JMMC-MAN-2200-0001

Revision 4.0

Date: 26 Feb 2013

JMMC

AppLauncher 1.1.1 - User Manual

Authors:
Sylvain LAFRASSE (SLa) <Sylvain.Lafrasse@obs.ujf-grenoble.fr> — IPAG

Author: JMMC Technical Group Institute: IPAG	Signature: Date: 26 Feb 2013
Approved by: Sylvain Lafrasse Institute: IPAG	Signature: Date: 26 Feb 2013
Released by: Guillaume Mella Institute: IPAG	Signature: Date: 27 Feb 2013

Change record

Revision	Date	Authors	Sections/Pages affected
	Remarks		
1.0	15 Mar. 2012	SLa	All
	First version.		
2.0	11 May 2012	SLa	All
	Updated main window description and screenshot, and first run description; Added preferences description; Included review comments, added conclusion.		
3.0	18 Sep. 2012	SLa	All
	Added description for automatic report and command-line native apps, updated screenshots.		
4.0	26 Feb. 2013	SLa	First Chapter
	Added interoperability introduction.		

Table of contents

1 Introduction **5**
1.1 What is Interoperability ? . . . 5
1.2 Presentation . . . 5
1.3 Installation . . . 5
1.4 Acknowledgments . . . 5
1.5 Method . . . 6
1.6 Abbreviations and acronyms . . . 6

2 How to use AppLauncher **7**
2.1 First run . . . 7
2.2 Usage example . . . 9
2.3 Configuration . . . 10
2.4 Adding applications . . . 11
2.5 Quitting . . . 11
2.6 Other functions . . . 13

3 Conclusion **13**

List of Tables

List of Figures

1 Main Window 7
2 AppLauncher Welcome Window 8
3 AppLauncher Auto-Test Window 9
4 AppLauncher Starter Popup 9
5 AppLauncher Applications Preference 10
6 AppLauncher Report Form 11
7 Command-line Dialog Box 12
8 AppLauncher Quit Alert 12

1 Introduction

1.1 What is Interoperability ?

Everybody's computer nowadays run multiple programs at once. And users are eager to easily pass data back and forth between those programs. To achieve that, programs must understand common data exchange formats, and also speak the same common protocol.

For example : You are preparing your next observation in your favorite tool. What if you could easily get calibrating stars directly in it from another software you don't even now yet ? What if you could directly fit simulated observations to verify your strategy in one click ?

This is interoperability : common data formats and communication protocols, allowing different programs to collaborate easily.

And *AppLauncher*[1] is here to simplify even more your experience !

1.2 Presentation

SAMP[2] is the dedicated Virtual Observatory protocol to ensure data exchange between compatible astronomical software running on personal computers. However, one *SAMP* weakness lies in its requirement to have interoperable applications already running in order to gracefully ensure communication between them.

To circumvent this requirement, the *JMMC*[3] created *AppLauncher*, our dedicated application that ensures transparent communication between (even not yet running) *SAMP* tools ! *AppLauncher* is a *SAMP* hub extension, able to automatically start dedicated applications 'on demand'. It also provides a 'Dock' window with all *JMMC* applications and VO compliant tools.

1.3 Installation

Availability *AppLauncher* is freely available for download on the *JMMC* website at `http://www.jmmc.fr/AppLauncher`.

Requirements *AppLauncher* is based on Java™1.5 or greater, so your computer must have it installed (see `http://www.jmmc.fr/apps.htm` for more details). An Internet connection is required to start applications the first time, and recommended later on, even if operation may work without any.

1.4 Acknowledgments

As with all other *JMMC* software, you must acknowledge *AppLauncher* usage if it was of any help in your research. The official text is available from the `Help` menu.

AppLauncher itself is based on multiple OpenSource libraries and public services. We especially would like to thank:

- *ASOV* and *IVOA*, for their dedicated support to make the astronomical software community a better place !
- Mark Taylor of Bristol University in England, for his huge commitment to ease astronomers everyday life with such great tools as *jSAMP*[4] library enabling easy yet reliable inter-application data exchanges;
- *jMCS*[5], the core *JMMC* graphical interface library, providing lots of missing functionalities in Java™, either developed internally or by incorporating third-party libraries, to help our end-users feel right at home using our apps.

