

A Survey of Open Source Installer Production Tools

Priyadharshini T.^{1*} and Sumathi V.P.²

^{1*,2} *Department of Computer Science and Engineering, Kumaraguru College of Technology, India,
priyadharshini.t.kct@gmail.com, sumathi.vp.cse@kct.ac.in*

www.ijcseonline.org

Received: 7 March 2014

Revised: 18 March 2014

Accepted: 25 March 2014

Published: 31 March 2014

Abstract— The Open source installer production tools are widely used for building software installer application. This survey paper deals with the most commonly occurring java based open source installer production tools in the software market and classified the tools according to the file packaging and distribution mechanism. This survey is mainly focused on open source installer production tools, which provides the framework for generating automated unattended installer. The automated installer is more generic installer for all types and versions of operating system and also provides the unique and clean way of distributing the software application during silent unattended installation. In the Introduction the reader is informed about the advantages of the Open source tools, furthermore discussed about types of installation, Open source Licenses, Java based installer production tools. At the end conclusion is drawn and outline some questions for future research.

Index Term— Distribution, Licenses, Installer, Opensource, Automated Installation, Application, Installer Building Tool

I. INTRODUCTION

The widespread adoption of Free and Open Source Software (FOSS) in many strategic contexts of the Information Technology Society has drawn the attention towards application development using Open Source Tools [8]. Open Source software are portably available on a wide range of platforms. In addition, the source code is openly available; people interested in another platform can contribute for the development of the open source software products. As a result, Open Source allows a wider choice of computing platforms and potentially easier upgrade to new technology. Open Source software tends to be free of dependency on related products, so avoiding lock-in to particular manufacturers [1, 19].

An open source installer is a system of tools used to create installer files, that users download and run in order to install an application physically on to a computer running an operating system. The installation is a complex process, as the software application could correctly interact with the operating system, the hardware and other software applications, without any conflicts. The open source installer is often an inexpensive and easy way for a software developer to take care of the installation process [8].

Today, many open source installer production tools are available in market. This installer production tools are avoiding the complexity of developing the installer from the scratch and provides the framework for developing the installers in the free of cost. Deployment of software application encompasses actions like the discovery of resources and enquiry about operating system types and versions then selection among the different versions of the software according to these states, installation, start-up and

eventually software installation monitoring and upgrading. Open Source Installer Production tools plays the major role in the deployment of the developed software application on the users' target in efficient way.

II. INSTALLATION

Installation is the process where a new operating system, computer program or driver is copied to the computer. After this process is possible use this installed component or components. Software makers seek maximum simplification of the installation process, which usually involves configuration and any other necessary activities. When installing an operating system in a dedicated storage area (typically a hard disk) created favorable environment (disk formatting and boot loader installation) and it is then copied the operating system kernel and other necessary components. Everything is configured so that the operating system can be easily used. Programs are usually delivered in the form of compressed packages containing a set of necessary files into a single file, because with them to better handle the distribution and sale. Before using the package has to be decompressed ("expanded"), the location of their seats and made other necessary settings. To simplify installation were developed universal installation programs whose activities can be set using the configuration file, so the programmer does not have to change the setup program [11].

According to the installation parameters can be defined several types:

- Clean installation
- Attended installation
- Silent installation
- Unattended Installation
- Headless installation

Priyadharshini T, priyadharshini.t.kct@gmail.com

- Scheduled installation
- Network installation
-

Clean Installation- The installation process does not affect unforeseen conditions which could cause a collision (such as the existence of files or settings from the previous installation of the same program due to repeated installations of the same program, or other circumstances that the creator did not foresee the installation process). An example might be a complete new installation of the operating system when the target disk (or partition) is formatted first and then will install on a blank disk and each operation is performed exactly as it was intended (of course do not enter into foreign influences, starting with the so-called . clear table).

Attended Installation- It is a most common type of installation in operating systems based on MS Windows. This type of installation requires the presence of the user who will handle the entire process. Installation wizard offers the user a choice of acceptance / rejection of End User License Agreement (EULA) and the choice of installation location, set passwords, etc. Graphical user interface (GUI) wizard is now a matter of course.

