

Department of Veterans Affairs

Open Source Electronic Health Record Services

Roll and Scroll Recorder Usage and Installation Documentation



**Version 0.3
January 2013**

Revision History

Date	Revision	Description	Author
12/29/2012	0.1	Initial Draft	Jimmy Spivey
01/02/2013	0.2	Peer Review	Meredith Watkins
01/03/2013	0.3	Peer Review	Kathleen Keating
04/22/2013	0.4	Removed JCTerm	Jimmy Spivey

Table of Contents

1.	Introduction.....	7
1.1.	Overview	7
2.	Installation.....	7
2.1.	Prerequisites.....	7
2.2.	Steps to Install.....	7
3.	Usage.....	12
3.1.	Connecting to a Remote System	12
3.2.	Recording a Test	14

1. Introduction

The Department of Veterans Affairs (VA) has contributed the latest U.S. Department of State Freedom of Information Act (FOIA) release of the Veterans Health Information Systems and Technology Architecture (VistA) codebase to Open Source Electronic Health Record Agent (OSEHRA), the custodial agent that serves as the central governing body of a new open source community. The Open Source Electronic Health Record (EHR) Services project includes VistA Data Comparison, VistA System Test Platform, VistA Refactoring, VistA System Test Scripts, Veterans Benefits Administration (VBA) System Test Platform, Eclipse Plug-In Tool, and VistA Meaningful Use Certification.

1.1. Overview

The Roll and Scroll Recorder (RASR) is a test automation tool for the Veterans Health Information Systems and Technology Architecture (VistA) platform. The tool is an Eclipse plug-in which connects to a remote terminal via Secure Shell (SSH). While connected, input from the user and the remote system are recorded as an automated test. These tests are exported as Python scripts and placed into a Python based test automation framework (The Open Source Electronic Health Record Agent [OSEHRA] Automated Testing Framework [ATF]).

2. Installation

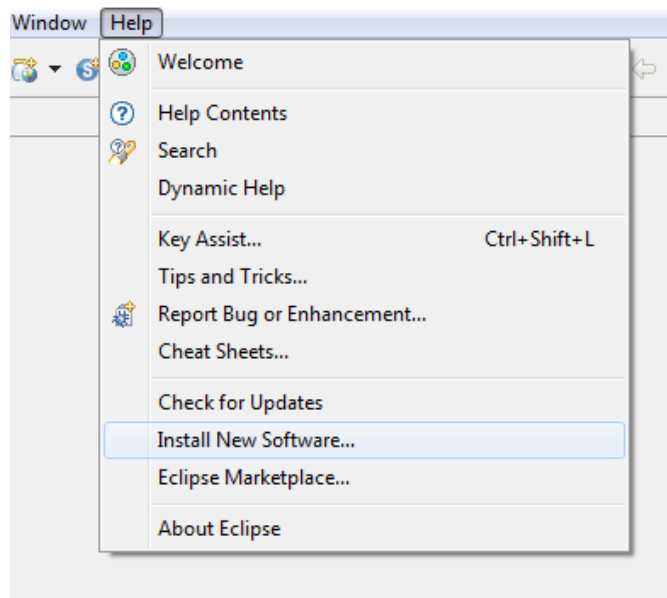
2.1. Prerequisites

Both RASR and the ATF support Windows and Linux.

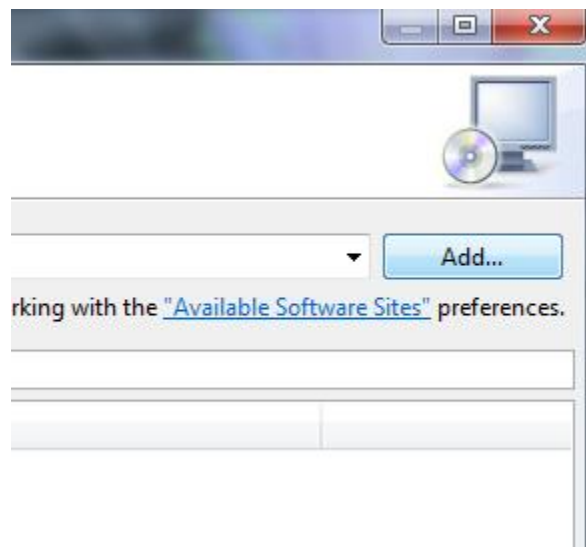
Application	Version	Notes
OSEHRA ATF	Latest	The current RASR compatible ATF is located at: https://github.com/JimDeanSpivey/ATF-RASR
Eclipse	Indigo	

