
FROM VIDEO TO VALUE
VIDEO ANALYTICS REDEFINED WITH OCI VISION AND ORACLE APEX
LUC BORS
APEX WORLD 2025

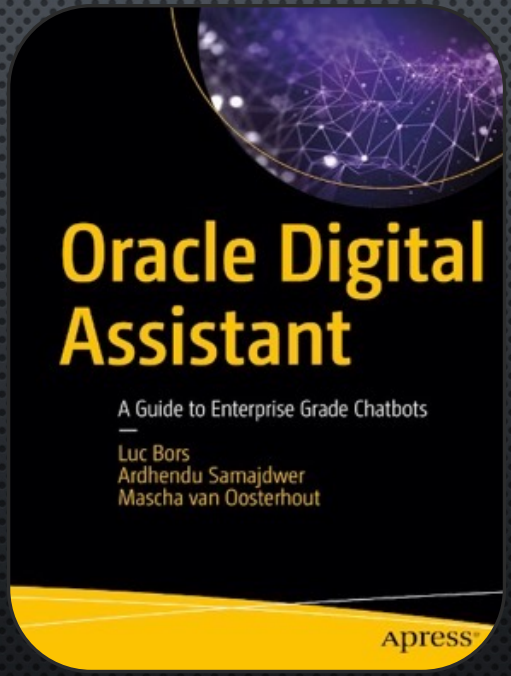
BOHORT

BOHORT

LUC BORS

NEW TECHNOLOGIES

Oracle ACE Director



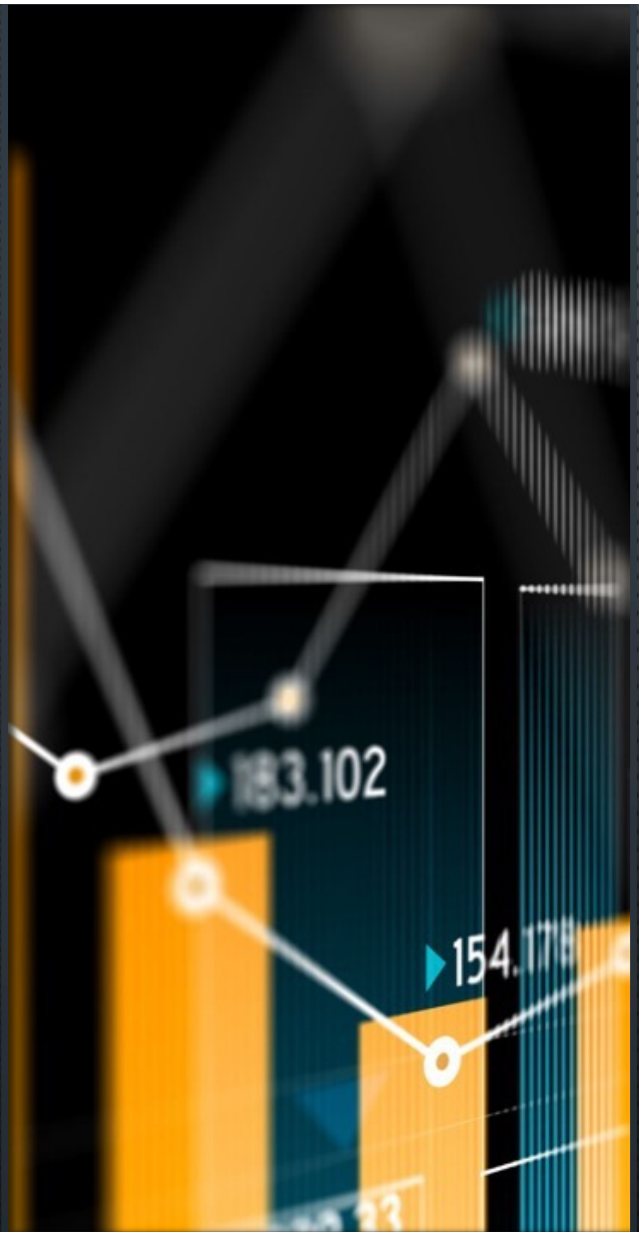
BOHORT

TODAYS TOPICS

Image recognition
and object detection

Technical
components

OCI Vision



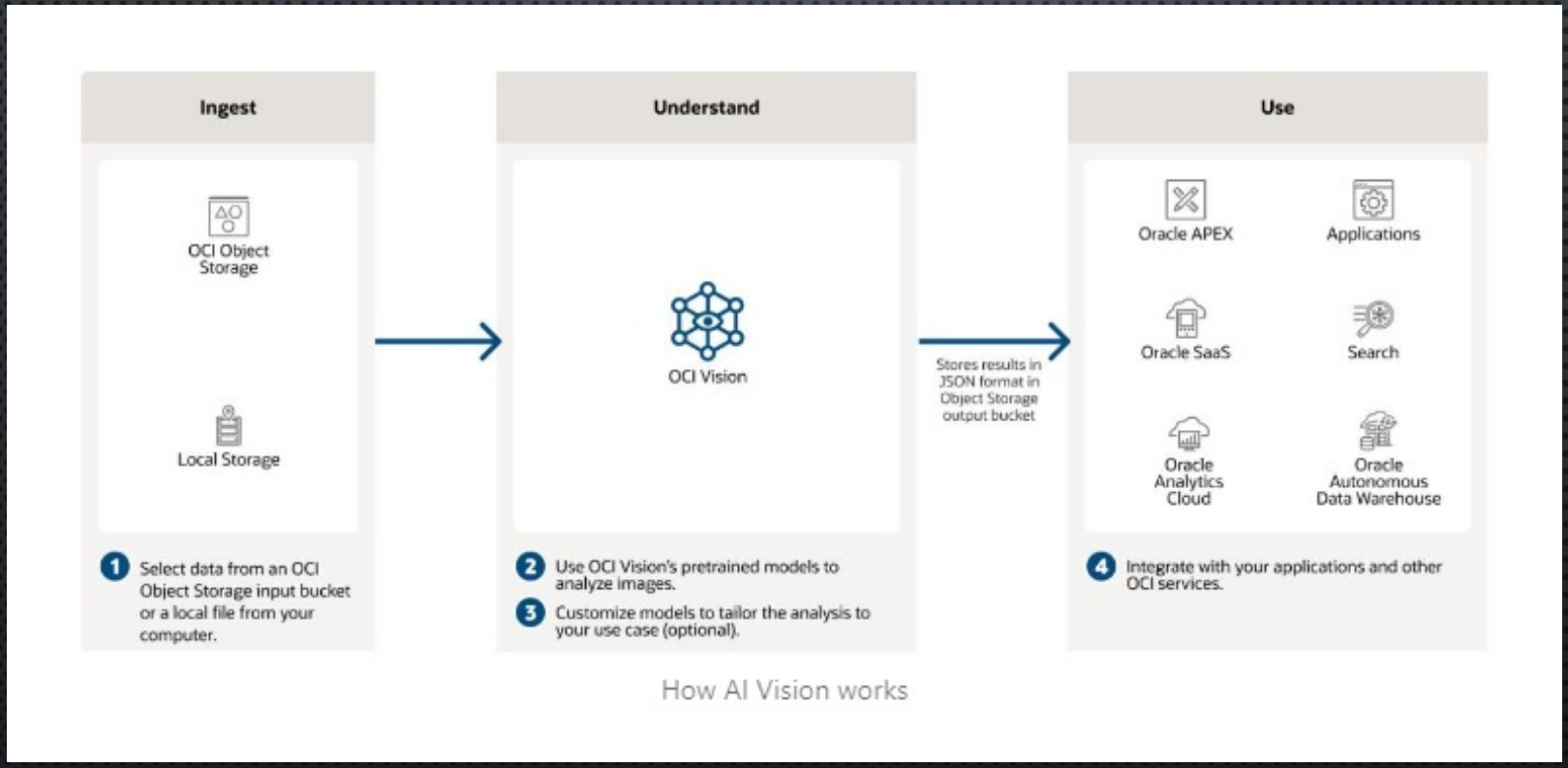
BOHORT

OCI VISION: AI-POWERED VIDEO ANALYSIS

- **KEY FEATURES:** OCI VISION OFFERS OBJECT DETECTION, CLASSIFICATION, AND ANOMALY DETECTION FUNCTIONALITIES FOR COMPREHENSIVE VIDEO ANALYSIS.
- **USE CASE APPLICATIONS:** MONITORING SECURITY FOOTAGE, ANALYZING RETAIL BEHAVIOR, AND TRACKING TRAFFIC EXEMPLIFY OCI VISION'S PRACTICAL APPLICATIONS.
- **ARCHITECTURE OVERVIEW:**

BOHORT

ARCHITECTURE OVERVIEW



How AI Vision works

OCI VISION OOTB

What can OCI Vision do ?

BOHORT

OOTB WORKS FINE……

…. BUT (HOW) CAN WE ENHANCE?

BOHORT

OCI VISION CUSTOM

WHAT CAN OCI VISION DO ?

BOHORT

NEXT LEVEL

PROCESSING VIDEOS

BOHORT

SETTING UP OCI VISION FOR ANALYSIS

- **SETUP OCI VISION INSTANCE:** INCLUDING API KEY RETRIEVAL AND IM SETUP.
- **PYTHON ANALYSIS SCRIPT:** A PYTHON CODE FOR VIDEO FRAME ANALYSIS, SHOWCASING PRACTICAL USAGE OF THE OCI VISION API.

WHAT IS OPENCV?

- OpenCV (Open Source Computer Vision Library)

- https://opencv.org/about/

BOHORT

EXTRACTING VIDEO FRAMES & SENDING TO OCI VISION

Video Frame Extraction: Extracting individual frames involves decoding video files and saving frames for subsequent analysis operations.

API Submission Process: Frames are formatted and transmitted through an API call to OCI Vision for efficient processing.

OpenCV Code Sample: A sample OpenCV script illustrates frame extraction and demonstrates a successful API integration with OCI Vision.

