



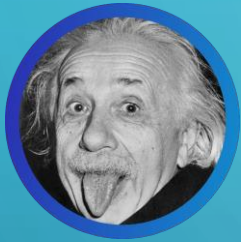
SIMPLE SOLUTIONS FOR COMPLEX PROBLEMS



ORACLE APEX PLUG-INS BY
BARTOSZ OSTROWSKI

ORACLE APEX PLUG-INS /

EINSTEIN NEVER SAID THAT, BUT

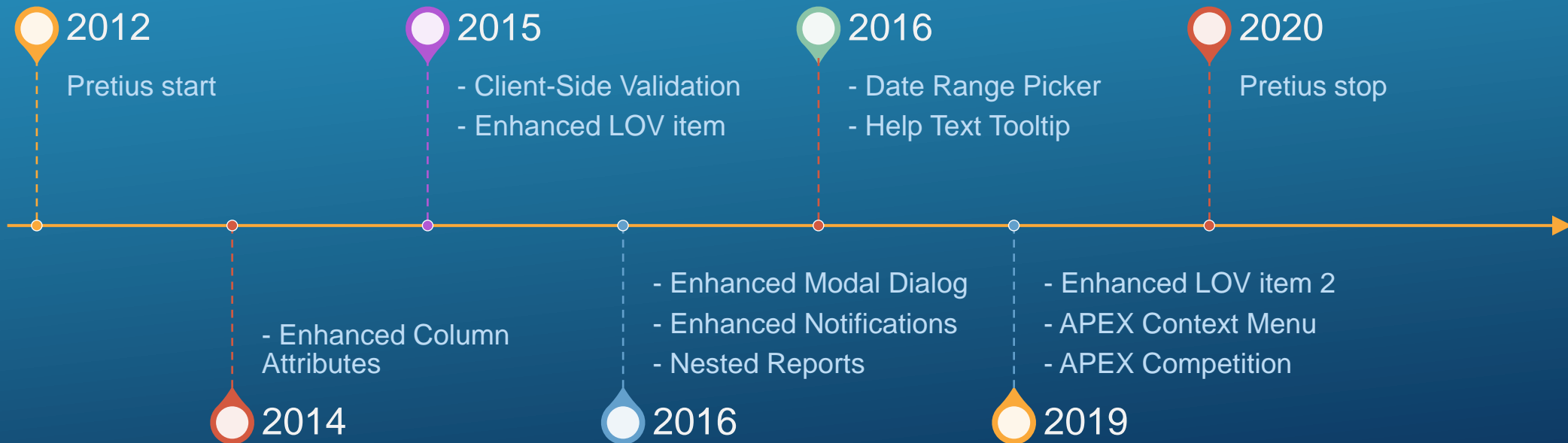


„The definition of insanity is doing the same thing over and over again and expecting a different result.”



„The definition of insanity *is implementing complex solution on multiple pages without wrapping it as a plug-in*”

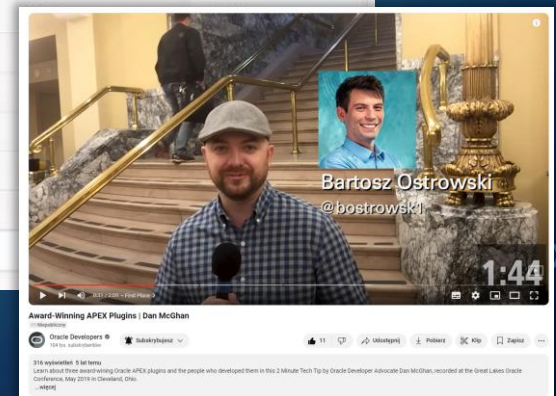
PRETIUS / 8 YEARS OF HISTORY





1ST PLACE / APEX COMPETITION @APEX CONNECT 2019

Search: All Text Columns Go Actions Edit Save Reset							
<input type="checkbox"/>	≡	Ename	Job	Mgr	Hiredate	Sal	Deptno
<input type="checkbox"/>	≡	CLARK	MANAGER	KING	1981-06-09	2450	ACCOUNTING
<input checked="" type="checkbox"/>	≡	MILLER	CLERK	BLAKE	1982-01-23	1300	ACCOUNTING
<input type="checkbox"/>	≡	SMITH	CLERK	JONES	1980-12-17	800	RESEARCH
<input type="checkbox"/>	≡	BLAKE	MANAGER	JONES	1981-05-01	2850	SALES
<input type="checkbox"/>	≡	SCOTT	ANALYST	SCOTT	1987-04-19	3000	RESEARCH
<input type="checkbox"/>	≡	FORD	ANALYST	SCOTT	1981-12-03	3000	RESEARCH
<input type="checkbox"/>	≡	TURNER	SALESMAN	SCOTT	1981-09-08	1500	RESEARCH
<input type="checkbox"/>	≡	WARD	SALESMAN	SCOTT	1981-02-22	1250	SALES
<input type="checkbox"/>	≡	MARTIN	SALESMAN	SCOTT	1981-09-28		
<input type="checkbox"/>	≡	JAMES	CLERK	SCOTT	1981-12-03		
<input type="checkbox"/>	≡	ALLEN	SALESMAN	FORD	1981-02-20		
<input type="checkbox"/>	≡	JONES	MANAGER	MARTIN	1981-04-02		
<input type="checkbox"/>	≡	KING	PRESIDENT		1981-11-17		
<input type="checkbox"/>	≡	ADAMS	CLERK		1987-05-23		
1 rows selected							



Source: https://youtu.be/CUk8_gC4DYs

UNITED CODES / COMMERCIAL PLUG-INS



REQUIREMENTS/ NESTED REPORTS


- „I want to click this row and get details. Can you do this?” -Przemysław Gołos
- Default template - data presented in table
- Custom template – iterate over fetched data and put into custom HTML
- Custom callback – support custom JavaScript Code after fetching data

Examples

Manual Events

This example presents 2 levels of nested reports created on top of the DEPT table.

The first level uses the default template with custom HTML (see Extend default template region) which uses built-in classes (see Built-in classes) that affect the nested report.

The second level uses a custom template (see Custom template region) which is bound with an APEX modal page through the  Edit button. The modal page allows users to change employee details, and after saving the changes the 2nd nested report is refreshed using the dialog close event triggered on the edit button. The modal page URL is generated using the second level SQL Query.

Extend default template


Custom template

Built-in classes





	Deptno	Dname	Loc
	10	ACCOUNTING	NEW YORK
	20	RESEARCH	DALLAS
	30	SALES	CHICAGO
	40	OPERATIONS	BOSTON
	80	kkk	ppp
	90	pop	-
	60	Tech	-
	70	-	-

1 - 8

Drilldown

This example shows how one dynamic action can drill-down into hierarchical data. The dynamic action is listening to the `Click` event on jQuery selector `.levelIsGetInferiors` which is a class of the anchor .