[1] `http://www.jmmc.fr/applauncher`
[2] Simple Application Messaging Protocol - `http://www.ivoa.net/samp/`
[3] Jean-Marie Mariotti Center (a.k.a as JMMC) - `http://www.jmmc.fr`
[4] `http://software.astrogrid.org/doc/p/jsamp/1.3-2/`
[5] `http://www.jmmc.fr/dev/jmcs`

1.5 Method

AppLauncher works by simulating well-known *SAMP*-compliant applications (such as *Aladin*, *TOPCAT* or *JMMC* tools). Further technical details are described in a paper (http://www.jmmc.fr/doc/approved/JMMC-PUB-2220-0001.pdf) and a poster (http://www.jmmc.fr/doc/approved/JMMC-POS-2220-0001.pdf) both presented in 2011 at ADASS XXI in Paris.

1.6 Abbreviations and acronyms

JMMC	Jean-Marie Mariotti Center, the french center for infrared and optical interferometry, providing support for the users of the astronomical interferometers currently in operation around the world
ASOV	Action Specifique Observatoires Virtuels France
IVOA	International Virtual Observatory Alliance
SAMP	Simple Application Messaging Protocol, defined by IVOA
JNLP	Java™Web Start application deployment technology from Sun
jMCS	Java™Mariotti Common Software
CDS	Centre de Donnees Astronomiques de Strasbourg, providing Aladin software
ADASS	Astronomical Data Analysis Software and Systems conference cycle

2 How to use AppLauncher

In Fig. 1 you can see *AppLauncher* main window, displaying some of the compatible applications available to you. If you click on any of these icons, the latest official version of the corresponding application will be downloaded from the Internet (if needed) and started. If you are not connected at this time, the last launched version will be started. *Please note that you at least need an active Internet connection the first time you want to start any application.*