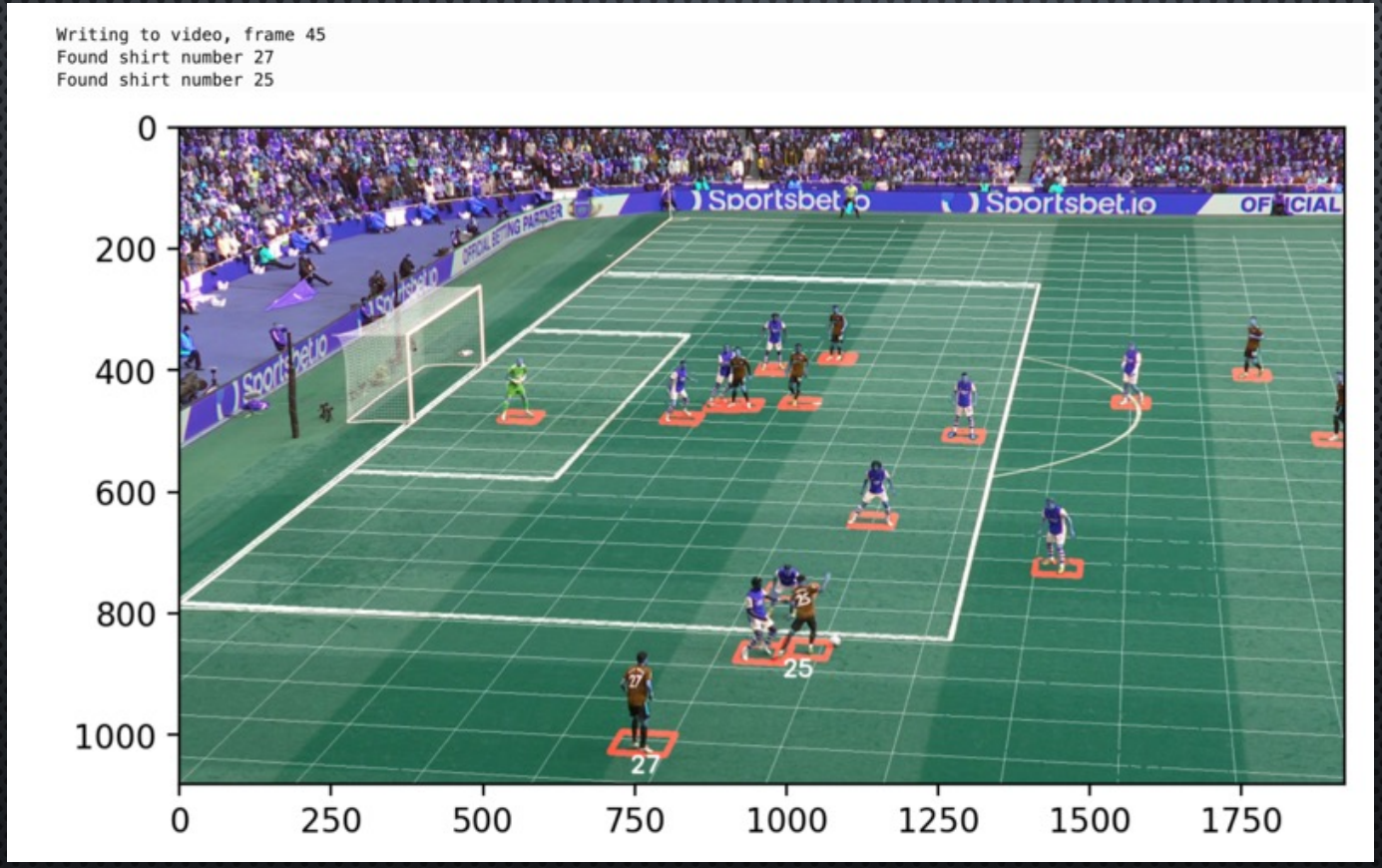
BOHORT

CALLING OIC VISION

```
# Video reader
filename = 'video.mp4'
cap = cv2.VideoCapture(filename)

# Video writer
out = cv2.VideoWriter('soccer_out.avi', cv2.VideoWriter_fourcc('M', 'J', 'P', 'G'), 10, (1920, 1080))

#Remove the first few frames (because it has a video overlay on top of the field)
for i in range(1, 15):
    ret, image_np = cap.read()

tracked_players = [] # Empty list

frame_number = 6
while (frame_number <= 80):
    # Read three frames from the video (we are skipping 2 frames, then processing the third)
    ret, image_np = cap.read()
    ret, image_np = cap.read()
    ret, image_np = cap.read()
    _, im_arr = cv2.imencode('.jpg', image_np)  # im_arr: image in Numpy one-dim array format.

    # Call the AI Vision service to do object recognition and text recognition
    im_bytes = im_arr.tobytes()
    im_b64 = base64.b64encode(im_bytes)
    inline_image_details.data = im_b64.decode('utf-8')
    analyze_image_details.image = inline_image_details
    analyze_image_details.features = features
    res = ai_service_vision_client.analyze_image(analyze_image_details=analyze_image_details)
    res_json = json.loads(repr(res.data))
```

BOHORT

```
Writing to video, frame 45
Found shirt number 27
