LEGAL

VERSION HISTORY

CHANGES IN V6

- Added button to display EA.exe.Config if present in EA installation directory

CHANGES IN V5

- Added DLL version to List of AddIns
- Added Registry Tree view complete with registry details
- Added context menu to Registry Tree view items to support queries based on contents of selected item
- Added Query results tab to display results from Registry Tree View queries as well as support user defined queries
- Output of queries to individual log files

CHANGES IN V4

- Added form to display **list of classes and methods** retrieved from the DLL

CHANGES IN V3

- Additional registry locations checked for add-in keys

CHANGES IN V2

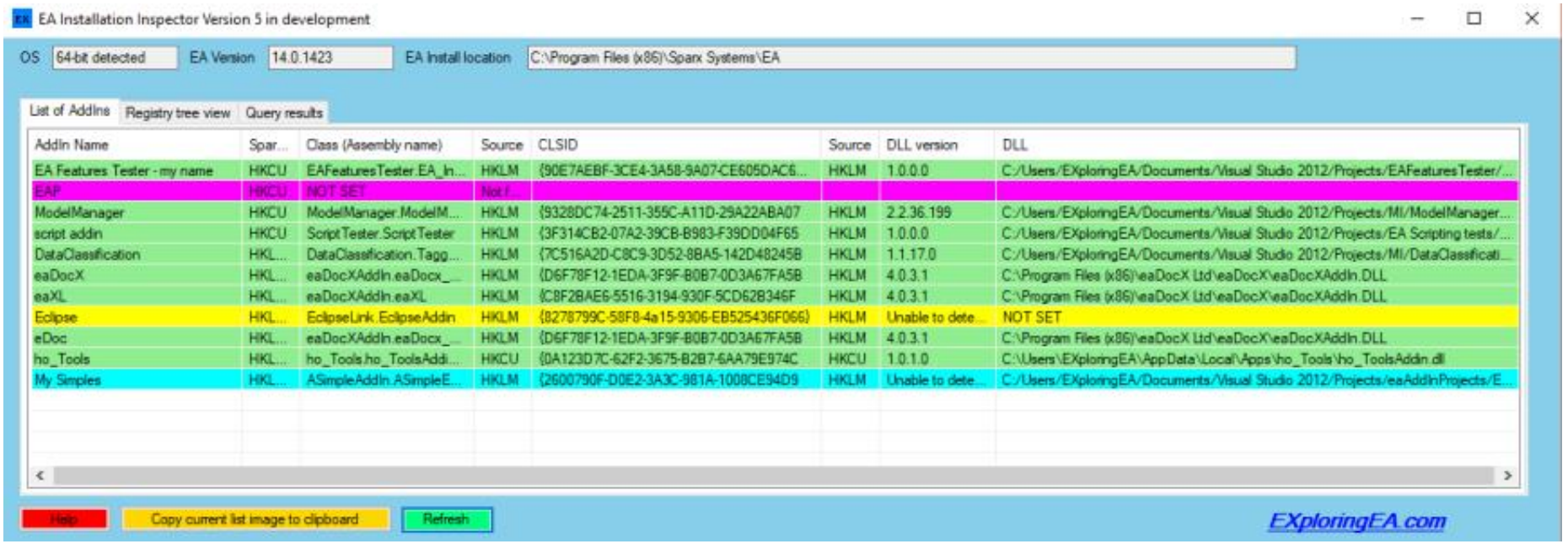
- Searches HKLM for Sparx keys as well as HKCU
- Pop-up added to make it easier to read

PURPOSE

EA Installation Inspector is a small utility for developers to search the registry to check information about current EA AddIn's.

RUNNING EA INSTALLATION INSPECTOR

The program is a windows application that will present the user with basic information about the installed environment and version together with a list of the EA Addin keys found in the registry with details of the relevant classes/DLLs - similar to the screen shot below.



