

# Contents

<b>Overview</b>	<b>2</b>
Introduction	2
Display	2
Features at a Glance	2
Found a bug	2
License	3
Legal Disclaimer	3
<b>Installation</b>	<b>3</b>
Install required package	3
Install plugin	4
Using in your application	5
<b>Configuration Settings</b>	<b>9</b>
Data Return	9
Include Page Items	9
List of Items to Include	9
Apex Collection Name	9
Maximum Rows	9
<b>Getting results</b>	<b>10</b>
<b>Installing sample application</b>	<b>10</b>

### Overview

#### Introduction

On the one hand, standard download functionality implemented into APEX Interactive Reports is very simple and supports only basic features. On the other hand, end users need eye-candy looking reach reports. Therefore report developers are forced to use third-party tools (BI Publisher, Apache FOP). While constructing such reports many problems with correct displaying of users customizations arise, because there are no standard way to get prepared dataset from Interactive Report.

#### Display

APEX\_IR.GET\_REPORT function from APEX-API has only restricted IR-feature support: features «compute», «aggregate», «calculations», «highlights» are not supported.

The "**GPV Interactive Report to XML**" APEX plugin provide simple access to IR-data and save them in XML-format, prepared to future processing. This solution is simple to use and don't require the knowledge of Interactive Report internal.

#### Features at a Glance

These Interactive Reports features are supported:

- Filtering and Sorting
- Control Break
- Highlights
- Computations
- Aggregations
- Simple data types of column (String/Date/Number)
- Column and header alignment

Possibly is to include in XML values of page or application items.

#### Found a bug

If you have found a bug, please send us debug information. To do this, set **Return Data** plugin options to "**Debug Information**". Try to get your XML data again. You'll get text a file with debug information. Please send this file to [apex.plugins@outlook.com](mailto:apex.plugins@outlook.com). We'll try to fix this error as soon as possible.

# GPV Interactive Report to XML

## License

The "GPV Interactive Report to XML" APEX plugin and IR\_TO\_XML package is currently available for use in all personal or commercial projects under both MIT and GPL licenses. This means that you can choose the license that best suits your project and use it accordingly. Both licenses have been included with this software.

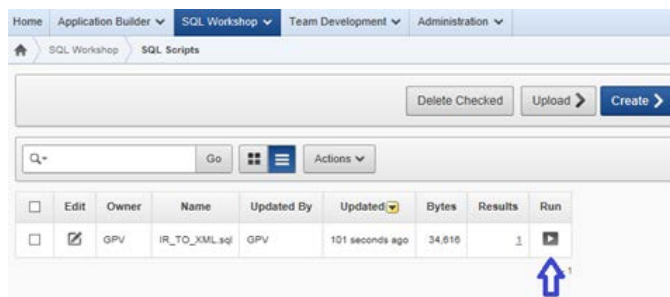
## Legal Disclaimer

The program(s) and/or file(s) are supplied as is. The author disclaims all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The author assumes no liability for damages, direct or consequential, which may result from the use of these program(s) and/or file(s).

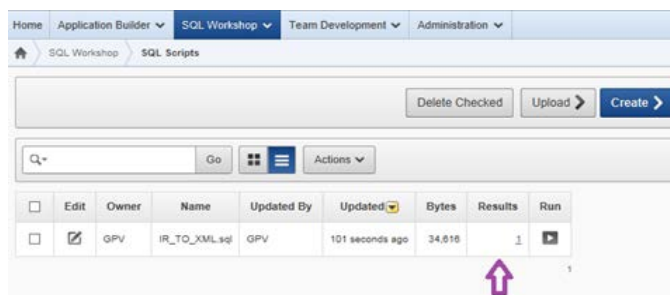
## Installation

### Install required package

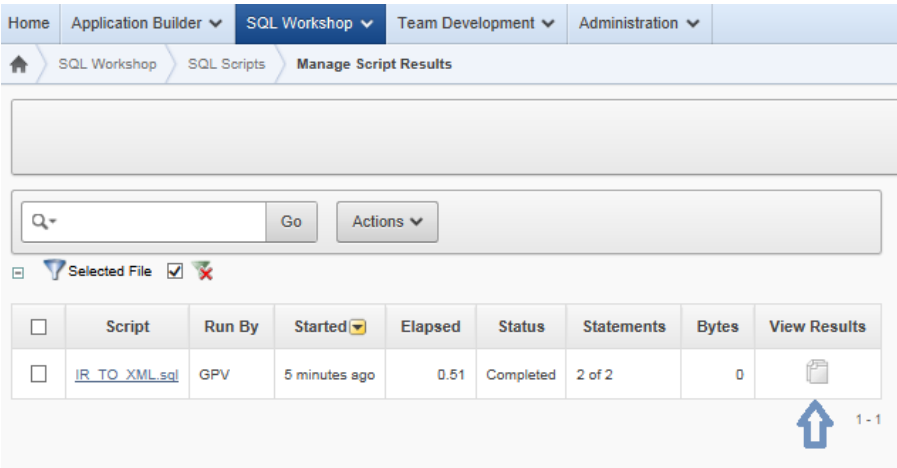
This plugin requires IR\_TO\_XML package. To install this package navigate to the "SQL-Workshop > SQL-Scripts". Upload file **IR\_TO\_XML.sql**.



