Oracle ACE

Tech Superstars Unite

Get worldwide recognition as an Oracle ACE

Oracle.com Profile Page

Travel Support

Swag & Certification Exam Credits

Networking Events

Exclusive Content from Product Development

Your Own Oracle Cloud Account

@oracleace

linkedin.com/groups/72183

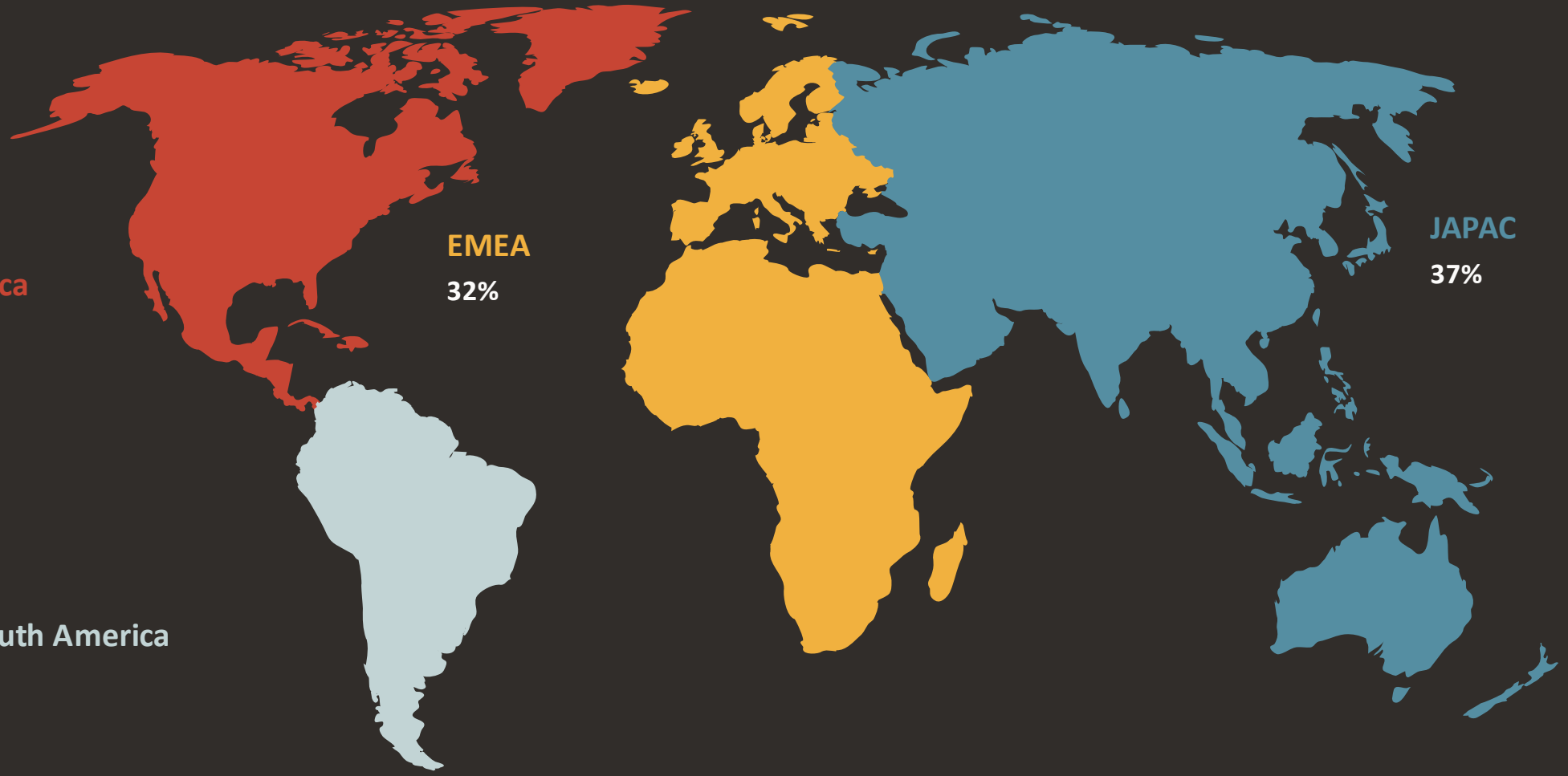
@oracleace.bsky.social

Learn more at:
ace.oracle.com

Oracle ACE

1,440+ Global ACE Members in 86 Countries

Members Directory: ace.oracle.com/directory



Oracle ACE

Mastering Design Patterns for Custom Business Processes in Oracle Fusion Applications: Strategies for CX, HCM, and SCM

Real-World Patterns: ESS + BIP + OIC | Groovy-based ESS | Emerging AI Agent Studio

Alex Rabinovich, Oracle ACE Pro

CEO, CID SOFTWARE SOLUTIONS LTD

Agenda

- Custom Business Processes Examples
  1. Parameterised Data Fetch and Update
  2. Event Driven Processing
- End User Interaction
  1. ESS Framework Utilization (standard UI, scheduling, monitoring)
  2. BI Publisher Catalog Report Parameter Page
  3. Classic UI Page Composer Links
  4. Application Composer Actions/CX Smart Actions
  5. Visual Builder Studio Extensions
  6. Dedicated Visual Builder Studio AppUI Screens
  7. AI Agent Exposure
- Covered Design Patterns for Custom Business Processes in FA
  1. BI Publisher as Data Fetch and UI Templating Tool
  2. ESS Job submitting Synchronous Processing in OIC
  3. ESS Job submitting Asynchronous Processing in OIC
  4. Standard/Custom Integration Event triggering an OIC Integration
  5. Groovy based ESS Job
  6. New: AI Agent Studio
- Design Pattern Selection
- Q & A

Confidential - Oracle Restricted

Custom Business Processes Examples

Real World Examples from CX, SCM and HCM Implementations

SaaS Data Manipulation Process Examples

- Data Manipulation Examples
  - Custom post clone HCM anonymization process
  - Enrich subscriptions automatically created from Order Management
  - Update Install Base following Service Logistics transactions
  - Auto create deliveries
  - Release order holds/pauses based on custom criteria
  - Upload sales orders from an excel file
  - Upload file of different formats – import customer product usage
- Data Extract Examples
  - Run a report based on an input file
- Event Driven Examples
  - Populate additional item defaults upon creation
  - Notify relevant business users with sales order status change

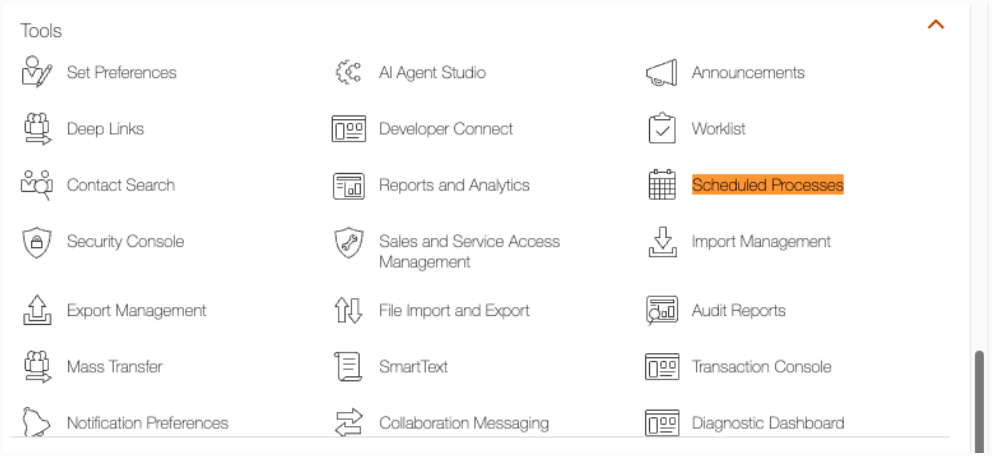
Confidential - Oracle Restricted

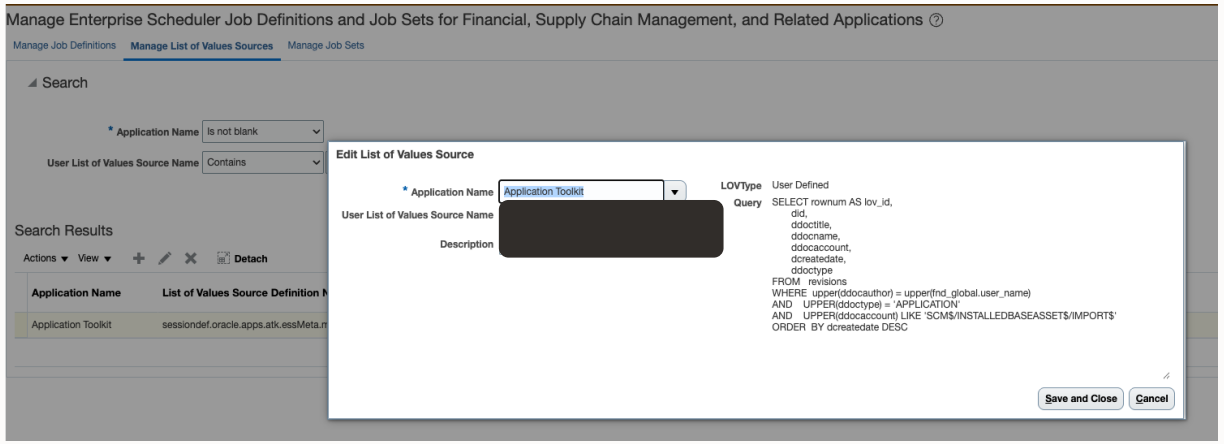
End User Interaction

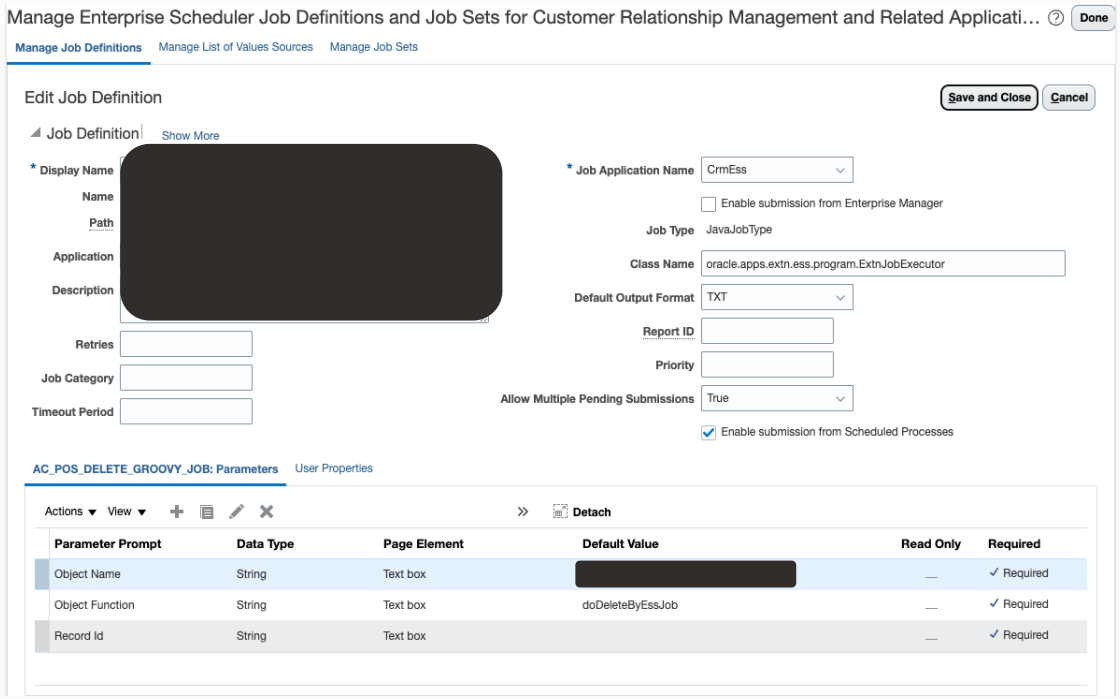
Expose Custom Business Processes To Users

1. End User Interaction – ESS Framework Utilization

- Expose in standard Schedule Processes Menu
- Use Checkbox/Date/Text/Number/LOV Parameters (without dependent SQL)
- Ability to connect to Report or Custom Groovy Function
- Schedule, Monitor Submission for Asynchronous Jobs (*)
- Schedule, Monitor Submission, Execution and Erros for Synchronous Jobs
- View Output for Synchronous/Groovy Jobs (manual screen refreshes)
- Per ESS Job Role in Security Console





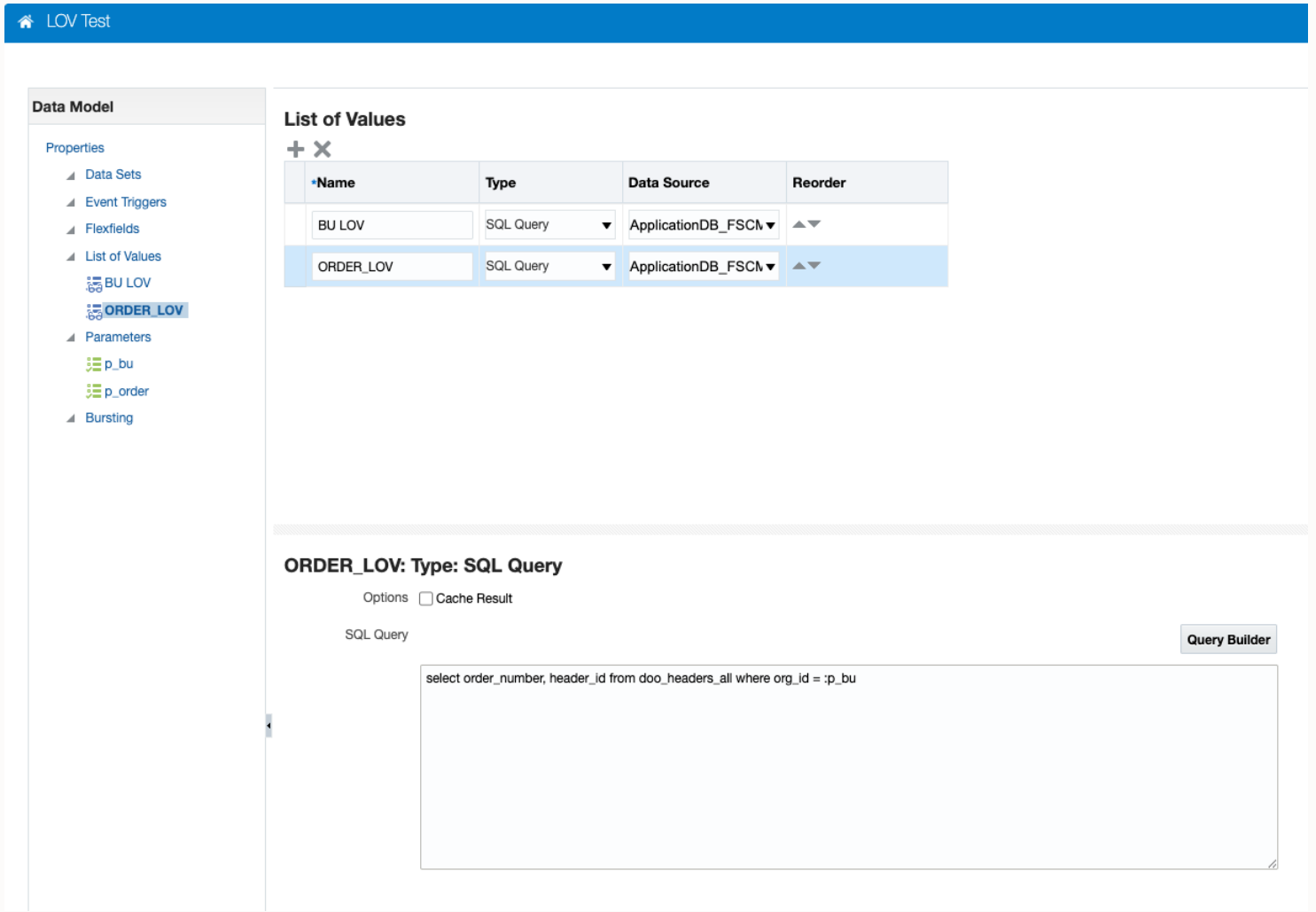


Oracle ACE

Confidential - Oracle Restricted

2. End User Interaction – BI Catalog Report Parameter Page

- Standard bookmarkable report link
- BI Publusher parameters utilization (including LOV dependency)
- Suitable for submission of async OIC calls
- Supports synchronous OIC calls
- Supports scheduled submissions
- BI Publisher template layout utilization (render Excel, HTML based on OIC results)
- BI Publisher report/folder security