Each row represents an AddIn entry – and for a valid entry will display all of:

- The AddIn Name
- The location of the Sparx AddIn key within the windows registry

- Class(Assembly name) - Addin entry class
- (CLSID) Source – the location where the ClassID is defined in the registry
- CLSID – Class ID – as defined when the class was registered
- (DLL) Source – the location where the DLL is defined in the registry
- DLL Version – read from the file information
- DLL – Full file name for the AddIn DLL

In some cases, and as illustrated in the screen shot not all entries are complet; it may be that the AddIn is working or correctly installed and to help see any issues each row is coloured to reflect the status of the entry.

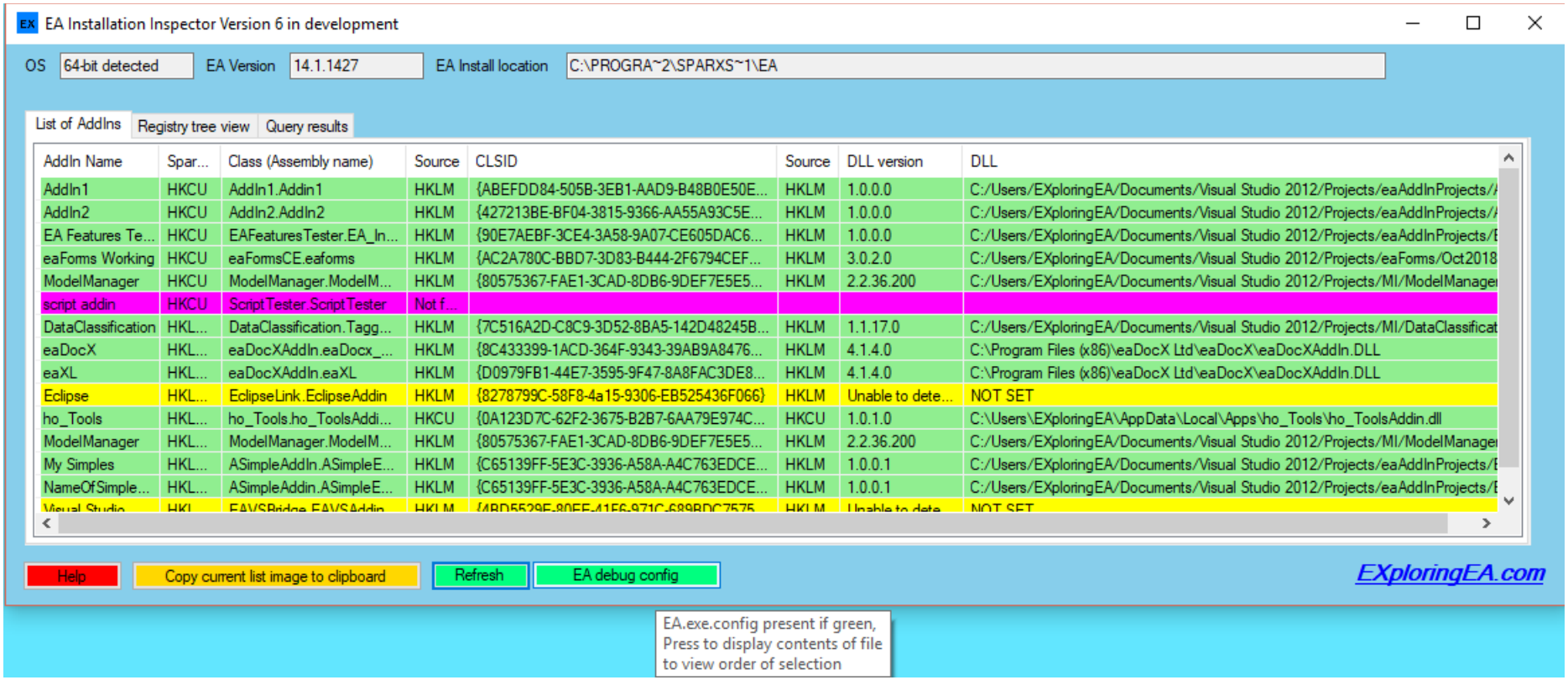
- Green – OK - the AddIn DLL has been found and the keys exist in the same hive; we assume that AddIn will be found by EA.
- Cyan – indicates that all the keys look fine but the DLL file does not exist at the specified location
- Magenta – means that no Class ID is set for the AddIn, hence the DLL cannot be indentified
- Red – indicates that CLSID and DLL are specified in different registry Key Sections
- Yellow – means that the DLL path is not set so cannot be found

FUNCTIONS

The program will automatically perform the search and present the list of AddIn's when run. The following functions are available by accessing the buttons at the bottom of the screen dialog.

