**Material Data Curator System:**

**Widows Setup Configuration**

This document explains how the automatic installation tool for Windows is created.

The goal of the installer is to make technologies and configurations transparent for the user. Instead of creating and configuring an environment on user’s computer, this method explains how to package the project with a preconfigured environment and make it easily deployable using a setup tool. User should also be able to make the MDCS run using an icon and not python command.

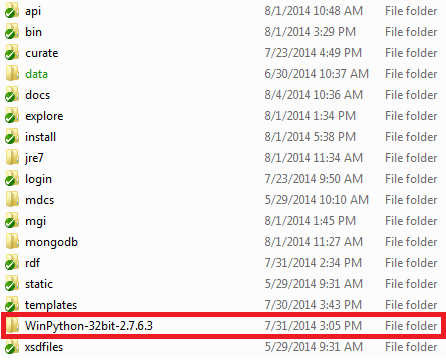
**WinPython**

**Source:** <http://winpython.sourceforge.net/>

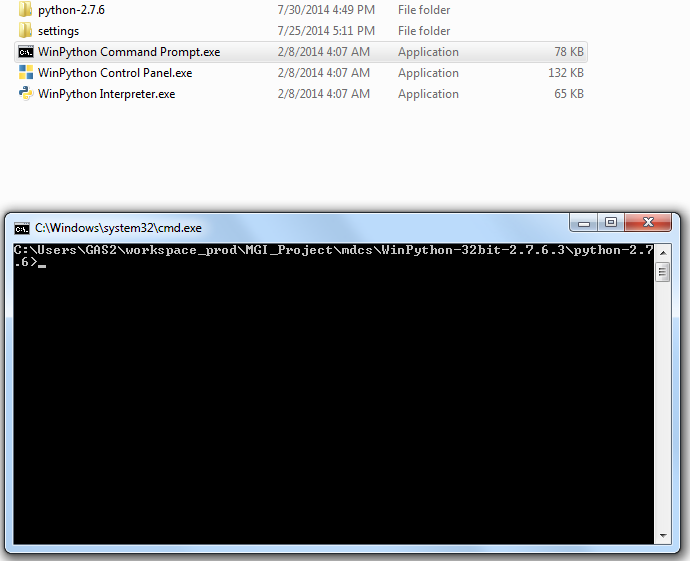
**Description:** WinPython is a portable version of python. Some python packages used in the project can have some troubles to install and run depending on the version of Python installed on the computer. Some of them also need configurations. So, using a portable version of Python allows to be sure the user is running the good version and that all packages are well installed.

**Instructions:**

* Download **WinPython 2.7.6.4 32bit** from <http://sourceforge.net/projects/winpython/files/>
* Copy the WinPython directory into the project directory.



* Open the WinPython Command Prompt. Pip is already installed. So you can use it to:
  + Uninstall every package unused in the project. These two commands can be useful.
    - pip list
    - pip uninstall <package\_name>
  + Install all packages used in the project. Follow the instructions related to package installation in the Zero-ConfigInstallationNotes-Win.docx doc. You don’t need to install and work with a virtual environment as this python version will only be used to run the MDCS.



* Remove all folders containing tools that won’t be used in the project.

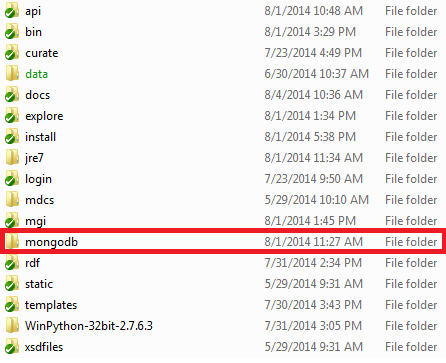
**MongoDB**

**Source:** <https://www.mongodb.org/>

**Description:** MongoDB is the backend database used by the project. To run the project, we need an instance of MongoDB running. To prevent the user to download and install it, it will also be embedded in the project.