Silent Installation- Installation which during its trial does not showing any messages or windows. Silent installation is not the same as unattended installation (see below). All silent installations are unattended installations, but not all unattended installations are silent. This type of installation is running in the background without the user's knowledge. The reason may be the installation convenience, the installation process is busy and can fully concentrate on their business, or as tricked the user knows what is in his system installed. This method uses most of Malware.

Unattended Installation- Installation that the installation process takes place without interaction with the user, or the user's presence is not required. Unattended installation does not require user to enter or otherwise provided all the necessary information for start of the installation process. All this information is already contained in any file which is part of installation source. These can be in the form of the command line or answer file.

Headless Installation- It is performed without the use of a display device or even without the graphics card, which is not plugged into the computer on which the installation is performed. Installation can be performed from another PC connected via LAN or via the serial port.

Scheduled Installation- It is also known as a planned installation. The installation process runs at a predetermined time, or generally under certain conditions. The normal installation process begins specifically at the initiative of the user. For example, a system administrator may want to install a newer version of the used software. It will be installed only if the software is not already in use. This event will start the installation process. Generally, this type of

installation is commonly used in most operating systems. The operating system automatically installs drivers for the connected device (plug and play).

Network installation- In this method, the installation process copies the files to the target directory from a shared network location. At this point, it may not always find the original installation files, but may also include a copy of your original installation media.

III. OPEN SOURCE LICENSES

To make the findings on licenses more comprehensible, we include a description of the open source licenses that will be referred to later in the paper. The *GNU General Public License (GPL)* [9] is a classic, often used open source license. Any user is free to make changes to the source code. If the changed version is only used privately, it is not a requirement that its source code is released. If it, however, is distributed in some way, then the source code must be made available under the GPL (i.e. also released as open source that any user is free to change and copy). It should be noted that a library released under the GPL will require any program that uses it to be licensed under the GPL. This is not the case when using *GNU Library General Public License (LGPL)* [10] which apart from that is much like the GPL. The *Common Public License (CPL)* [6] was developed by IBM as an open source license. Like the GPL, the CPL requires that the source code for a modified version of a program is made publicly available if the new version is distributed to anyone. Programs and libraries released under the CPL may be used from and integrated with software released under other (also closed source) licenses. The *Mozilla Public License* [18] is also an open source license that requires the code for any distributed modified works to be made publicly available. It is allowed to use a library under the Mozilla Public License from a closed source application. Thus the license has some similarities with the LGPL. The *Apache License* [3] allows the code to be used both in open source, free programs and in commercial programs. It is also possible to modify the code and redistribute it under another license under certain conditions (e.g. the use of the original code should be acknowledged). Version 1.0 and 1.1 of the Apache License included requirements about the use of the name "Apache" in documentation and advertising materials. That meant that the license should be modified for use in non-Apache projects. This was changed with version 2.0 [4]. The *BSD License* [5] is a very liberal open source license. It is permitted to use source code from a BSD licensed program in a commercial, closed source application. As long as any copyright notices remain in the modified code, there are no requirements saying that modifications of the code should be BSD licensed or open source.

IV. INSTALLER BUILDING TOOLS

Java based application has cross-platform support, that can be directly run on any platform without special preparation,

e.g., application written in an interpreted language or pre-compiled portable byte code for which the interpreters or run-time packages are common or standard components of all platforms.

Lift off Java-Lift off Java Installer is an installer for Java applications and distributed under GNU GPL License. The important features are self-extracting class files, cross platform installation and minimal requirements of the Java installation on the target system [16].

VAInstall- VAInstall is a multi-platform installer, which is written in Java and distributed under the GNU GPL License. Through a simple configuration file, it allows the packager to pick files and directories from various places on disk, and not only one root directory like other installers but also the packager can completely reorganize the files and directory on the destination host. The installer can choose the install mode for the destination host (graphic, xtra, text or ANSI) [21].