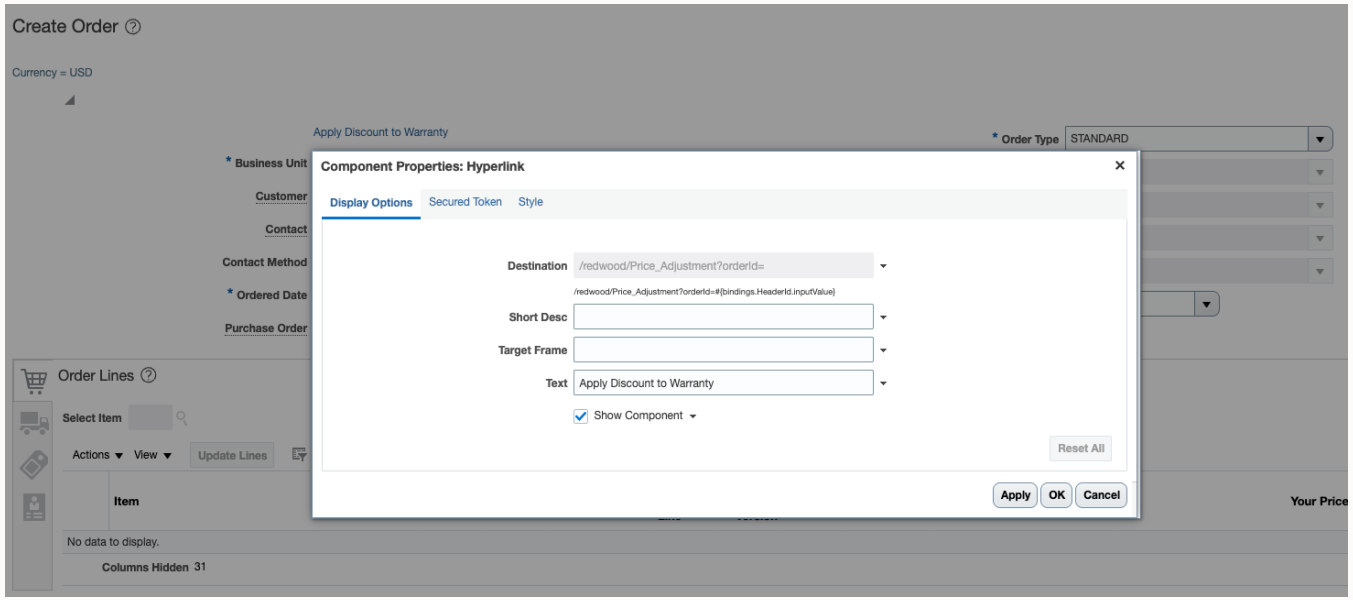


Oracle ACE

Confidential - Oracle Restricted

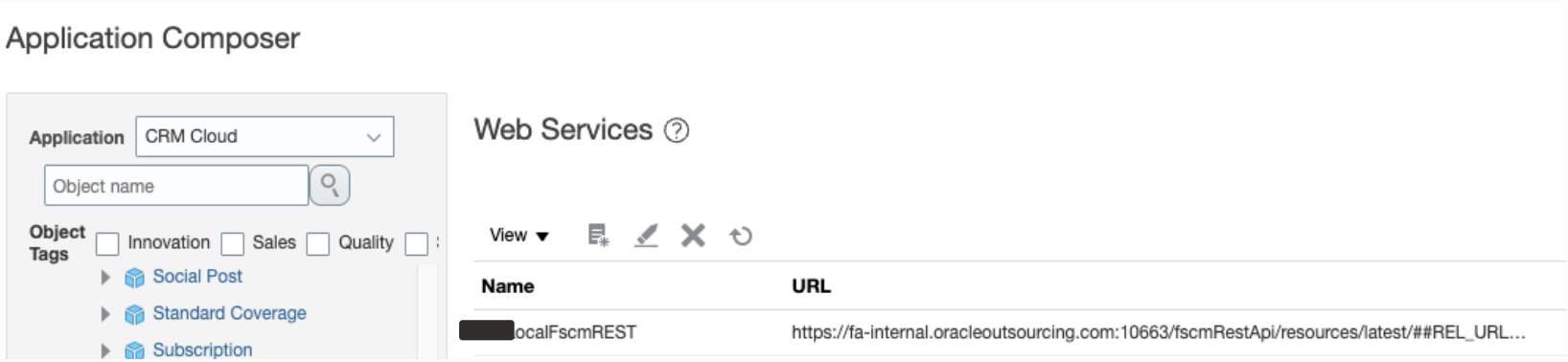
3. End User Interaction – Classic UI Page Composer Links

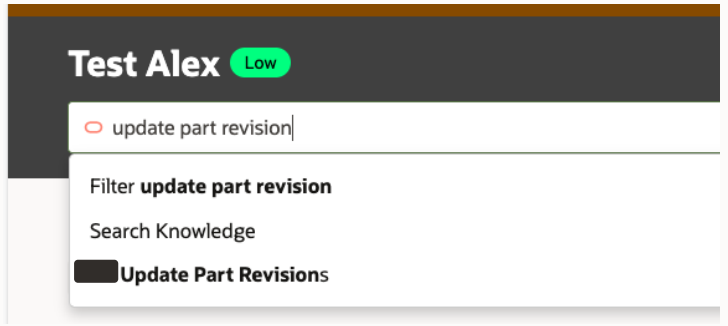
- Setup through Page Composer Sandbox
- Allows current record context passing
- Navigate to BI Publisher report (including report submitting an OIC)
- Navigate to Visual Builder AppUI/Redwood URL
- Keep URL definition P2T safe (//analytics, //xmlpserver)



4. End User Interaction – Application Composer Actions/CX Smart Actions

- Setup through Application Composer Sandbox
- Allows current record context passing
- Navigate to BI Publisher report (including report submitting an OIC)
- Navigate to Visual Builder AppUI/Redwood URL
- Keep URL definition P2T safe (//analytics, //xmlpserver)
- Navigate to OIC (Webservice Definition) (can be made P2T safe with OCI API Gateway)

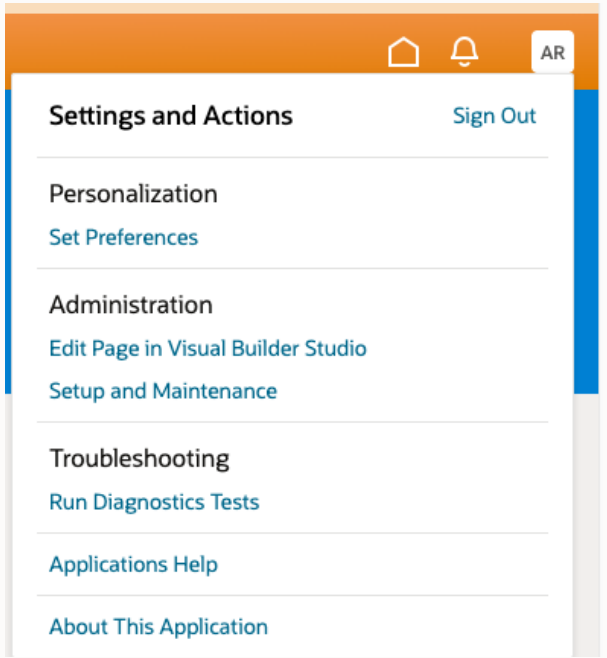


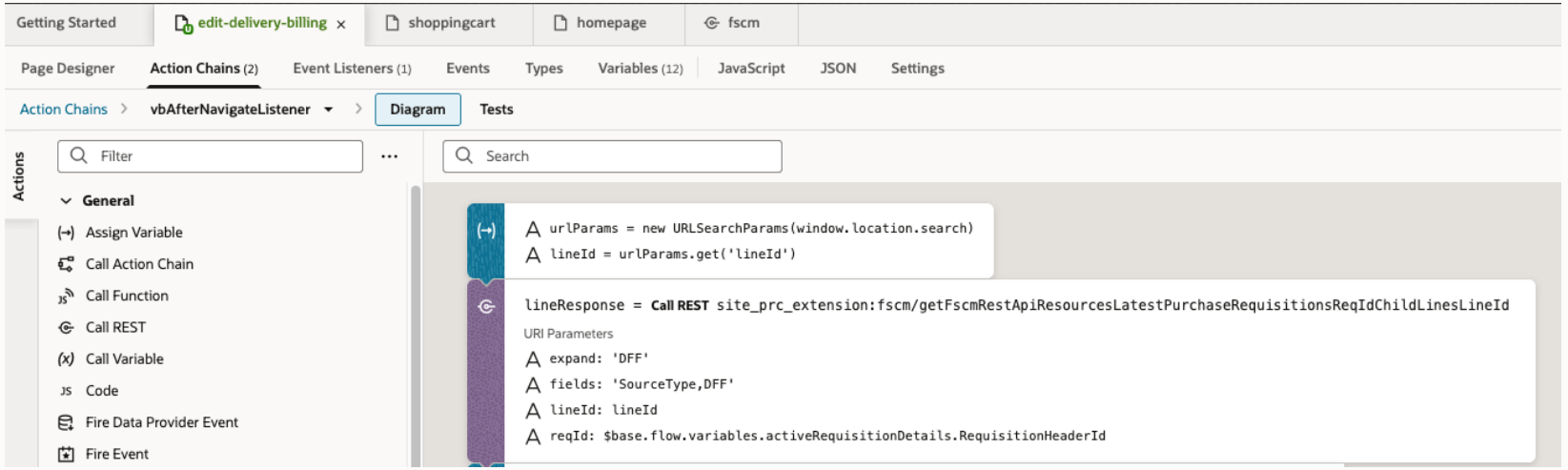


Oracle ACE

5. End User Interaction – Visual Builder Studio Extensions

- Setup through Edit Page in Visual Builder Studio
- Regular link addition is possible only in Oracle defined components (new section in Edit Purchase Requisition Delivery Details)
- Allows current record context passing (*)
- Navigate to BI Publisher report (including report submitting an OIC) (*)
- Navigate to Visual Builder AppUI/Redwood URL
- Keep URL definition P2T safe (//analytics, //xmlpserver)
- Advanced solutions – utilization of Lifecycle Events



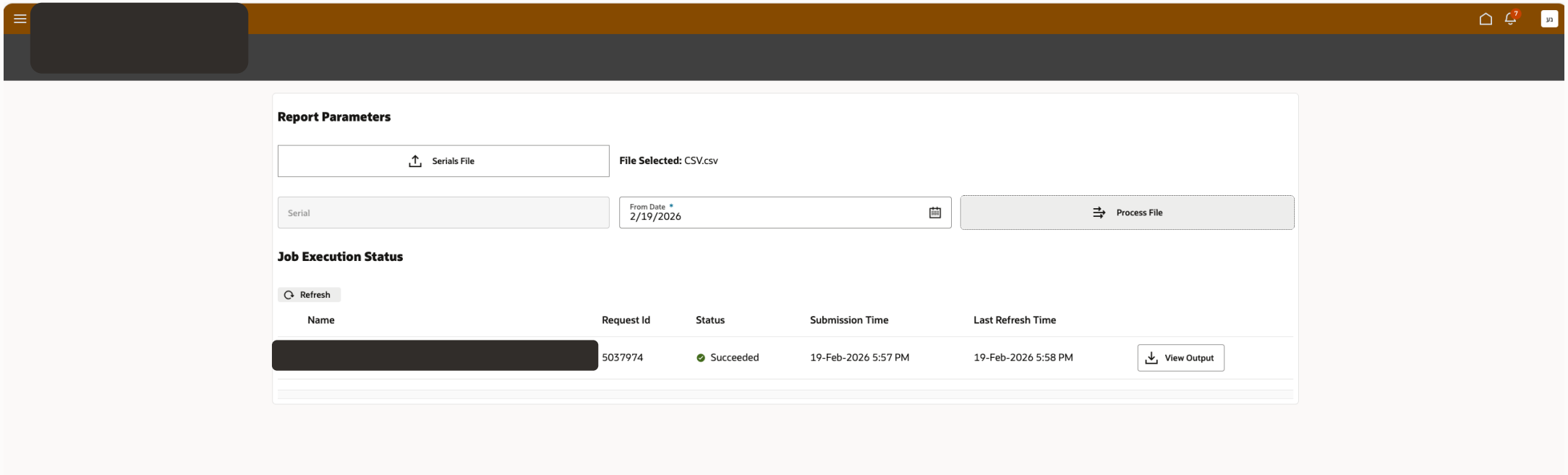


Oracle ACE

Confidential - Oracle Restricted

6. End User Interaction – Dedicated Visual Builder Studio AppUI Screens

- Requires VB expertise in addition to BIP

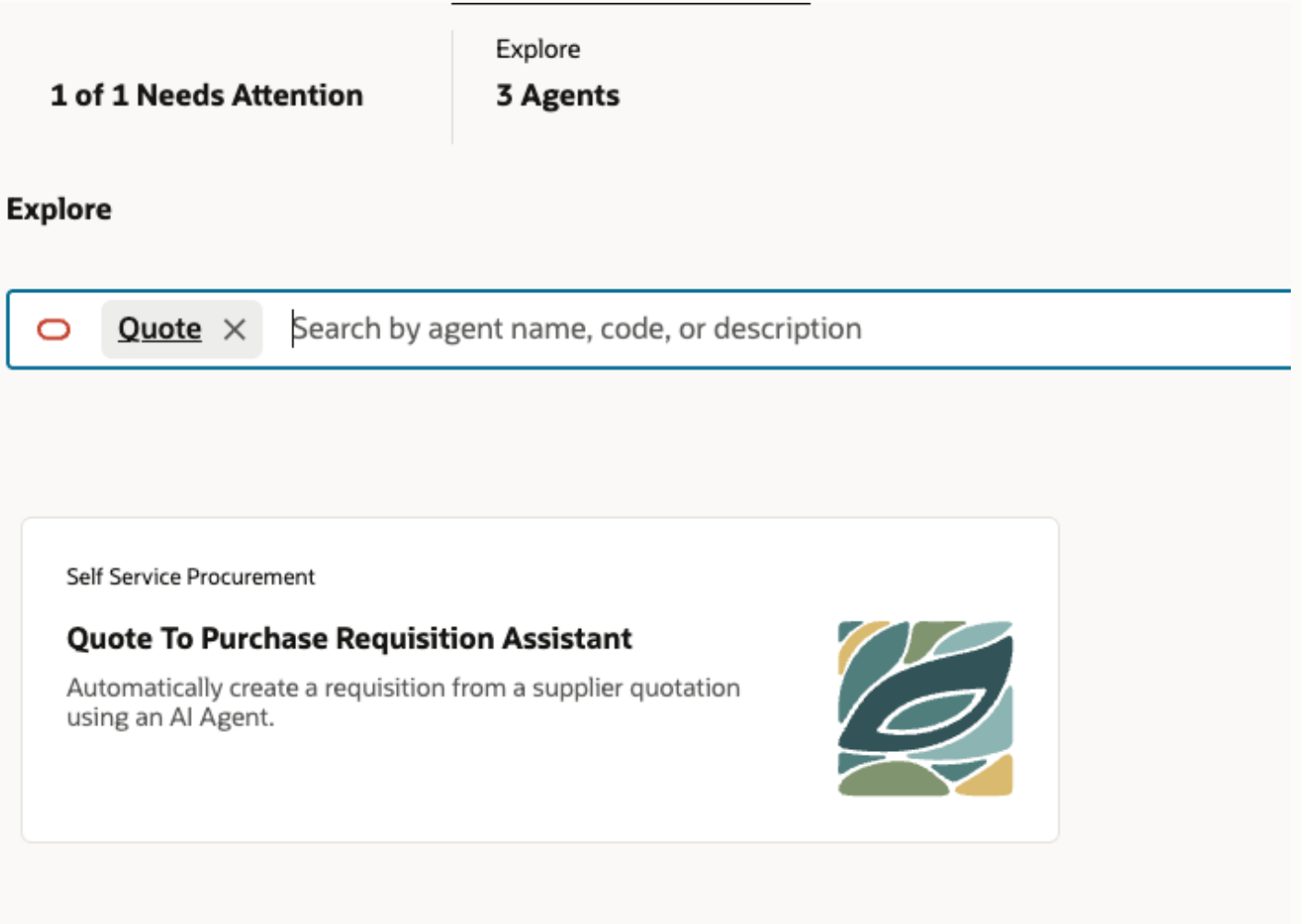


Oracle ACE

Confidential - Oracle Restricted

7. End User Interaction – AI Agent Exposure

- Role based AI Agents launch screen /fscmUI/redwood/human-resources/ai-studio/agent-explore
- Guided Journeys
- Smart Actions (CX)