Found shirt number 25
```



BOHORT

STORING THE RESULT IN A TABLE

```
def store_tracking_information(image_objects, frame_number, connection, cur):
    # Stores the tracking information that was derived from the image in the database
    for index, image_object in enumerate(image_objects):
        world_coordinate = image_object["world_coordinates"]
        if "display" in image_object:
            sql = ""
            if image_object["name"] == "Person":
                shirt_number = "null"
                if "shirt_number" in image_object:
                    shirt_number = image_object["shirt_number"]
                sql = "insert into tracking(framenumber, object, shirt_number, position_x, position_y) values (" + str(frame_number) + ", 'Player', " + s
            if image_object["name"] == "Football":
                sql = "insert into tracking(framenumber, object, position_x, position_y) values (" + str(frame_number) + ", \'Ball\', " + str(image_obje
            cur.execute(sql)
    connection.commit()
```

BOHORT

INTEGRATING OCI VISION API WITH ORACLE APEX

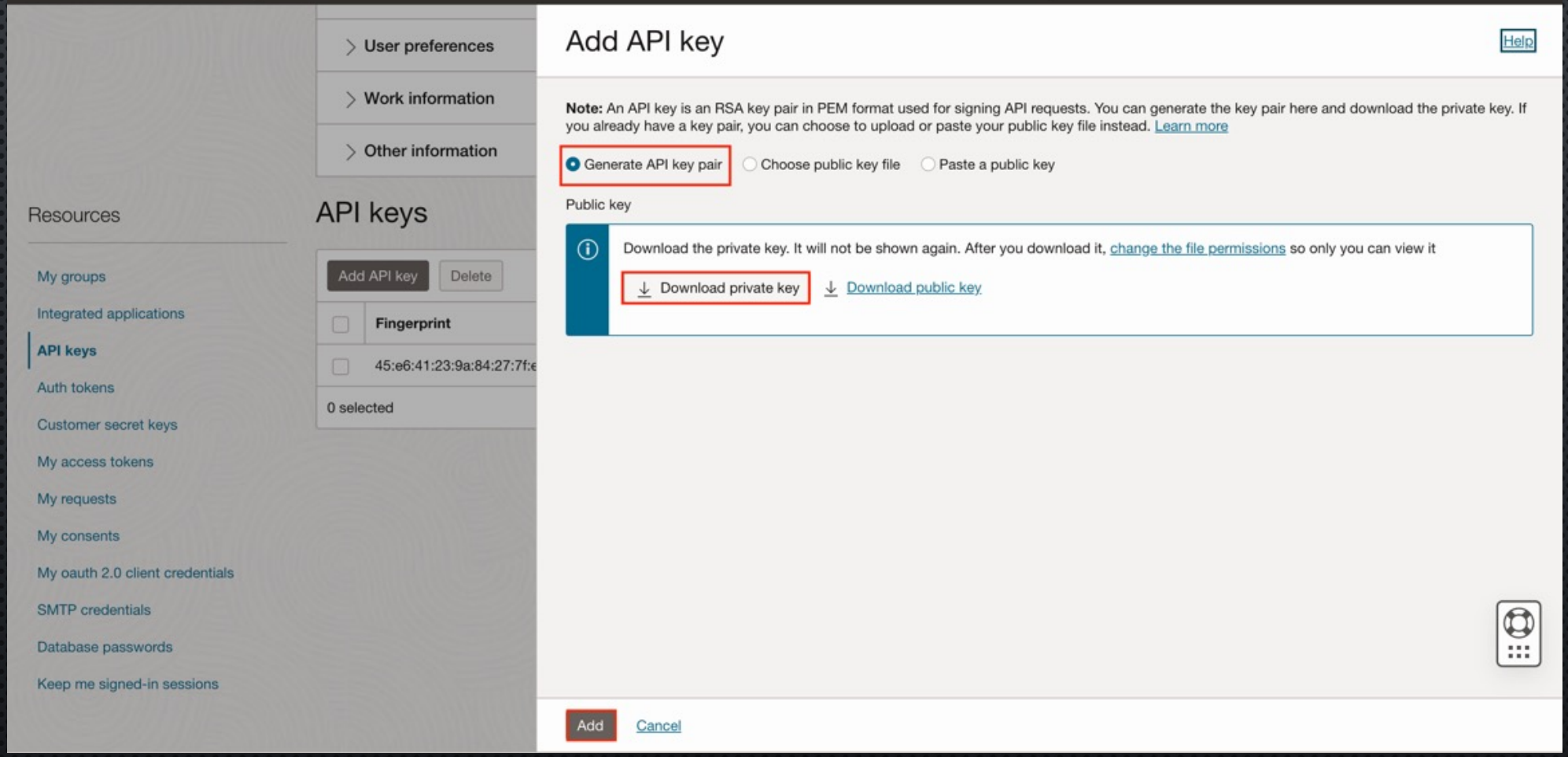
Creating RESTful Web Service: Developing a RESTful Web Service in APEX enables seamless integration with the OCI Vision API.