Packlet- Packlet is a simple installer tool for Java and distributed under GNU Library or Lesser General Public License. A packlet is similar to a compressed zip or tgz file: it packages an application into a single compressed file. As the Packlet files are Java class files, the installer can be executed on any machine with JRE installed. Executing a packlet opens an installer GUI to configure and unpack the application [20].

Mini-installer- Mini-installer is a simple package to build Java self installation programs. The process of installation is script driven and the result is a unique JAR file that the user can execute the installer by double click to execute a wizard-like installation process. MiniInstaller is distributed under GPL License [17].

Freeinstaller- FreeInstaller is an Open Source installer program in Java and it has a self extracting class file. The Freeinstaller is able to install Java and other software on UNIX and Windows New Technology [7].

Jsmooth- JSmooth is a Java Executable Wrapper that makes a standard Windows executable binary from a jar file. Jsmooth installers are distributed under GNU GPL License. It makes java deployment much user-friendly and smoothly. Jsmooth is able to find a Java Virtual Machine by itself. When no Virtual Machine is available, it guides the user by providing feed-back to the users, and can launch the web browser to a webpage that explains how to download a Java Virtual Machine [14].

Antigen- Antigen (Ant Installer Generator) is a tool to take an Ant build script, that combine with a GUI and wrap the installer as an executable java archive file. The primary purpose of Antigen is to create powerful graphical installers using Ant scripts. Antigen is under Academic Free License (APL) [1].

Java-service-wrapper- The Java-service-wrapper makes it possible to install a Java Application as a Windows NT Service. The scripts provided with the Wrapper makes the Java Applications installation process as daemon processes on UNIX systems. The Wrapper handles user log outs under Windows and the ability to run services which interact with the desktop [13].

AntInstaller- AntInstaller is an installer framework which uses a simple XML file to define the User Interface, and back end as Apache Ant. The installers have Ant's capability and are also compatible with Ant's extension mechanism. Installers can be packaged as a self extracting Jar and can run from the command line or through GUI. AntInstaller can also be used as a GUI front end for existing Ant builds. AntInstaller is distributed under GNU GPL License [2].

Launch4j- Launch4j installer is Windows native executable (.exe) java application wrapper and distributed under GNU Library or Lesser General Public License (LGPL). Launch4j Offers native application icon, splash screen, feedback on startup failure, search for Java Runtime Environment if not present use bundled one and passes command line arguments [15].

Izpack- IzPack is an installer generator for the Java platform. This installer production tool produces lightweight installers that can be run on any operating system where a Java virtual machine is available. The installer is java archive file and depending on the operating system, it can be executed by a double clicking or a simple 'java -jar installer.jar' on a command prompt. IzPack is under GNU GPL License. The main benefit of IzPack is to provide a unique and clean way of distributing a project to users using different types of operating systems [12].

V. CLASSIFICATION OF INSTALLER

Today's mainstream open-source production tools of installers are IzPack, VAInstall, Packlet, Antigen and some other software. The production tools build the installer by packaging applications and the corresponding resource file. Then it enables application software to be installed on the user's machine conveniently and successfully. According to the file packaging and distribution mechanism, the open-source installers divided into four categories:

Script File Driven- Script-based installer generally put all the file paths, the installation steps and the installation information into a script file. When generating executable file, the program builds the installer by packaging all the resources in sequence based on the script file. The feature of this kind installer is that the configuration information is clear, and all the operations are carried out around the script in sequence [10].

Configuration File Driven- The configuration file of Configuration file driven installer allows the packager to pick files and directories from various places on disk. The packager can completely reorganize the files and directories on the destination host. This type of installer can also choose the install mode for the destination host (graphic, text or ansi).

GUI Driven- GUI driven installer packages an application into a single compressed file that can be distributed. Executing an installer production tool brings up an installer GUI to configure and unpack the application. The features of this type of installer are to take an Ant build script, combine with a GUI and wrap it up as an executable file. Its primary purpose not only to create powerful graphical installers from Ant script but also be used as a GUI front end for existing Ant builds.