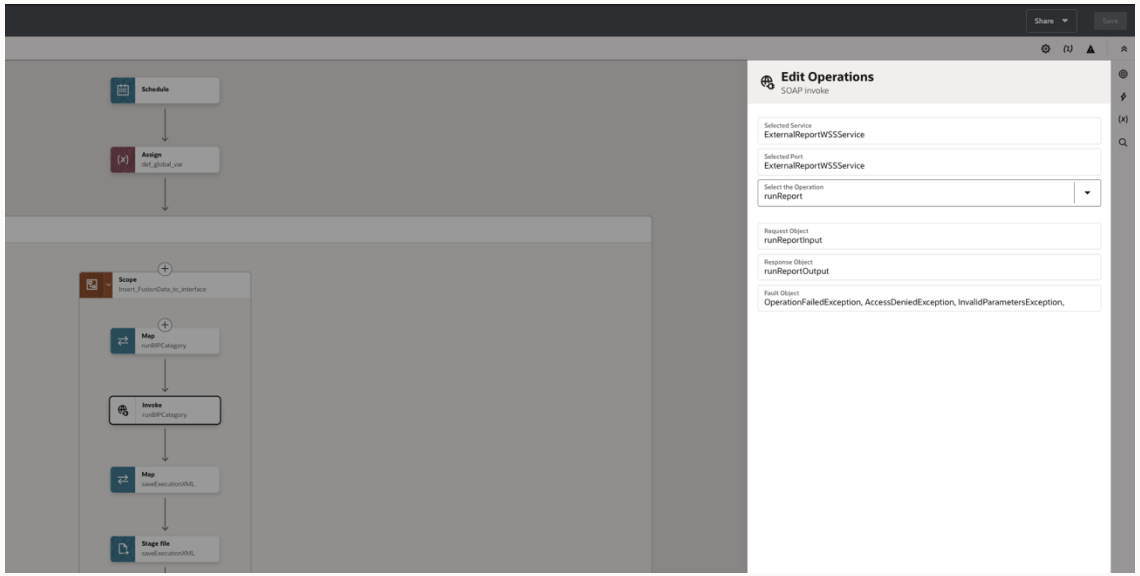
Oracle ACE

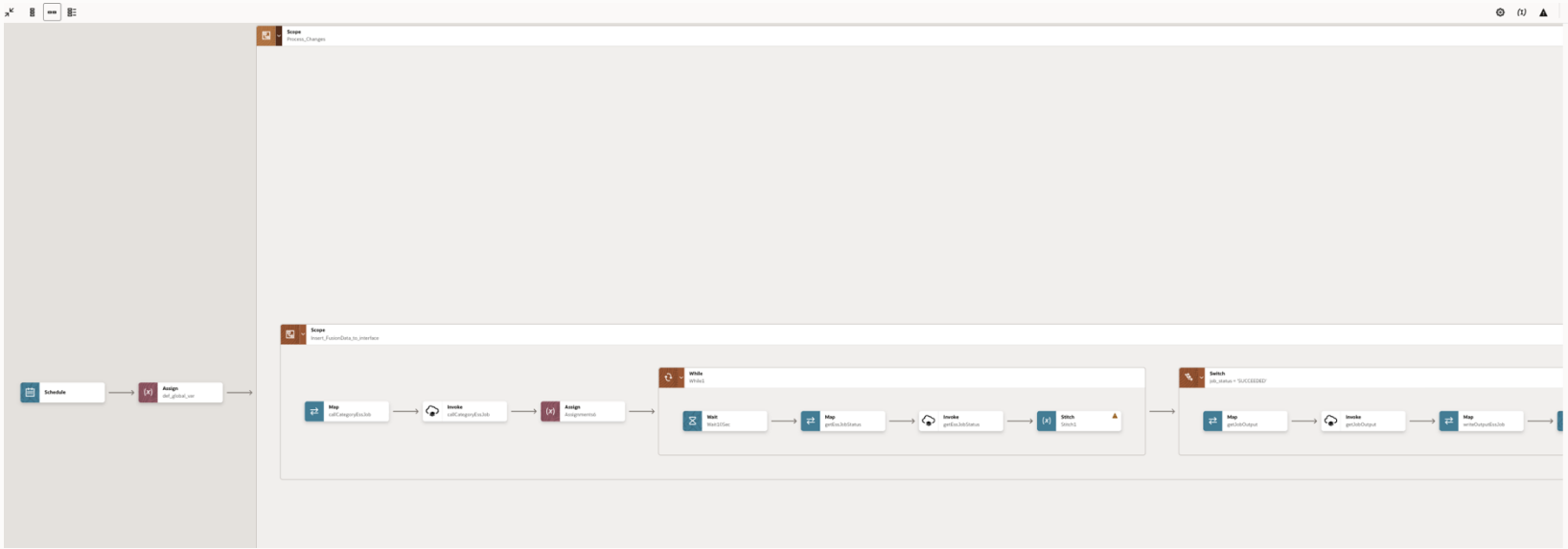
Covered Design Patterns for Custom Business Processes in FA

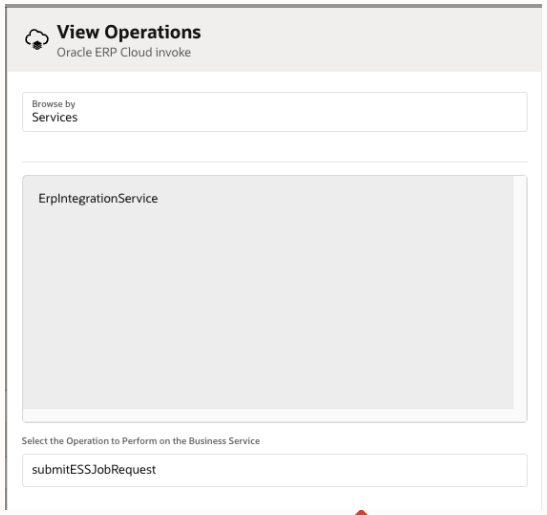
Cloud native design patterns for custom business logic implementation in FA

1. Covered Design Patterns for Custom Business Processes in FA – BI Publisher as Data Fetch and UI Templating Tool

- Identify fetched data volume and query complexity
- Prefer ESS submission vs. online SOAP BIP call
  - Execution timeout
  - Result output size
  - Impact on online FA users
- ESS + Wait vs. Enable Events = Y
  - Single OIC Process
  - Submitter + Callback



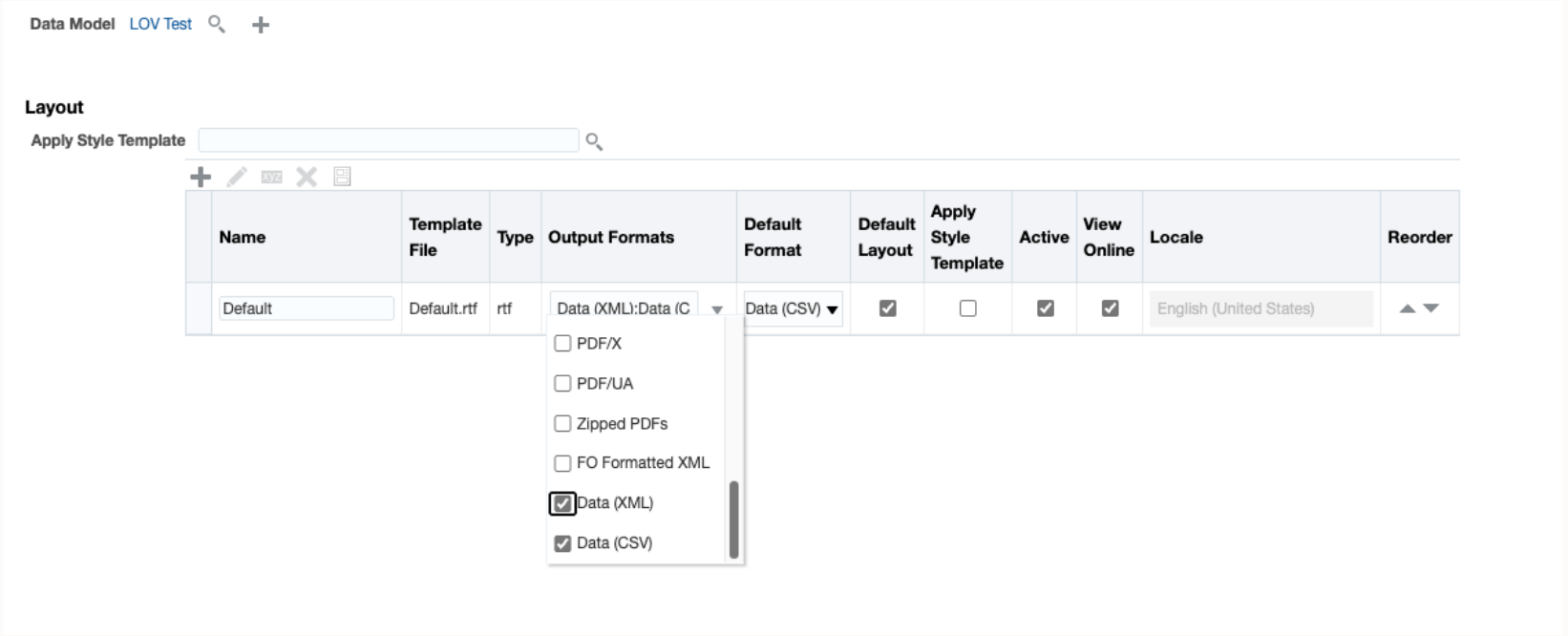


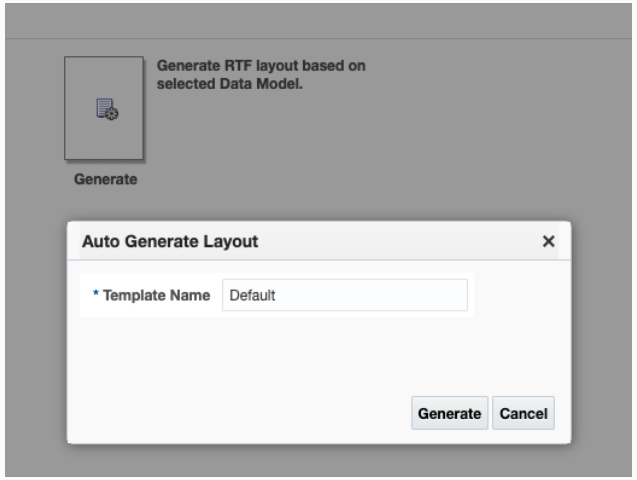


Oracle ACE

1. Covered Design Patterns for Custom Business Processes in FA – BI Publisher as Data Fetch and UI Templating Tool – Cont'd

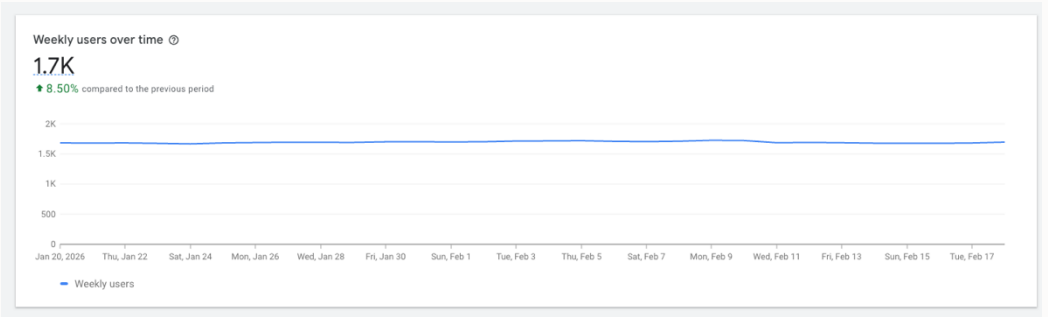
- Use Data (XML) and Data (CSV) to skip layout calculation
- Utilize lexical variables to generate dynamic SQL
  (example: &P_QUERY_TEXT – useful for Planning modules, custom object access)
- Use Explain Plan (including AI)

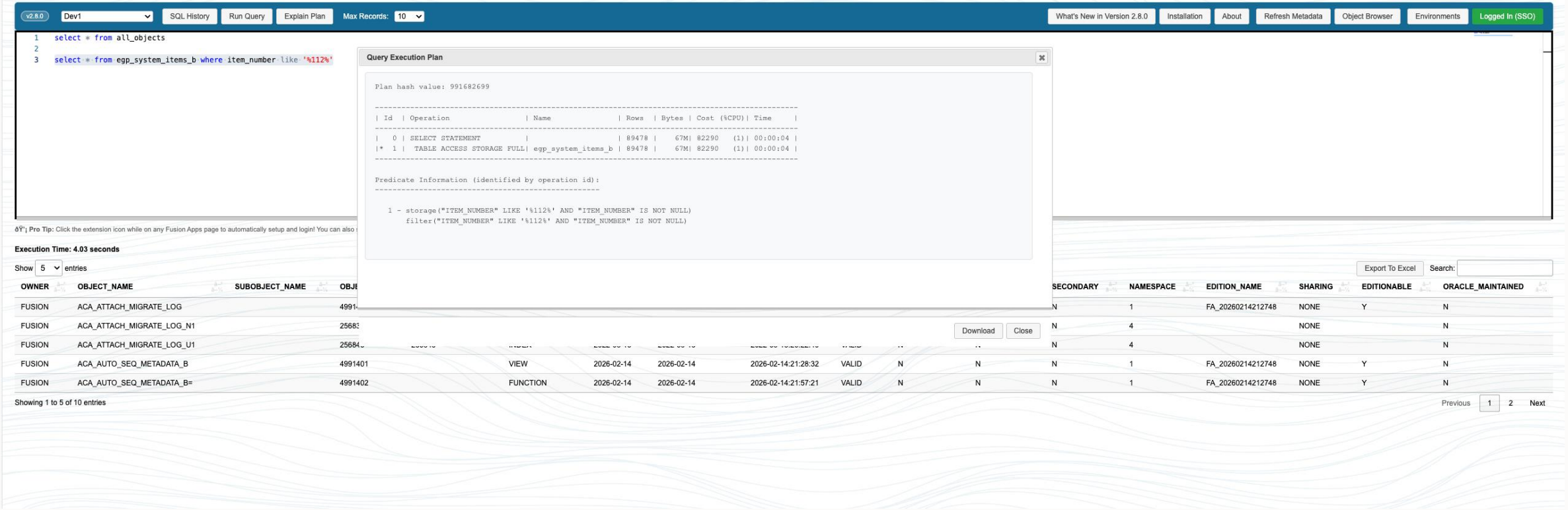




1. Covered Design Patterns for Custom Business Processes in FA – BI Publisher as Data Fetch and UI Templating Tool – Cont'd

- Use free/paid tools for easy SQL access
  - Free "CID Fusion Query" chrome extension
  - Recent addition – support for 320k query text