Drilldown SQL Query

	Empno	Ename	Job	Sal	Comm	Inferiors
	7839	KING	PRESIDENT	1000	-	0
	7839	KING	PRESIDENT	1000	-	0
	7839	KING	PRESIDENT	1000	-	0
	7839	KING	PRESIDENT	1000	-	0
	7839	KING	PRESIDENT	1000	-	0

STATISTICS & INFO/ NESTED REPORTS

- ~1 300 JavaScript lines of code
- ~550 PL/SQL lines of code
- ~ 250 e-mails from around the world (outside Pretius support)
- 39 stars on Github
- Most liked and popular at apex.world
- Used internally by Oracle employees

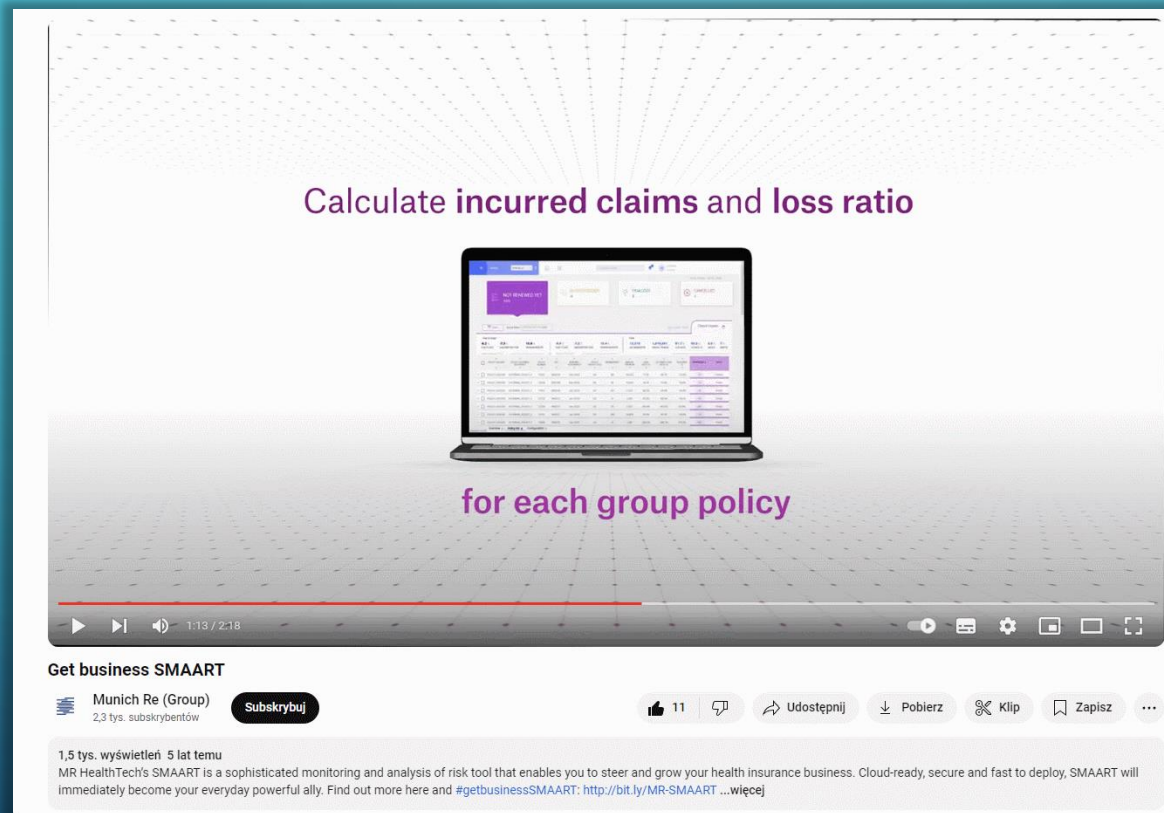
The screenshot shows the Oracle APEX Page Designer interface. The top navigation bar includes 'App Builder', 'SQL Workshop', and 'Team Development'. The main toolbar shows 'Application 176 \ Page Designer' with various icons for file operations and a 'Save' button. The left sidebar contains 'Action', 'Layout', 'Component View', and 'Messages'. The main content area is titled 'United Codes Nested Reports [Plug-In]' and is divided into 'Identification' and 'Settings' sections. The 'Settings' section is highlighted with a red border and contains the following configuration:

- Mode:** Default template & default callback
- Default callback options:**
 - ☒ Cache results
 - ☐ Collapse expanded
 - ☐ Set max height
 - ☒ Add animation
 - ☒ Loading indicator
- Default template options:**
 - ☒ Stretch report
 - ☒ Sorting data
 - ☐ Extend default template
- Loading indicator style:** Aligned to report
- Background color:** #EBEBEB
- Border color:** #c5c5c5
- Highlight row color:** #e3e3e3
- Details query:**

```
select
case
when count(EMPNO)-1 = 0 then '-'
else '
  <a href="javascript: void(0)" class="levelsGetInferiors">
    <span class="fa fa-sitemap-vertical"></span>
  </a>
  <span class="EMPNO hidden">' || EMPNO || '</span>
' end
EMPNO      as derived01_fa_sitemap_vertical,
EMPNO      as "Empno",
ENAME      as "Name",
```
- When no data found:** No data found

The bottom section, 'Affected Elements', shows a table with columns 'Selection Type' and 'Region', and a row with '.....Example'.

NESTED REPORTS: CUSTOM TEMPLATE + IFRAME/ MRHT SMART

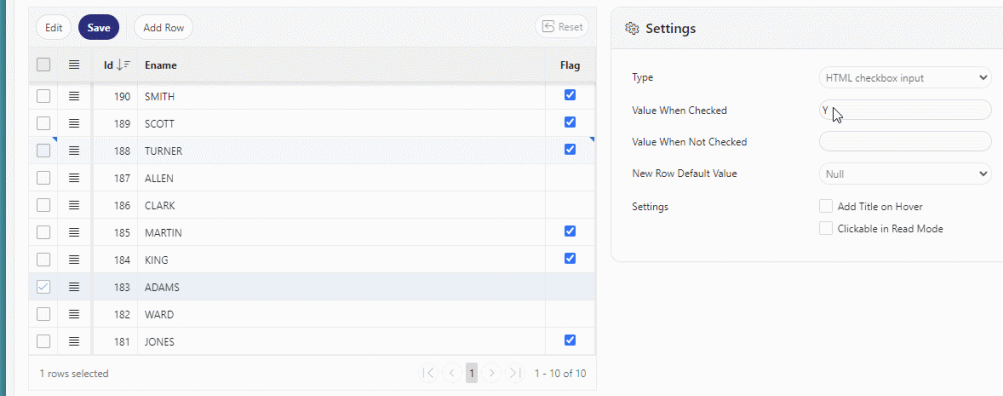


Source: <http://youtu.be/S8W3hECXXuE>

REQUIREMENTS/ IG CHECKBOX PRO

- Support for Oracle APEX 19.2 onwards
- Support three checkbox states „Yes”, „No” and NULL value
- Support custom icons instead of HTML checkbox
- Enable checkbox interaction outside IG edit mode
- Integrate using Interactive Grid JS APIs

Example 1



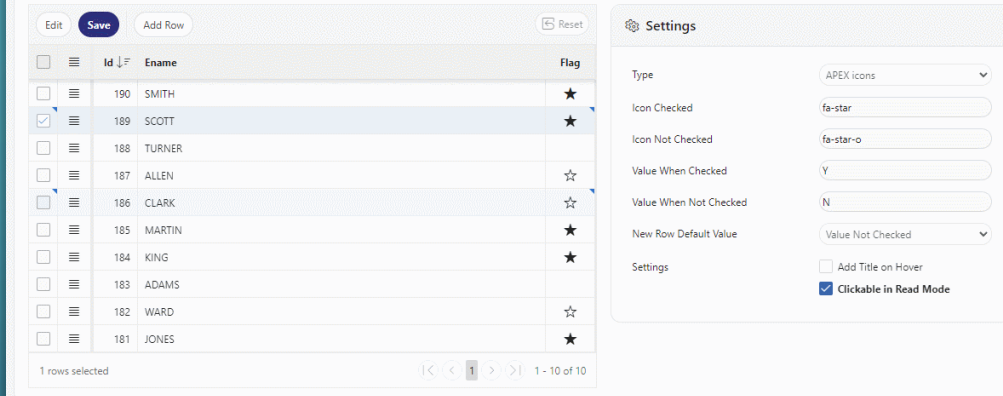
Example 1 shows an Interactive Grid (IG) with columns: Id, Ename, and Flag. The Flag column uses an HTML checkbox input. The settings panel on the right shows the configuration for the Flag column.