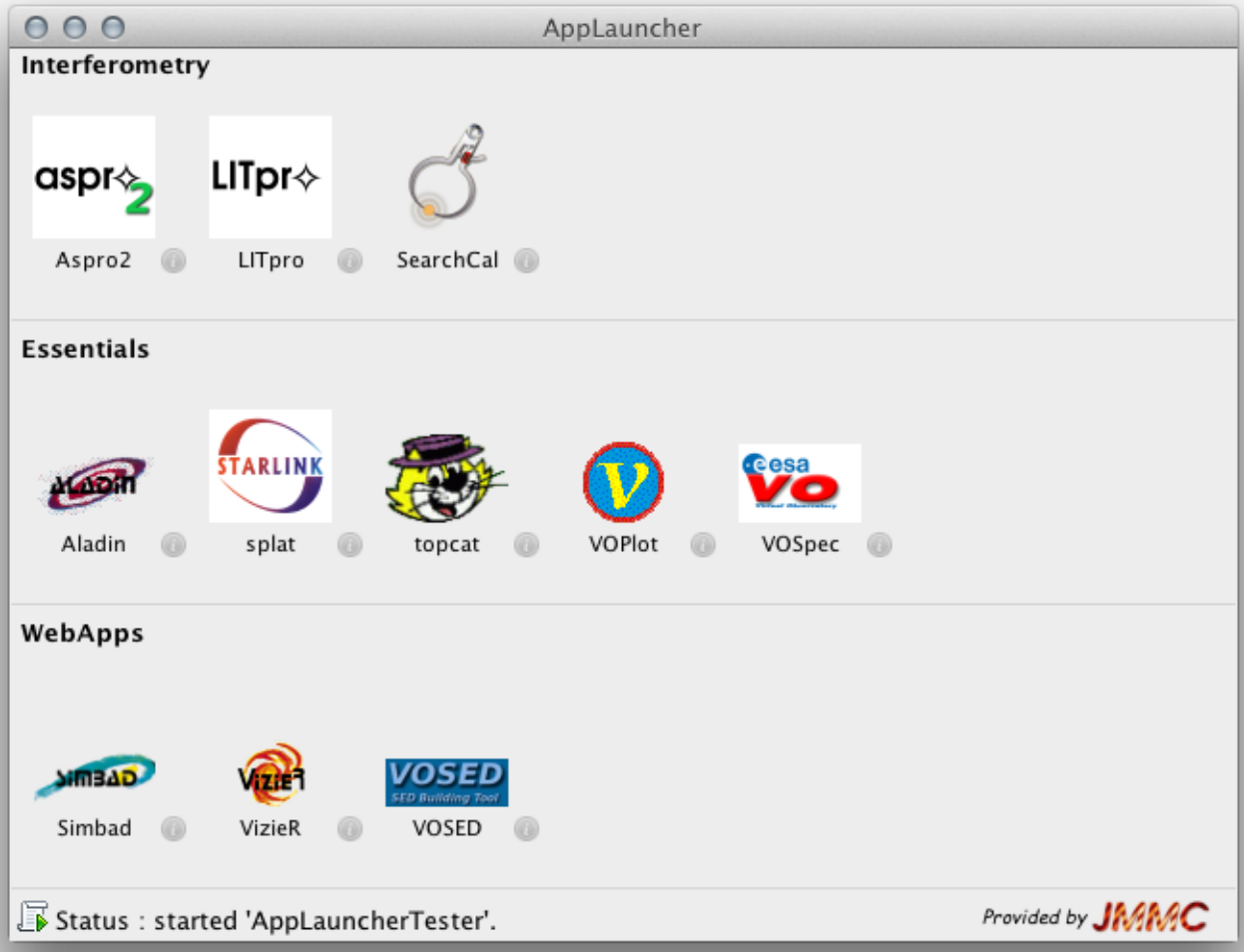

Figure 1: Main Window

If you click on the small `Info` icon next to each application icon, a detailled description of the corresponding application will be displayed.

2.1 First run

The first time you launch *AppLauncher*, a `Welcome` window (see Fig. 2) will be shown, briefly explaining *AppLauncher* purpose and main options.

Figure 2: AppLauncher Welcome Window

Once you click `OK`, an auto-test routine will automatically proceed to ensure everything is correctly set up to start *SAMP* applications on demand. If the test went fine, you should see the window in Fig. 3. You can also manually relaunch this test later, and also further configure Java WebStart, from the `Help` menu.

Figure 3: AppLauncher Auto-Test Window

2.2 Usage example

To better illustrate the way *AppLauncher* works, start by clicking the `SearchCal` icon. You should see the window in Fig. 4 stating that *SearchCal* is being started. Once *SearchCal* is available, click the `Get Calibrators` button and wait while results appear onscreen. Once done, use *SearchCal* `Interop` menu to send those results to any of the available applications (i.e either `Aladin` or `TOPCAT`) and watch *AppLauncher* automatically starting the desired application, then forwarding *SearchCal* results after a while !

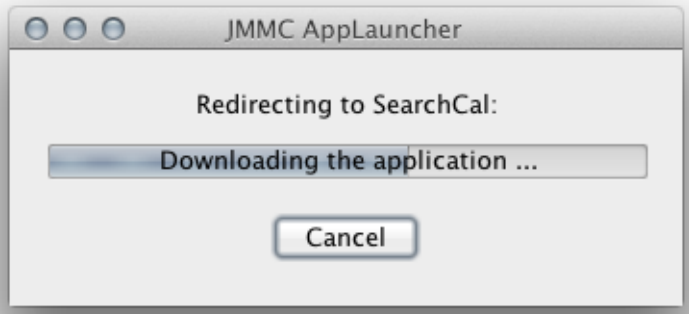

Figure 4: AppLauncher Starter Popup

2.3 Configuration

AppLauncher offers a limited set of applications by default. You can easily customize this using the `Preferences` window as shown in Fig. 5. Some applications can also be used in beta version (when provided by their authors). You can also disable a whole category if needed.