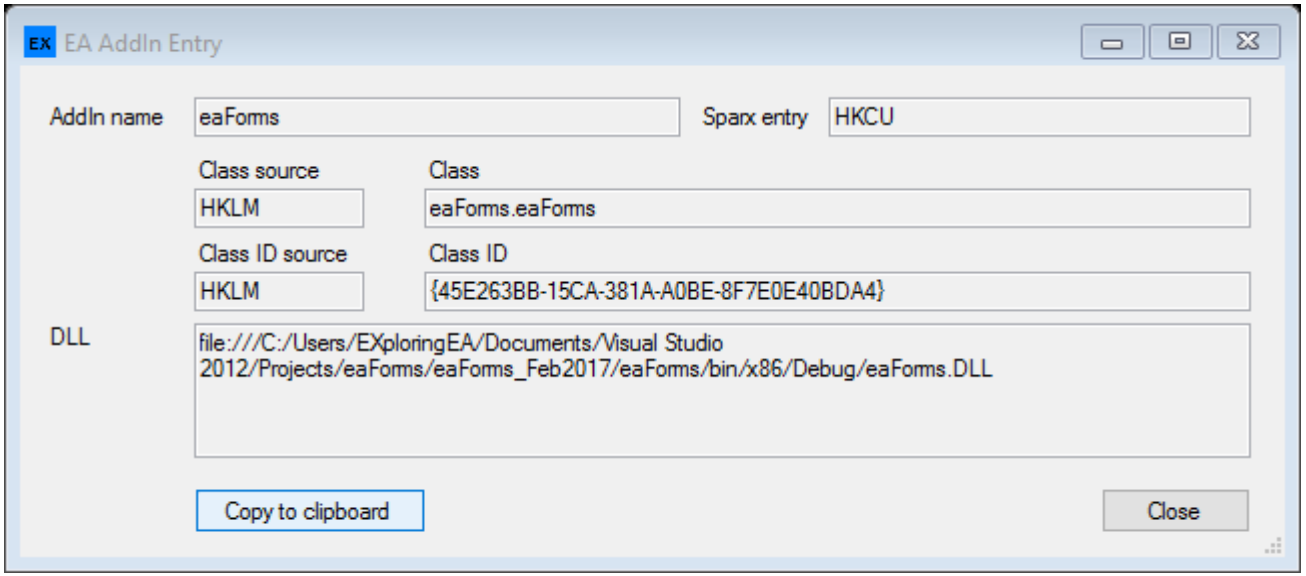
- **Help** – will present this document
- **Refresh** – will redo the search to reflect any changes that the user many have made to their system
- **Copy current list image to clipboard** – will copy the current list as an image to the clipboard for those times when the user may wish to forward to others.

- **EXploringEA.com –** will launch the default web browser with our blog page. You can also find contact information on this site should you wish to contact us
- **EA debug config –** this button is only presented if there is an EA.exe.config file present in the EA install path; the file is typically used to specify the .NET run time version to use. The screen shot below illustrates a case where the button is present.