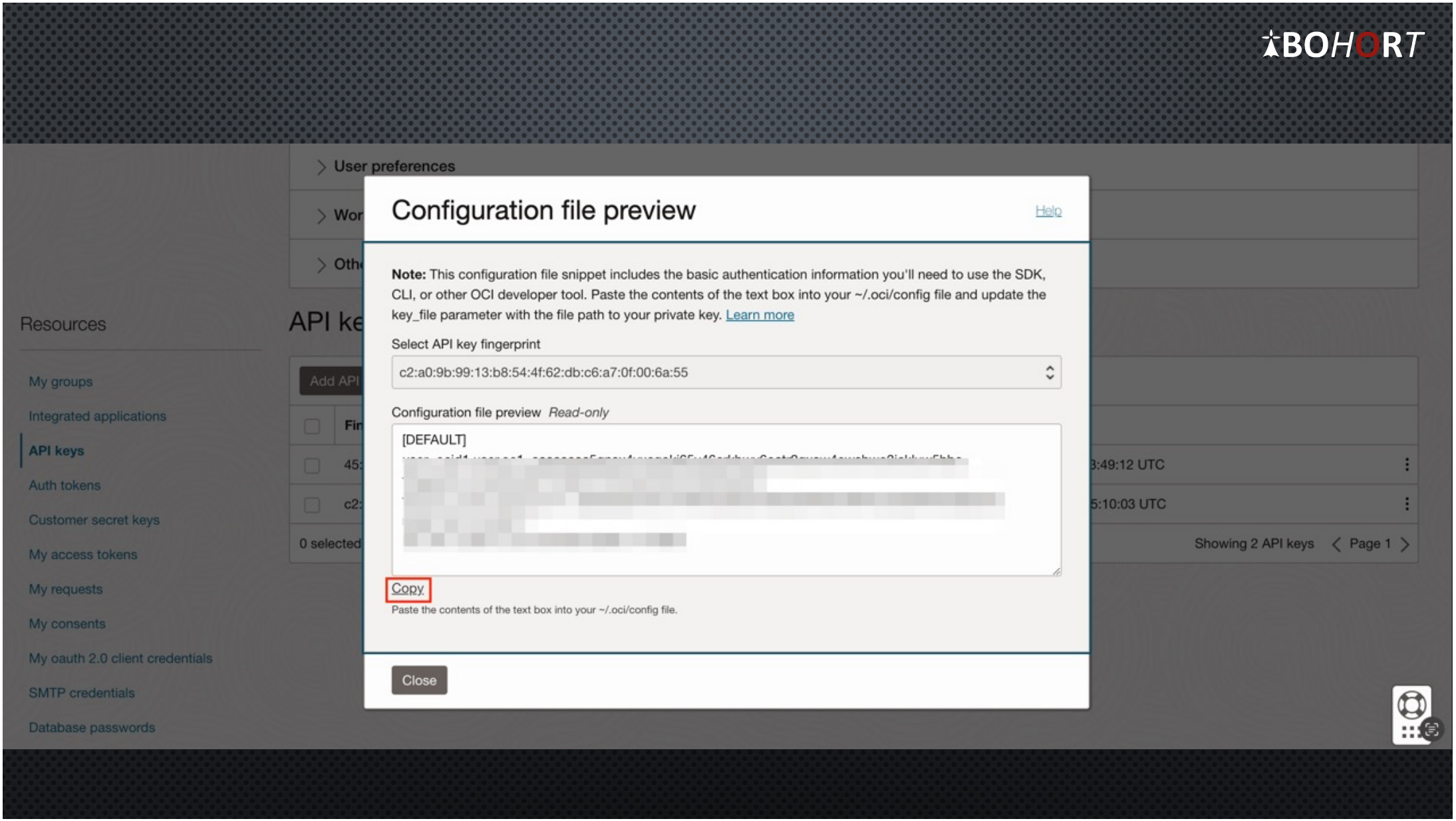
API Interaction Diagram: A visual representation shows the interaction between APEX, OCI Vision API, JSON parsing, and UI display.

APEX REST API Example: Sample code demonstrates how to call the OCI Vision service within an APEX application efficiently.