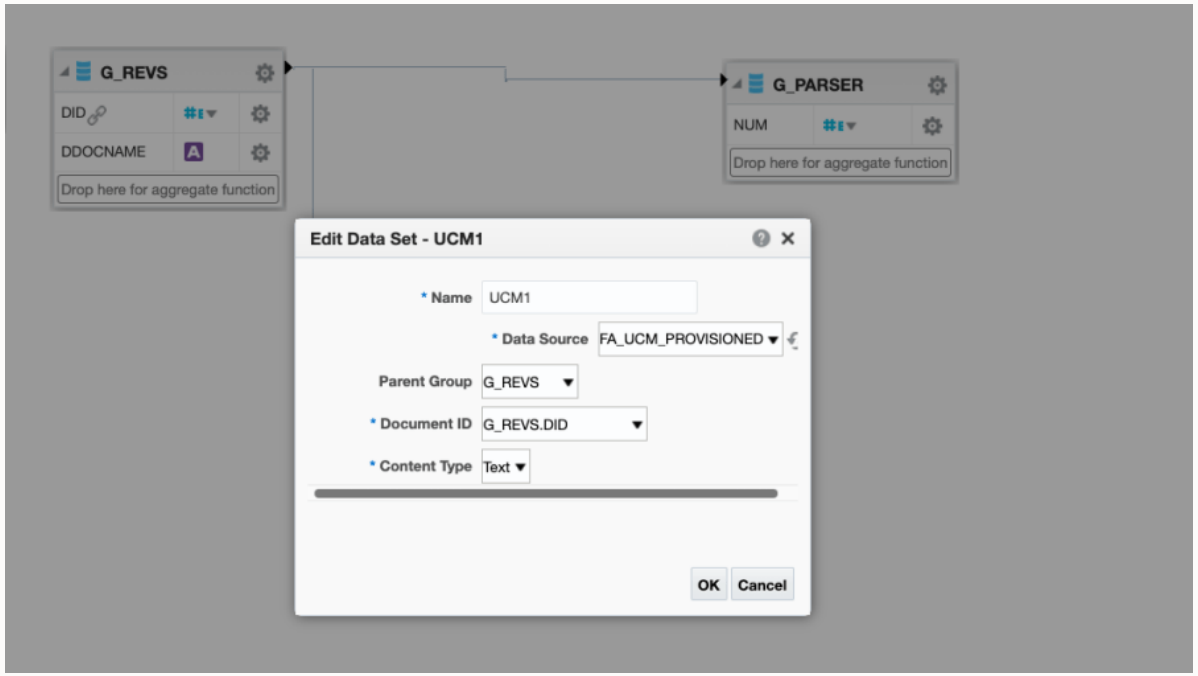


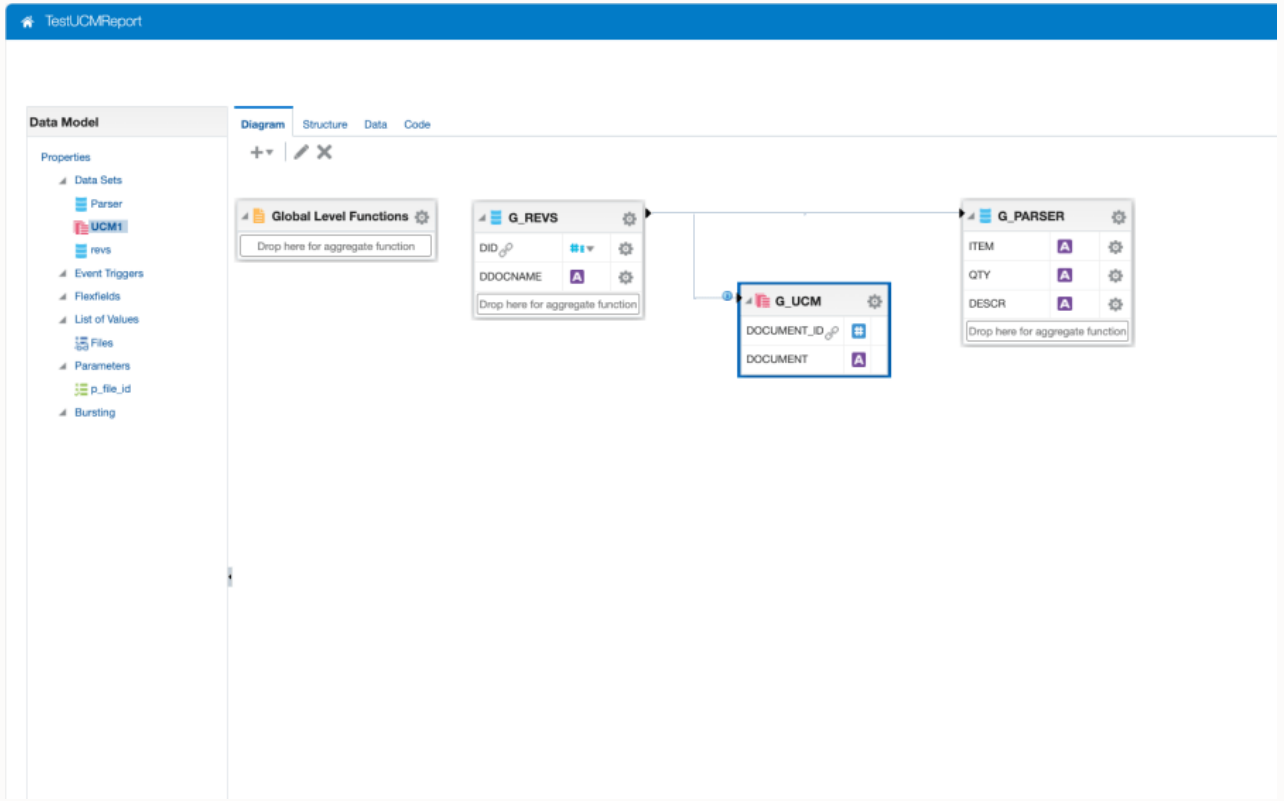
Oracle ACE

1. Covered Design Patterns for Custom Business Processes in FA – BI Publisher as Data Fetch and UI Templating Tool – Cont'd

- Access UCM text documents by UCM connection
- Access Query Example:

```
with rws as (select :DOCUMENT str from dual)
select REGEXP_SUBSTR(column_value, '[^,]+', 1, 1) AS item,
      REGEXP_SUBSTR(column_value, '[^,]+', 1, 2) AS qty,
      REGEXP_SUBSTR(column_value, '[^,]+', 1, 3) AS desc
from  (select trim(regexp_substr(str, '[^' || chr(10) || ']+', 1,level)) column_value
      from rws
      connect by level <= length(str) - length(replace(str, chr(10))) + 1
      ) file_data
```





Oracle ACE

Confidential - Oracle Restricted

1. Covered Design Patterns for Custom Business Processes in FA – BI Publisher as Data Fetch and UI Templating Tool – Cont'd

- Utilize Oracle's database powerful XML and JSON functions together with CLOB fields in Custom Objects
- Combine fully qualified JSON payload generated by DB with OIC binary input

```
WITH
 FileUploadRequest_sql as (
        select 'select ' || listagg(c.column_name || ' ' || cu.attribute_name, ',') ||
               ' from '  || max(t.table_name) ||
               ' where attribute_category = ''FileUploadRequest_c'' and id=' || :p_upload_rec_id code
        from   adf_extensible_table_usage tu,
               adf_extension_column_usage cu,
               adf_extension_column       c,
               adf_extensible_table       t
        where  tu.context_column_value = 'FileUploadRequest_c'
               and cu.table_usage_id = tu.usage_id
               and cu.column_id = c.column_id
               and c.table_id = t.table_id
    ),
    FileUploadRequest as (
        select c.*
        from   FileUploadRequest_sql s,
               xmltable(
                 '/ROWSET/ROW' passing dbms_xmlgen.getxmltype(s.code)
                columns
                    FileContent_c clob
               ) c
    ),
fileContent as (
    select FileContent_c from FileUploadRequest
),
```

```
),
getHeader as (
select t.SourceOrderNumber OrderNumber, any_value(t.Action) Action,
       JSON_OBJECT(
        'SourceTransactionNumber'   value t.SourceOrderNumber,
        'TransactionalCurrencyCode'     value any_value(t.CurrencyCode),
        'SourceTransactionId'       value any_value(id.header_id+rownum),
        'SourceTransactionSystem'   value 'OPS',
        'BusinessUnitName'          value any_value(t.BusinessUnit),
        'RequestingBusinessUnitId'  value any_value(ou.organization_id),
        'BuyingPartyName'           value any_value(t.Customer),
        'BuyingPartyNumber'         value any_value(t.Customernumber),
        'SubmittedFlag'             value decode(any_value(t.SOstatus),'Draft','false','true'),
        'DemandClass'               value any_value(t.DemandClass),
        'billToCustomer'            value
          JSON_ARRAY(
            JSON_OBJECT(
              'PartyName'     value any_value(t.BilltoCustomer),
              'AccountNumber' value any_value(t.Billtoaccount)
              returning clob
            ) returning clob
          ),
        'shipToCustomer'            value
          JSON_ARRAY(
            JSON_OBJECT(
              'PartyName'     value any_value(t.ShiptoCustomer),
              'Address1'      value any_value(t.Shiptoaddress)
              returning clob
            ) returning clob
          ),
        'TransactionType'           value any_value(t.OrderType),
        'SalesChannel'              value any_value(t.SalesChannel),
        'Salesperson'               value any_value(t.PrimarySalesPerson),
        'CustomerPONumber'          value any_value(t.PurchaseOrder),
 'additionalInformation'     value
          JSON_ARRAY(
            JSON_OBJECT(
              'Category'              value 'DOO_HEADERS_ADD_INFO',
              'HeaderEffBCustomer__NotesprivateVO'  value
                JSON_ARRAY(
                  JSON_OBJECT(
                    'ContextCode'       value 'Customer Notes',
                    'deliveryComments1' value any_value(t.DeliveryComments1),
                    'deliveryComments2' value any_value(t.DeliveryComments2)
                    returning clob
```


Oracle ACE

Confidential - Oracle Restricted

1. Covered Design Patterns for Custom Business Processes in FA – BI Publisher as Data Fetch and UI Templating Tool – Cont'd

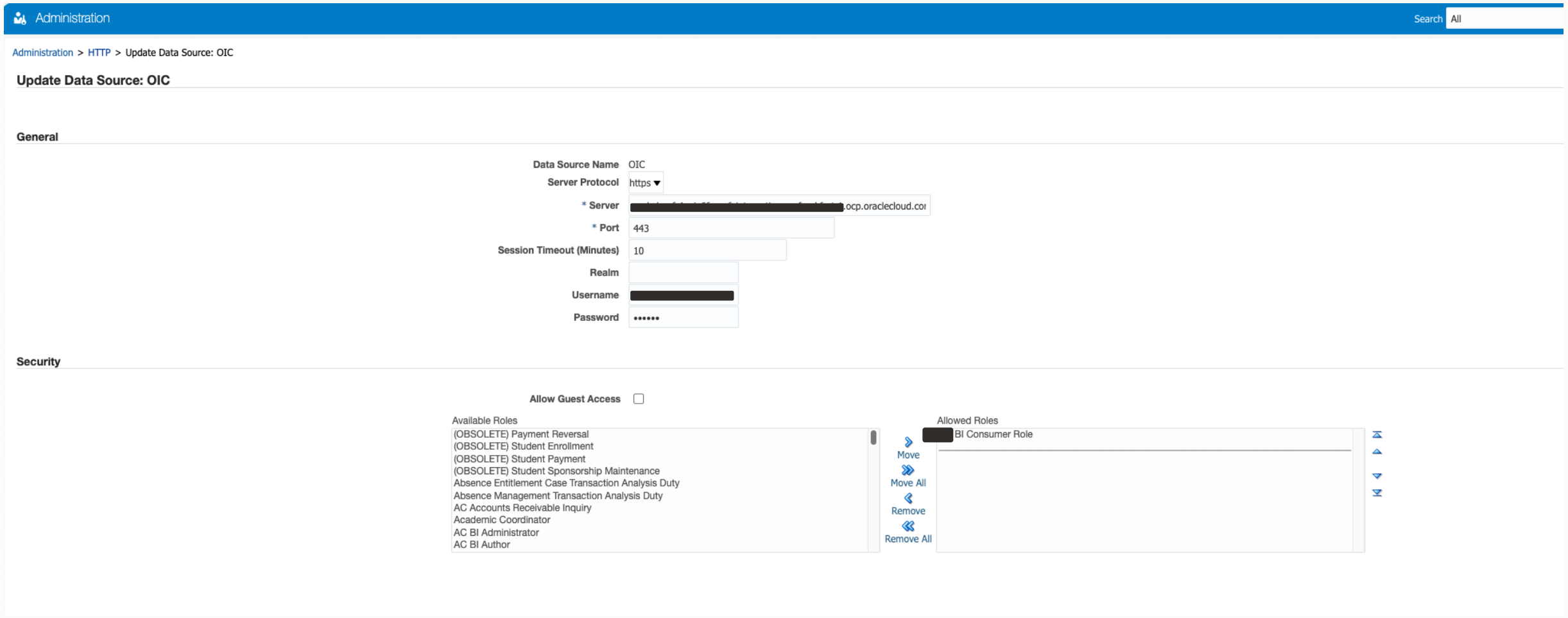
- Utilization of layout templates:
  - Documents generation (PDF, HTML, Excel)
    (access to Excel/HTML can be done by Scheduling Service SOAP by BIP Job ID)
    Example can be found in the blog article
  - XSL-TEXT + eText for structured text files (integrations, label printing, etc.)
- Bursting options
  - Email (use carefully after P2T)
  - SFTP with/without encryption
  - UCM – can be combined with OIC processing
    (Run a report and attach its output to every order as an attachment)

2. Covered Design Patterns for Custom Business Processes in FA – ESS Job submitting Synchronous Processing in OIC

- Triggered by a dedicated connection in BIP data model
  - Utilize a generic HTTP connection to OIC (one place for P2T change)
  - Possibility of creating a SOAP connection to OIC (additional P2T change)
- Ability to map report parameters to OIC integration (can not be done from query)
- Ability to map **fusionapps_request_id** built in parameter passing current ESS Job Id
- Design a synchronous OIC integration (GET, POST with response, request/response SOAP)
- Pros
  - ESS appears running as long as OIC process is running
  - Resulting XML from HTTP/SOAP connection can be used in BIP template
  - Output/Errors are exposable in ESS's output
- Cons
  - 5 mins limit on sync OIC call
  - Not suitable for large volume processing
  - Last Update user it OIC service user

Confidential - Oracle Restricted

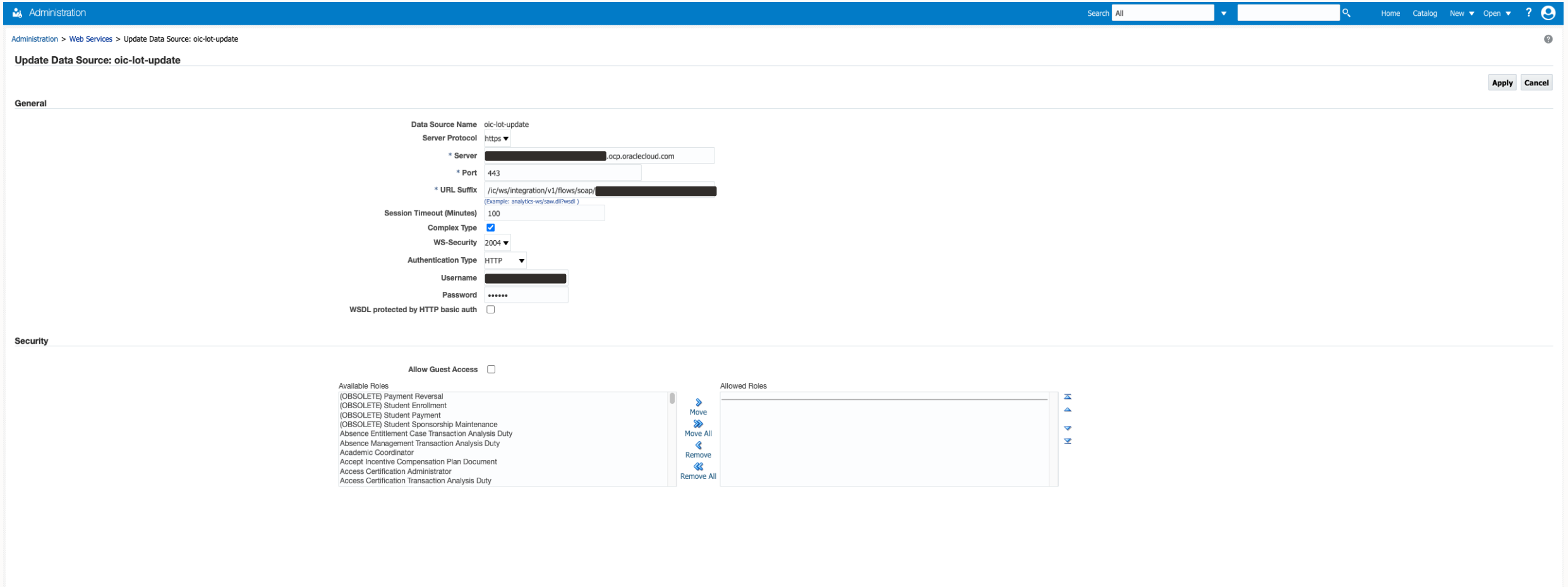
2. Covered Design Patterns for Custom Business Processes in FA – ESS Job submitting Synchronous Processing in OIC – Cont'd