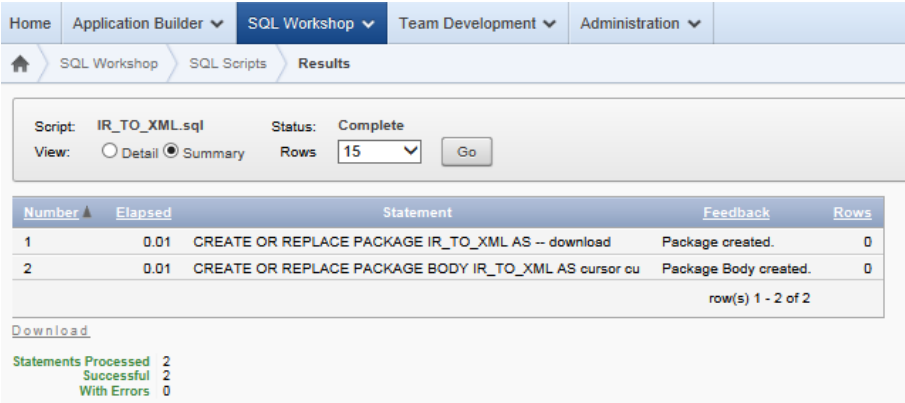
This script will be created IR\_TO\_XML package. Run this script by pressing Run-icon.



# GPV Interactive Report to XML



After execution take a look at execution results. At first open “*Manage Script Results*” tab by pressing on link in **View Results** column.



There should be no errors.

### Install plugin

Navigate to “*Shared Components > Plug-ins*” and click **Import**. Herefrom you can follow the menu to upload plugin-file “**process\_type\_plugin\_gpv\_ir\_xml.sql**” and to install the plug-in using the file above.

## GPV Interactive Report to XML

### Using in your application

1. At first you should create the button will start the plugin on page with Interactive Report.

The screenshot shows the 'Create Button' wizard in the Oracle APEX Application Builder. The breadcrumb trail is: Home > Application Builder > Application 102 > Page 1 > Create Button. The wizard has six steps: Button Region, Button Position, Button Attributes (current), Display Properties, Action When Clicked, and Conditional Display. The 'Page' is '1 - Plugin Demonstration' and the 'Region' is 'Sample Interactive Report'. The 'Button Name' is 'YOUR\_BUTTON\_NAME' and the 'Label' is 'Get XML from Interactive Report'. The 'Button Style' is 'HTML Button'. There are 'Cancel' and 'Create Button' buttons, and a 'Next >' button.

Enter “*Button name*” and “*Label*”.

**Please note the name you given; this will be used later.**

The screenshot shows the 'Create Button' wizard in the Oracle APEX Application Builder, now at the 'Display Properties' step. The breadcrumb trail is: Home > Application Builder > Application 102 > Page 1 > Create Button. The wizard has six steps: Button Region, Button Position, Button Attributes, Display Properties (current), Action When Clicked, and Conditional Display. The 'Page' is '1 - Plugin Demonstration' and the 'Region' is 'Sample Interactive Report'. The 'Button Name' is 'P1\_GET\_XML\_FROM\_IR' and the 'Sequence' is '10'. There are 'Cancel' and 'Create Button' buttons, and a 'Next >' button.

Press “**Next**”.

## GPV Interactive Report to XML

Home Application Builder SQL Workshop Team Development Administration Search Application

Application Builder Application 102 Page 1 Create Button Page 1

Button Region Button Position Button Attributes Display Properties **Action When Clicked** Conditional Display

Cancel Create Button Next

Page: 1 - Plugin Demonstration

Region: Sample Interactive Report

Button Name: P1\_GET\_XML\_FROM\_IR

Action: Redirect to Page in this Application

\* Page: &APP\_PAGE\_ID.

