons: 'Yes' and 'No'.

The screenshot shows the 'Errors' dashboard with a blue header. Below the header, there are filters for 'Track Errors' and 'In Retention Period'. A filter for 'Integration is AsyncDemo (1.0)' is applied. Below the filter, there are two buttons: 'Select All' and 'Resubmit'. The main content area shows a list of failed instances. Each instance has a checkbox, an envelope icon, the text 'id:undefined', the integration name 'AsyncDemo | 1.0', and the error message 'Failure processing integration'. The first instance has 'Instance ID: 6400015' and the second has 'Instance ID: 6000011'.



# Automated Bulk Recovery

A dynamic solution would be to create one or more integration flows designed to automatically resubmit faulted instances in bulk at specified intervals. High-level steps include the following:



- Create a scheduled orchestration flow.
- Add schedule parameters to define:
  - Min/Max Batch Sizes
  - An Error Query Filter
  - Retry Count
- Use the REST API to retrieve error instances.
- Use the REST API to resubmit error instances.
- Create appropriate email notifications.
- Define a schedule to run (*i.e. once per hour*).
- Activate the integration and monitor.

Define which specific subset of faulted instances to process.

# Setting Logging Levels

For additional troubleshooting assistance, you may wish to increase logging levels for content sent to the diagnostic logs.

- Navigate to **Settings > Logging Levels**
- Select a level for one or more loggers.
- Restore back to defaults once your troubleshooting is completed.

Changing from the default values can fill up disk space and degrade system performance.

**Note:** *Loggers & diagnostic log access is no longer supported with newer OCI Generation 2–provisioned OLC instances.*

Settings

Certificates

Notifications

Database

Logging Levels

Recommendations

API Platform

Tracing

ORACLE Integration

Logging Levels

Apr 2, 2020 8:26:03 PM UTC

Change the logger level of any logger under 'oracle' directory. This may cause disk space to fill up and degrade system performance.

Components and Services	Default Level	Current Level
oracle	ERROR	ERROR:1 (Error)
oracle.adf	ERROR:1 (Error) - Inherited from parent	ERROR:1 (Error) - Inherited from parent
oracle.adfdebugdiagnostics	ERROR:1 (Error) - Inherited from parent	ERROR:1 (Error) - Inherited from parent
oracle.adfdiagnostics	ERROR:1 (Error) - Inherited from parent	ERROR:1 (Error) - Inherited from parent
oracle.adfdt	ERROR:1 (Error) - Inherited from parent	ERROR:1 (Error) - Inherited from parent
oracle.adfdtinternal	ERROR:1 (Error) - Inherited from parent	ERROR:1 (Error) - Inherited from parent
oracle.adfinternal	ERROR:1 (Error) - Inherited from parent	ERROR:1 (Error) - Inherited from parent
oracle.fabric	WARNING:1 (Warning) - Inherited from parent	ERROR:1 (Error) - Inherited from parent
oracle.ics	ERROR:1 (Error) - Inherited from parent	ERROR:1 (Error) - Inherited from parent
oracle.integration	ERROR:1 (Error) - Inherited from parent	INCIDENT_ERROR:1 (Incident error) ERROR:1 (Error) - Inherited from parent WARNING:1 (Warning)
oracle.jbo	ERROR:1 (Error) - Inherited from parent	NOTIFICATION:1 (Info) NOTIFICATION:16 (Config) NOTIFICATION:32 (Notification)
oracle.jrf	WARNING:1 (Warning) - Inherited from parent	TRACE:1 (Fine) TRACE:16 (Finer) TRACE:32 (Finest)
oracle.mds	WARNING	
oracle.odl	WARNING:1 (Warning) - Inherited from parent	ERROR:1 (Error) - Inherited from parent