Id	Ename	Flag
190	SMITH	<input checked="" type="checkbox"/>
189	SCOTT	<input checked="" type="checkbox"/>
188	TURNER	<input checked="" type="checkbox"/>
187	ALLEN	<input type="checkbox"/>
186	CLARK	<input type="checkbox"/>
185	MARTIN	<input checked="" type="checkbox"/>
184	KING	<input checked="" type="checkbox"/>
183	ADAMS	<input type="checkbox"/>
182	WARD	<input type="checkbox"/>
181	JONES	<input checked="" type="checkbox"/>

Settings for Example 1:

- Type: HTML checkbox input
- Value When Checked: Y
- Value When Not Checked:
- New Row Default Value: Null
- Settings: ☐ Add Title on Hover, ☐ Clickable in Read Mode

Example 2



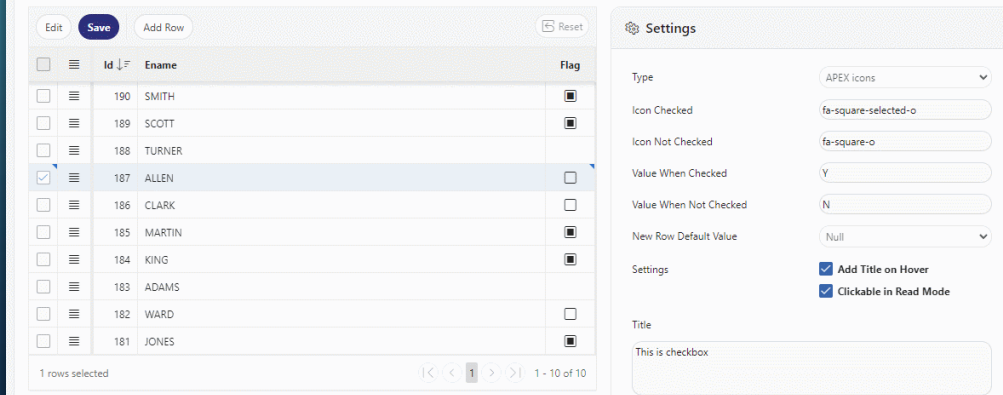
Example 2 shows an Interactive Grid (IG) with columns: Id, Ename, and Flag. The Flag column uses APEX icons. The settings panel on the right shows the configuration for the Flag column.

Id	Ename	Flag
190	SMITH	★
189	SCOTT	★
188	TURNER	★
187	ALLEN	☆
186	CLARK	☆
185	MARTIN	★
184	KING	★
183	ADAMS	★
182	WARD	☆
181	JONES	★

Settings for Example 2:

- Type: APEX icons
- Icon Checked: fa-star
- Icon Not Checked: fa-star-o
- Value When Checked: Y
- Value When Not Checked: N
- New Row Default Value: Value Not Checked
- Settings: ☐ Add Title on Hover, ☒ Clickable in Read Mode

Example 3



Example 3 shows an Interactive Grid (IG) with columns: Id, Ename, and Flag. The Flag column uses APEX icons. The settings panel on the right shows the configuration for the Flag column.

Id	Ename	Flag
190	SMITH	■
189	SCOTT	■
188	TURNER	■
187	ALLEN	■
186	CLARK	■
185	MARTIN	■
184	KING	■
183	ADAMS	■
182	WARD	■
181	JONES	■

Settings for Example 3:

- Type: APEX icons
- Icon Checked: fa-square-selected-o
- Icon Not Checked: fa-square-o
- Value When Checked: Y
- Value When Not Checked: N
- New Row Default Value: Null
- Settings: ☒ Add Title on Hover, ☒ Clickable in Read Mode
- Title: This is checkbox

FUN FACTS/ IG CHECKBOX PRO

- Oracle introduced native component half year later for APEX 20.2
- Today, the native component still doesn't support icons and three states of checkbox



3 comments:

[John Snyders](#) said...



Looks very nice.

"but it's unclear if [new APEX checkbox] has all of the features our checkbox plug-in has"

It does not. No custom icons for example. Also no null value.

Going forward you may be interested in some of the upcoming changes to the item API for this and other plugins.

[9/14/2020 8:43 PM](#)

[Dimitri Gielis](#) said...



Hi John,

Thanks so much for the feedback.

Looks like our plug-in will still be useful even in APEX 20.2 :)



Gemma Wood

@gemmawood

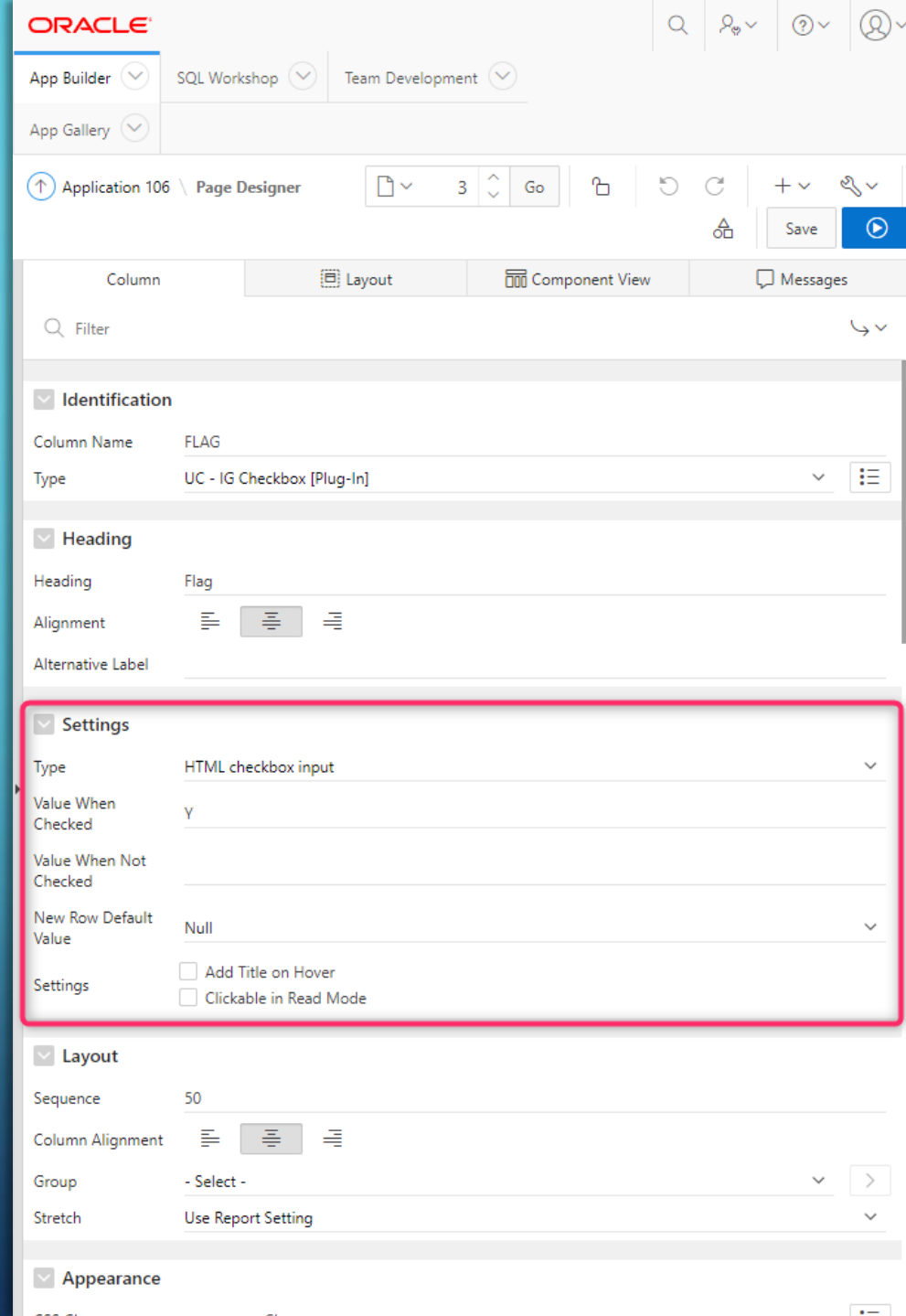
6 months I go, I spent several very frustrating days creating a simple checkbox for an [#orclapex](#) interactive grid. Today I created one in 5 minutes thanks to [@Plug_ins_Pro](#) (the guys from [@apexofficeprint](#)). Great job guys, can't wait to use more of your plugins for my client!