☐ reset pagination for this page

Request: YOUR\_BUTTON\_NAME

Clear Cache: (comma separated page numbers)

Set these items: (comma separated name list)

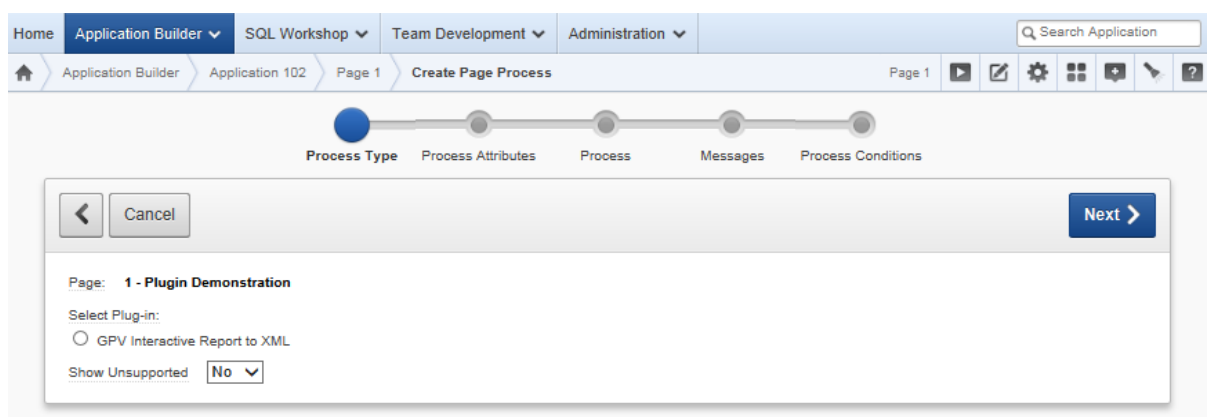
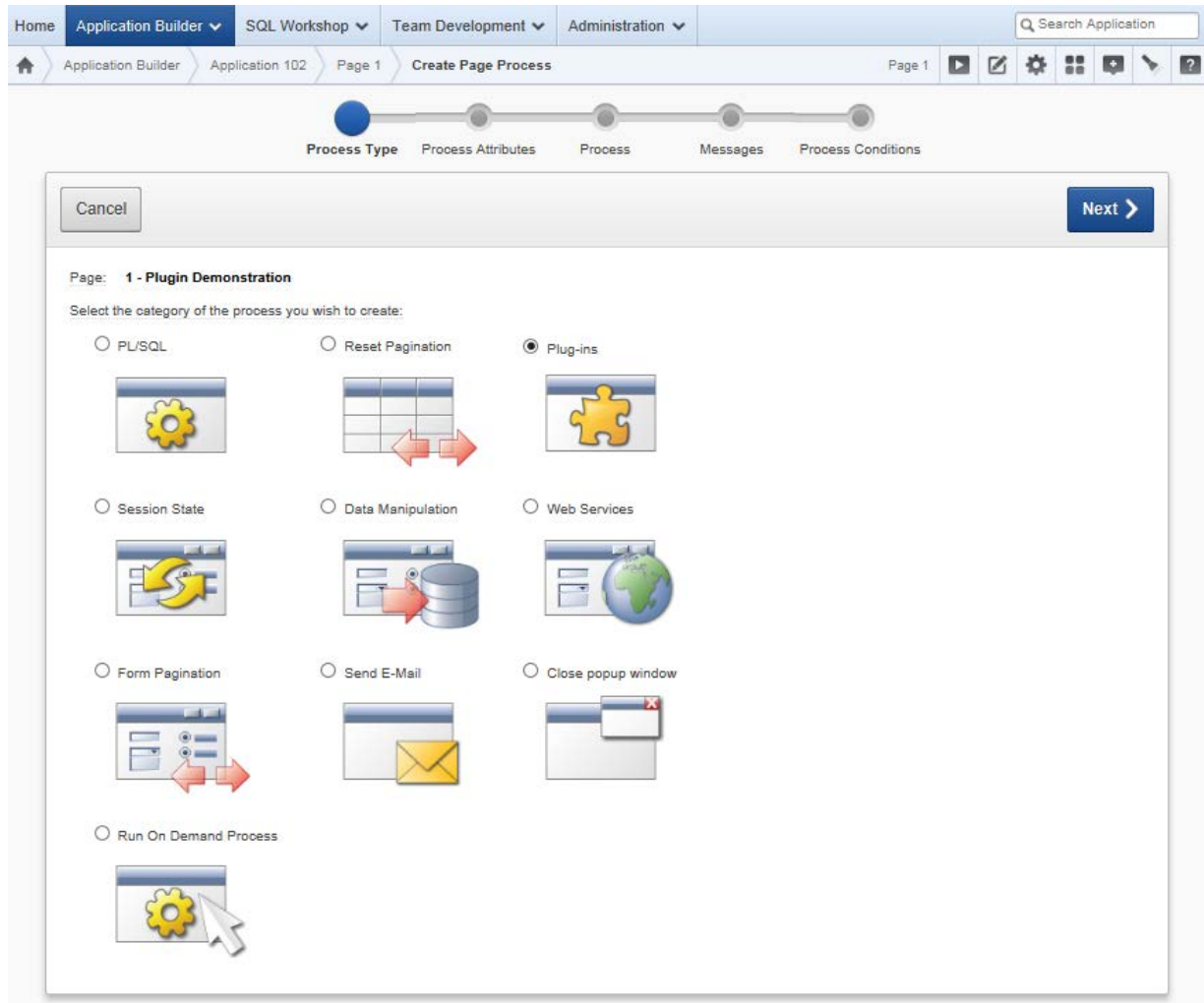
With these values: (comma separated value list)