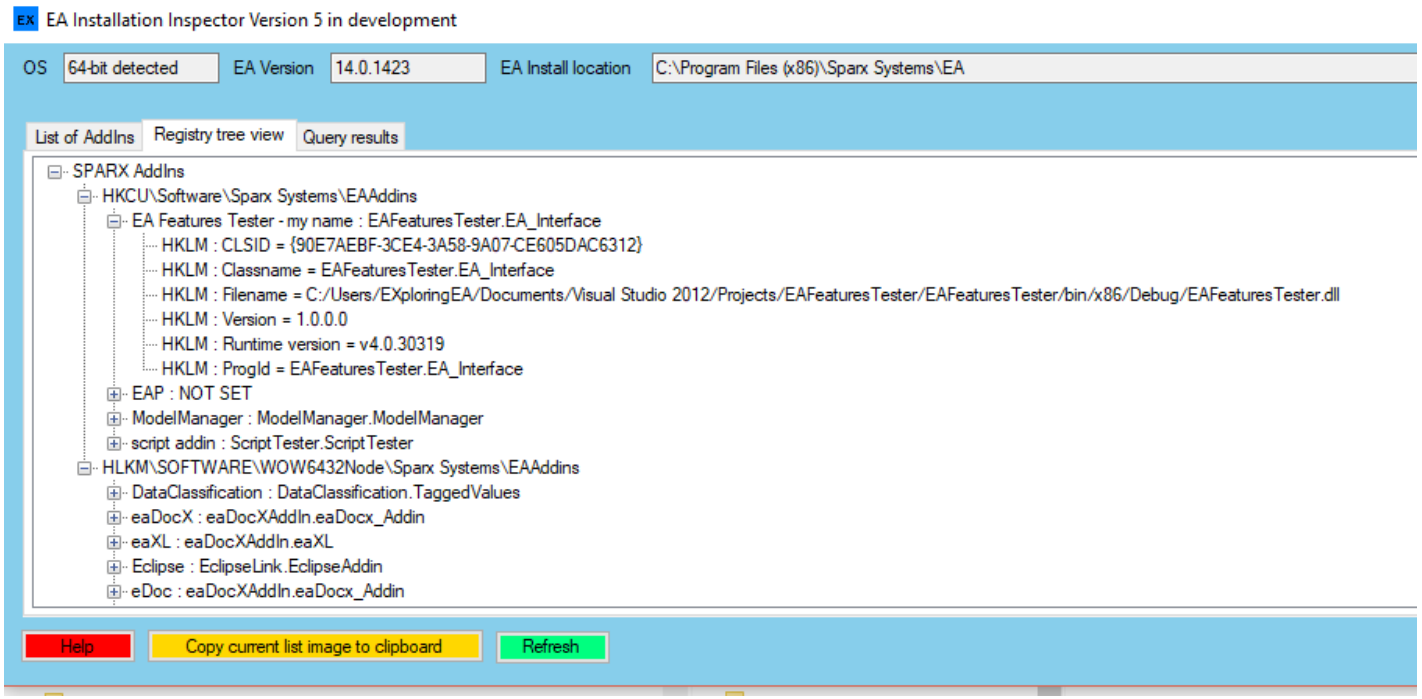
POP-UP ENTRY DETAILS FORM

It can sometimes be difficult to see all the information in a row so you can now ***double_click*** an entry and a form (similar to that illustrated below) is presented with the values more readily seen.