[Przetłumacz wpis](#)

9:29 PM · 1 lip 2020

STATISTICS & INFO/ IG CHECKBOX PRO

- ~100h
- ~1 300 JavaScript
- ~760 PL/SQL

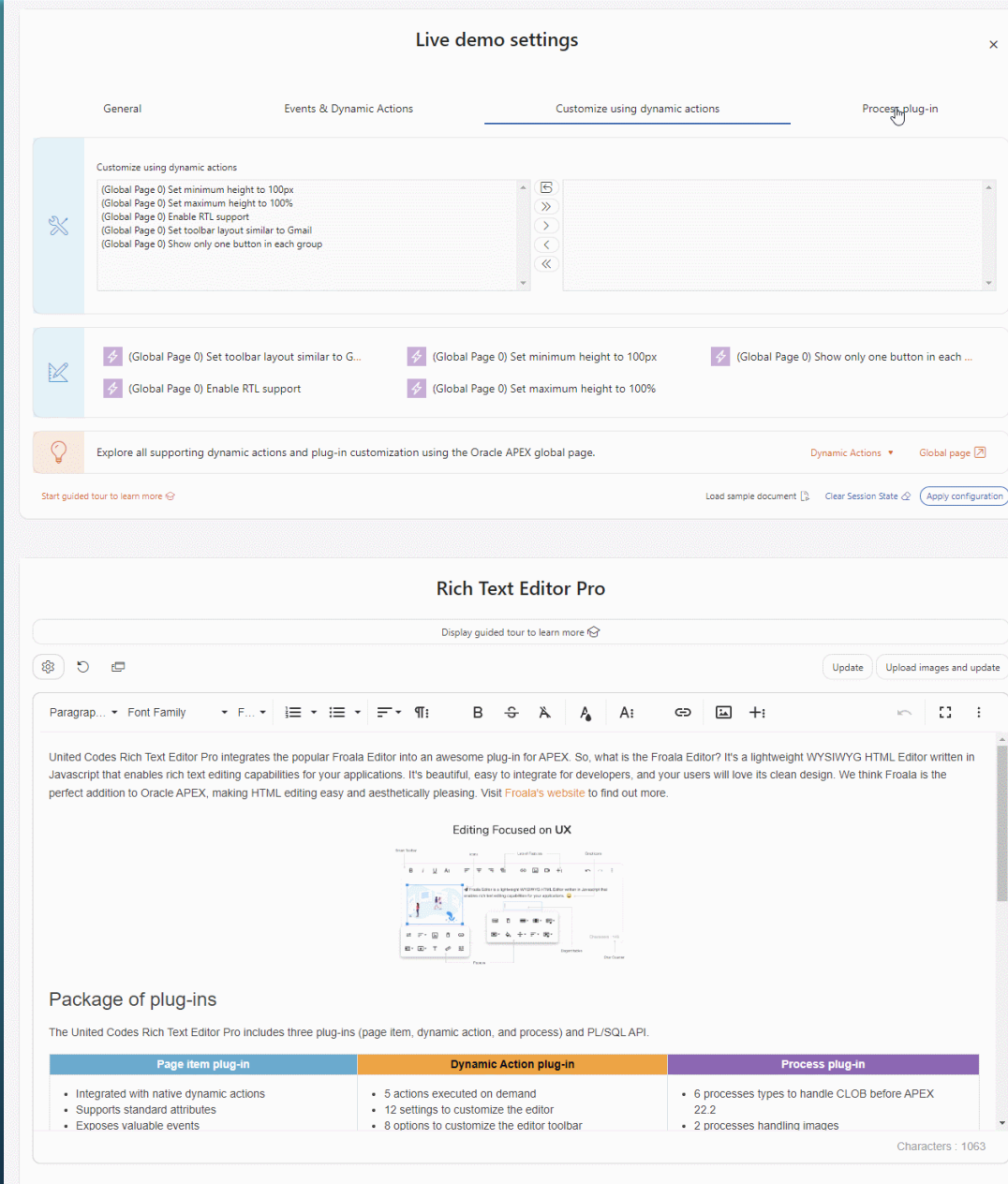


The screenshot displays the Oracle APEX Page Designer interface for 'Application 106' in 'Page Designer' mode. The 'Component View' tab is active, showing the configuration for a column named 'FLAG' of type 'UC - IG Checkbox [Plug-In]'. The 'Settings' section is highlighted with a red box and contains the following configuration:

- Identification:**
 - Column Name: FLAG
 - Type: UC - IG Checkbox [Plug-In]
- Heading:**
 - Heading: Flag
 - Alignment: Center (selected)
 - Alternative Label: (empty)
- Settings:**
 - Type: HTML checkbox input
 - Value When Checked: y
 - Value When Not Checked: (empty)
 - New Row Default Value: Null
 - Settings:
 - ☐ Add Title on Hover
 - ☐ Clickable in Read Mode
- Layout:**
 - Sequence: 50
 - Column Alignment: Center (selected)
 - Group: - Select -
 - Stretch: Use Report Setting
- Appearance:** (partially visible)

REQUIREMENTS/ RICH TEXT EDITOR PRO

- Support CLOB values for APEX 19.2
- Integrate with native components (forms)
- Upload images on page submission
- Dynamic Action configuration cross application pages (global page)
- Deleting images in database after removing from rich text document



The screenshot displays the 'Live demo settings' dialog for the 'Rich Text Editor Pro'. The dialog has four tabs: 'General', 'Events & Dynamic Actions', 'Customize using dynamic actions', and 'Process plug-in'. The 'Customize using dynamic actions' tab is active, showing a list of dynamic actions for the 'Global Page 0' that can be applied to the editor. Below this, there are buttons for 'Start guided tour to learn more', 'Load sample document', 'Clear Session State', and 'Apply configuration'.