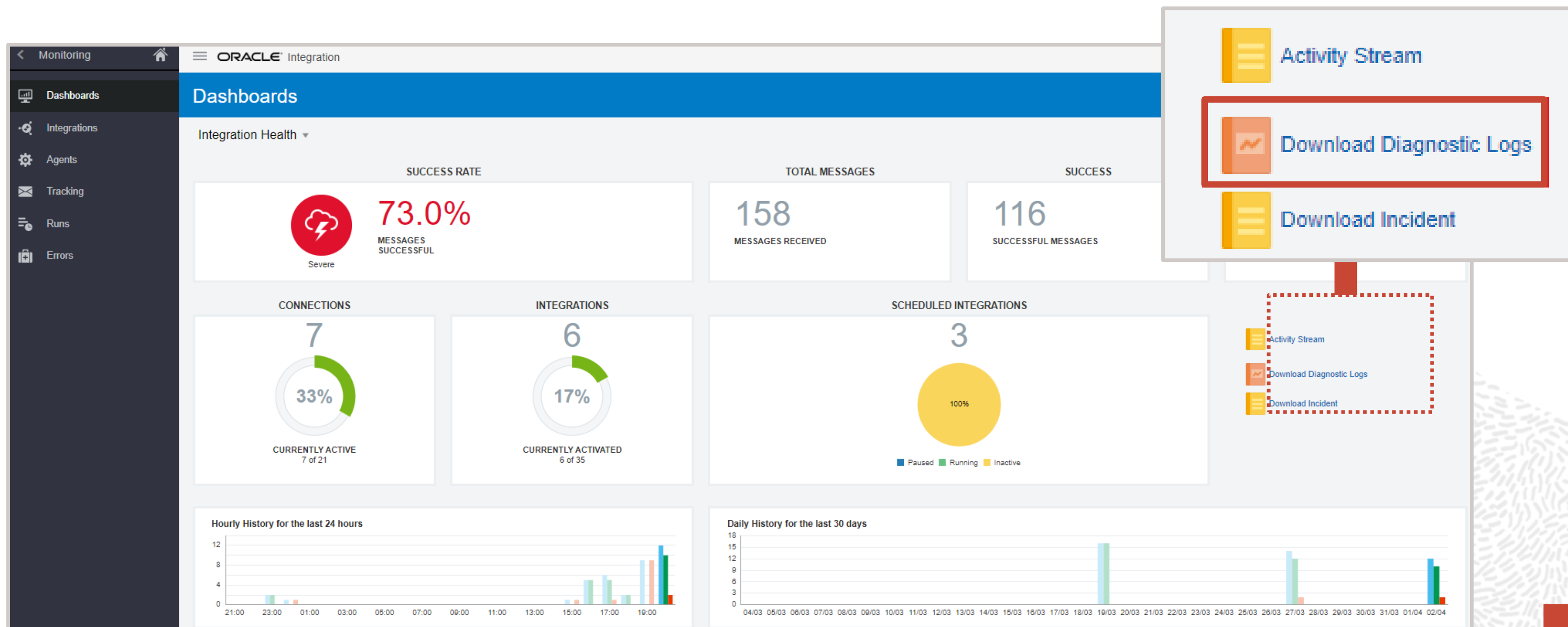




# Accessing Logs

**Note:** *Loggers & diagnostic log access is no longer supported with newer OCI Generation 2–provisioned OIC instances.*

In the **Monitoring** portal, click the **Download Diagnostic Logs** link on the **Dashboards** page.





# Downloaded Contents

**Note:** *Loggers & diagnostic log access is no longer supported with newer OCI Generation 2–provisioned OIC instances.*

The extracted archive will include folders containing the log files for each WebLogic Server instance for this OIC environment (*WLS domain*).

- The `_server_x_diagnostic.log` file is used for all OIC loggers.
- Each log entry uses the standard Oracle logging format.