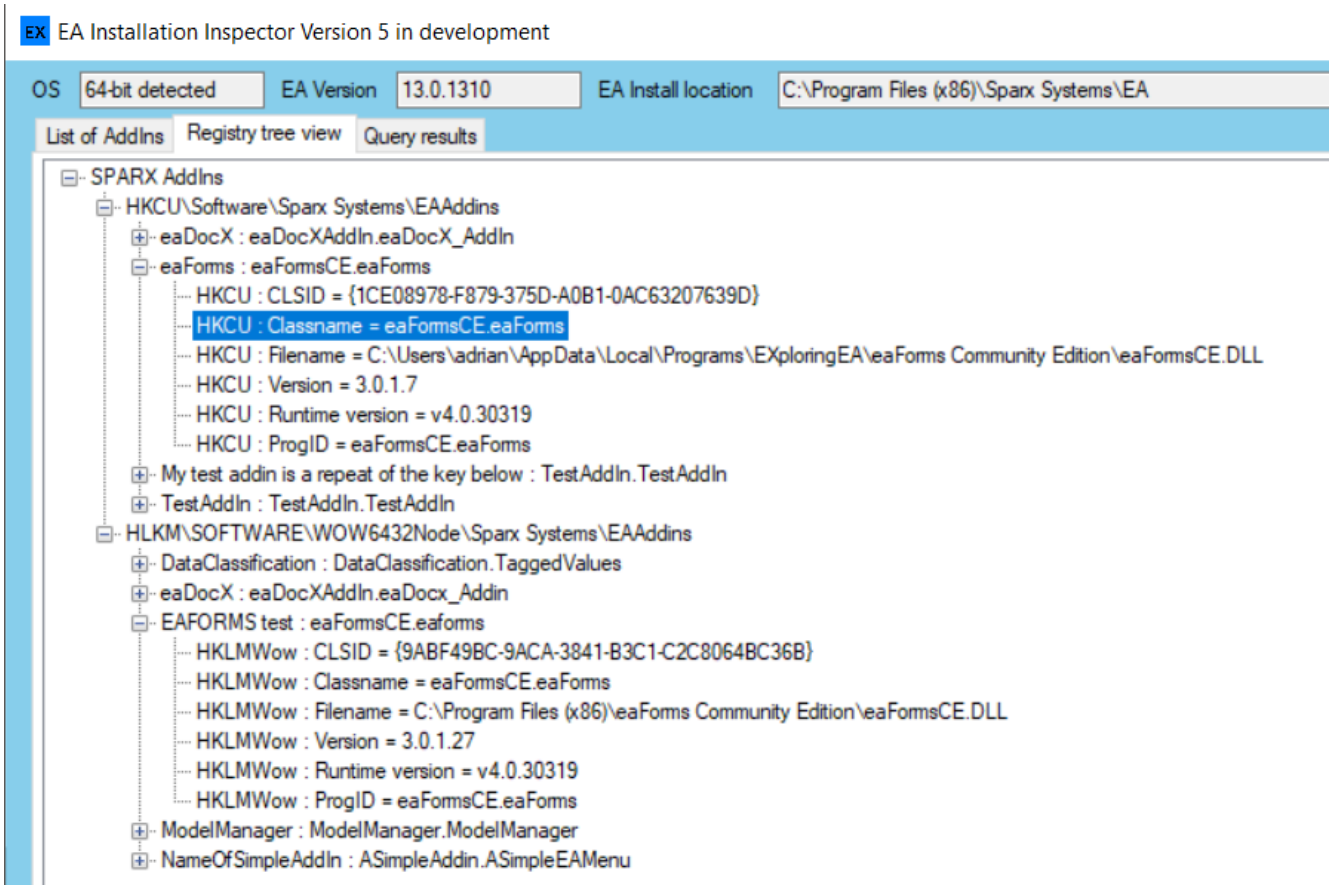


REGISTRY TREE VIEW

This tab presents the AddIn information as a tree format, based on the location of the information stored within the registry



This means that if there are entries for an AddIn that is both listed in HKLM and HKCU then the differences will be visible. See example screenshot below in which HKCU eaForms and HKLM EAFORMS Test - are the same AddIn, but could be different files, especially during development.