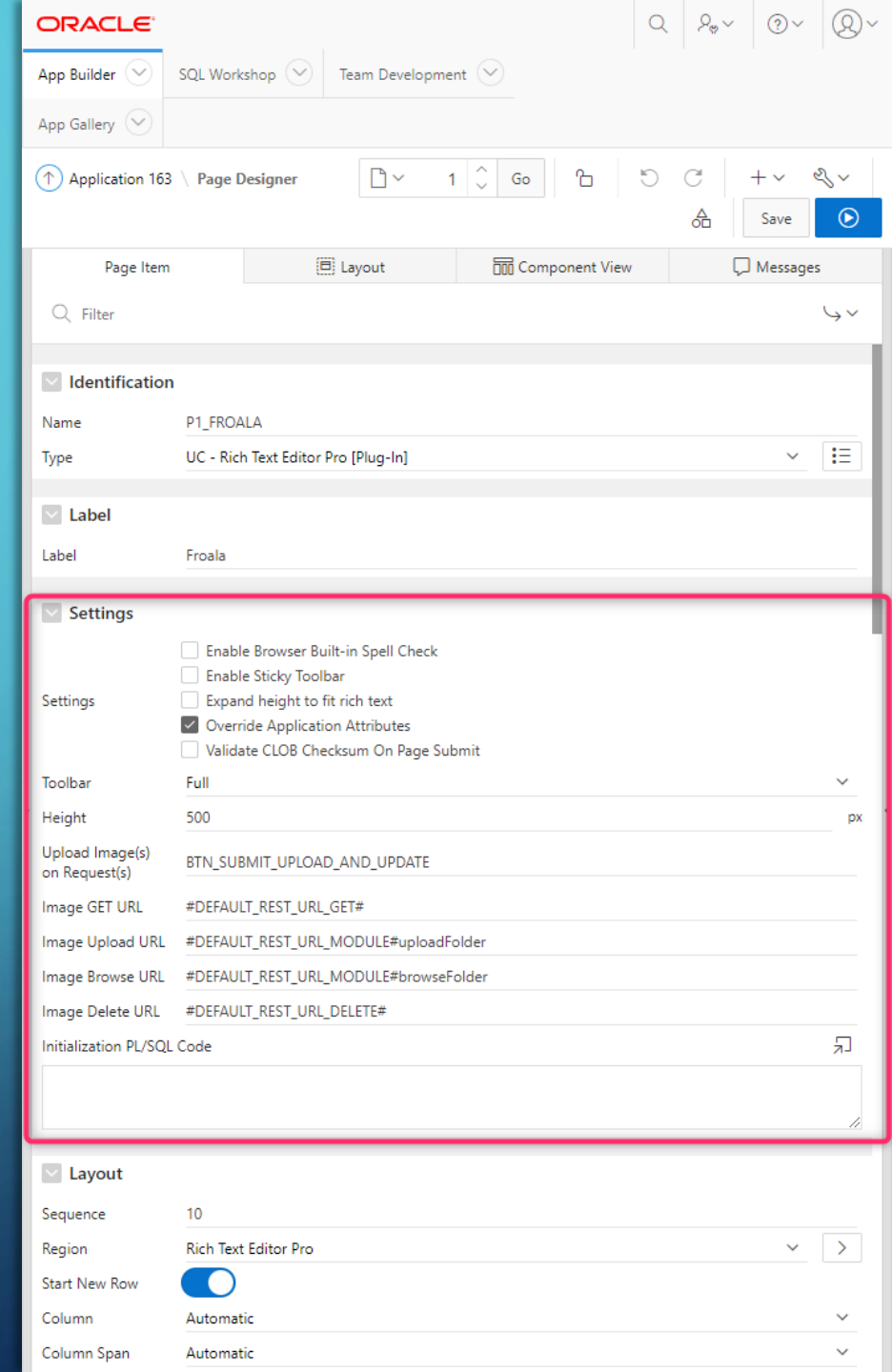
Below the settings dialog, the 'Rich Text Editor Pro' interface is shown. It features a top toolbar with various editing options like Paragraph, Font Family, Font Size, Bold, Italic, Underline, Link, and Image. The main text area contains a paragraph of text about the editor's integration with APEX. Below the text area, there is a diagram titled 'Editing Focused on UX' showing the workflow of the editor. At the bottom, a section titled 'Package of plug-ins' lists the features of the editor, including its integration with native dynamic actions, support for standard attributes, and exposure of valuable events.

Page item plug-in	Dynamic Action plug-in	Process plug-in
<ul style="list-style-type: none">Integrated with native dynamic actionsSupports standard attributesExposes valuable events	<ul style="list-style-type: none">5 actions executed on demand12 settings to customize the editor8 options to customize the editor toolbar	<ul style="list-style-type: none">6 processes types to handle CLOB before APEX 22.22 processes handling images

Characters : 1063

STATISTICS & INFO/ RICH TEXT EDITOR PRO

- Page item plug-in
- 2 500 h
- 15 000 lines in JavaScript
- 10 000 lines in PL/SQL



STATISTICS & INFO/ RICH TEXT EDITOR PRO

- ~100 documentation pages
- Concepts behind plug-in explained
- Step by step guiliness
- Code examples ready to copy

united codes
Plug-ins Pro

Rich Text Editor Pro

Docs

Guides

Live demo

Buy now

Links

Guidelines

Download

Installation

Implementation

Customizations

Toolbar

Images

Upload Workflow

RESTful service

Upload an image

Display images

Browse images

Delete and image

Customizations

RESTful service

Upload an image

Template handler uploading images

This document covers sample implementation for upload template POST handler [uploading image submission](#) or on demand using supporting dynamic action [Upload Images on Demand](#). All example images to the plug-in sample application table [UC_FROALA_SAMPLE_BLOBS](#). The difference between complexity of PL/SQL code, whether handler uses default query parameters, custom parameters or checksum to secure display handler.

Simple

Default parameters

Custom parameters

Image checksum

1

HANDLER FEATURES

- can be combined with handler displaying an image Simple
- can be combined with handler browsing images Simple
- doesn't use default parameters and the URL to displaying image handler (page item attribute URL) is hardcoded in the code

Requirements

The upload handler must meet the following requirements:

1. must use POST method
2. must be compilable at the run time
3. must return 200 HTTP status code
4. must print JSON describing uploaded image

Handler definition

Simple

Default parameters

Custom parameters

Image checksum

Attribute	Value
RESTful Service Module	A custom RESTservice
Module Base Path	/customrest/
URI Template	upload-simple
Full URL	https://example-domain.com/ords/workspacename/customrest/ords/upload-simple
Handler Method	POST
Source Type	PL/SQL
Mime Types Allowed	Not defined

Handler parameters

The upload handler doesn't require registering any resource parameters as long as REST specific `status_code` is used to set request HTTP status.

Handler Source

Simple

Default parameters

Custom parameters

Image checksum

```
declare
v_image_blob blob := :body;
v_image_mime_type varchar2(4000) := :content_type;
v_image_link varchar2(4000);
v_file_id number;

/* without default parameters image filename is not available */
v_file_name varchar2(4000) := 'filename';
begin

insert into UC_FROALA_SAMPLE_BLOBS(
FILE_NAME,
FILE_CONTENT,
FILE_MIMETYPE,
SESSION_ID
) values(
v_file_name,
v_image_blob,
v_image_mime_type,
null
) returning ID into v_file_id;

/* The URL to RESTful service GET handler is hardcoded */
v_image_link := 'https://example-domain.com/ords/workspacename/customrest/display';
```

Documentation

Plug-in Documentation

The Rich Text Editor Pro documentation covers all plug-in components available after installing it in Oracle APEX application and explains concepts on how the plug-in works.

Concepts

Learn on how plug-in works, what information is available through the plug-in session state, how the plug-in handles images while working with a document and while submitting a page. Also, learn what are the plug-in toolbar capabilities for customization, and how APEX features such as authorization schemes and global page can be used to customize the plug-in.

1

TOPICS

- Learn about the plug-in session state and information stored in APEX collection
- Learn about the plug-in image handling capabilities
- Learn how the plug-in upload and update workflow preparing document to be saved in the database
- Learn about the plug-in toolbar capabilities
- Learn about the plug-in capability to be configured depending on user roles
- Learn on how to leverage APEX application global page to customize the plug-in

Page Item plug-in

Learn about page item plug-in implementing Rich Text Editor Pro. What attributes are exposed in application component settings, and what can be configured on page level using APEX page designer.

1

TOPICS

- Explore which standard page item attributes are used by the plug-in
- Explore the plug-in custom attributes exposed in the page designer
- Explore the plug-in JavaScript events exposed for the page designer
- APEX Integration

Process plug-in

Learn about the supporting process plug-in exposing processes to handle CLOB values across APEX versions, and two processes for image handling in the database after submitting a page.

1

TOPICS

- Clear CLOB
- Load CLOB
- Update CLOB
- Image Handling

Dynamic Action plug-in

Learn about the supporting dynamic action plug-in exposing dozen options to customize the plug-in using dynamic action interface. The dynamic action plug-in exposes also actions on demand to manipulate the plug-in session state or upload images on end-user demand without submitting a page.

1

TOPICS

Configuration:

- Learn how to use dynamic action attributes
- Change Query Parameters
- Change Settings
- Change Toolbar

Action on demand:

- Learn how to use dynamic action attributes
- Reset Value
- Update CLOB
- Upload Images on demand

Database Objects

Learn about all database objects created in APEX schema after installing the plug-in.