Class File Driven- Class file driven installer is mainly for distributing Java applications. This installer has a self extracting class file, that are able to install both Java and non-Java Software. This installer makes java deployment much easier and faster by its ability to find a Java Virtual Machine by itself. When no Virtual Machine is available, the installer paves a way to the users and launches the help to explain how to download a Java Virtual Machine.

VI. CONCLUSION

This article researches and realizes the identification and building system for specific installer. This survey provides the basis for automating the process of software installation. Currently there is no sufficient research regarding the automated installation. This paper aims to provide the information regarding the open source installer production tools. This study is not deep enough, the productions tools are classified according to the file extracting and distribution mechanism and not concentrated on other aspects. We sincerely hope researchers to participate in order to further improve and resolve deficiencies. We hope that the survey regarding the tools for automated software installation will be more reasonable and more standardized in future.

REFERENCES

- [1] Antigen Installer, <http://antigen.sourceforge.net> as of 05/15/2014.
- [2] AntInstaller, <http://antinstaller.sourceforge.net> as of 05/15/2014.
- [3] Apache SW License v1.1, www.apache.org/licenses/LICENSE-1.1 as of 01/15/2014.
- [4] Apache SW License, v2.0, www.apache.org/licenses/LICENSE-2.0 as of 01/15/2014.
- [5] BSD license, opensource.org/licenses/bsd-license.php as of 01/15/2014.
- [6] CPL v1.0, opensource.org/licenses/cpl1.0.php as of 01/15/2014.
- [7] FreeInstaller, <http://www.xenonsoft.demon.co.uk/products/freeinstaller> as of 05/15/2014.
- [8] Free Open Source Software, http://freepackages.org/index.php?title=Main_Page as of 05/15/14.
- [9] GNU GPL License, www.gnu.org/copyleft/gpl.html as of 01/15/2014.
- [10] GNU LGPL License, www.gnu.org/copyleft/lgpl.html as of 01/15/2014.
- [11] Installation, <http://en.wikipedia.org/wiki/Installation> as of 05/15/14
- [12] IzPack installer, <http://www.izforge.com> as of 05/15/2014.
- [13] Java-service-wrapper, <http://wrapper.tanukisoftware.org> as of 05/15/2014.
- [14] Jsmooth Installer, <http://jsmooth.sourceforge.net> as of 05/15/2014.
- [15] Launch4j installer, <http://launch4j.sourceforge.net> as of 05/15/2014.
- [16] LiftOff installer, <http://liftoff.sourceforge.net/index.en.html> as of 05/15/2014.
- [17] MiniInstaller, <http://www.yagga.net/java/miniinstaller/index.shtml> as of 05/15/2014.
- [18] Mozilla Public License v1.1, www.mozilla.org/MPL/MPL-1.1.html as of 05/15/2014.
- [19] Open source, http://en.wikipedia.org/wiki/Open_source as of 05/15/14.
- [20] Packet installer, <http://packet.sourceforge.net> as of 05/15/2014.
- [21] VAIInstall installer, <http://vainstall.sourceforge.net> as of 05/15/2014.

AUTHORS PROFILE

Priyadharshini was born on November, 21, 1988 in Coimbatore, TamilNadu. She completed her B.E (CSE) degree in the year 2010 from SNS College of Technology, Coimbatore, TamilNadu. She cleared GATE 2012 and pursuing M.E (CSE) at Kumaraguru College of Technology, Coimbatore, TamilNadu. She is doing Internship at Robert Bosch Engineering and Business Solutions Ltd.



Sumathi is working as Senior Grade Assistant Professor at Kumaraguru College of Technology, TamilNadu. She was born on February, 05, 1976. She completed her B.Sc. degree and M.C.A from Bharatiyar University, Coimbatore, TamilNadu. She also completed M.E in the stream of Computer Science and Engineering from Kumaraguru College of Technology, Coimbatore, TamilNadu. . She has about 11 years of teaching experience. She is doing research in Ontology. She has published research papers at national and international Journals.