BOHORT

APEX WEB CREDENTIALS

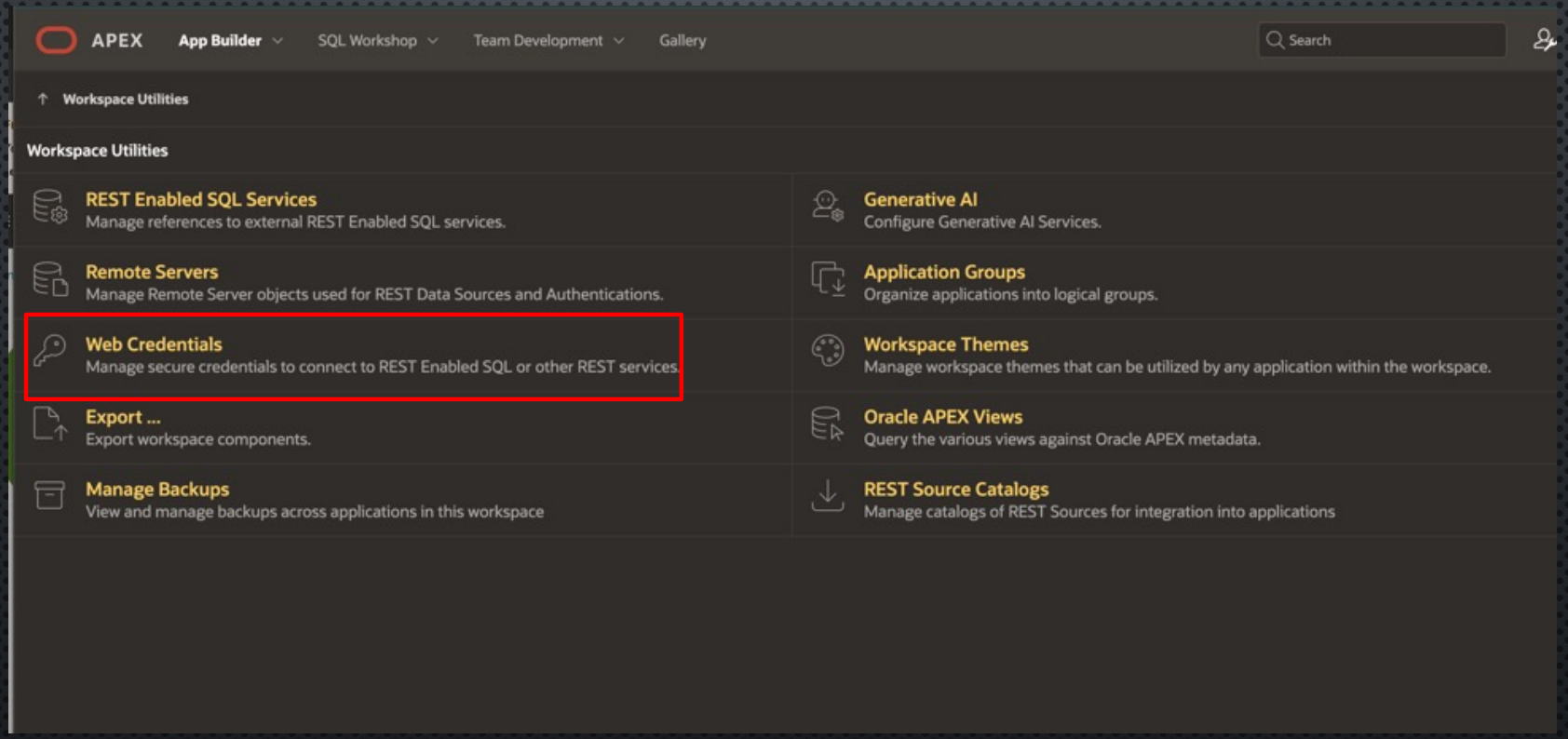



BOHORT
User preferences
Configuration file preview
Help
Note: This configuration file snippet includes the basic authentication information you'll need to use the SDK, CLI, or other OCI developer tool. Paste the contents of the text box into your ~/.oci/config file and update the key_file parameter with the file path to your private key. Learn more
Select API key fingerprint
c2:a0:9b:99:13:b8:54:4f:62:db:c6:a7:0f:00:6a:55
Configuration file preview Read-only
[DEFAULT]
Copy
Paste the contents of the text box into your ~/.oci/config file.
Close
Resources
My groups
Integrated applications
API keys
Auth tokens
Customer secret keys
My access tokens
My requests
My consents
My oauth 2.0 client credentials
SMTP credentials
Database passwords
API ke
Add API
0 selected
Showing 2 API keys
Page 1

APEX SETUP

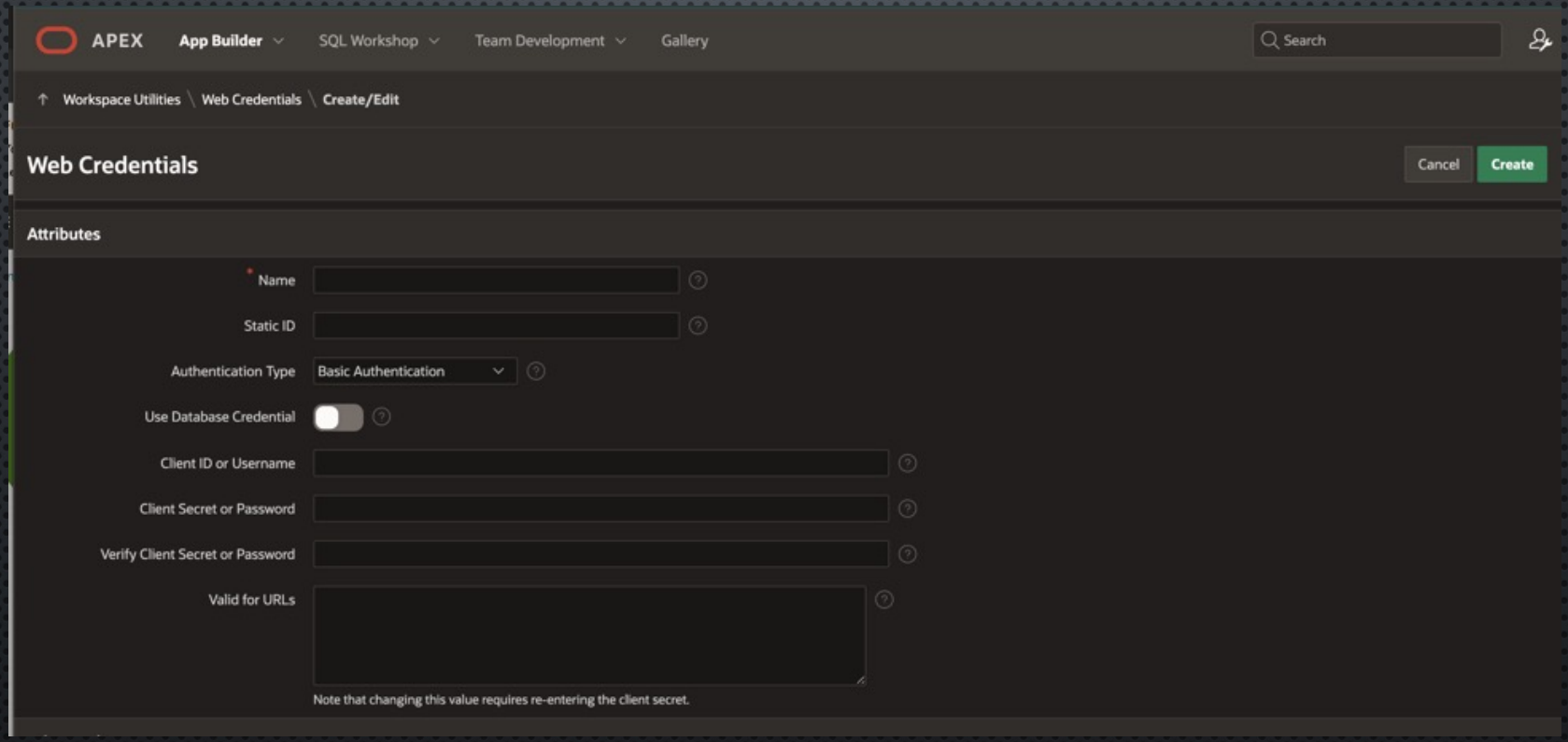
- WEBCREDENTIALS
- REST DATA SOURCE

BOHORT


APEX
App Builder
SQL Workshop
Team Development
Gallery
Search
Workspace Utilities
Workspace Utilities
REST Enabled SQL Services
Manage references to external REST Enabled SQL services.
Generative AI
Configure Generative AI Services.
Remote Servers
Manage Remote Server objects used for REST Data Sources and Authentications.
Application Groups
Organize applications into logical groups.
Web Credentials
Manage secure credentials to connect to REST Enabled SQL or other REST services.
Workspace Themes
Manage workspace themes that can be utilized by any application within the workspace.
Export ...
Export workspace components.
Oracle APEX Views
Query the various views against Oracle APEX metadata.
Manage Backups
View and manage backups across applications in this workspace
REST Source Catalogs
Manage catalogs of REST Sources for integration into applications