Logs >

☐ Name

☒ oiciad5l\_domain-diagnostic-logs-0402201107.zip

environment.txt

oiciad5l\_server\_1

oiciad5l\_server\_2

Logs > 

oiciad5l\_server\_1

☐ Name

access.log

ics-audit.log

ics-flow.log

oiciad5l\_server\_1.log

oiciad5l\_server\_1.out

☒ oiciad5l\_server\_1-diagnostic.log

oiciad5l\_server\_1-ics-resiliency.log

oiciad5l\_server\_1-ics-resiliency-1.log

oiciad5l\_server\_1-ics-resiliency-2.log

oiciad5l\_server\_1-nodemanager.log

```
value:RichPanelGroupLayout[UIXFacesBeanImpl, id=dc_cli_pgl19] for
key:PropertyKey[usagePgl]
[2020-03-31T19:23:29.788+00:00] [oiciad5l_server_1] [ERROR] []
[oracle.cloud.cpi.agent.transport.AQChannel] [tid: pool-124-
thread-1] [userId: ora028] [ecid: 665fc7b5-1801-4176-8c52-
f31bd31312eb-000e73f6,0:2] [APP: icswebapp] [partition-name:
DOMAIN] [tenant-name: GLOBAL] No response received within
response time out window. Connectivity Agent may not be running,
or temporarily facing connectivity issues to Oracle Integration
Cloud Service. Please check the health of the Connectivity Agent
in Agent Monitoring page. For additional insight into request
a22c7fc98d210948e053345b080a5095 processing refer to the
Connectivity Agent logs.
[2020-03-31T19:23:37.727+00:00] [oiciad5l_server_1] [WARNING] []
[org.apache.myfaces.trinidad.component.UIXComponentBase] [tid:
```





# Agenda

- Creating Scheduled Integrations
- Monitoring Instances and Message Tracing
- Managing Error Instances and Troubleshooting
- OIC REST API Overview



# REST API for Oracle Integration

- Leverage the OIC REST APIs to automate administration and operations tasks.
- All REST API operations:
  - Are executed over secure sockets layer (SSL)
  - Provide detailed documentation with example usage

 Help Center

Cloud / Cloud Platform / Integration

**REST API for Oracle Integration**

**Connections**

- Delete a Connection
- Refresh Metadata for a Connection
- Retrieve a Connection
- Retrieve Connection Property Attachment
- Retrieve Connections
- Test a Connection
- Update a Connection
- Upload Connection Property Attachment
- Validate a Connection

**Packages**

- Delete a Package
- Export a Package
- Import Sample Packages
- Import(Add) a Package
- Import(Replace) a Package
- Retrieve a Package
- Retrieve Packages

**Monitoring**

- Discard an Error Integration Instance
- Discard Error Integration Instances
- Download a Log File
- Download an Incident
- Resubmit an Error Integration Instance
- Resubmit Error Integration Instances
- Retrieve Activated Integrations
- Retrieve an Activated Integration
- Retrieve an Error Integration Instance
- Retrieve an Integration Instance
- Retrieve Audit Records
- Retrieve Error Integration Instances
- Retrieve Integration Instances
- Retrieve the Activity Stream

**Integrations**

- Clone an Integration
- Delete an Integration Version
- Export an Integration
- Import(Add) an Integration
- Import(Replace) an Integration
- Retrieve an Integration
- Retrieve Integration Activation Errors
- Retrieve Integrations
- Update(Activate/Deactivate) an Integration

**Scheduled Integrations**

- Pause an Integration Schedule
- Resume an Integration Schedule
- Retrieve Scheduled Integration Run Status
- Run a Scheduled Integration Now
- Start an Integration Schedule
- Stop an Integration Schedule
- Update Scheduled Integration Parameters





# Example Use Cases

- Activating/deactivating integrations
- Exporting/importing integrations
- Updating connection URL or security credentials
- Downloading log files

Each use case is supported with specific documentation details:

## Update(Activate/Deactivate) an Integration

POST

`/ic/api/integration/v1/integrations/{id}`

# Moving Assets Between Environments

- You can export OIC data objects as an archive and import that archive into another OIC instance.
- Activated integrations in OIC do not need to manually configure connection endpoints and passwords – integration activations can occur automatically.

