

# Contents

**Overview ..... 3**

**Introduction ..... 3**

**Features at a Glance ..... 3**

**License ..... 3**

**Legal Disclaimer ..... 3**

**Found a bug ..... 3**

**Installation ..... 4**

**Install required package ..... 4**

**Install plugin..... 5**

**Using in your application ..... 6**

**Configuration Settings..... 8**

**Return Data..... 8**

**Maximum Rows ..... 8**

**Download when click on (jQuery Selector)..... 8**

**Installing sample application ..... 9**

**FAQ (How To) ..... 9**

**Replace default file name (Excel.xlsx) to the custom file name..... 9**

**Easy implement this functionality to all Interactive report in your application9**

**Set custom columns widths .....10**

**Adding custom download button .....11**

**How to exclude reports columns from export/make columns that are visible in Excel but not visible in Interactive Report .....12**

# GPV Interactive Report to MSeXcel

---

## Overview

### Introduction

The "GPV Interactive Report to Excel" APEX plugin provide simple export of IR-data into Microsoft Excel.

### Features at a Glance

These Interactive Reports features are supported:

- Correct export of data types (String/Date/Number)
- Filtering and Sorting
- Control Break
- Computations
- Aggregations
- Highlighting
- Auto adjustable Column Width
- Date/Number Formats
- Custom downloaded file name (see FAQ)
- Can be easy implemented for all Interactive reports in application (see FAQ)

### License

The "GPV Interactive Report to MSeXcel" APEX plugin and IR\_TO\_XML, XML\_TO\_XSLX, IR\_TO\_MSEXCEL packages are 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).

### Found a bug

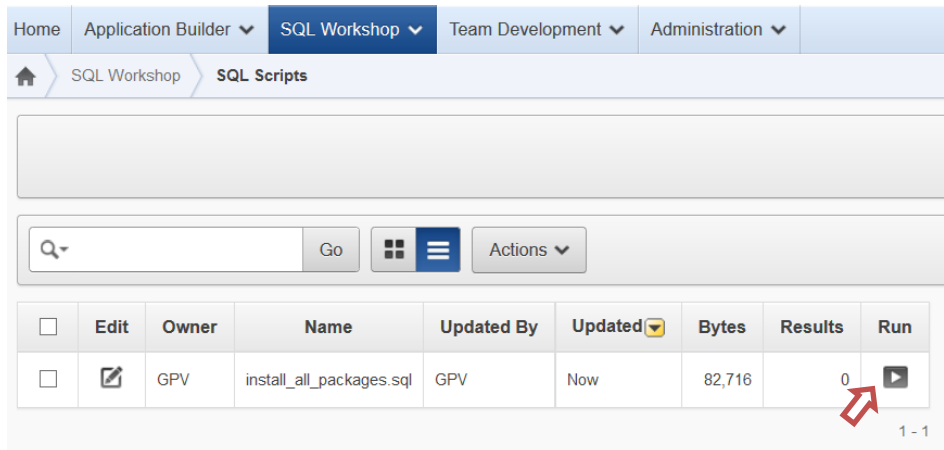
If you have found a bug, please send us debug information. To do this, set **Return Data** plugin options to **"Debug Information"**. Try to get your XML data again. You'll get text a file with debug information. Please send this file to [pavel.glebov@outlook.com](mailto:pavel.glebov@outlook.com). We'll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

### Installation

#### Install required package

This plugin requires IR\_TO\_XML, XML\_TO\_XSLX, IR\_TO\_MSEXCEL and AS\_ZIP packages. To install these packages, please use install\_all\_packages.sql all-in-one installation script.

To do this, navigate to the “SQL-Workshop > SQL-Scripts”. Upload file **install\_all\_packages.sql**.