REGISTRY TREE VIEW QUERIES

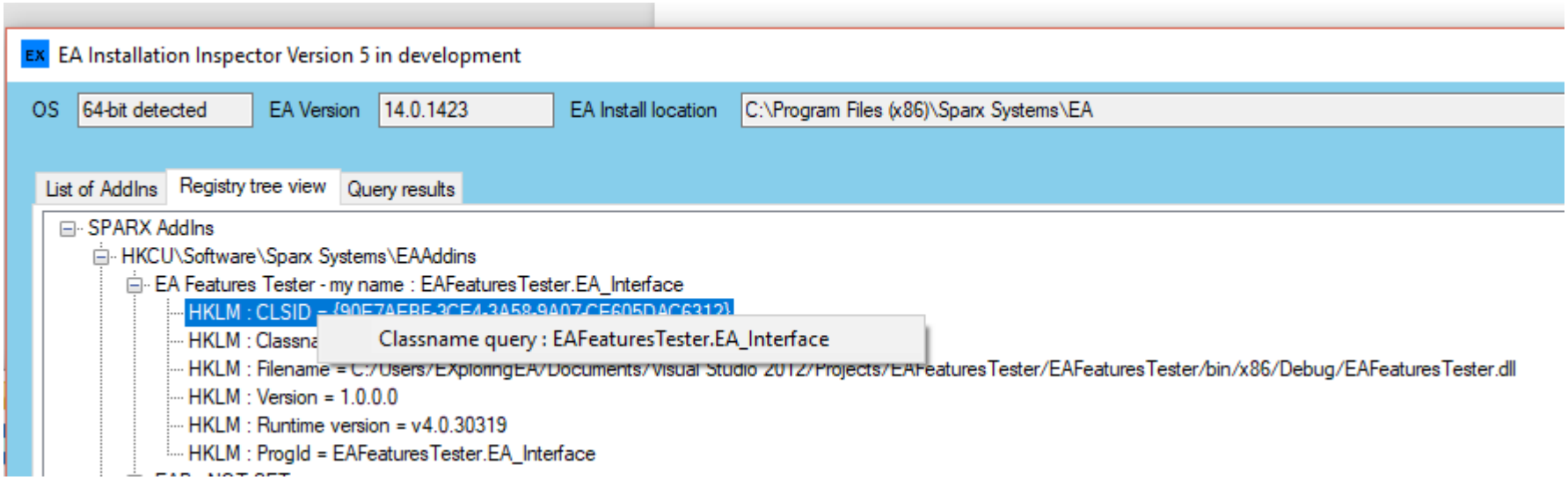
The user can select the context menu for some items:

- CLSID
- Classname
- Filename
- ProgID

to initiate a query for value within the selected line i.e. GUID, classname, filename, classname respectively. Menu items that provide an immediate response will have a pop-up dialog, whilst those that query the registry will be initiated in the background (so other actions can continue) - with the current registry query present in the Query info text box and a "Query active" indicator present on its right hand side, and results presented in the Query tab, where each query executed is listed before its results are displayed.

CLASSNAME QUERY

For CLISD and Classname entries a classname query can be performed. This will search the registry and output details of ANY key which contain the classname.



FILENAME ENTRIES:

- Open file location in windows explorer - will open windows explorer at the location specified in the selected key
- FileInformation - will present a pop-up window displaying information about the DLL.
- Filename query - will initiate a query across all the registry for the specific DLL.

An example of the query is **Query started: reg query HKLM\SOFTWARE\CLASSES /reg:32 /s /f EAFeaturesTester.dll** The results are display in the query results tab. Note some queries may take some time - they will execute in the background.