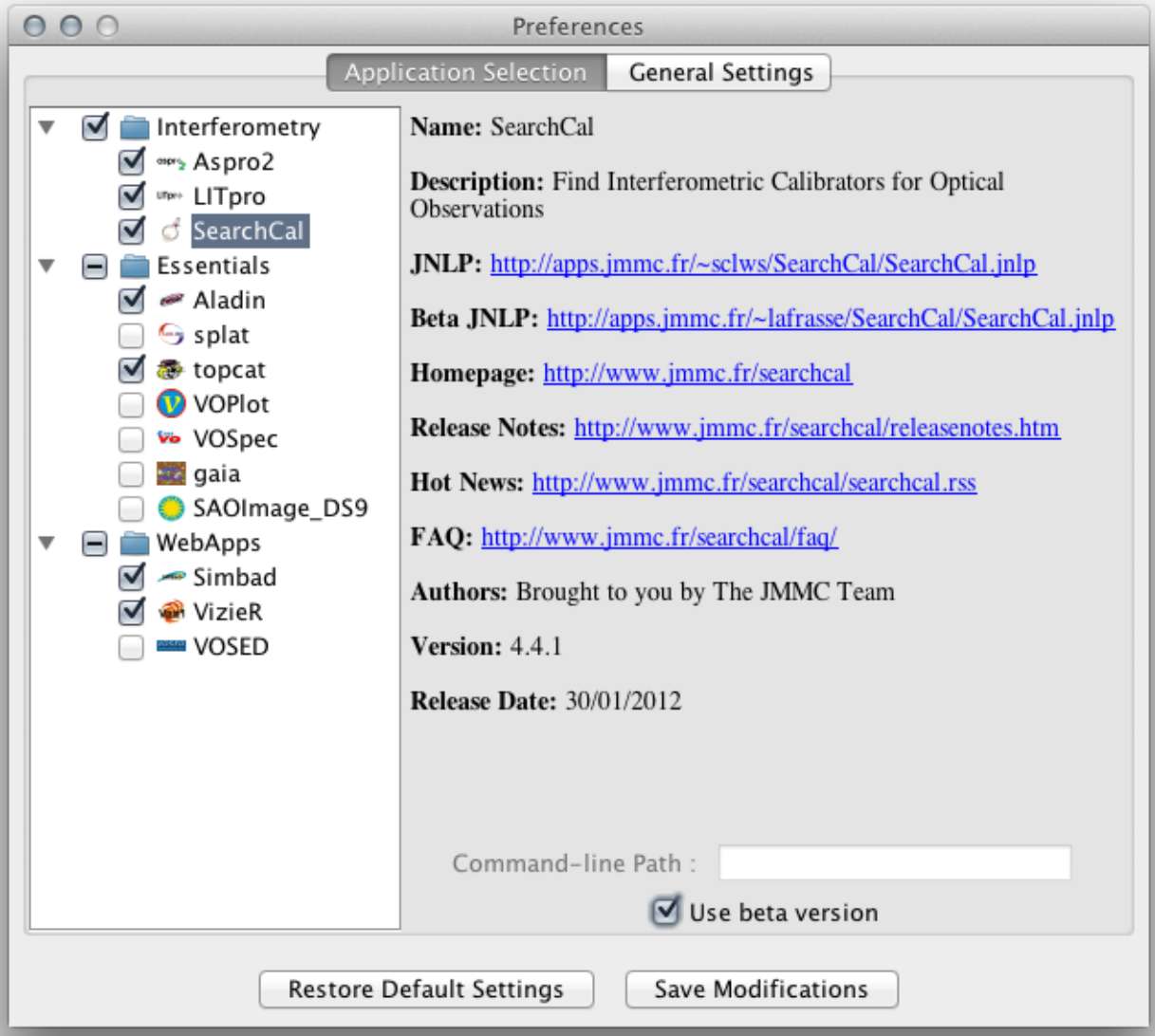

Figure 5: AppLauncher Applications Preference

Some general settings are also available to:

- show or hide the Dock window on startup (if you only need the *SAMP* simulation aspect of *AppLauncher* and not the visual dock);
- automatically report any future unknown applications detected, without asking for explicit permission before transmission;

- restrict *SAMP* simulation to your selected application set, or to all known application (very useful to discover new tools);
- show or skip the warning message when you are about to quit *AppLauncher* and kill the *SAMP* hub by the way.

Please note that AppLauncher must be restarted for those general settings to take effect.

2.4 Adding applications

In case you use some *SAMP*-compliant applications that *AppLauncher* does not know of yet, it is smart enough to discover them, and offers you to report them using the form presented in Fig. 6. We are eager to add all kinds of applications to *AppLauncher*, so don't hesitate to report ! To do so, you can also choose to automatically report any future unknown applications.

