Mt Wilson

Installation

# Background

The Mt Wilson installation procedure must be fully automated in order to decrease development and testing costs and to make it easy for partners to integrate it into their own products.

# Architecture

In this document, installation refers to copying necessary files to and running necessary commands on the target server such that when the installation is complete, the Mt Wilson setup tasks are ready to be executed and the server can be started.

An environment file is a file containing shell variable definitions. If present, the environment file is read by the self-extracting installer in order to automate its choices and avoid the need for user input.

## Distribution

### Self-extracting installer

Mt Wilson is comprised of numerous Java archives, shell scripts, HTML content, and executable binaries. These are packaged into a zip file. This zip file is wrapped with an installation script and a self-extracting archive.

### Linux Package

Mt Wilson may also be available as a Linux package in Ubuntu, RedHat, SUSE, etc. Each Java archive, executable binary, and shell script collection might be packaged separately to let the package manager automatically install them as dependencies of the main package.

### Windows Package

Mt Wilson may also be available as a Windows Installer (.msi) package.

## Installation

The installer or package should take the following general steps:

1. Intel copyright notice and other required copyright and legal notices (subcomponents, etc)
2. Obtain required information from existing configuration files, environment variables, a .env file in user’s home directory, or prompt user for required information for which there is no default
3. Copy files to destination
4. Run setup tasks using the installed software
5. Integrate with other system services as necessary, for example monit and iptables
6. Services only: Start the service

## Un-installation

A regular uninstall should:

* Delete application code and binaries
* Remove startup scripts
* Remove any integration with other system components

A “purge” uninstall should perform above steps and also:

* Delete configuration files
* Delete logs
* Delete user data maintained by the application (database or repository folder)

It is the user’s responsibility to export and backup their configuration, data, and logs before they uninstall with the purge option.

# Development Roadmap

Currently the “functions.sh” file contains functions to assist in the installation and setup of many features. This needs to be separated into one shell script per feature. For better performance, these separate scripts can be combined into a single functions.sh every time they are updated.

## Windows

On the build server, we can install WiX, Mono, and Wine together to allow us to integrate Windows Installer (.msi) generation into the build process.

WiX <http://wixtoolset.org/> license “Microsoft Reciprocal License (MS-RL)” for the tool which allows us to use and copy the WiX tool for free, and makes no claims on the output (our installer). <http://wixtoolset.org/about/license/>

Mono <http://www.mono-project.com/> is used while running WiX only and would not be distributed as part of our application. License <http://www.mono-project.com/docs/faq/licensing/>

Wine <https://www.winehq.org/> is used while running WiX only and would not be distributed as part of our application. License <https://www.winehq.org/license>