Enter **&APP\_PAGE\_ID.** in field "Page".

Set "Request" : **name of the button** given on step 1. Press **Create Button**.

## GPV Interactive Report to XML

2. At second create “On Load - Before Header”-process “**Plug-Ins**” type.



Select “**GPV Interactive Report to XML**”.

## GPV Interactive Report to XML

The screenshot shows the 'Process Attributes' step of a five-step wizard. The steps are: Process Type, Process Attributes (current), Process, Messages, and Process Conditions. The 'Process Attributes' section contains the following fields:

- Page: 1 - Home
- Name: GetXMLfromIR Plugin
- Sequence: 10
- Point: On Load - Before Header
- Type: GPV Interactive Report to XML [Plug-in]

Navigation buttons include a back arrow, 'Cancel', and 'Next >'.

Give this plugin a name (for example *"GetXMLfromIR Plugin"*).  
Select execution *"Point"*: *"On Load - Before Header"*

The screenshot shows the 'Process' step of the wizard. The steps are: Process Type, Process Attributes, Process (current), Messages, and Process Conditions. The 'Process' section contains the following fields:

- Page: 1 - Home
- Point: On Load - Before Header
- Return Data: Generated XML-data
- Include Page Items: Yes
- List of Items to Include: (empty text box)
- APEX Collection Name: (empty text box)
- Maximum Rows: 1000

Navigation buttons include a back arrow, 'Cancel', 'Create Process', and 'Next >'.

> Page Items

Press **"Next"**.

Skip next Page by pressing **"Next"**.



## GPV Interactive Report to XML

The screenshot shows the 'Process Conditions' step of a wizard. At the top, a progress bar indicates five steps: Process Type, Process Attributes, Process, Messages, and Process Conditions (the current step). Below the progress bar, there are navigation buttons: a back arrow, 'Cancel', and 'Create Process'. A text box explains: 'This process executes in sequence and only if the conditions identified in this step are met. You can specify a condition by making a selection from the When Button Pressed list. Any condition specified must be met before the page process executes. When Button Pressed conditions are only relevant for processes of type After Submit.'

The configuration fields are as follows:

- Page:** 1 - Plugin Demonstration
- Point:** On Load - Before Header
- Condition Type:** A dropdown menu showing 'Request = Expression 1'. Below it, a list of available condition types is shown: [PL/SQL], [item / column=value], [item / column not null], [item / column null], [request=e1], [page in], [page not in], [exists], [never], [none].
- Expression 1:** A text area containing the text 'YOUR\_BUTTON\_NAME'.

At the bottom, there is a link '> Page Items'.

Set **"Condition Type"** : **"Request = Expression 1"**

Set **"Expression 1"** : the **button name** given in step 1.

Press **"Create Process"**.

## Configuration Settings

### Data Return

Choose "Generated XML-data" to return result of Interactive Report in XML format.

Choose "Debug information" to get debug information. Use this option to get more info in case of error.

### Include Page Items

Include all items with their values belonging to current page in XML.

### List of Items to Include

Comma delimited list of page or application items with their values will be included in XML.

### Apex Collection Name

Enter the name of APEX collection where XML-data will be saved.

When empty - result will be downloaded as file.

### Maximum Rows

Rows greater than this value will not be exported.

To export all the rows set the value to 1000000000.

### Getting results

You can get your XML-data as a file ready to download or as a CLOB in APEX-Collection

To get the result as XML file leave plugin parameter **APEX Collection Name** empty.

To get the result in CLOB just enter the name of your APEX Collection in **APEX Collection Name** plugin parameter field. If Collection does not exist, the plugin will create it automatically.

To get your XML please use this query:

```
SELECT clob001
FROM apex_collections
WHERE collection_name = 'YOUR_COLLECTION_NAME'
AND seq_id = 1
```

More info about APEX Collections you can find in [Oracle APEX Documentation](#)

### Installing sample application

You can find sample application in SAMPLE\_APPLICATION folder in plugin archive.

To install sample application please follow this [Instructions](#).