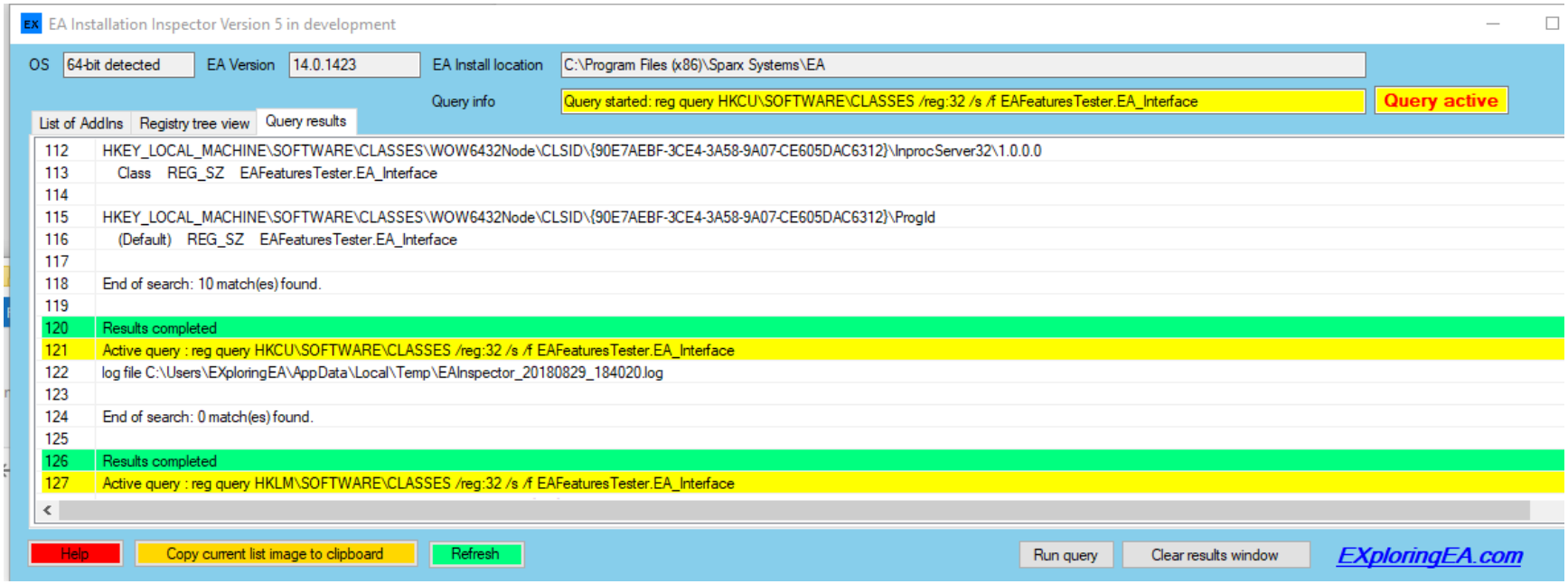
PROG ID ENTRIES

The Prog ID query will search the whole registry for ProgID's that match the selected entry. An example of the query is **Query started: reg query HKCU\SOFTWARE\CLASSES /reg:32 /s /f EAFeaturesTester.EA_Interface**