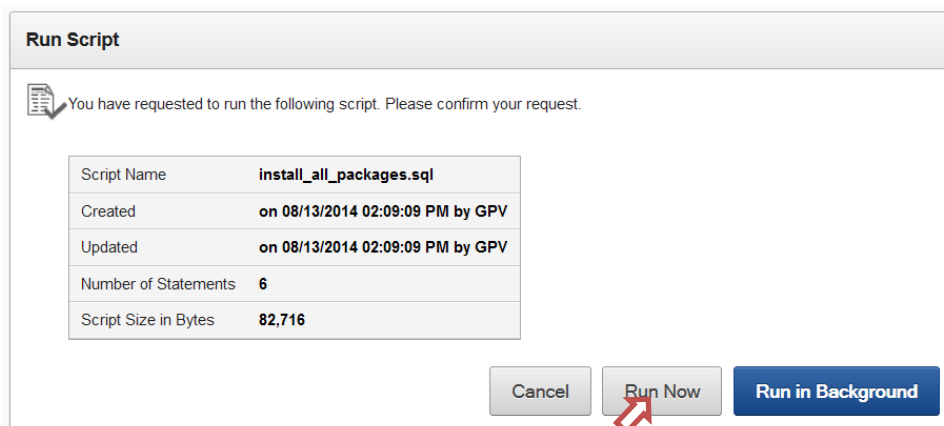


The screenshot shows the Oracle SQL Workshop interface. The breadcrumb navigation is Home > Application Builder > SQL Workshop > SQL Scripts. Below the navigation is a search bar with a 'Go' button and an 'Actions' dropdown. A table lists the scripts:

<input type="checkbox"/>	Edit	Owner	Name	Updated By	Updated	Bytes	Results	Run
<input type="checkbox"/>		GPV	install_all_packages.sql	GPV	Now	82,716	0	

A red arrow points to the 'Run' icon in the table. The page number '1 - 1' is visible at the bottom right.

This script will be created all required packages. Run this script by pressing Run-icon.



The 'Run Script' dialog box displays the following information:

Script Name	install_all_packages.sql
Created	on 08/13/2014 02:09:09 PM by GPV
Updated	on 08/13/2014 02:09:09 PM by GPV
Number of Statements	6
Script Size in Bytes	82,716

At the bottom, there are three buttons: 'Cancel', 'Run Now', and 'Run in Background'. A red arrow points to the 'Run Now' button.


Press “Run Now” button.

# GPV Interactive Report to MSExcel

HomeApplication BuilderSQL WorkshopTeam DevelopmentAdministration

SQL WorkshopSQL ScriptsManage Script Results

QGoActions

<input type="checkbox"/>	Script	Run By	Started	Elapsed	Status	Statements	Bytes	View Results
<input type="checkbox"/>	<a href="#">install_all_packages.sql</a>	GPV	2 minutes ago	1.06	Completed	6 of 6	0	

1 - 1

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

HomeApplication BuilderSQL WorkshopTeam DevelopmentAdministration

SQL WorkshopSQL ScriptsResults

Script: install\_all\_packages.sqlStatus: Complete

View: ☐ Detail ☒ SummaryRows: 15Go

Number	Elapsed	Statement	Feedback	Rows
1	0.63	CREATE OR REPLACE package ir_to_xml as --ver 1.4. --	Package created.	0
2	0.80	CREATE OR REPLACE package body ir_to_xml as subtype	Package Body created.	0
3	0.06	CREATE OR REPLACE package as_zip is /*****	Package created.	0
4	0.01	CREATE OR REPLACE package body as_zip is -- c_LOCAL_FILE_H	Package Body created.	0
5	0.05	CREATE OR REPLACE package xml_to_xslx -- ver 1.0. IS WIDTH	Package created.	0
6	0.35	CREATE OR REPLACE PACKAGE body XML_TO_XSLX is STRING_HEIGH	Package Body created.	0
7	0.02	CREATE OR REPLACE package IR_TO_MSEXCEL as function get_x	Package created.	0
8	0.03	CREATE OR REPLACE package body IR_TO_MSEXCEL as XLS_DOW	Package Body created.	0

row(s) 1 - 8 of 8

Download

Statements Processed 8  
Successful 8  
With Errors 0

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 MSExcel

### Using in your application

1. Create “On Load - Before Regions”-process “**Plug-Ins**” type.

Process Type Process Attributes Process Messages Process Conditions

Cancel Next >

Page: 1 - Plugin Demonstration

Select the category of the process you wish to create:

☐ PL/SQL ☐ Reset Pagination ☒ Plug-ins

☐ Session State ☐ Data Manipulation ☐ Web Services

Process Type Process Attributes Process Messages Process Conditions

< Cancel Next >

Page: 1 - Plugin Demonstration

Select Plug-in:

☒ GPV Interactive Report to XML

Show Unsupported: No

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

Process Type Process Attributes Process Messages Process Conditions

< Cancel Next >

Page: 1 - Plugin Demonstration

\* Name: Get Excel

\* Sequence: 10

\* Point: On Load - Before Regions

\* Type: GPV Interactive Report to XML [Plug-in]

Give this process a name (for example “Get Excel”).  
Select execution “Point”: “**On Load - Before Regions**”

## GPV Interactive Report to MSEXcel

Process Type   Process Attributes   **Process**   Messages   Process Conditions

< Cancel   Create Process   Next >

Page: **1 - Plugin Demonstration**

Point: **On Load - Before Regions**

Return Data: **Generated XML-data** ▾

Include Page Items: **Yes** ▾

List of Items to Include:

APEX Collection Name:

\* Maximum Rows:

"try again" message:

> Page Items

Press **"Create Process"**.

## GPV Interactive Report to MSeExcel

### Configuration Settings

#### Return Data

Choose "Excel XLSX" to return result of Interactive Report as MS-Excel file.

Choose "XML (Debug)" to return result of Interactive Report in XML format.

Choose "Debug TXT" to view debug information.

#### Maximum Rows

Rows greater than this value will not be exported.

To export all the rows set the value to 1000000000.

When empty - value from Interactive Report Attributes-> Maximum Row Count will be used.

Home Application Builder SQL Workshop Team Development Administration

Application Builder Application 26482 Page 1 Interactive Report Attributes

Home Region Definition Report Attributes Saved Reports Print Attributes

Region Name: Sample Interactive Report

**Pagination**

Pagination Type: Row Ranges X to Y

Pagination Display Position: Bottom - Right

Show Null Values as: -

Maximum Row Count: 1000

When more than maximum row data found message

The maximum row count for this report is #MAX\_ROW\_COUNT# rows. Please apply a filter to reduce the number of records in your query.

#### Download when click on (jQuery Selector)

Download starts when clicking on objects that are selected with this JQuery selector.

#### Replace standard IR XLS download

Replace standard IR download XLS menu functionality to download XLSX feature.

Download

Choose report download format:

CSV HTML Email XLS

Cancel

This functionality will be replaced



## GPV Interactive Report to MSeExcel

### Installing sample application

You can find sample application in SAMPLE\_APPLICATION folder in plugin archive.  
To install sample application please follow this [Instructions](#).

### FAQ (How To)

#### Replace default file name (Excel.xlsx) to the custom file name

When not empty, value from *Report Attributes-> Download-> Filename* field will be used.

Home Application Builder SQL Workshop Team Development Administration

Application Builder Application 26482 Page 1 Interactive Report Attributes

Region Definition Report Attributes Saved Reports Print Attributes

**Download**

Download Formats ( \* for authenticated pages only):

☒ CSV ☒ HTML ☒ Email \* ☒ XLS ☐ PDF ☐ RTF

CSV Separator  CSV Enclosed By

Filename

#### Easy implement this functionality to all Interactive report in your application

1. Install plugin and required packages.
2. Create “On Load: After Header” application process.

Home Application Builder SQL Workshop Team Development Administration

Application Builder Application 26482 Shared Components Application Processes Page 1

Application Processes History

Go

No data found.

