

Plug-INS and their Use in Desktop and Online Applications

Samer Barakat

Department of MSE, Applied Science Private University, Amman, Jordan

Abstract: Web applications such as Content Management Systems (CMS) as well as standalone software applications deployed on computers are installed with standard settings which include minimal functionalities. The applications are powerful enough on their own to do the job meant for them to do however, to enhance the capabilities of these software applications we need to install add-ons called Plug-ins. In this study, we shall shed the light on what is a plug-in and why and it is used in relation to online and offline software applications.

Key words: Plug-in, content management systems, compute applications, capabilities, enhance

INTRODUCTION

Although, plugins are a new expression for to many of us, the term Plug-in started to be used in the 1975 in reference to EDT text editor that was installed on Unisys operating system used by mainframe computers (EDT text editor reference manual) (UCC, 1975). The plugin then allowed for accessing the applications buffer and thereby allowing other programs to have a direct access to the editor's memory session. The plug-in was able to send user calls through the operating system to the text editor in order to conduct text-editing functionalities (EDT text editor reference manual).

Some of the software applications that has employed the use of plug-in the 1980s included HyperCard, QuarkXPress (Huttenlock *et al.*, 2006). In the late 80s, some early applications and computer games incorporated the use of plugins to extend the applications functionalities and abilities. Starting in the 1990s the terms started gaining even more popularity under different names such as extensions and add-ons (Seadle, 2006). These extensions and add-ons were developed either by third parties and sold separately to enhance the applications abilities or were generated by the original software development company.

However, according to Mozilla the terms plug-in, extensions and add-ons are not similar nor mean doing the same functionalities (Sunny, 2008). Microsoft and Adobe where the first organizations to employ the use of plugins in their desktop applications. They were meant to be extensions that does not need elaborative user interface tools (Sunny, 2008). According to Microsoft Windows, Plug-ins add custom functionality and are equivalent to Dynamically-Linked Libraries (DLLs) used on their operating system.

WHAT ARE PLUG-INS?

It is a well-known fact among system developers and designers that Plug-ins provides a wide variety of features which enhance the functionalities of applications (Open Source CMS). It is an application add-on that is once installed or activated it enablesthe program to perform additional capabilities (Seadle, 2006). Flash and PDF viewers are an excellent example of plugins downloaded and used through a container applications such as web browsers (<http://www.cmsmatrix.org>).

Plug-ins research on providing different functionalities such as multimedia functionalities, compression and decompression utilities and tailored applications that perform certain tasks (<http://www.opensourcecms.com>).

Plugins are not only used by desktop applications and users. There are thousands and thousands of plugins used by content management system applications such as Word Press, Joomla, Drupal, Moodle, etc (Steve White and Andy Wallace).

These plugins are installed by the web developer and are typically used to enable the users to use extra functionalities or enhancing the website including adding contact forms, login screens, photo galleries, social networking links and much more (Drupal Installing Modules).

HOW PLUG-INS WORK

Plugins work in a browser or in an online application system. If it works in a browser it can be client side plug-in and this depends on the web browser hosting the plug-in and the operating system support (<http://wordpress.com>). If it was an online application

plug-in, it could be designed by third party developers for online systems such as Word Press, Joomla and Drupal to name a few. Plugins function on a host application which provides the platform and the infrastructure for the plugin to function properly and do what it was designed to do (<http://wordpress.com>). The system allows the plug-in to register itself through the host application and start exchanging data with the system. This indicates clearly that plug-ins rely on the functionalities provided by the system and does not work by themselves as a standalone functionality as the host system do (Barakat, 2013). This enables applications users to add, update and delete a plug-in without worrying about the basic functionalities of the original system being affected by the un-install, update or delete of the plug-in (Barakat, 2013).

Plugins developed for content management systems rely on shared libraries already installed on the host system. Most plugins developed today are programmed using PHP since the majority of content management systems are based on PHP and My Sql database (Barakat, 2013).

Plugins costs and availability: We now know the logic behind the use of plugins but why do they exist and who is behind them (Joomla! Administrative Manual). The answer is pretty much straight forward. Third party companies and programmers build plug-ins in the hope that the end user would like what it does and hopefully one day upgrade or purchase the full package.

They live as long as the application or their host environment exists. Therefore, some programmers make sure that their plug-ins are cross platform and function on several other applications (Seadle, 2006). This is not an easy thing to do since building a plugin requires studying the existing system to use its already available services.

The prices of plug-ins is usually very moderate compared to building a customized plug-in. The idea is since they are going to sell the plug-in to a large number of users then charging a moderate fee shall be beneficial in the long run. Some plugins are provide for free in the hope that the end user would employ the technical expertise of its programmers. Some do it just to be well known in the plug-in programming domain. All this made

the number of available plug-ins is in the thousands and it shall continue growing in the future since we shall be more reliant on modular programming.

CONCLUSION

The purpose of this study was to explain what are plug-in's and how they do operate and what is their use and availability. Our research focused found out that the use of plug-ins enabled users to add even more functionalities to existing applications and allowed web developers to enhance the services they provide through their online systems. Plug-ins are becoming a must for all desktop and online system and is gaining popularity every day. In conclusion we like to stress the fact that plug-ins enhanced and expanded the lives of existing systems desktop and online alike. Developers should understand how plug-ins work and start to rely heavily on the use of third party plug-ins to expedite and reduce the delivery costs.

ACKNOWLEDGEMENT

Researcher is grateful to Applied Science Private University, Amman, Jordan, for the full financial support granted to this research project.

REFERENCES

- Barakat, S., 2013. Agile programming: As a CMS development Methodology. *Innovative Syst. Des. Eng. Int. Inst. Sci.*, 4: 13-16.
- Huttenlock, T.L., J.W. Beaird and R.W. Fordham, 2006. Untangling a tangled web: A case study in choosing and implementing a CMS. *Lib. Hi Tech*, 24: 61-68.
- Seadle, M., 2006. Content management systems. *Lib. Hi. Tech*, 24: 1-5.
- Sunny, S.K., 2008. Evaluation of Open Source Content Management System: A Comparative Study. *Inflibnet Center, India*, ISBN:978-81-902079-6-6,.
- UCC., 1975. EDT text editor reference manual cinnaminson. Unisys Computer company, New Jersey, USA.