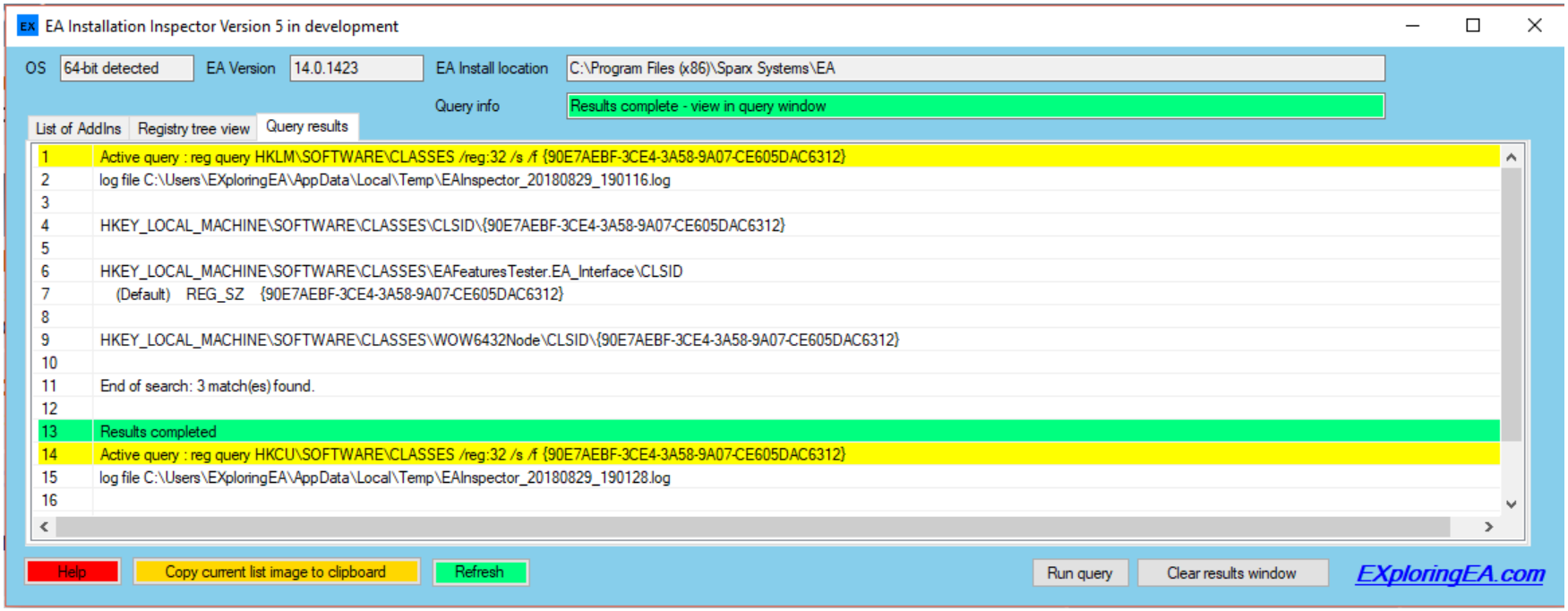
QUERY RESULTS TAB

The main function of this tab is to display results from queries. The output is display in the main area as shown below.

NB: Queries of the registry can take a lot of time hence each query is queued and the results output as they are obtained. The current query is displayed in the **query info** box which will have a yellow background with red text whilst queries are running, also a **Query active** label to the right of the query info box will be present. When all queries are completed then the **query info** box background turns green and the **query active** label disappears.



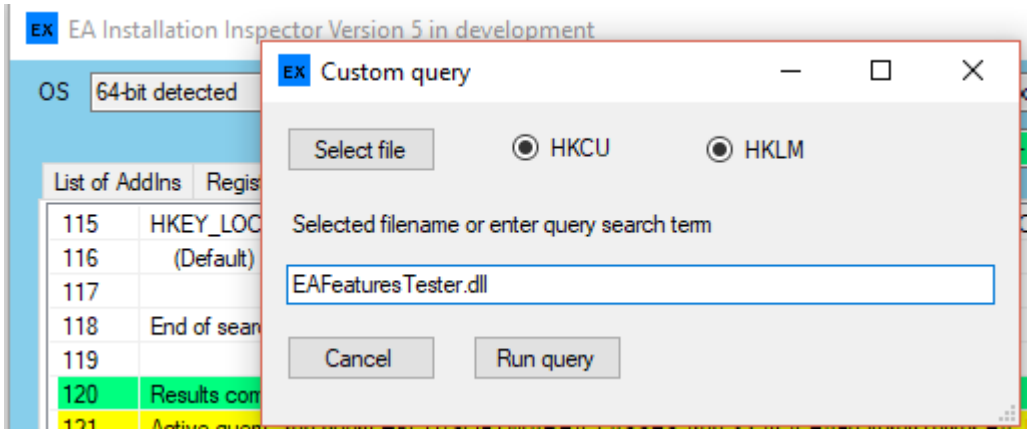
Below is a screenshot of all queries complete - also note the name of each log file should the user wish to get a hard copy of the results for a specific query.



Note: Unless the user specifically clears the results, they are appended to the page so there may be a need to scroll down.

The **Clear results window** button can be used to clear the contents.

In addition to queries initiated by the context menu on the **Registry Tree View** a user can create their own query using the **Run query** button.



QUERY LOG FILE

A log file for each query is produced and saved in the users AppData area in the directory for example:

C:\USERS\USERNAME\APPDATA\LOCAL\TEMP\EAINSPECTOR_DATE_TIME.LOG

OTHER BUTTONS

The program will automatically perform the search and present the list of AddIn's when run. The following functions are available by accessing the buttons at the bottom of the screen dialog.

- Help – the local help file may not be up to date.
- Refresh – will redo the search to reflect any changes that the user many have made to their system
- Copy current list image to clipboard – will copy the current list as an image to the clipboard for those times when the user may wish to forward to others.
- EXploringEA.com – will launch the default web browser with our blog page. You can also find contact information on this site should you wish to contact us

ASSUMPTIONS AND METHOD

As we don’t have access to the Sparx Code which loads the AddIn’s, it is assumed that AddIn’s registry keys are specified in locations under "Sparx Systems\EAAddins" within either in HKCU or HKLM in the registry (if running on 64-bit systems this means also under "Wow6432Node"). Using the list of Addins found in those locations a search is made in the registry for the referenced classes and their DLL’s.