Oracle ACE

Confidential - Oracle Restricted

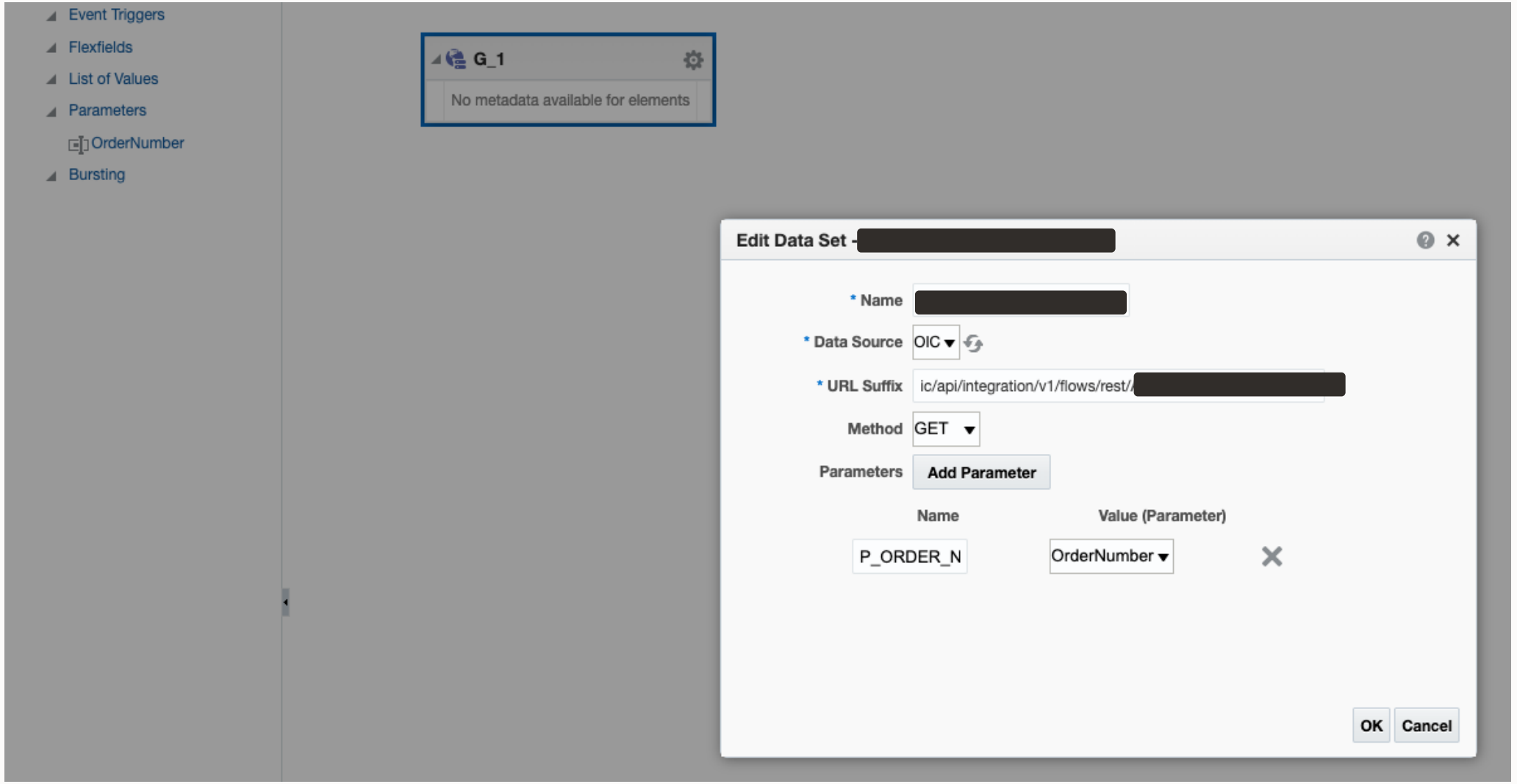
2. Covered Design Patterns for Custom Business Processes in FA – ESS Job submitting Synchronous Processing in OIC – Cont'd



Oracle ACE

Confidential - Oracle Restricted

2. Covered Design Patterns for Custom Business Processes in FA – ESS Job submitting Synchronous Processing in OIC – Cont'd



Oracle ACE

Confidential - Oracle Restricted

2. Covered Design Patterns for Custom Business Processes in FA – ESS Job submitting Synchronous Processing in OIC – Cont'd

- Small volume/single record processing pattern:
  - Submitting ESS Job
  - Caller BIP with HTTP connection, HTML/PDF output
  - Synchronous OIC returning XML in case of success or failure
  - Online REST APIs/BIP queries if needed

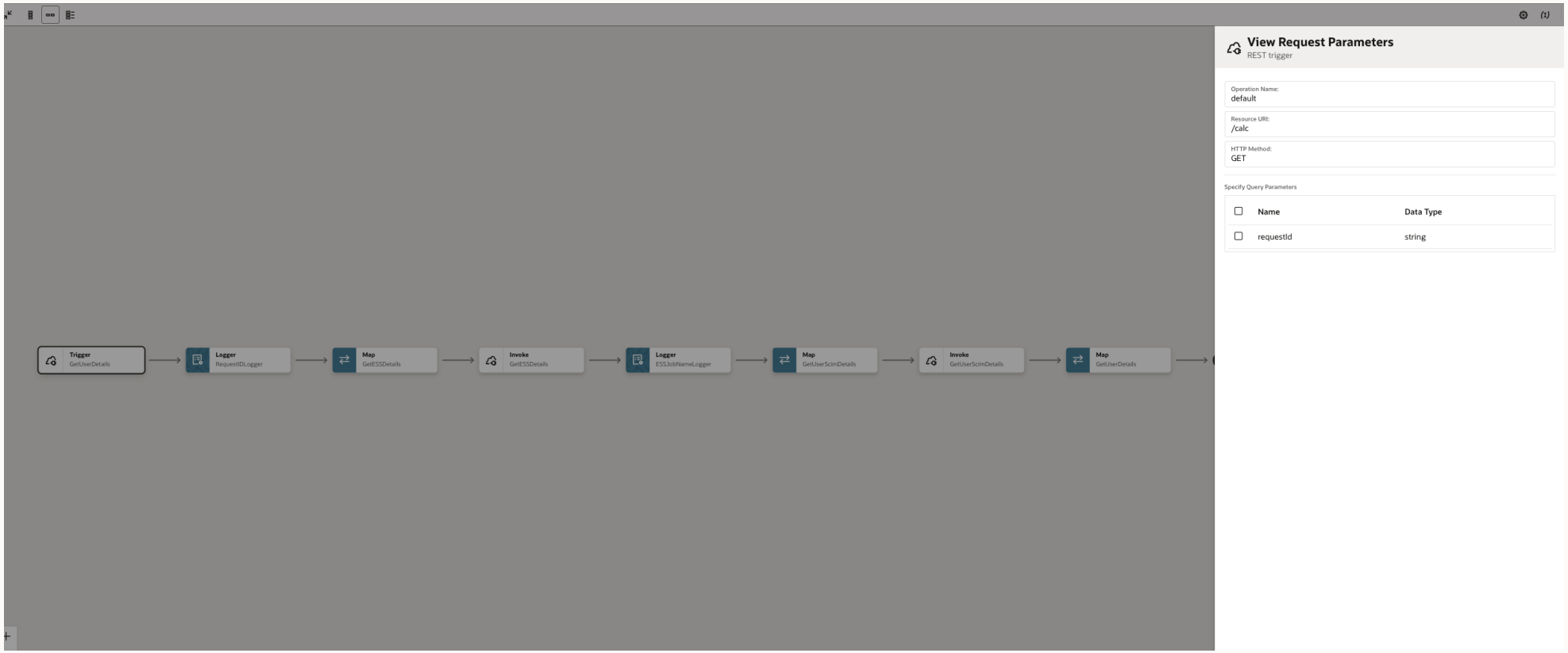
3. Covered Design Patterns for Custom Business Processes in FA – ESS Job submitting Asynchronous Processing in OIC

- ESS Job triggering a synchronous OIC integration by GET
- GET integration can return submitted integration id or an error
- Pass **fusionapps_request_id** – allows current user/email identification
- Design a caller synchronous OIC integration (same as before)
- Design an asynchronous OIC integration (POST without response, one-way SOAP, scheduled)
  - Scheduled integration is a preferred pattern (multiple submission will be placed in a queue)
- Pros:
  - Execution time is up to 6 hours
  - Utilization of full OIC capabilities
- Cons:
  - ESS completes immediately, no indication of progress, errors, feedback (possible to create SQL dataset with **DBMS_LOCK.sleep** to wait until OIC completion)
  - Last Update user it OIC service user
  - Outputs of BIPs executed as part of OIC are not visible by submitter user and should be emailed or attached to accessible business record

Confidential - Oracle Restricted

3. Covered Design Patterns for Custom Business Processes in FA – ESS Job submitting Asynchronous Processing in OIC – Cont'd

- Example of an OIC utility integration fetching user's email for **fusionapps_request_id**
- Utilize the following endpoints:
  - /ess/rest/scheduler/v1/requests/{requestId}/
  - /hcmRestApi/scim/Users



Oracle ACE

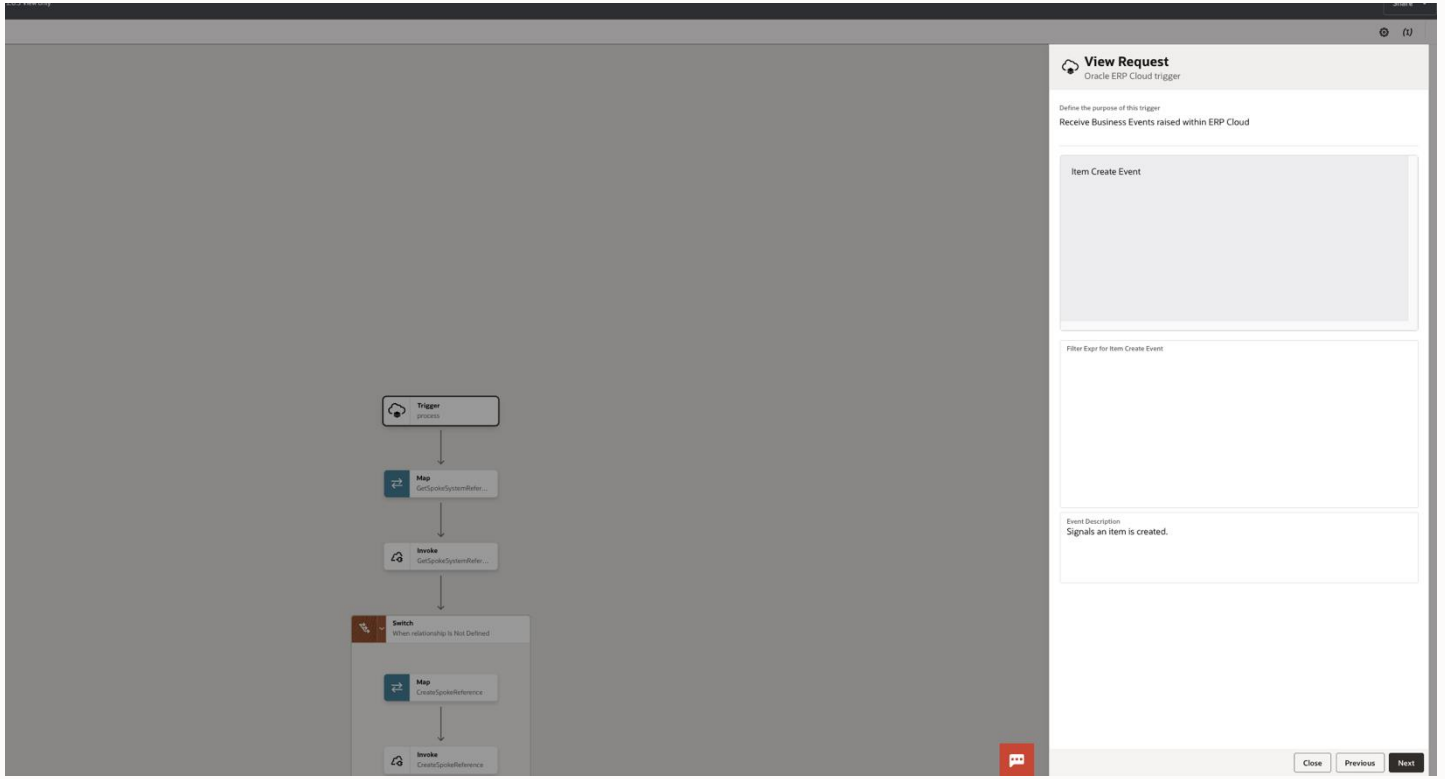
Confidential - Oracle Restricted

3. Covered Design Patterns for Custom Business Processes in FA – ESS Job submitting Asynchronous Processing in OIC – Cont'd

- Large volume data manipulation pattern:
  - Submitting ESS Job
  - Caller BIP with HTTP connection
  - Trigger OIC GET integration
  - Scheduled OIC processing integration
  - Data fetch by BIP processed by additional ESS (can be defined not user triggerable)
  - REST API/FBDI/HDL/Import Management processing
  - Securely expose generated data files to users
  - Error handling and user's progress/error notification
- Additional large volume processing example:
  - User uploads to sharepoint an excel file with sales order for import
  - ESS Job is submitted
  - File fetched from sharepoint, converted to JSON and stored in a custom object CLOB field
  - CLOB field is parsed by XMLTABLE/JSON_TABLE, records are processed
  - User notified/result files published to sharepoint

4. Covered Design Patterns for Custom Business Processes in FA – Standard/Custom Integration Event triggering an OIC Integration

- Use OIC to subscribe to pillar specific FA standard business events
- Events examples:
  - ESS/FBDI load completion
  - Item creation/update
  - Work Order change
  - Sales Order status update
- No impact on P2T process
- No SLA for for triggering time
- Event processing can be lost due to password change
- Backup process to sync unprocessed events

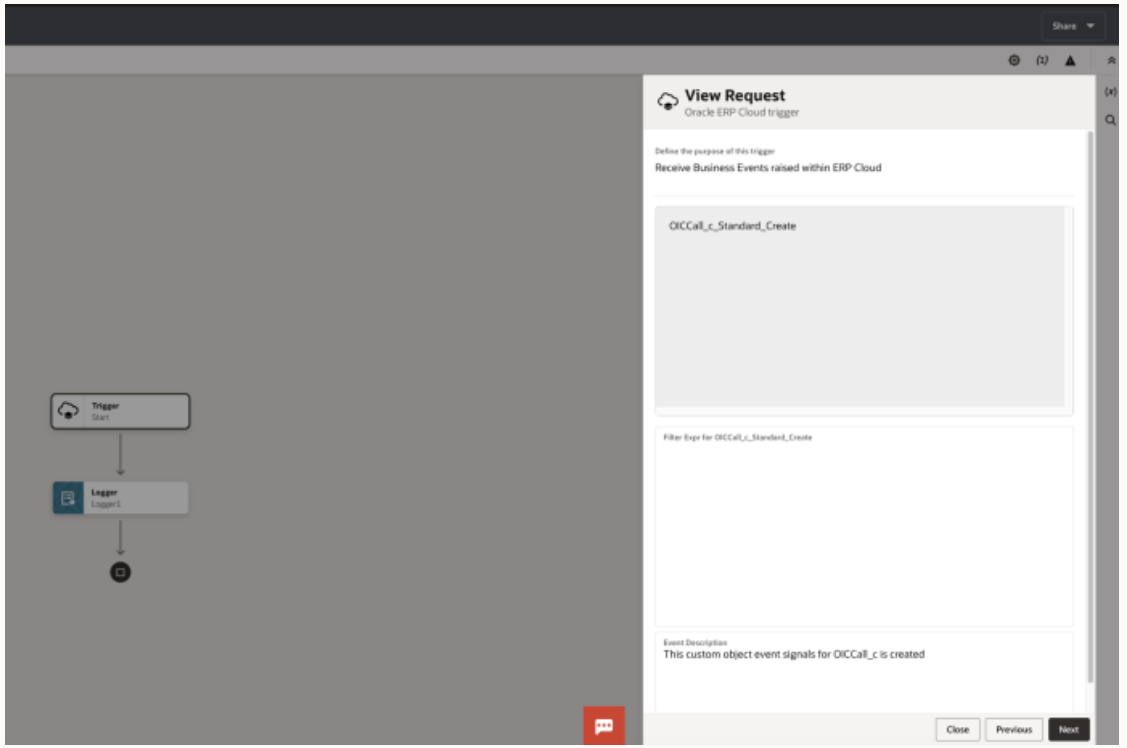


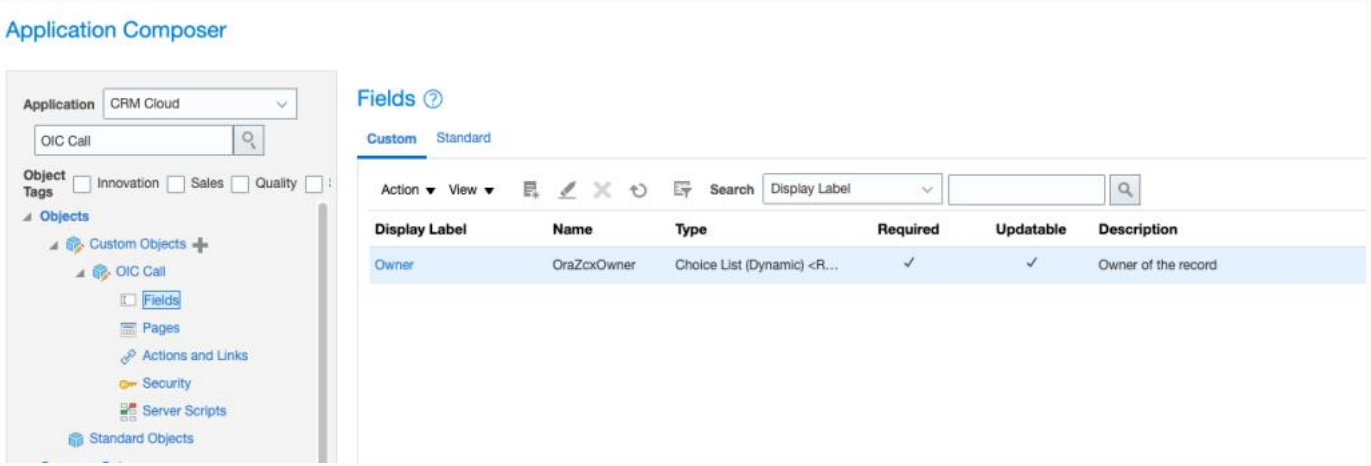
Oracle ACE

Confidential - Oracle Restricted

4. Covered Design Patterns for Custom Business Processes in FA – Standard/Custom Integration Event triggering an OIC Integration – Cont'd