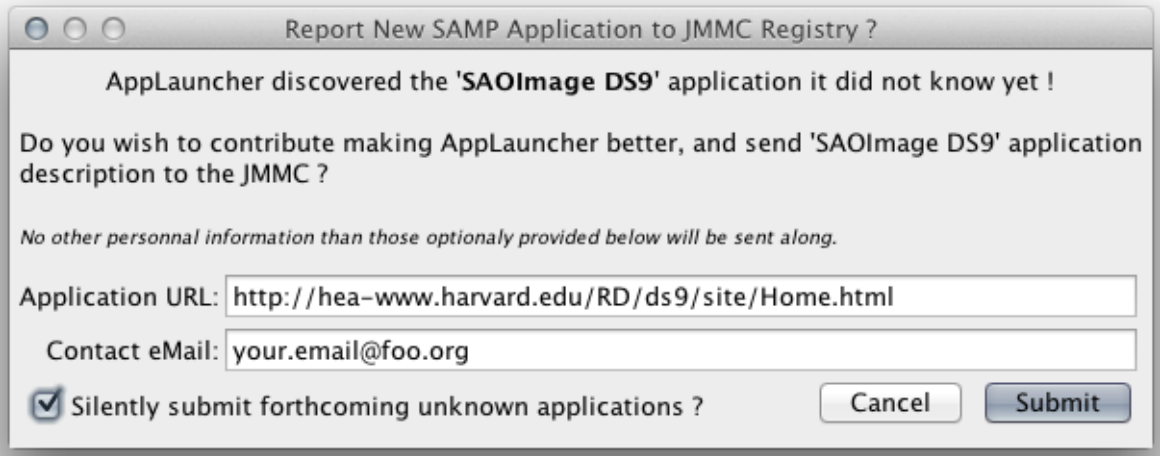

Figure 6: AppLauncher Report Form

Please note that preliminary support for native applications (i.e applications not directly available from the Internet as *JNLP*, thus requiring you to download and install it manually) has been added in version 1.1. In order to handle such applications, you have to provide *AppLauncher* with the command-line path to the application executable once installed, using either:

- the application's `Command-line Path` text field in the `Preferences` window (see Fig. 5);
- the dedicated dialog box (see Fig. 7) that will appear the first time you solicit the application through *SAMP*.

2.5 Quitting

AppLauncher is meant to be the first application you launch beginning your work session. If so, it takes the responsibility to initiate all the *SAMP*-related environment (i.e. the hub) for the other applications. So when you are about to quit *AppLauncher*, shutting down the hub by the same time, you are informed of this with the window shown in Fig. 8. You then have the possibility to change your mind if you feel so !

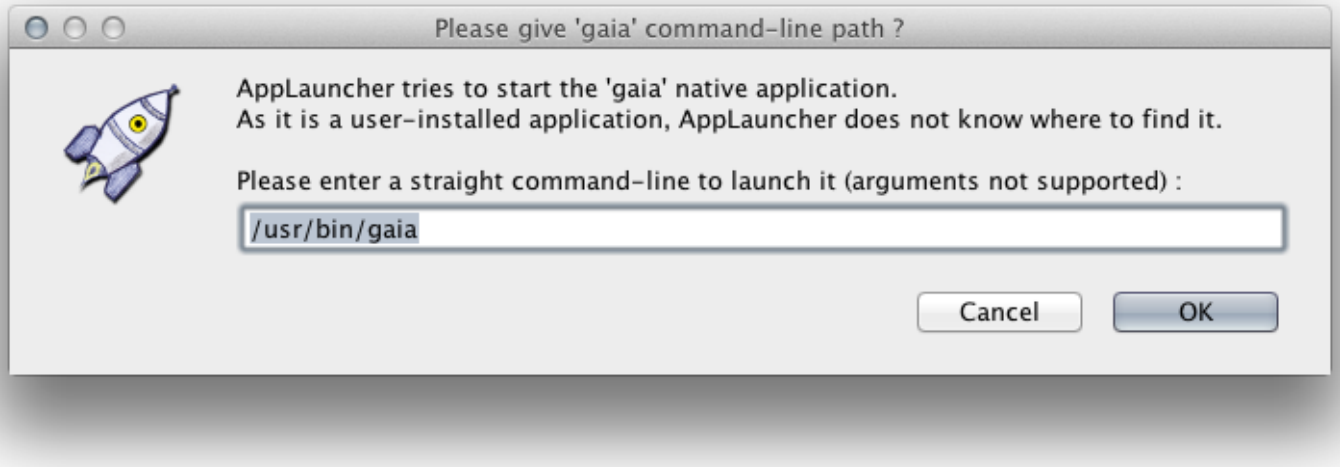

Figure 7: Command-line Dialog Box

Figure 8: AppLauncher Quit Alert

2.6 Other functions

As acknowledged in section 1.4, *AppLauncher* makes extensive use of *jMCS*, a *JMMC* OpenSource initiative that provides lots of neat features, such as:

- a deep platform integration to make applications feel native on any of the 3 main desktop operating system that are Mac OS X, Linux or Windows;
- an embedded user manual viewer;
- the ability to send feedback reports (either to signal software bugs, documentation typos, user questions or evolution requests), that will aways be very welcome by the *JMMC Technical Team*;
- a dedicated console window to monitor application execution and statuses;
- a way to directly copy-paste application acknowledgements to your scientific papers;
- a standardized `About Box` for further application details.

3 Conclusion

As usual, the *JMMC Technical Group* put its best in the creation of *AppLauncher*. We hope it will be of great help for your everyday work, and once more don't hesitate to give us feedback !