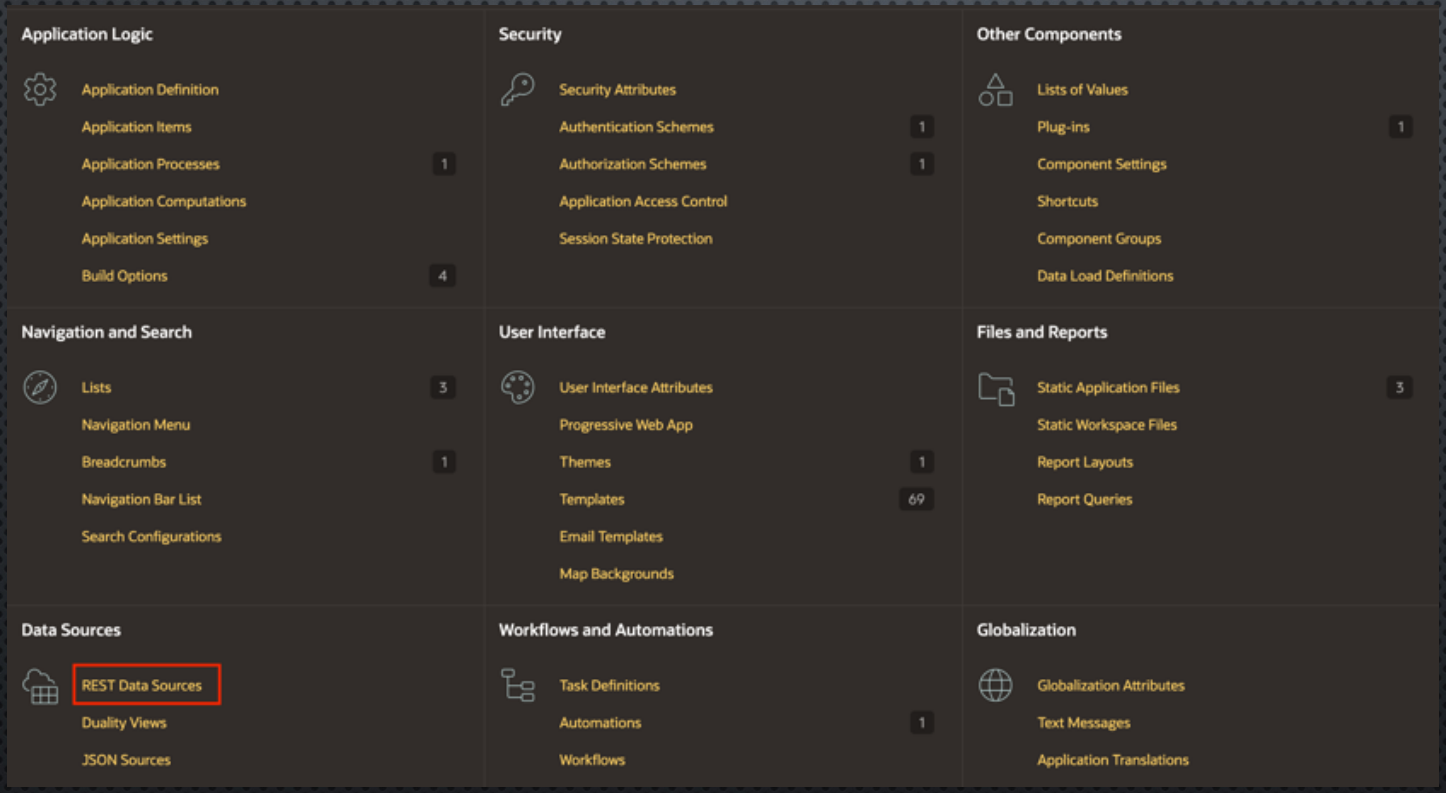
BOHORT

WEBCREDENTIALS (SEE CONFIG FILE)