- Use OIC to subscribe to FA custom business events
- Prerequisites:
  - Set Profile option **ZCX_CUSTOM_OBJECT_EVENTS** to **Yes** on site level
  - Navigate to **Application Composer > Common Setup** (outside of sandbox)
  - Right-click and select **Generate Integration Event**.
- Create/Use custom object in **Application Composer**
- Create an OIC Integration to Listen to CRUD Events





Oracle ACE

Confidential - Oracle Restricted

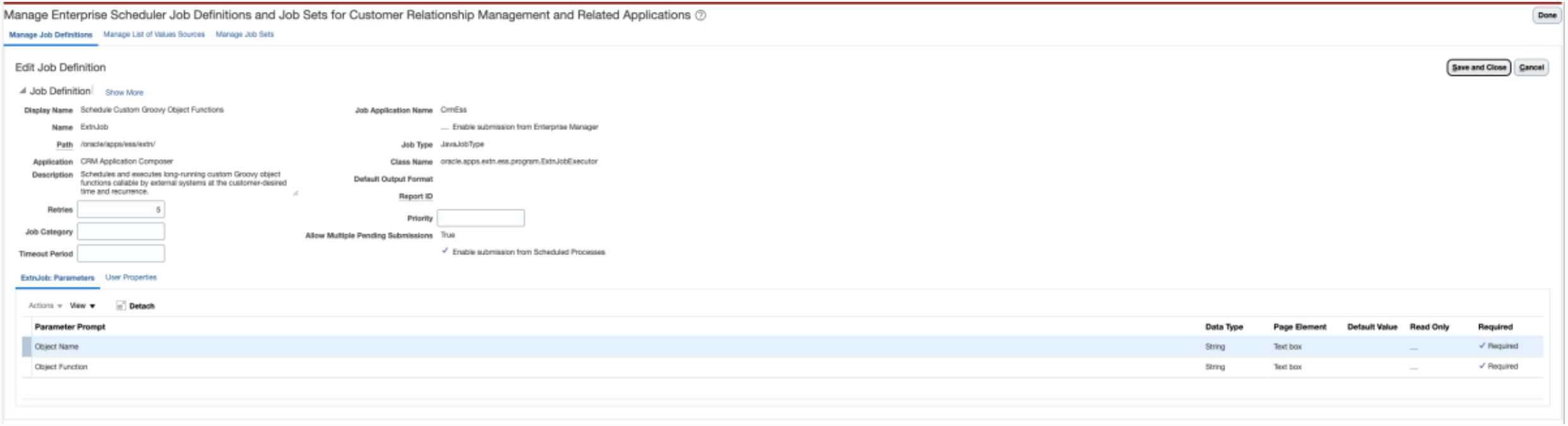
5. Covered Design Patterns for Custom Business Processes in FA – Groovy based ESS Job

- Expose as regular ESS
- Implement logic inside FA without need for OIC
- Useful for HCM implementation without OIC
- Limited to 30 mins execution time (more than triggers, workflows)
- Implementation in Groovy by using:
  - Custom Object Object Function
  - Based on official documentation the parameters for Scheduled Groovy are not supported
  - Identify ESS parameters by current session available in Groovy
  - BIP Reports for data fetch
  - Data manipulation by using Web Services defined against P2T safe URL https://fa-internal.oracleoutsourcing.com:10663
  - Text output for results and error display
- Submission user is used for DML, proper license and access required
- Real life example can be found in the blog article

Confidential - Oracle Restricted

5. Covered Design Patterns for Custom Business Processes in FA – Groovy based ESS Job – Cont'd

- Create a custom ESS job definition by cloning the standard **Schedule Custom Groovy Object Functions** job

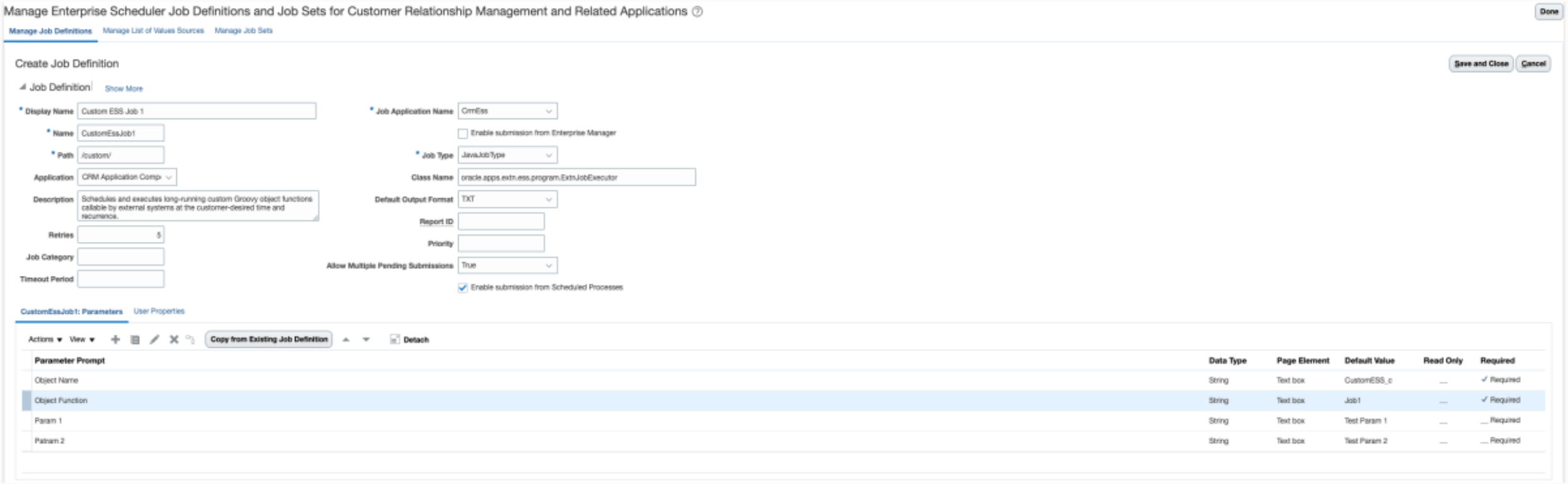


Oracle ACE

Confidential - Oracle Restricted

5. Covered Design Patterns for Custom Business Processes in FA – Groovy based ESS Job – Cont'd

- Add parameter fields **after the two required parameters:**

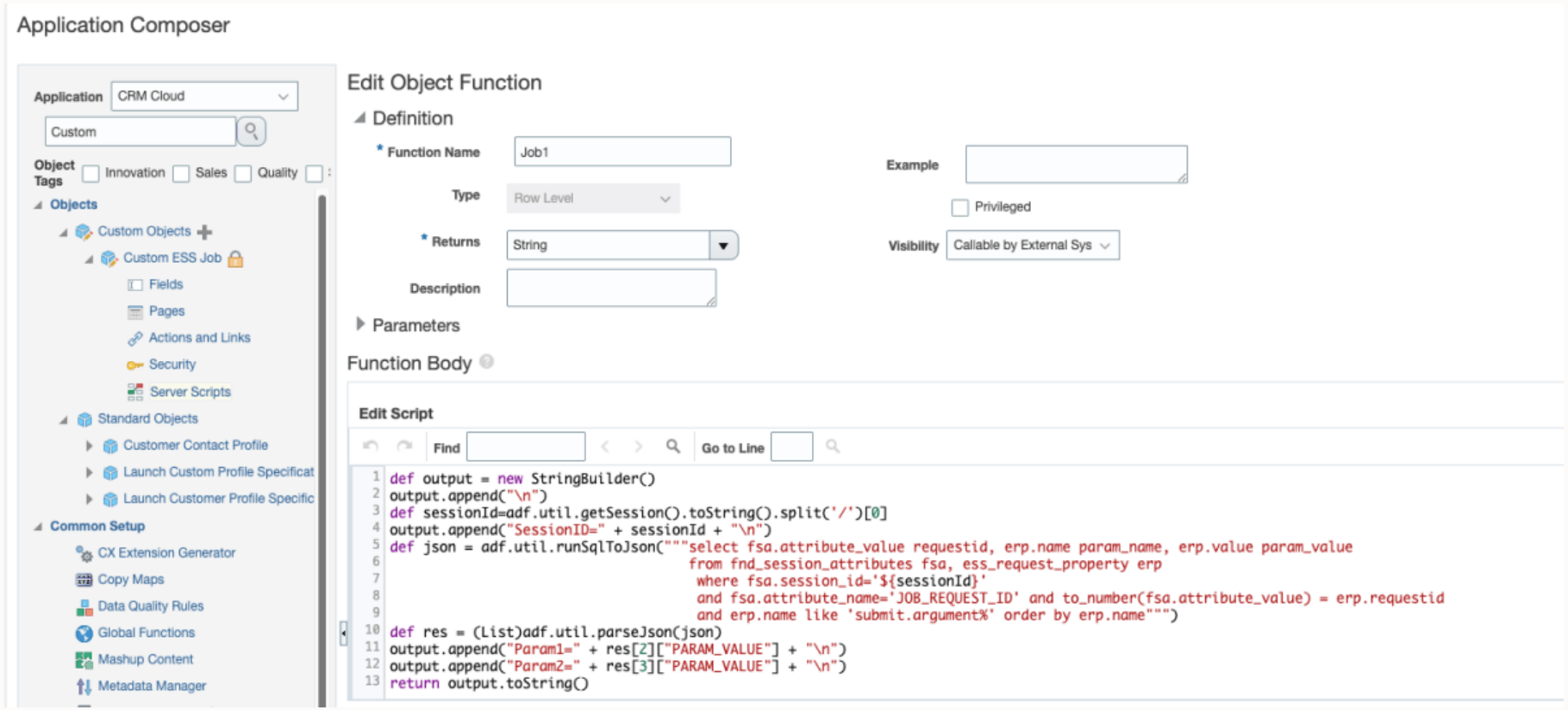


Oracle ACE

Confidential - Oracle Restricted

5. Covered Design Patterns for Custom Business Processes in FA – Groovy based ESS Job – Cont'd

- Use Groovy code to retrieve the current **sessionId**
- Build a BIP report that queries **FND_SESSION_ATTRIBUTES** and **ESS_REQUEST_PROPERTY** to fetch passed parameters
- Execute the report from Groovy and parse the parameters for use in logic

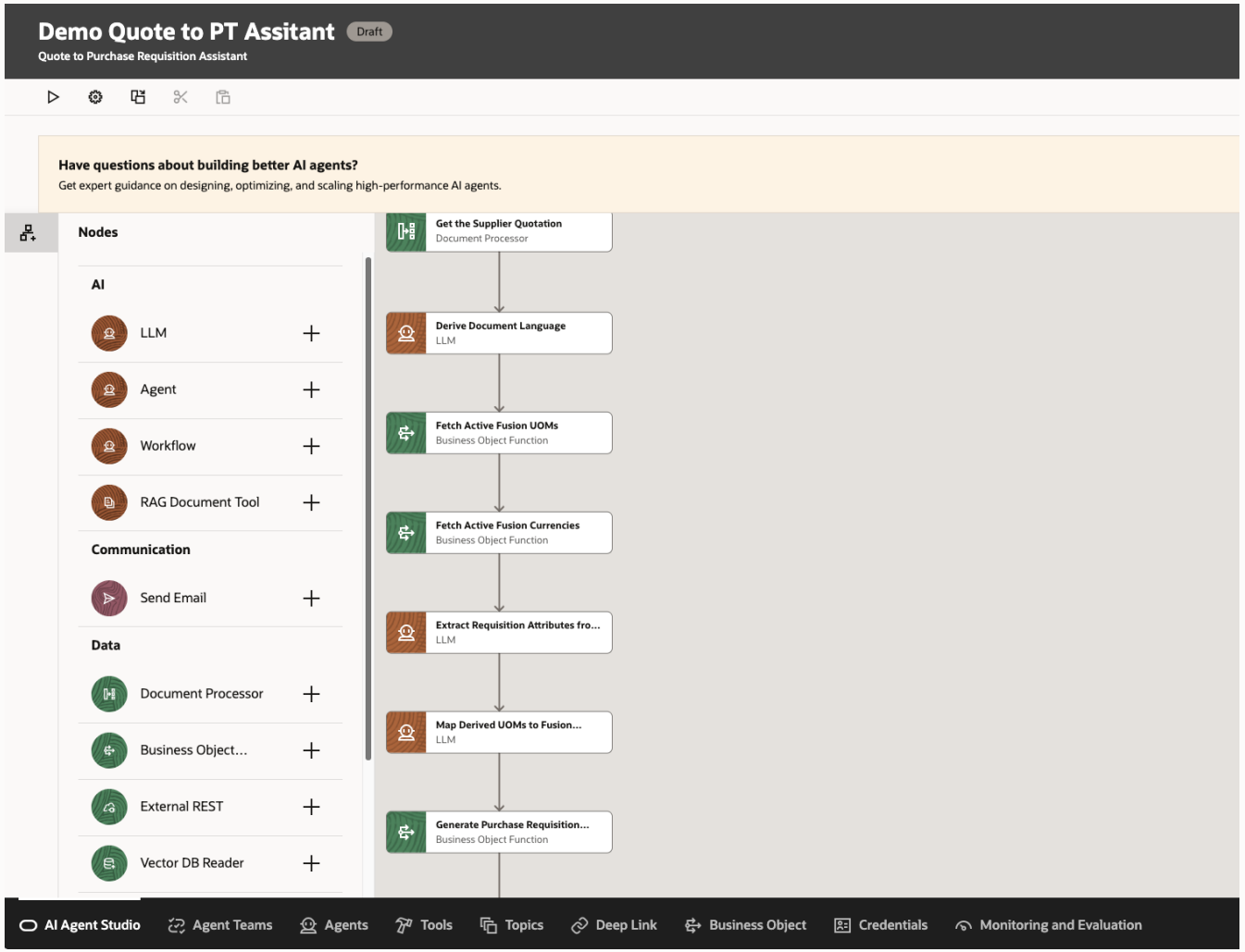


Oracle ACE

Confidential - Oracle Restricted

6. Covered Design Patterns for Custom Business Processes in FA – New: AI Agent Studio

- Start using AI Agent Studio
- Utilize and learn from pre-existing templates
- Use the power of LLM for user interaction, unstructured files processing
- Enable in non-prod to start exploration
- Start user adoption with out-of-the-box agents
- Proceed with custom Tools for advanced processes