## GPV Interactive Report to MSeExcel

The top screenshot shows the 'Identification' tab of the APEX application process configuration. It includes a progress bar with three steps: Identification, Source, and Conditionality. The 'Identification' step is active. The configuration details are as follows:

- Application: 26482 Sample Application
- Name: Get Excel from IR
- Sequence: 1
- Point: On Load: After Header (page template header)
- Type: PL/SQL Anonymous Block

The bottom screenshot shows the 'Source' tab of the same configuration. It includes a text area for the PL/SQL code:

```
get_xlsx_from_ir_ext(p_maximum_rows => null,
                    p_jquery_selector => null,
                    p_download_type => 'E', -- E -> Excel XLSX, X -> XML (Debug), T -> Debug TXT
                    p_replace_xls => 'Y', --Y/N
                    p_custom_width => null
                    );
```

Please set your custom parameters or leave default settings.

```
get_xlsx_from_ir_ext(
    p_maximum_rows => null,
    p_jquery_selector => null,
    p_download_type => 'E', -- E -> Excel XLSX,
                        -- X -> XML (Debug),
                        -- T -> Debug TXT
    p_replace_xls => 'Y', --Y/N
    p_custom_width => null
);
```

Parameters description:

- `p_maximum_rows` - see *Maximum Rows* plugin parameter
- `p_jquery_selector` - see *Download when click on (jQuery Selector)* plugin parameter
- `p_download_type` - see *Return Data* plugin parameter
- `p_replace_xls` - see *Replace standard IR XLS download* plugin parameter
- `p_custom_width` - see FAQ

### Set custom columns widths

This functionality is only available, when plugin is implemented as application process.

For this please set `p_custom_width` parameter with required value. This is comma-delimited string with values, each value consist of column name and column width.

## GPV Interactive Report to MSExcel

So, this string

PROJECT=151,TASK\_NAME=319,START\_DATE=133,

set

width = 151 to the column with name = "PROJECT",

width = 319 to the column with name = "NAME",

width = 133 to the column with name = "START\_DATE"

**Please do not forgive comma at the end of the string!**

Width are set in special units that are **approximately** equal to pixels.

### Adding custom download button

1. First create a button with certain **Static ID** and Action **"Defined by Dynamic Actions"**

The screenshot shows two configuration panels for a button. The top panel, titled 'Attributes', contains the following fields: 'Static ID' with the value 'GETEXCEL' (highlighted with an orange box), 'Button Style' set to 'Template Based Button', '\* Button Template' set to 'Interactive Report Button', 'Button Type' set to 'Normal', 'Button CSS Classes' (empty), and 'Button Attributes' (empty). The bottom panel, titled 'Action When Button Clicked', contains: 'Action' set to 'Defined by Dynamic Action' (highlighted with an orange box), 'Execute Validations' set to 'Yes', and 'Database Action' set to '- No Database Action -'.

2. Edit "Download when click on (JQuery Selector)" property of Process with type "GPV Interactive Report to MSExcel [Plug-in]". Add **Static ID** of the button with '#'-prefix.

The screenshot shows the 'Settings' panel for the 'Download when click on (JQuery Selector)' property. It includes: 'Replace standard IR XLS download' set to 'Yes', 'Maximum Rows' set to '500', 'Download when click on (JQuery Selector)' set to '#GETEXCEL' (highlighted with an orange box), and 'Download Type' set to 'Excel XLSX'.

## GPV Interactive Report to MSEExcel

### How to exclude reports columns from export/make columns that are visible in Excel but not visible in Interactive Report

Use **Conditional Display** column property for this.

**Conditional Display**

Condition Type

PL/SQL Expression

[PL/SQL] [item / column=value] [item / column not null] [item / column null] [request=e1] [page in] [page not in] [exists] [never] [none]

Expression 1

```
nvl(:REQUEST,'N') not like 'GPV_IR_TO_MSEXCEL%'
```

☐ Do not validate code (parse code at runtime only).

Use ***nvl(:REQUEST,'N') not like 'GPV\_IR\_TO\_MSEXCEL%'*** Expression to exclude column from export, and ***nvl(:REQUEST,'N') like 'GPV\_IR\_TO\_MSEXCEL%'*** Expression to make column visible on export only.