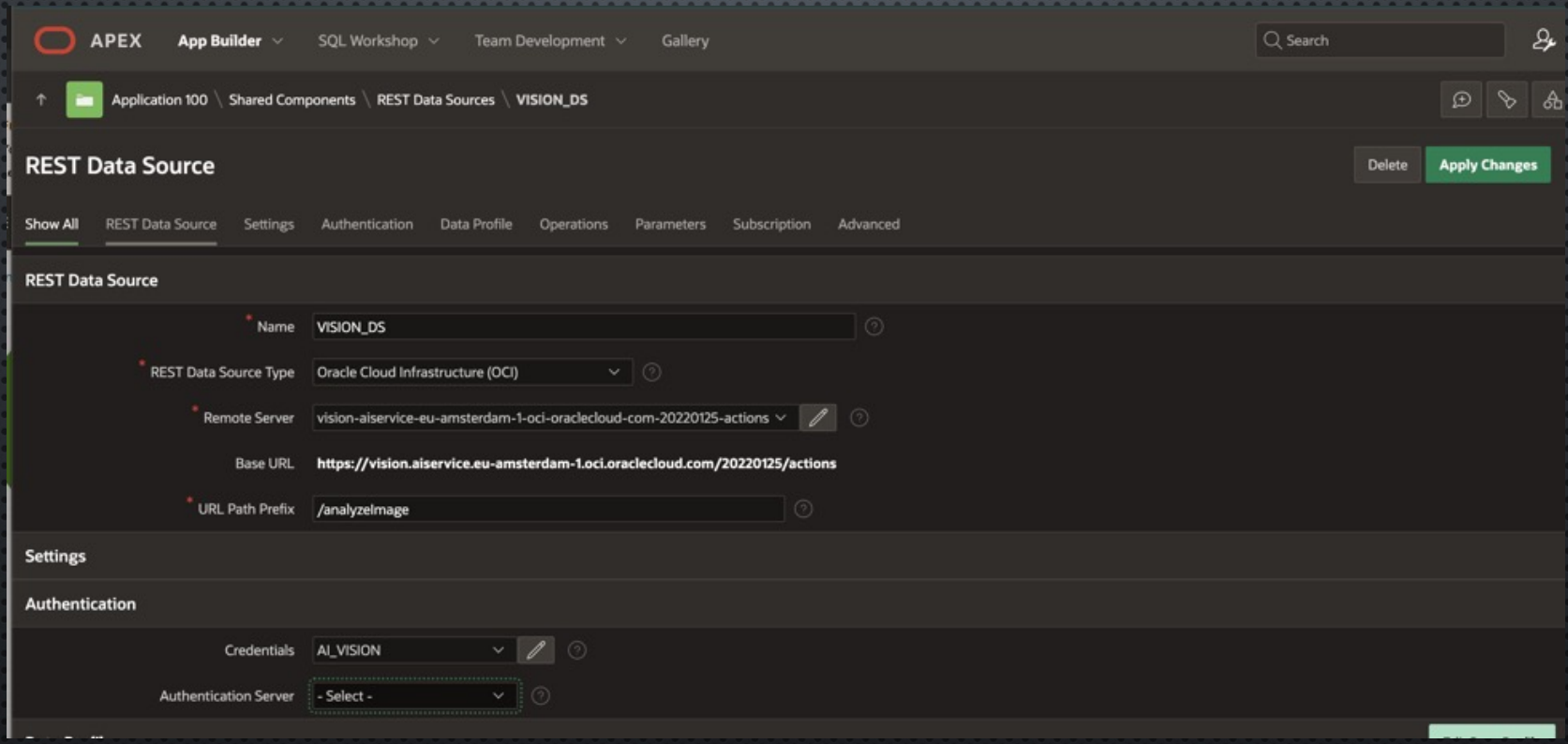


BOHORT

USE REST DATA SOURCE

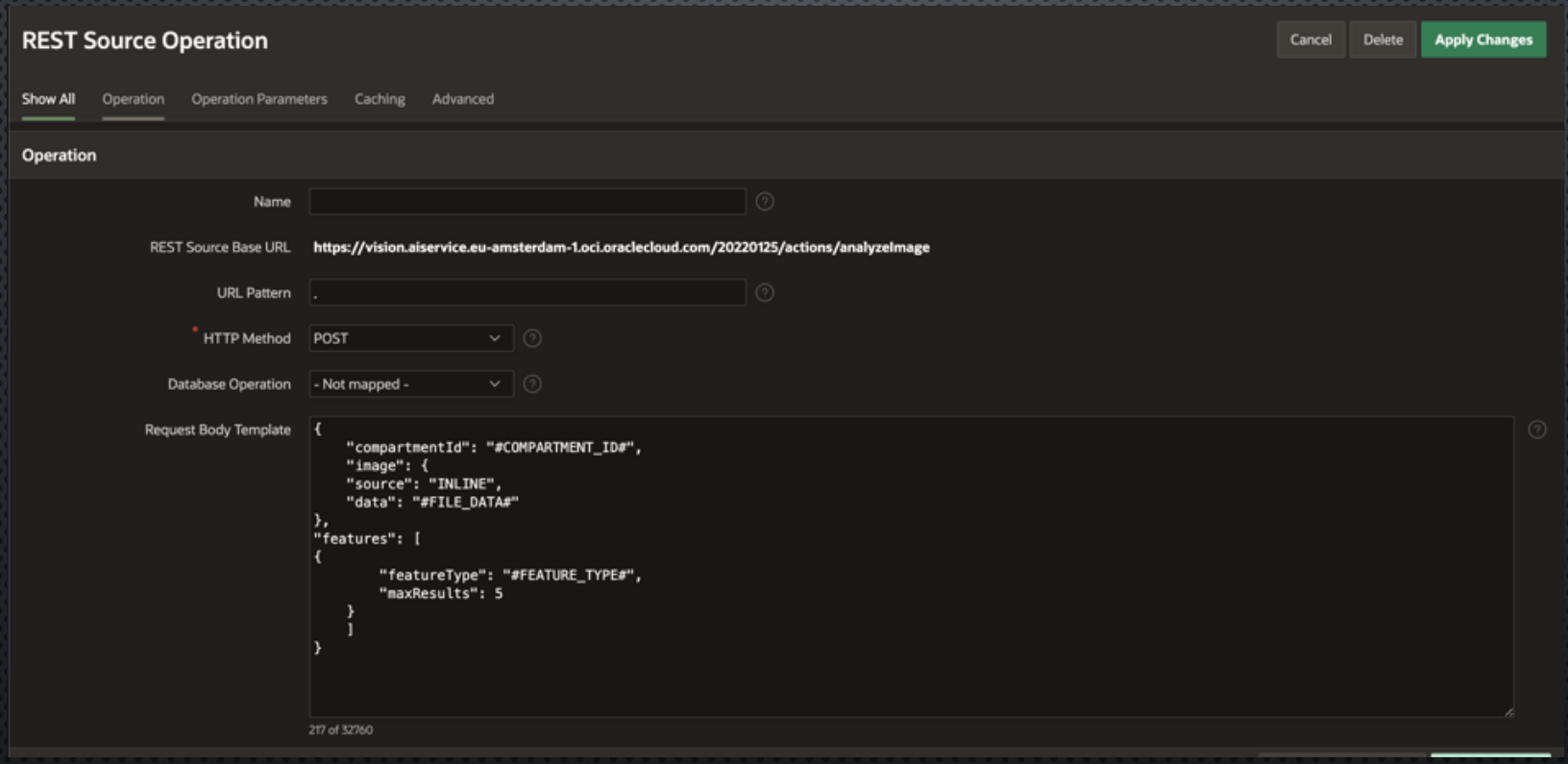


BOHORT


APEX
App Builder
SQL Workshop
Team Development
Gallery
Search
Application 100
Shared Components
REST Data Sources
VISION_DS
REST Data Source
Delete
Apply Changes
Show All
REST Data Source
Settings
Authentication
Data Profile
Operations
Parameters
Subscription
Advanced
REST Data Source
Name
VISION_DS
REST Data Source Type
Oracle Cloud Infrastructure (OCI)
Remote Server
vision-aiservice-eu-amsterdam-1-oci-oraclecloud-com-20220125-actions
Base URL
https://vision.aiservice.eu-amsterdam-1.oci.oraclecloud.com/20220125/actions
URL Path Prefix
/analyzeImage
Settings
Authentication
Credentials
AI_VISION
Authentication Server
- Select -

BOHORT

CONFIGURE PARAMETERS!