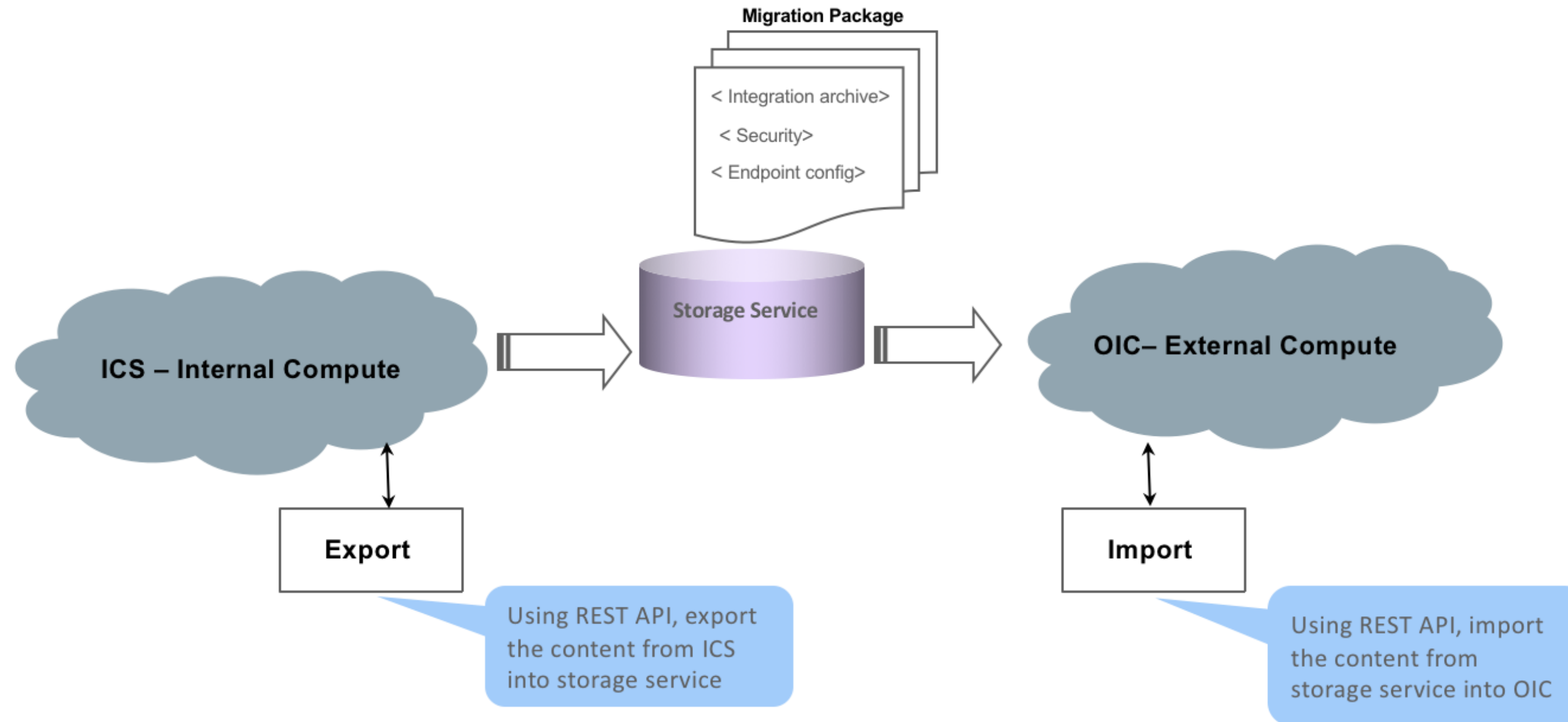


**Oracle Integration Cloud**



**Oracle Integration Cloud**

# Migration Overview



Export REST API: **POST**  
`/ic/api/common/v1/exportServiceInstanceArchive`

Import REST API: **POST**  
`/ic/api/common/v1/importServiceInstanceArchive`



# Summary

---

In this lesson, you should have learned how to:

- Define basic and advanced schedules for a scheduled orchestration
- Execute scheduled or ad hoc runs in scheduled orchestrations
- Resubmit failed scheduled orchestration run instances
- Leverage schedule parameters for multiple run instances
- Monitor integration instances and message details
- View, resubmit, or discard integration error instances
- Access diagnostic logs for troubleshooting
- Describe the OIC REST API capabilities



# Practice 10-1: Creating a Scheduled Orchestration

This practice includes:

- Defining a new Scheduled Orchestration
- Asynchronously Invoking an App Driven Orchestration
- Executing and Monitoring an Ad Hoc Run