Oracle ACE

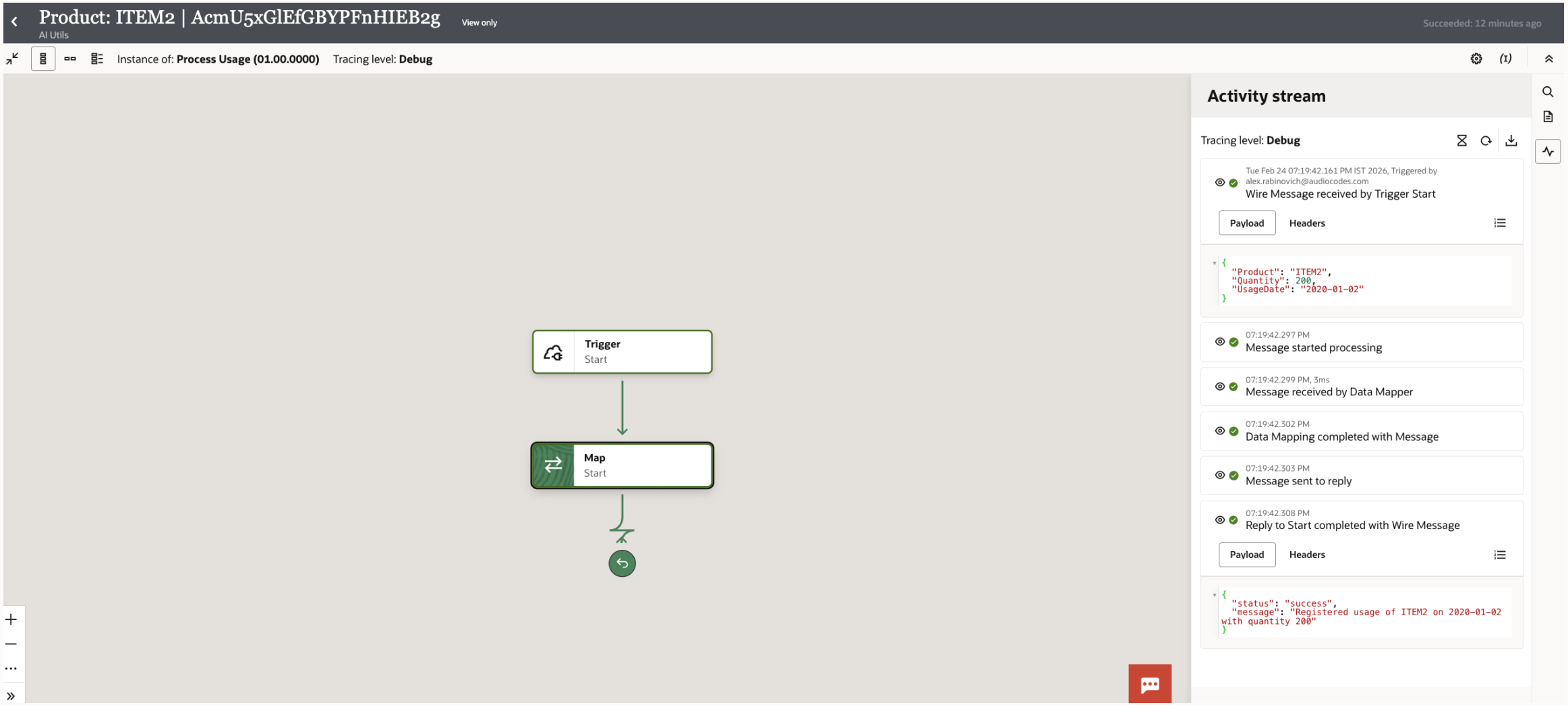
Confidential - Oracle Restricted

6. Covered Design Patterns for Custom Business Processes in FA – New: AI Agent Studio – Example

- Business Scenario:
  - User receives multiple formats of product usages from customers
  - Files could be Excel, CSV, PDF
  - Column names could be different
  - The column's order and location is not the same
  - Goal – extract from the files common data and upload it to billing system (exposed by OIC)
- Demo Solution:
  - OIC Integration receiving JSON with single usage payload and returns register status
  - AI Agent Studio – Custom Tool of External URL pointing to OIC integration is defined
  - AI Agent Studio – Custom Agent using pre-built MultiFileProcessor tool, prompt defined with the parsing logic
  - AI Agent Studio – Custom Agent Team of type Workflow using the above Custom Agent, record loops and External URL calls

Confidential - Oracle Restricted

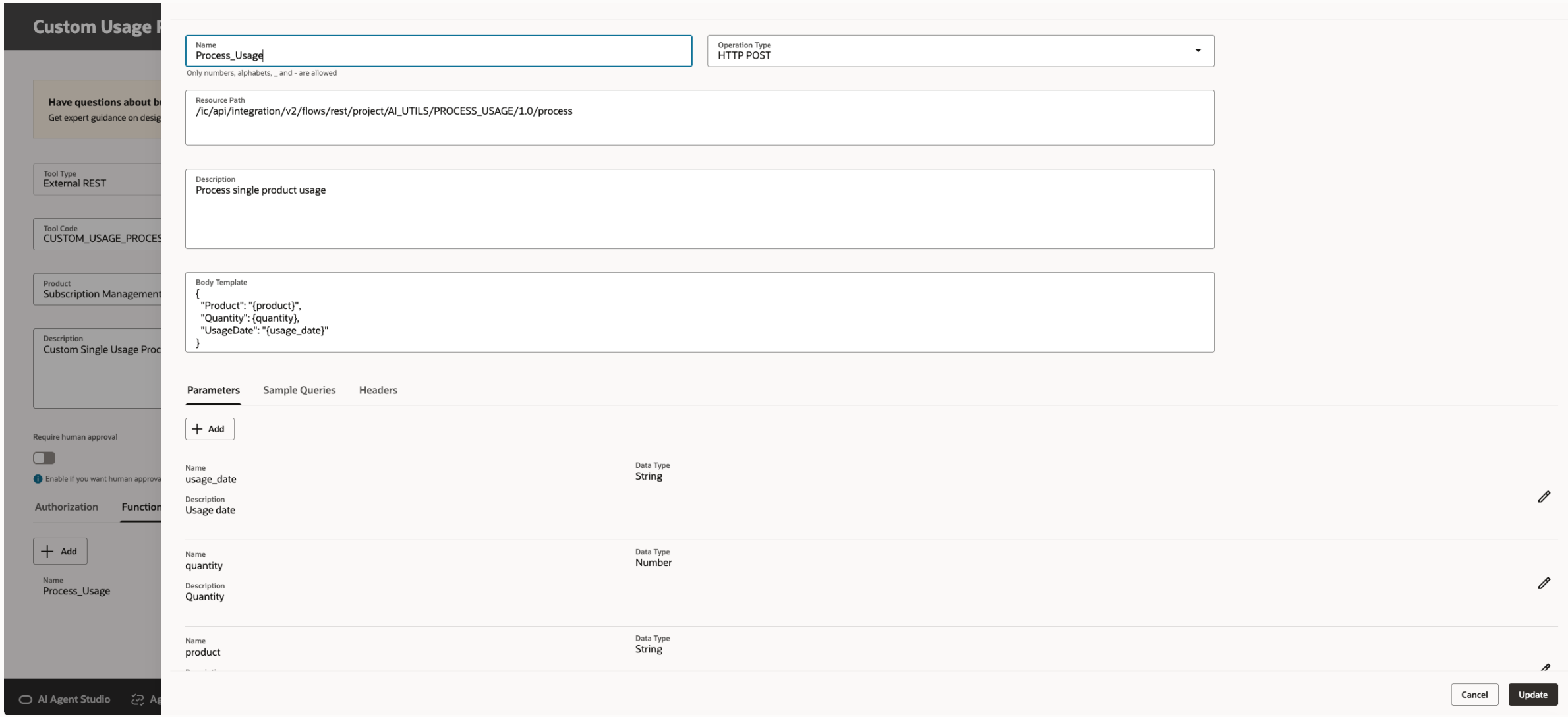
6. Covered Design Patterns for Custom Business Processes in FA – New: AI Agent Studio – Example – Cont'd



Oracle ACE

Confidential - Oracle Restricted

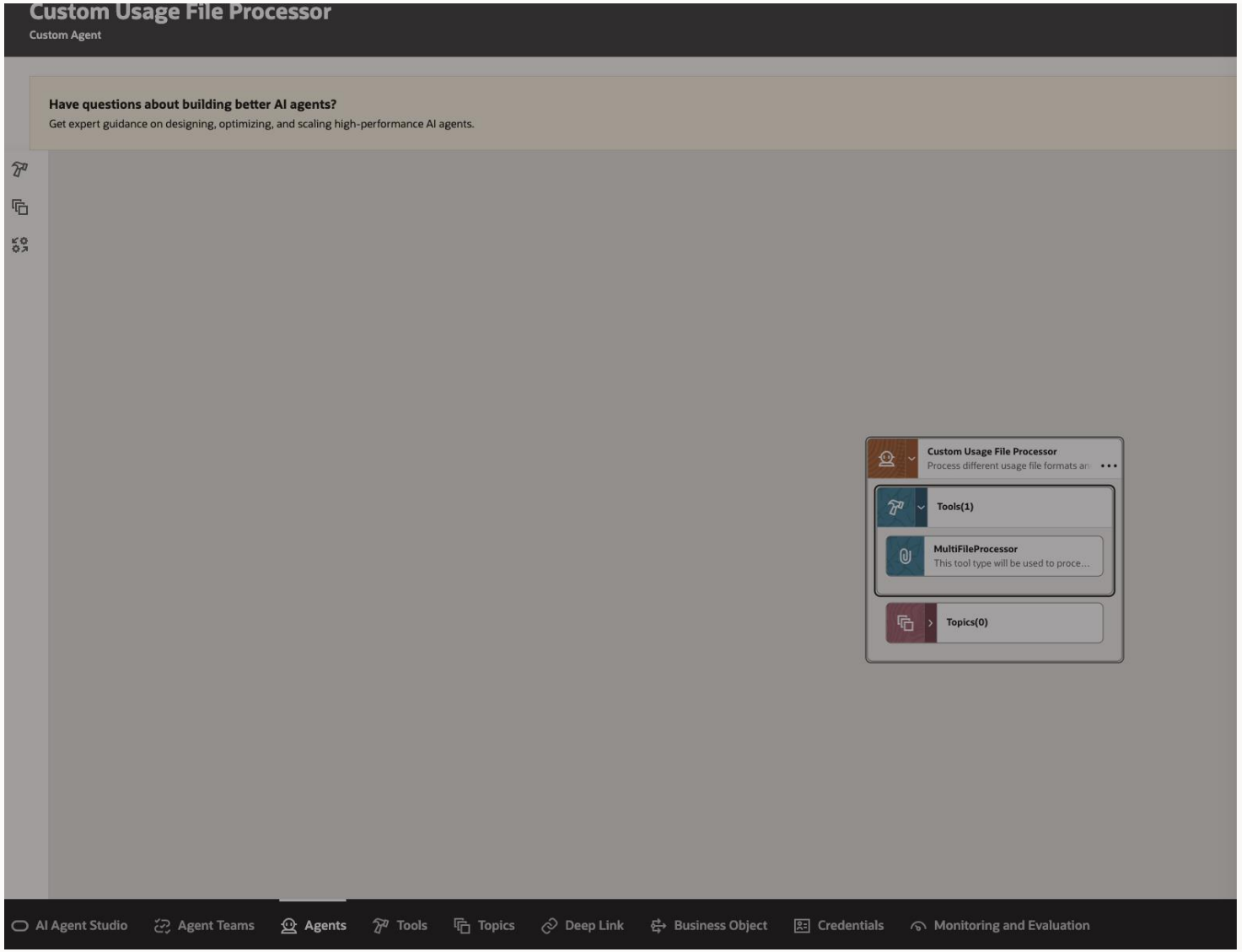
6. Covered Design Patterns for Custom Business Processes in FA – New: AI Agent Studio – Example – Cont'd