**Instructions:**

* Download **MongoDB** from <https://www.mongodb.org/downloads>
* Install the downloaded file
* Move the MongoDB folder to the project folder



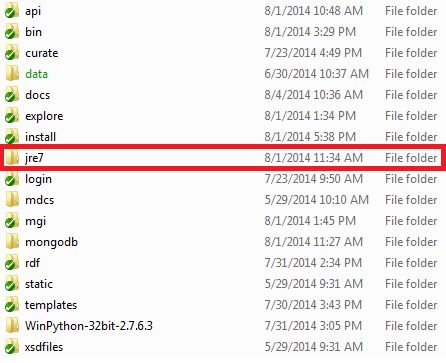
**JRE**

**Source:** <http://www.oracle.com/technetwork/java/javase/downloads/java-se-jre-7-download-432155.html>

**Description:** The Jena servers are written in Java so we need a JRE to run them. For now the JRE is also embedded in the project until we find a better solution.

**Instructions:**

* Download the JRE 7 for windows from <http://www.oracle.com/technetwork/java/javase/downloads/java-se-jre-7-download-432155.html>
* Install the downloaded file
* Move the JRE folder to the project folder



**Inno Setup:**

**Source:** <http://www.jrsoftware.org/isinfo.php>

**Description:** Inno allows to package everything that is in the project forlder and to make an automatic install tool for it.

**Instructions:**

* Download **Inno Setup** from <http://www.jrsoftware.org/isdl.php>
* Install it
* Use Inno Setup Compiler to write a script that will create the installer. You can use the assistant or write the script by yourself

; Script generated by the Inno Setup Script Wizard.

; SEE THE DOCUMENTATION FOR DETAILS ON CREATING INNO SETUP SCRIPT FILES!

**[Setup]**

; NOTE: The value of AppId uniquely identifies this application.

; Do not use the same AppId value in installers for other applications.

; (To generate a new GUID, click Tools | Generate GUID inside the IDE.)

**AppId**={{EE247CA0-EF41-48C8-988D-8A009BF94485}

**AppName**=mdcs

**AppVersion**=1.0

;AppVerName=mdcs 1.0

**AppPublisher**=NIST

**DefaultDirName**=**{pf}**\mdcs

**DefaultGroupName**=mdcs

**OutputBaseFilename**=setup

**Compression**=lzma

**SolidCompression**=yes

**[Languages]**

**Name**: "english"; **MessagesFile**: "compiler:Default.isl"

**[Files]**

**Source**: "..\\*"; **DestDir**: "{app}"; **Flags**: ignoreversion recursesubdirs createallsubdirs

; NOTE: Don't use "Flags: ignoreversion" on any shared system files

**[Icons]**

**Name**: "{commondesktop}\mdcs"; **Filename**: "{app}\bin\mdcs.bat";

Here is an explanation of the purpose of each section:

* [Files]: describes from where source files are gotten and where they should be copied.
  + Source: the script is in mdcs/install, so we use “../\*” to tell to copy everything that is in mdcs folder (sources + winpython + mongodb + jre).
  + DestDir: set to {app}. The constant represents the place where the application is installed on the system.
* [Icons]: describes the application icon and what it should run.
  + Name: name of the icon. The constant commondesktop is used to place the icon on the desktop.
  + Filename: name of the file to run.
* [Setup]: this section is generated by the assistant. Not all fields are required. The DefaultDirName property describes that the project will be installed in Program Files\mdcs by default.

**One batch file to run them all**

**Description:** To run MDCS, the user has to run the mongo daemon, the Jena servers and the Django application in this order. In order to make it easier for him, a batch file was created to do so.

**Instructions:**

* Write a bat file able to launch MongoDB, the Jena Servers and the Django Server at the same time.

@setlocal enableextensions

@**cd** /d "**%~dp0**"

start cmd /k ..\mongodb\bin\mongod.exe --dbpath ..\data\db

start cmd /k ..\jre7\bin\java.exe -jar ..\rdf\JenaServers.jar -rdfserver\_endpoint "tcp://127.0.0.1:5555" -sparqlserver\_endpoint "tcp://127.0.0.1:5556" -tdb\_directory ..\data\ts -project\_uri "http://www.example.com"

start cmd /k ..\WinPython-32bit-2.7.6.3\python-2.7.6\python.exe ..\manage.py runserver

* Make the Inno shortcut run this batch file.