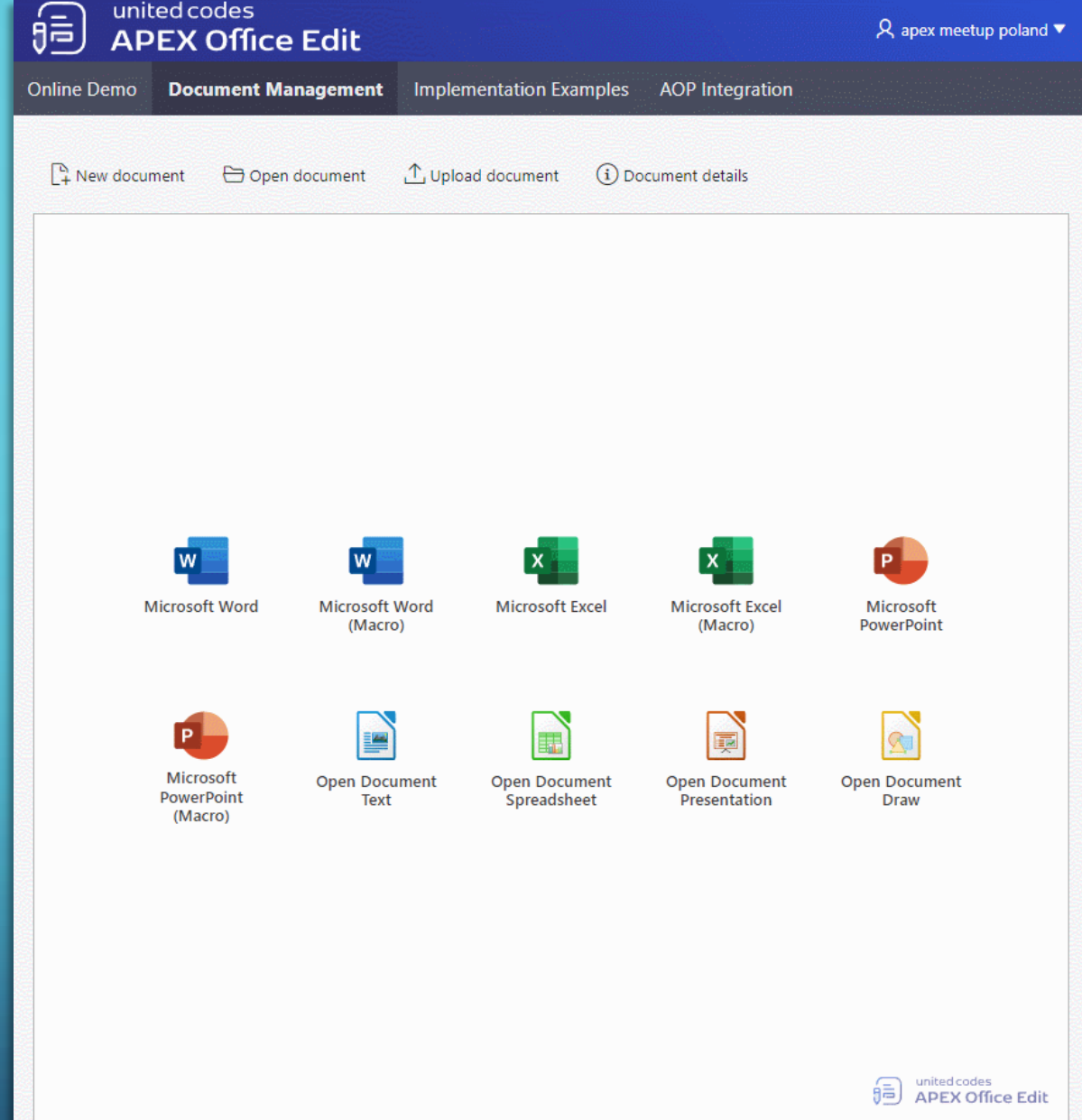
1

TABLE OF CONTENTS

- Tables, sequences and triggers
- Access Token Type
- Package UC_FROALA_RTE
- Package UC_FROALA_SETTINGS
- Package UC_FROALA_SAMPLE_APP
- Package UC_FROALA_SETTINGS_APP

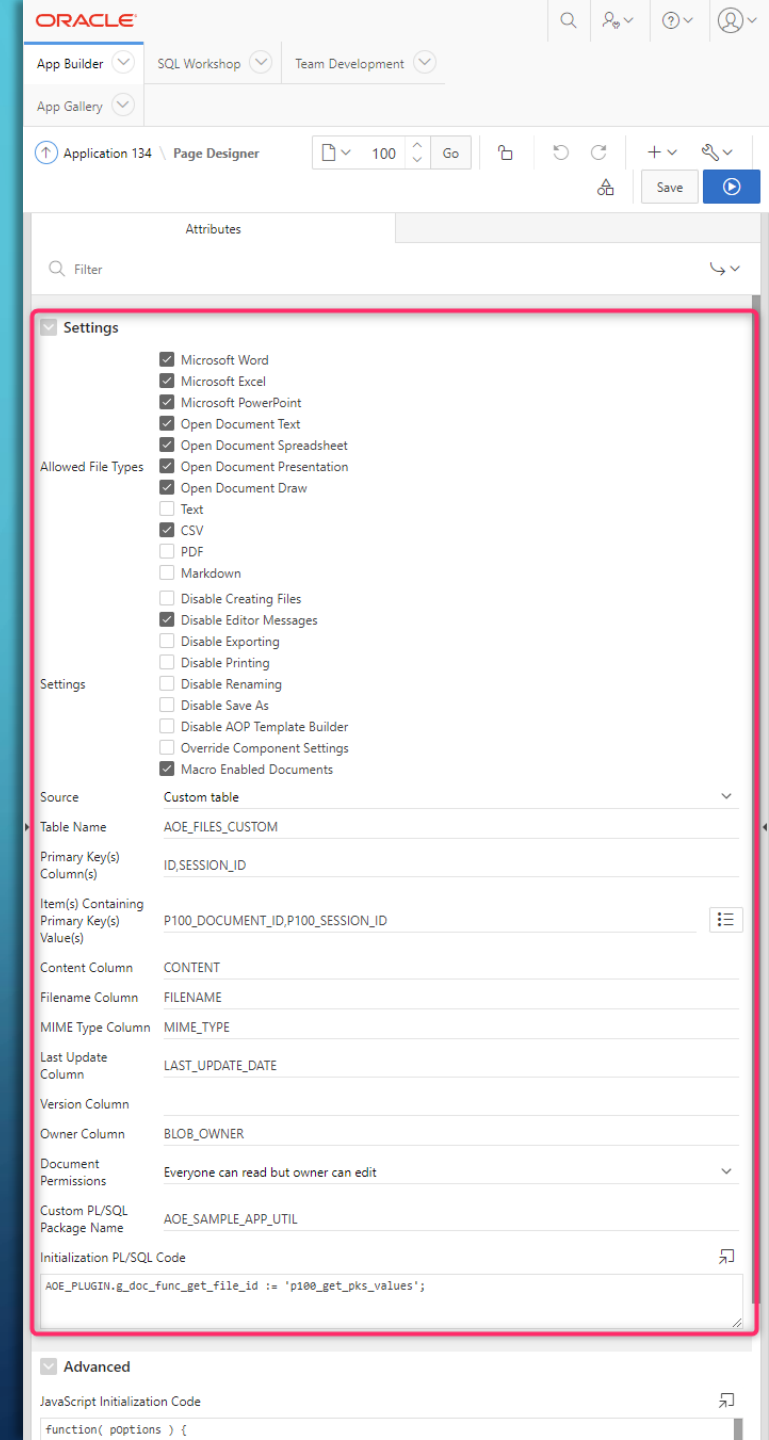
REQUIREMENTS/ APEX OFFICE EDIT

- Integrate Libre Office
- Handle communication between APEX – ORDS – AOE server
- File BLOB processing based on custom table definition
- File BLOB processing using developer's PL/SQL code