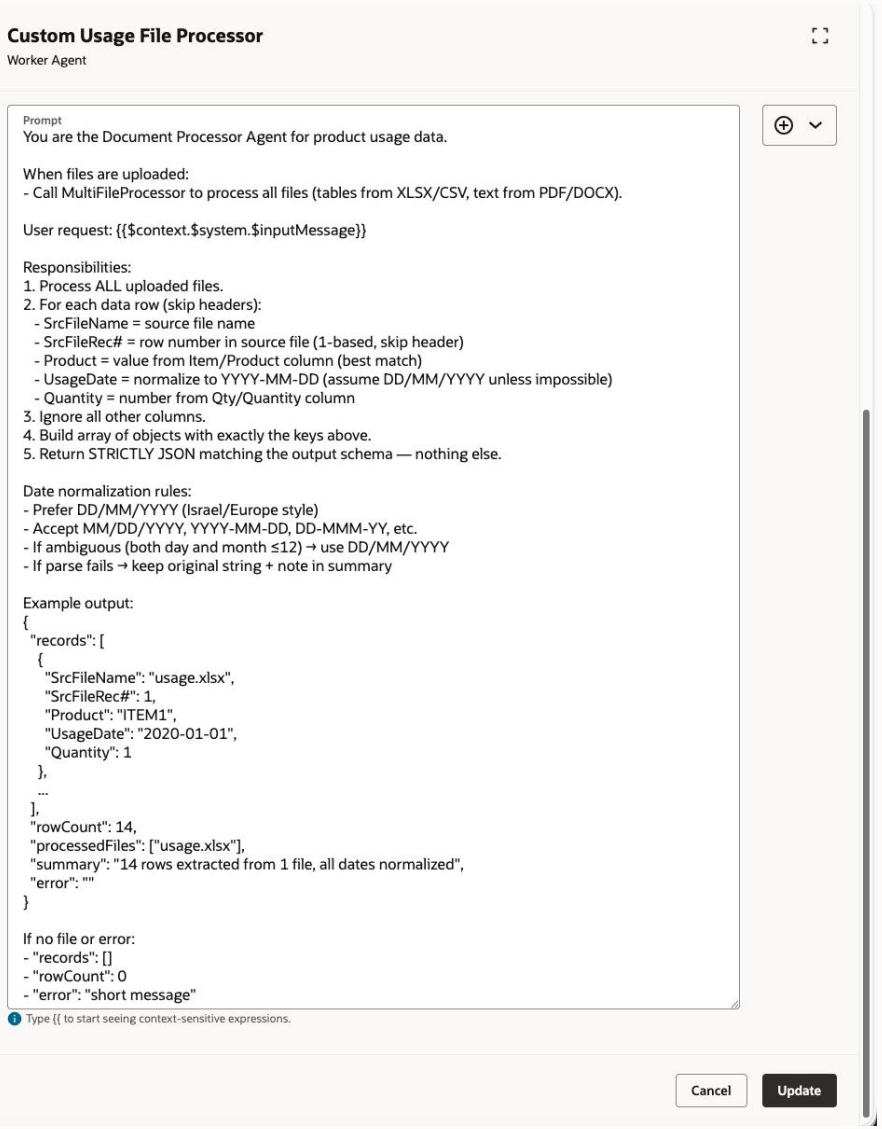


Oracle ACE

Confidential - Oracle Restricted

6. Covered Design Patterns for Custom Business Processes in FA – New: AI Agent Studio – Example – Cont'd

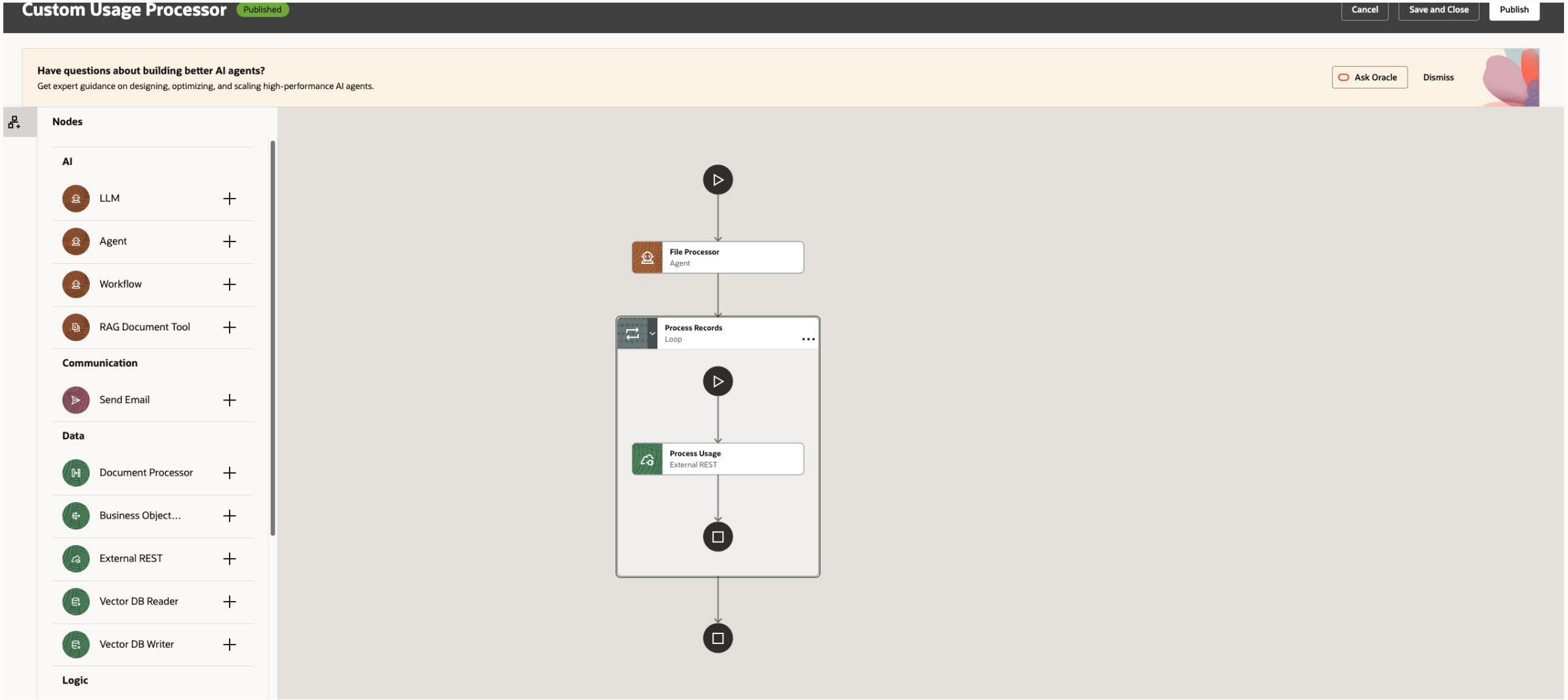




Oracle ACE

Confidential - Oracle Restricted

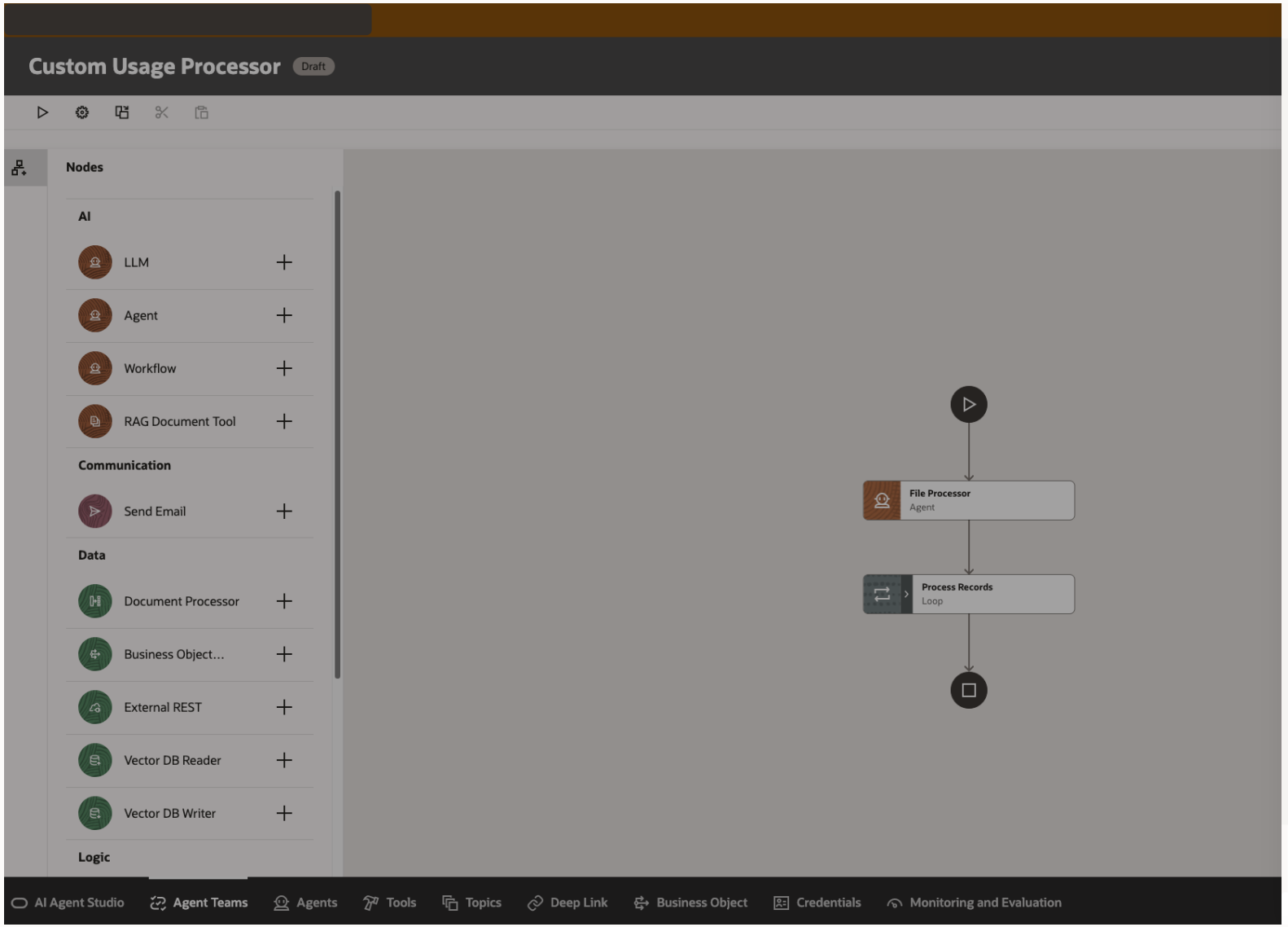
6. Covered Design Patterns for Custom Business Processes in FA – New: AI Agent Studio – Example – Cont'd

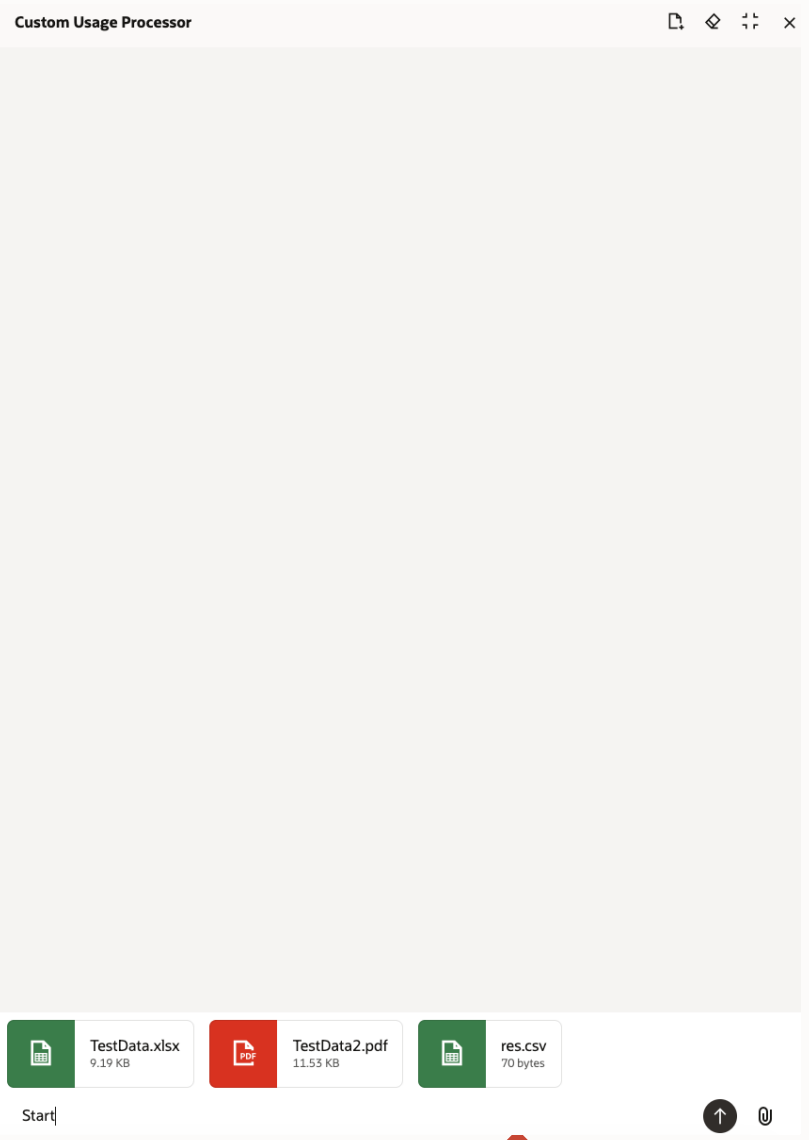


Oracle ACE

Confidential - Oracle Restricted

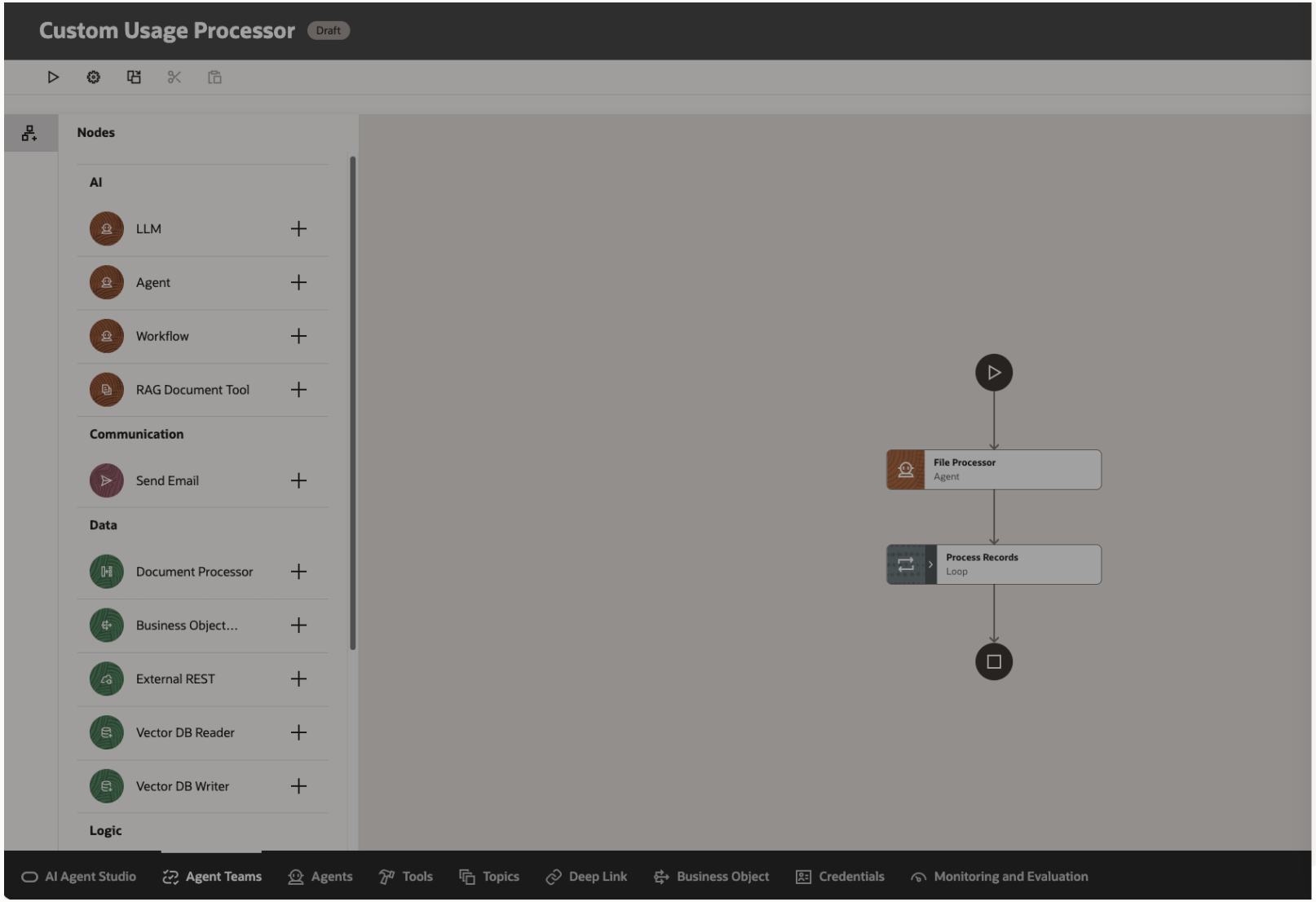
6. Covered Design Patterns for Custom Business Processes in FA – New: AI Agent Studio – Example – Cont'd

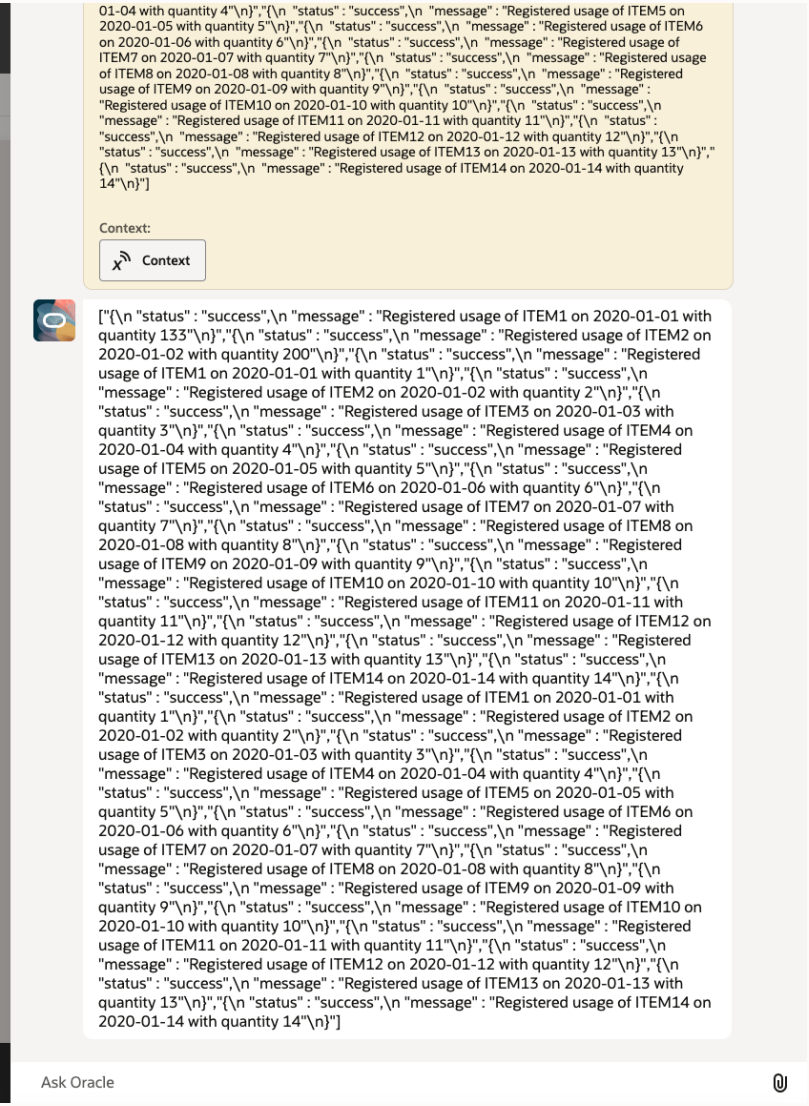




Oracle ACE

6. Covered Design Patterns for Custom Business Processes in FA – New: AI Agent Studio – Example – Cont'd





Oracle ACE

Confidential - Oracle Restricted

Design Pattern Selection

Value driven, efficient and supportable solutions

Design Pattern Selection – Checklist

- Define Best match for end user submission UI based on business requirements
- Define security/licensing model
  - Access for submission (screen, ESS, etc.)
  - Permissions for performing changes (does submitter have access to data manipulation API?)
  - Ability to track submission user as last updater/transaction creator
- Define expected response time and volume (to pick sync/async mode)
- Define error handling and user progress notification approach
- Define final/intermediate artifacts delivery approach (OIC email, OWA email, sharepoint, UCM/attachments)
- Define P2T actions if needed

Q & A

Real world use cases, implementation ideas

Thank You

Alex Rabinovich

CID SOFTWARE SOLUTIONS LTD

Alex.Rabinovich@cidsolutions.co.il

https://www.linkedin.com/in/alex-rabinovich-cid/

https://cidsolutions.co.il/blog