BOHORT

BUILDING A FULL-STACK VIDEO ANALYTICS APP

Full-Stack Architecture Overview: ...

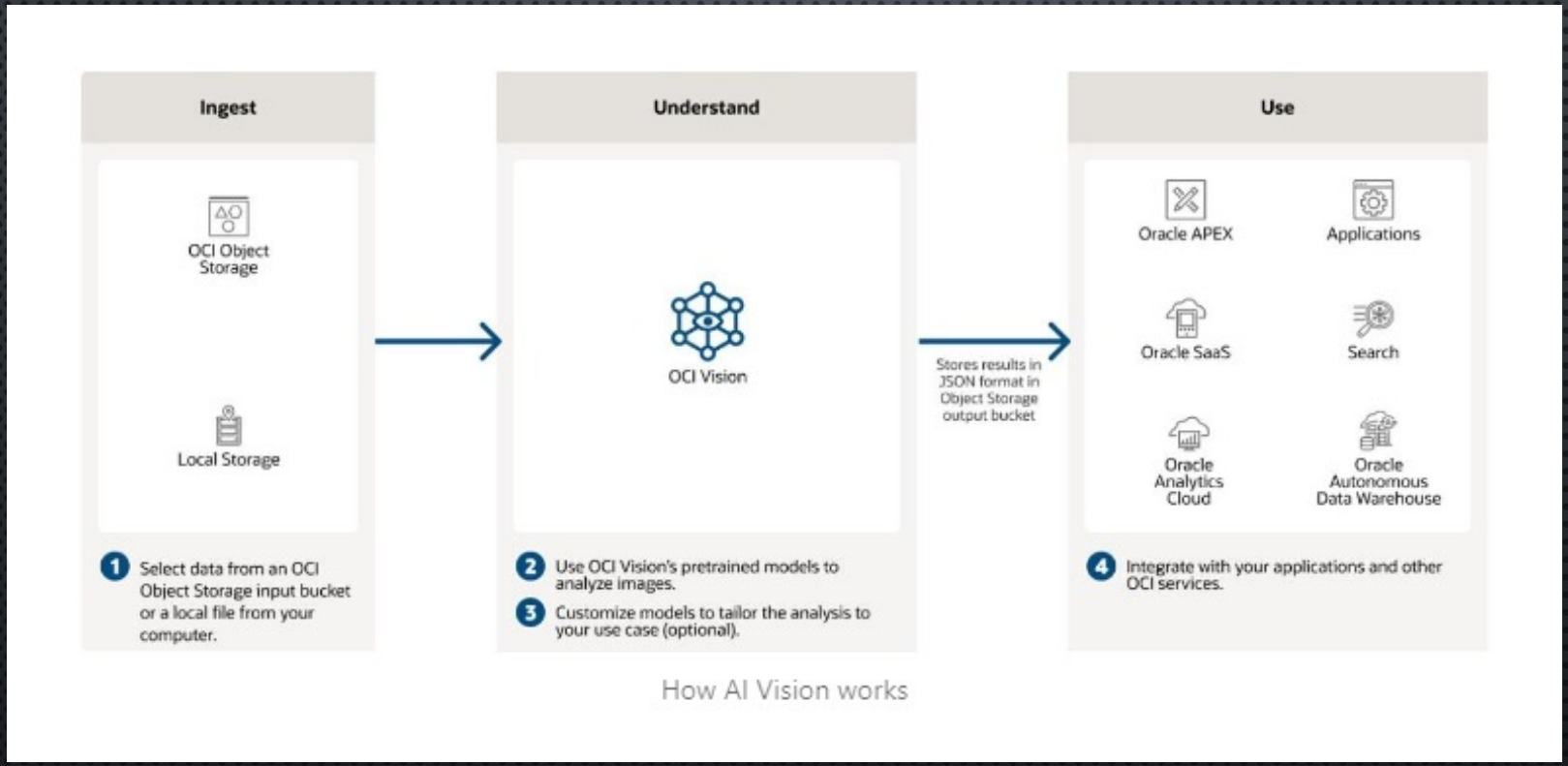
Data Storage in OCI: Utilize OCI Object Storage for secure video storage, ensuring scalability and durability for large datasets.

Processing & UI Display: Leverage backend functions for frame processing, delivering results to user-friendly APEX dashboards for visualization.

BOHORT

THE FLOW

- **Video** (or image) **Upload** → OCI Object Storage
- **Frame Processing** → Sent to OCI Vision API
- **AI Processing** → JSON Response stored in Oracle DB
- **Visualization** → APEX application



How AI Vision works

CALL OCI VISION FROM APEX

```
DECLARE
    l_blob      BLOB;
    l_clob      CLOB;
    l_response  CLOB;
BEGIN
    -- Retrieve the selected image BLOB
    SELECT file_data
    INTO l_blob
    FROM images
    WHERE id = :P_SELECTED_ID;

    -- Convert BLOB to Base64 if API requires it
    l_clob := APEX_WEB_SERVICE.BLOB2CLOBBASE64(l_blob);

    -- Call REST Data Source
    l_response := APEX_EXEC.EXECUTE_REMOTE(
        p_connection => 'VISION_DS', -- Your REST Data Source
        p_operation  => 'POST',
        p_request_body => JSON_OBJECT(--> see below <--)
    );

    -- Show response message
    APEX_UTIL.SET_SESSION_STATE('P_MESSAGE', 'Upload Successful! Response: ' || l_response);
END;


{
    "compartmentId": "#COMPARTMENT_ID#",
    "image": {
    "source": "INLINE",
    "data": l_clob
},
"features": [
{
        "featureType": "OBJECT_DETECTION",
```

BOHORT

SUMMARY

- Oracle OCI Vision:
  - Object Detection works OOTB
  - Custom Models can enhance detections
  - Integration via API
  - Advanced Video Analyses still needs processing and custom code
  - Analyses can be triggered from APEX applications
  - Analytics data stored in DB can be exposed in APEX applications

BOHORT

Q&A & NEXT STEPS

Key Takeaways:
Integration of OCI Vision and Oracle APEX fosters efficient development of sophisticated video analytics applications.

Additional Resources:
Explore comprehensive resources available online for deeper understanding of video analytics, OCI, and APEX.

GA of AI Vision Stored Video Analyses:
Currently only available in, Ashburn. Phoenix and London

Generated on AIDOCMAKER.COM

BOHORT

www.bohort.nl

@lucb_

@bohort_nl

@official.bohort_nl

https://www.linkedin.com/company/bohort