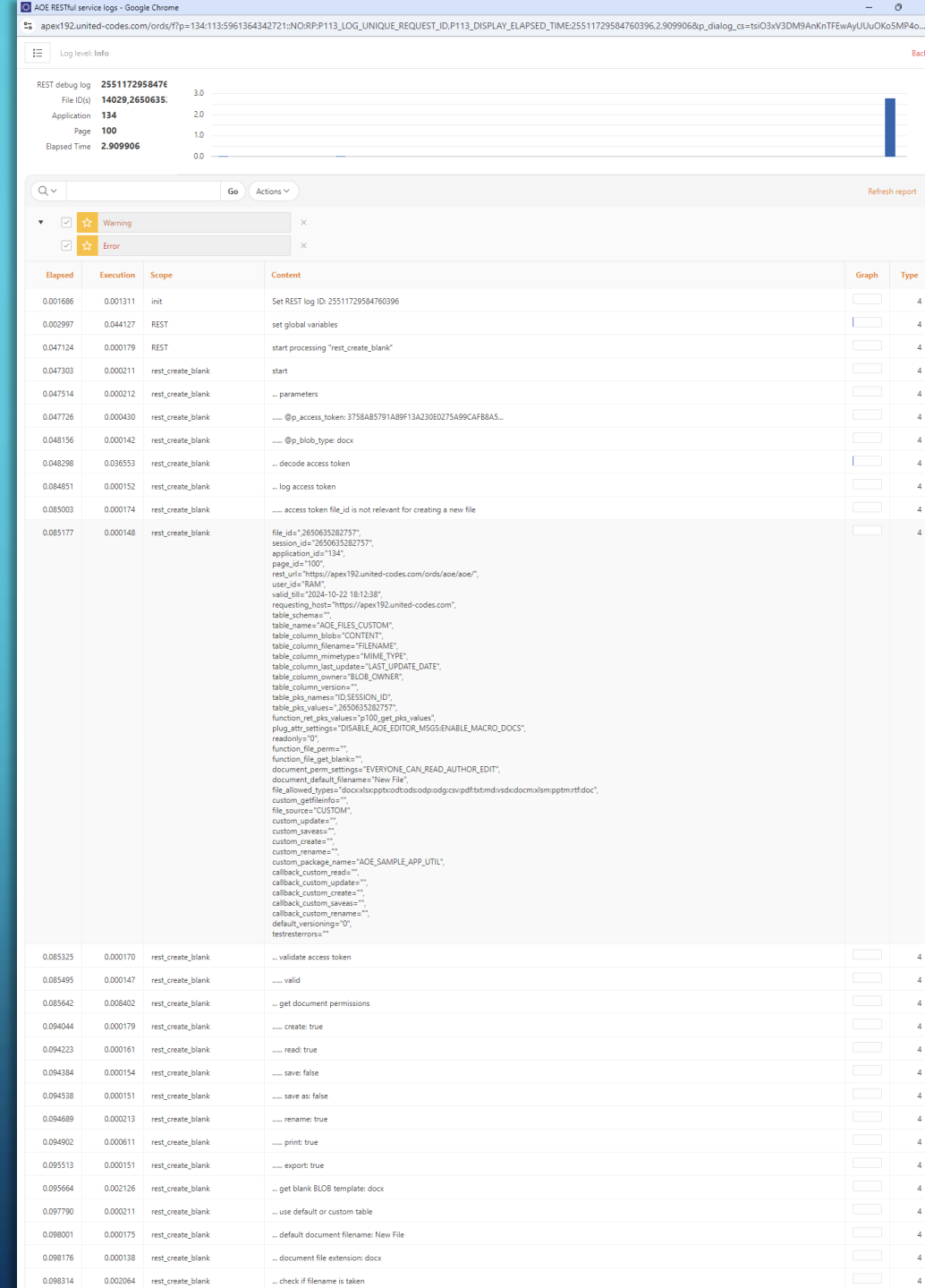
STATISTICS & INFO/ APEX OFFICE EDIT

- Region plug-in
- Allows editing MS documents in APEX
- ~1 200 h
- ~15 100 JavaScript
- ~11 500 PL/SQL



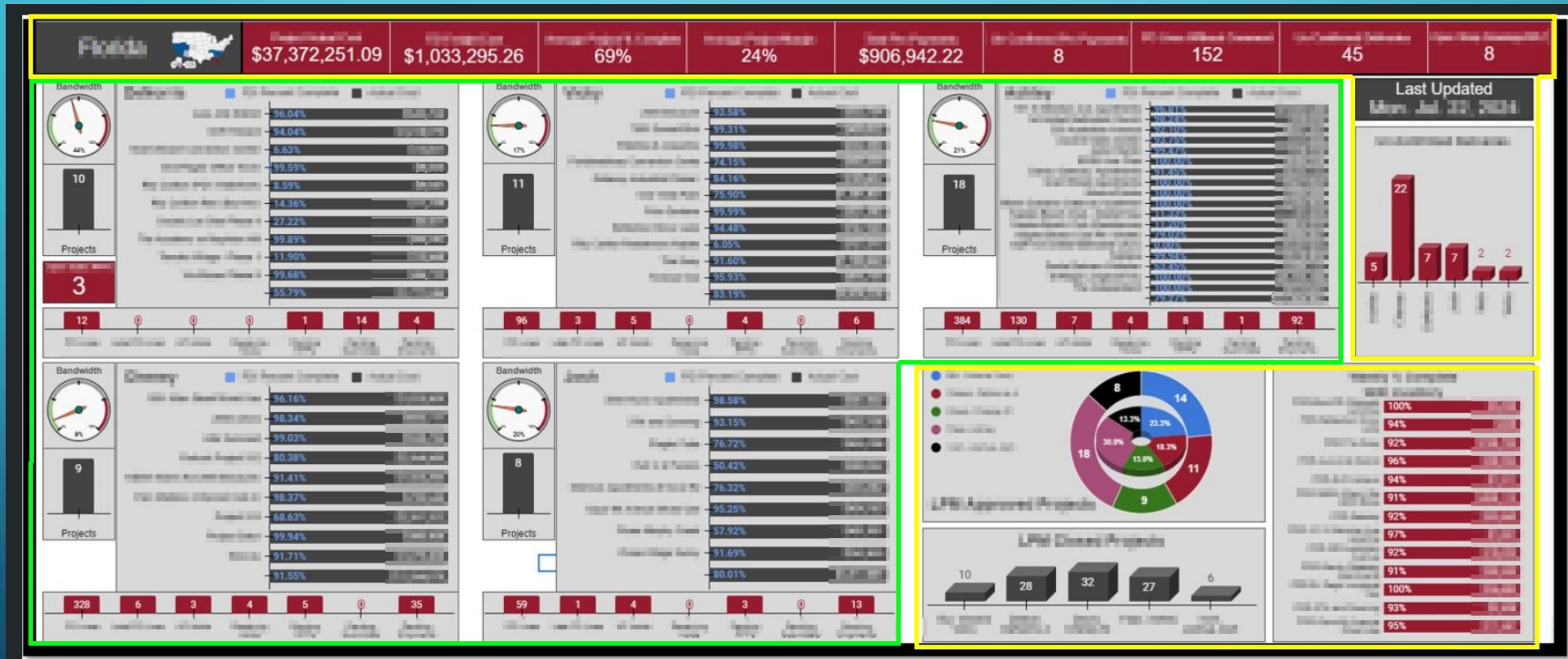
FUN FACTS/ APEX OFFICE EDIT

- Don't build your own debug message for ORDS RESTful handlers
- Debug messages must be „saved“ „apex_session.detach;“



