

# APEX 5 plug-ins for everyone

Alan Arentsen  
Ordina  
The Netherlands

## Keywords:

APEX, Plug-in, Application Express

## Introduction

Do you think it's hard to create your own plug-in? Never took the time to look into it? Or just don't see the added value of using plug-ins?

At DOAG you might have seen my presentation where I explained what a plug-in does and how easy it can be to create one yourself.

It isn't like waking up in the middle of the night while thinking: "I'm going to create a plug-in." Often you spent a lot of time puzzling and working out to a good solution. Lots of time could have been saved by creating a plug-in.

## What is a plug-in?

In APEX you work with components all the time. Whilst creating your page you add regions, items buttons and so on. Beside all the components you see rendered on your screen, there are also components which are adding functionality to your page like dynamic actions and processes.

With plug-ins it is possible to create APEX components with your own layout and custom functionalities. You will be creating a container with all the parts you need to create a component. So all your HTML, CSS, JavaScript and of course PL/SQL will be packed together in this component.

There were no plug-ins before APEX 4.0, so there was no possibility to create your own APEX component. Small changes on existing APEX component almost always lead to pieces of custom JavaScript or CSS in your page.

It therefore was a big change when plug-in's were introduced. From then it was a lot clearer by keeping your component organized in a plug-in.

You can use your plug-in on your page like you would use a standard APEX component. But it's also possible to export a plugin and use it in another application.

Many plug-in's made by APEX evangelists or commercial companies can be found on the internet. At the moment of writing there were three important sites to visit if you want to use plug-in's:

- OTN website for APEX Development Team Plug-Ins
- <http://www.apex-plugin.com>
- <http://apex.world>

## Where to start?

It is possible to create a plug-in from scratch as your first plugin. And maybe you would manage to do that.

Globally there are three ways of creating a plugin:

- download an existing plug-in from a plug-in repository and use it
- copy an existing plug-in and edit it until it fits your needs
- create a plug-in from scratch

The easiest way is to download a plug-in from a repository and import it into your application. You can use the plugin right away but it maybe not will fill all your needs.

You can also edit the plugin so it will do what you want but you will have to be aware that the license maybe won't allow you to do this. Many plug-ins are open source and this often shouldn't be a problem. Though it's always advised to contact the owner before editing a plug-in. Maybe you can help the community by making it better of pay the owner a fee .

If you are more familiar in using plug-ins you can start by making your own plug-in. When you have created a custom HTML component on your site sometimes it is the only option to create a plug-in from scratch. By creating a plug-in of it you can use it on all your pages in your application and share it with the community if you like.

When you start by creating a plug-in from scratch make sure you write down what you would expect from a plugin and how it should look like. After that create a working example in a separate HTML file or APEX page. Then create an empty plugin with just the basic stuff. And at the end, copy pieces from that page one by one to your plugin and test if the adjustments work.

If you work like this you will notice immediate when something is wrong. You will also learn a lot how a plugin is constructed.

Whatever method you use make sure you always have a plan! Without a plan you will end with a badly working plugin. Or worst, a lot of code and no plugin.

## And now?

First thing to do when creating a plugin is implementing a render function. This function will be used by APEX when the plugin is rendered to the page. There is an example render function specification for every plug-in type you can create. The function is written in PL/SQL, so in PL/SQL code you will determine how your plugin will look like when it's rendered.

From this function you also include the JavaScript and CSS code which is needed for your plug-in.

You are not obligated to upload your files to a web server. You can write all your code in the plug-in and upload your files in your plug-in. In APEX 5 it is even possible to upload a ZIP file where all your files are stored in. APEX will automatically extract the file. If you want to make a portable plug-in it is advised to upload your files within the plugin. Otherwise you can also upload your files to a web server an link to that server instead. It can be faster because of the webserver caching mechanism.

If your plug-in is ready to use you can also move your PL/SQL from the plug-in to your database. In this way it is more transparent where your code is and it is easier to maintain the plug-in PL/SQL code.

Beside the rendering function there are a few things you can define declaratively. Like when you are planning to use your plugin on different pages and you want the behavior or the appearance to be different based on a certain variable. There is a possibility to parameterize this and use those variables to e.g. change the way the plug-in is rendered.

### **What if I fail?**

During plug-in development you will experience a lot of issues which may look unsolvable at the moment. Though by reading the online instructions and help many of those issues will be solved. Working step by step in constructing a plug-in will lead you easily to the issue.

On almost every variable or setting in the plugin there is help available by pushing the help button next to the setting. There are even pieces of example code written in the help.

If you are a bit more experienced in creating plug-ins, you can find the description of some important packages in the online help. A few packages which are often used in creating plug-ins are:

- APEX\_PLUGIN
- APEX\_PLUGIN\_UTIL
- APEX\_JAVASCRIPT
- APEX\_CSS

There are more packages in the 'Oracle Application Express API Reference' which you can use in creating your plug-in. Every package is heavily documented and explained with examples.

### **Final words**

The possibilities in creating plug-ins are endless. On one hand you easily can get stuck, on the other hand plug-ins are maybe the best documented feature of APEX ever.

There are also a lot of working examples with source code in the plug-in repositories.

As a final tip I advise you to just download a working plugin and look how other people create plugins. Learn from it and start by creating a simple plug-in. Work through the lines of code with the internet documentation and the help function next to it. Start with minor adjustments and you will see creating plug-ins can be very addictive.

### **Contact address:**

**Alan Arentsen**

Ordina

Ringwade 1

3438 MN, Nieuwegein

Phone: +31 (0) 6 13 01 88 32

Email [alan.arentsen@ordina.nl](mailto:alan.arentsen@ordina.nl)

Internet: [alanarentsen.blogspot.nl](http://alanarentsen.blogspot.nl)