Elapsed	Execution	Scope	Content	Graph	Type
0.001686	0.001311	init	Set REST log ID: 25511729584760396		4
0.002997	0.044127	REST	set global variables		4
0.047124	0.000179	REST	start processing "rest_create_blank"		4
0.047303	0.000211	rest_create_blank	start		4
0.047514	0.000212	rest_create_blank	... parameters		4
0.047726	0.000430	rest_create_blank @p_access_token: 3750A85791A89F13A230E0275A99CAF8BA5...		4
0.048156	0.000142	rest_create_blank @p_blob_type: docx		4
0.048298	0.036553	rest_create_blank	... decode access token		4
0.084851	0.000152	rest_create_blank	... log access token		4
0.085003	0.000174	rest_create_blank access token file_id is not relevant for creating a new file		4
0.085177	0.000148	rest_create_blank	file_id="2650635282757", session_id="2650635282757", application_id="134", page_id="100", rest_url="https://apex192.united-codes.com/ords/aoe/aoe", user_id="SAM", valid_till="2024-10-22 18:12:38", requesting_host="https://apex192.united-codes.com", table_schemas="*", table_name="AOE_FILES_CUSTOM", table_column_blobs="CONTENT", table_column_filenames="FILENAME", table_column_mimetypes="MIME_TYPE", table_column_last_updates="LAST_UPDATE_DATE", table_column_owners="BLOB_OWNER", table_column_versions="*", table_pk_names="ID,SESSION_ID", table_pk_values="2650635282757", function_get_pk_values="p100_get_pk_values", plug_attr_settings="DISABLE_AOE_EDITOR_MISSES_ENABLE_MACRO_DOCS", readonly="0", function_file_perms="*", function_file_get_blank="*", document_perm_settings="EVERYONE_CAN_READ_AUTHOR_EDIT", document_default_filename="New File", file_allowed_types="docxdocxpptpptodtdodsodpdocxpdfbtxtdmdvsdtdocxslmpptmrtfdoc", custom_get_filename="*", file_source="CUSTOM", custom_updates="*", custom_saves="*", custom_create="*", custom_rename="*", custom_package_name="AOE_SAMPLE_APP_UTIL", callback_custom_read="*", callback_custom_update="*", callback_custom_create="*", callback_custom_save="*", callback_custom_rename="*", default_versioning="0", testresterrors="0"		4
0.085325	0.000170	rest_create_blank	... validate access token		4
0.085495	0.000147	rest_create_blank valid		4
0.085642	0.008402	rest_create_blank	... get document permissions		4
0.094044	0.000179	rest_create_blank create: true		4
0.094223	0.000161	rest_create_blank read: true		4
0.094384	0.000154	rest_create_blank save: false		4
0.094538	0.000151	rest_create_blank save as: false		4
0.094689	0.000213	rest_create_blank rename: true		4
0.094902	0.000611	rest_create_blank print: true		4
0.095513	0.000151	rest_create_blank export: true		4
0.095664	0.002126	rest_create_blank	... get blank BLOB template: docx		4
0.097790	0.000211	rest_create_blank	... use default or custom table		4
0.098001	0.000175	rest_create_blank	... default document filename: New File		4
0.098176	0.000138	rest_create_blank	... document file extension: docx		4
0.098314	0.002064	rest_create_blank	... check if filename is taken		4

REQUIREMENTS/ DYNAMIC DASHBOARD

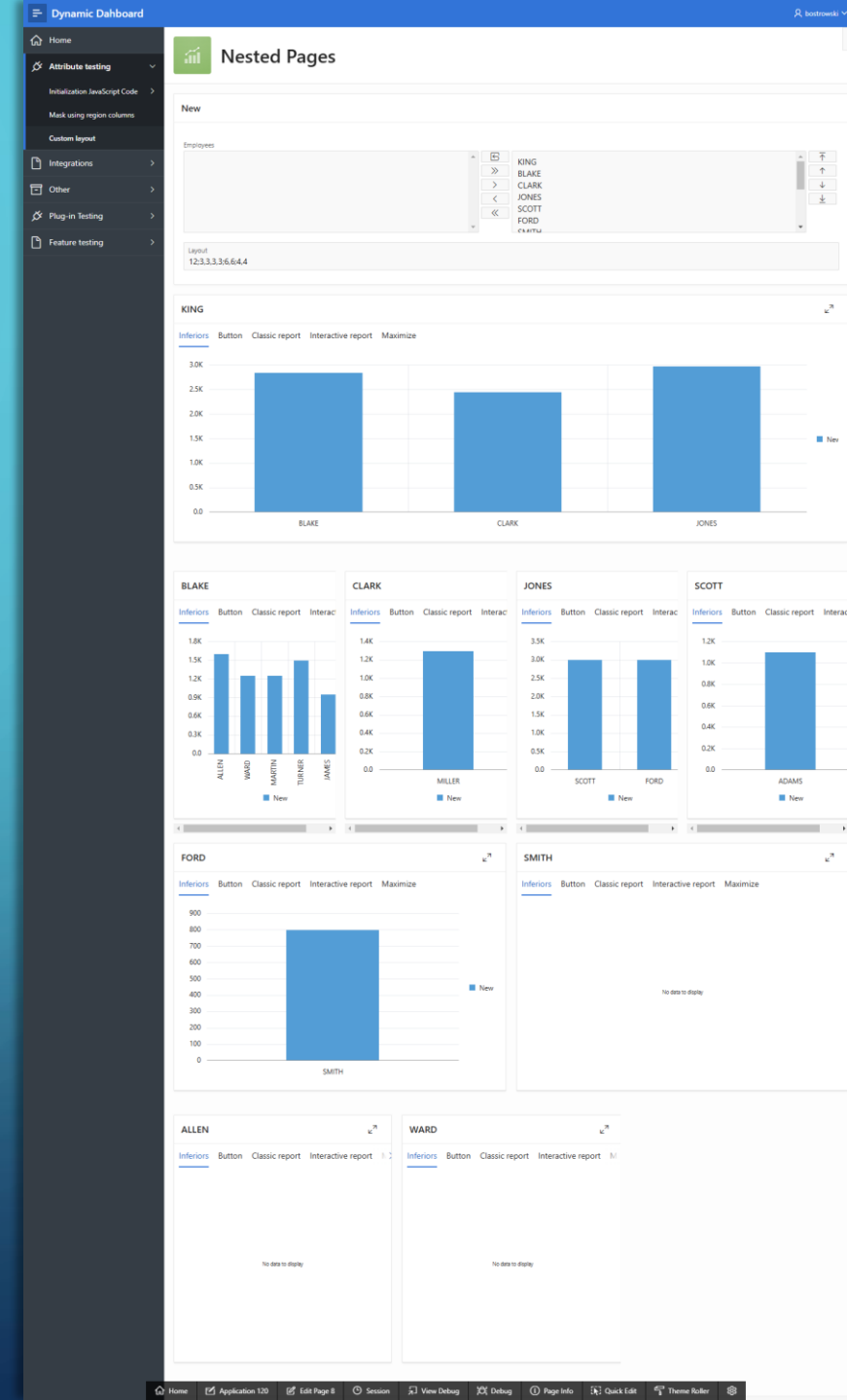


Region number depends on data in the database

Static components

WORK IN PROGRESSS/ DYNAMIC DASHBOARD DEMO

- Data driven dashboard
- Native components support
- Citizen Development
- Dynamic layout using APEX 12 column grid
- ACL support



END/ LAST THOUGHTS

- JavaScript and PL/SQL debug messages are your friend
- Explore APEX internals yourself (tables, views, APIs)
- Learn from work of other developers (plug-ins at apex.world)
- APEX community will help you at [twitter](https://twitter.com/apexworld) / apex.world slack

The background is a blue gradient. In the corners, there are white line-art illustrations of circuit boards or neural networks, with lines and small circles representing nodes.

THANK YOU FOR YOUR TIME

Twitter: [@bostrowsk1](#)

E-mail: bartosz@united-codes.com

E-mail: ostrowski.bartosz@gmail.com