2.2. Steps to Install

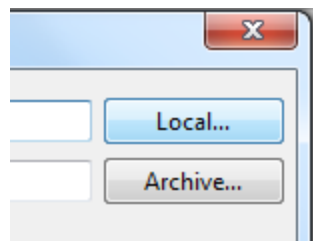
- 1) Install the OSEHRA ATF located at <https://github.com/JimDeanSpivey/ATF-RASR/archive/master.zip>. This version is modified to work with RASR.
- 2) Open Eclipse (Indigo).
- 3) Install RASR:
 - a. Download the latest RASR from <https://github.com/OSEHR/RASR/archive/master.zip>
 - b. Unpack the zip file
 - c. Open Eclipse
 - d. Click Help → Install New Software



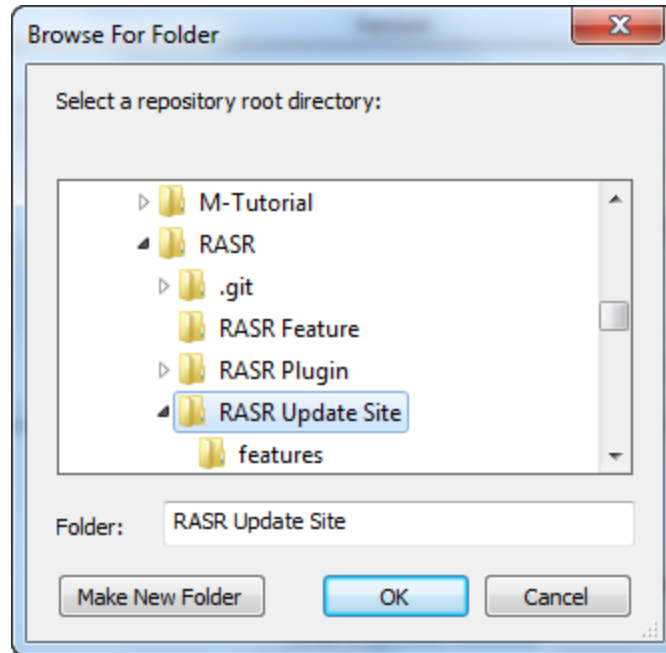
- e. Then click Add at the top right



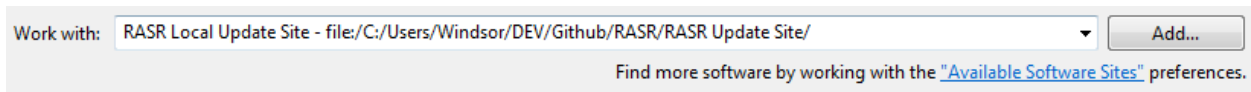
- f. Select Local towards the top right



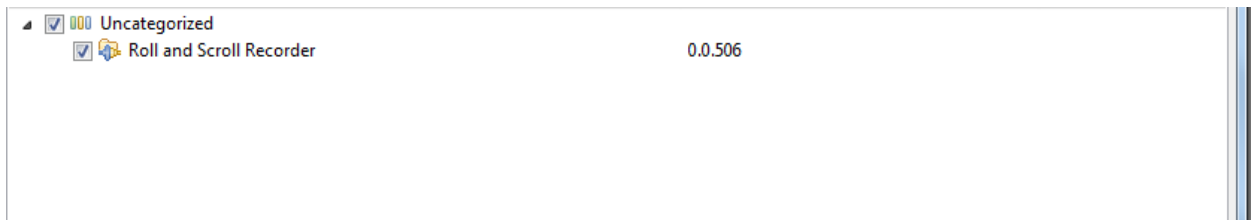
- g. Select the directory to where the RASR zip file was unpacked, then choose the "RASR Update Site" directory



h. After clicking OK, the newly added update site will already be selected:



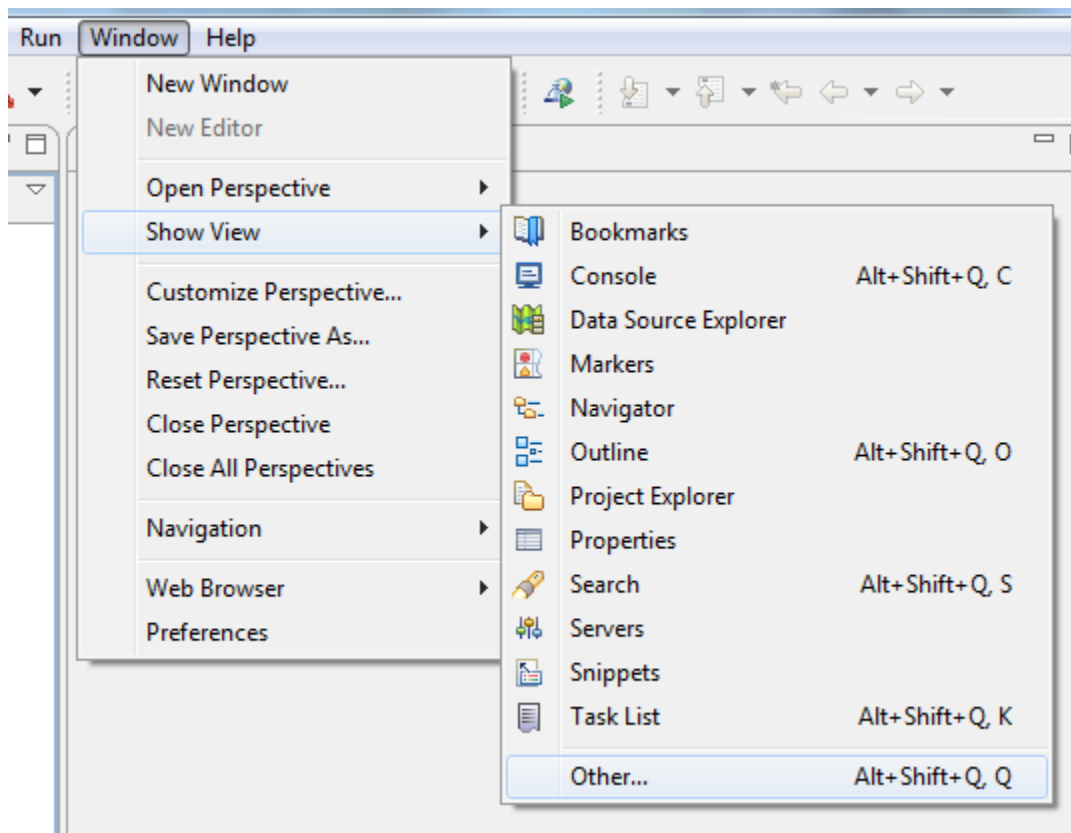
i. Select the RASR plugin.



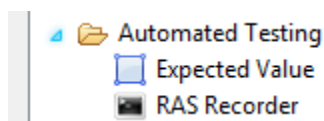
j. Click Next and follow the prompts to install the plugin. Eclipse will restart when complete.

4) Setup RASR:

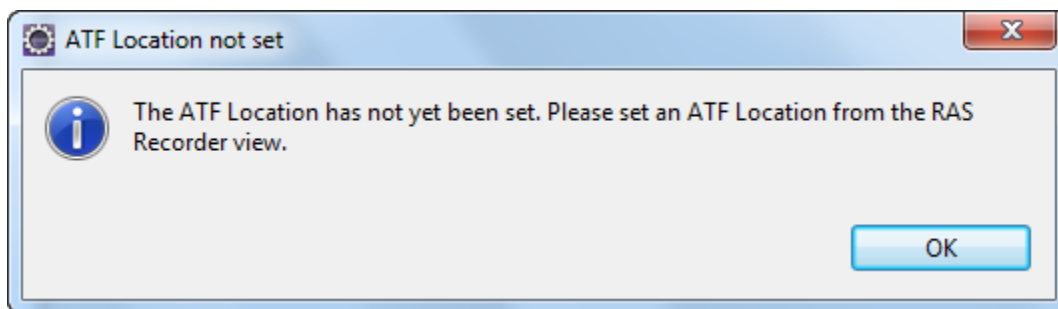
a. To operate RASR, both the “RAS Recorder” and “Expected Value” view must be displayed. To display both of these views click Window → Show View → Other



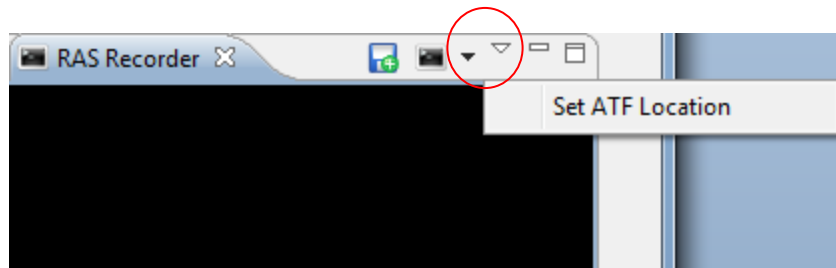
Both views will be present under the “Automated Testing” category (both views are required to be displayed while RASR is running):



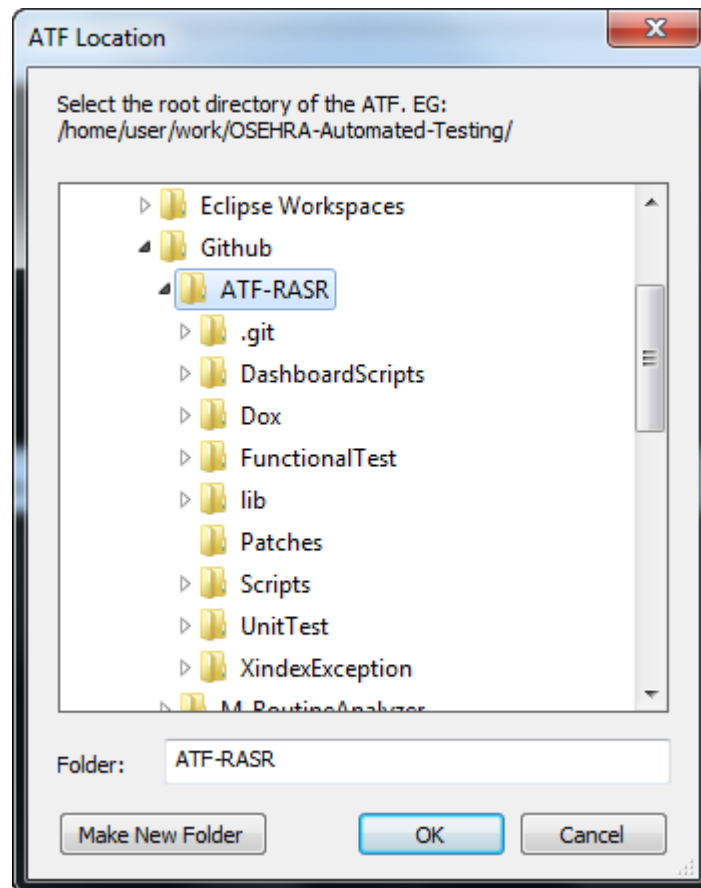
b. RASR must know where the ATF exists. You will be promoted to set this:



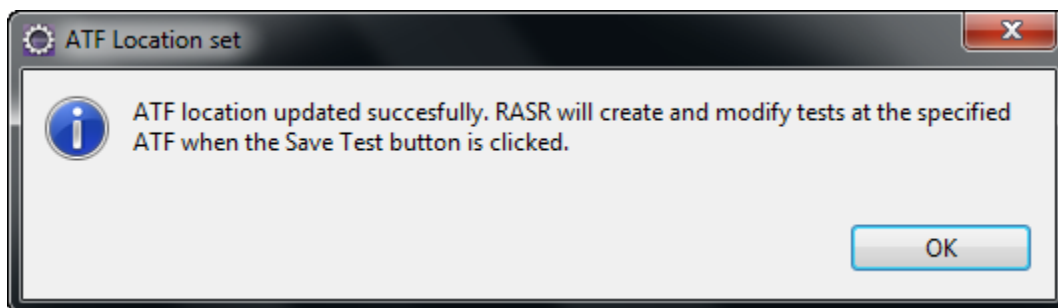
To set this, click on the white drop-down arrow on the RAS Recorder Tab:



Select the location of the ATF's root directory.



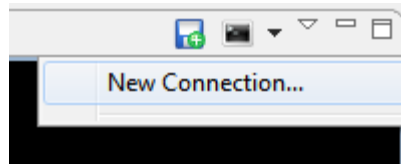
If the incorrect directory is chosen it will display a warning. If a valid ATF directory is set it will display an informational message:



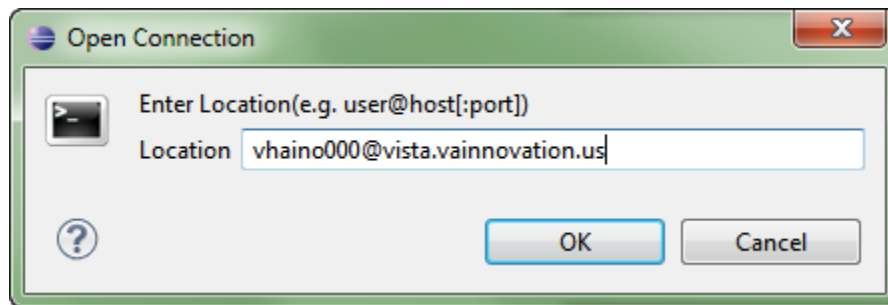
3. Usage

3.1. Connecting to a Remote System

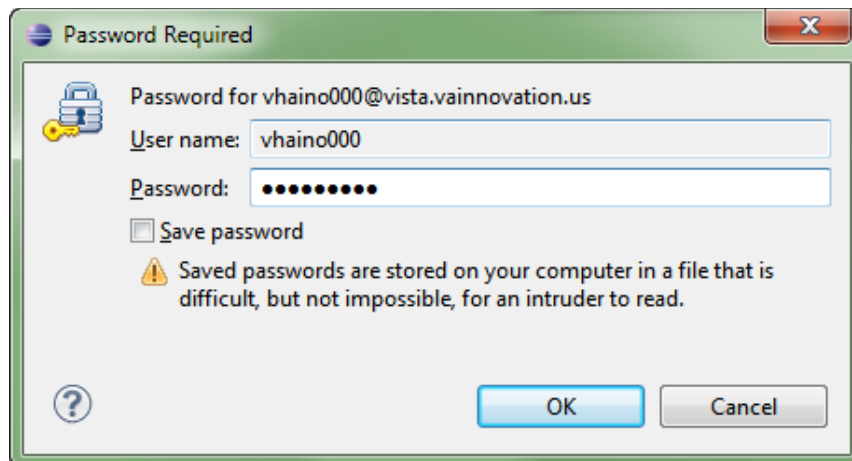
- 1) Begin by opening a new connection from the drop down menu:



- 2) Enter the connection details in the format of 'SSH-Username@Remote-Address':



- 3) Enter the SSH password on the next prompt:



- 4) Once connected, both views will display the terminal output. Access codes are not recorded:

```
Expected Value 83

MISUSE of this system constitutes a Federal Crime.

*****

>>>> CPRS is installed in this account <<<<
>>  CPRS GUI IS AT VERSION 27.90 (OR*3*296)  <<
    Copy Date:
*****

MISUSE of this system constitutes a Federal Crime.
CPM  CPM  CPM  CPM  CPM  CPM  CPM  CPM  CPM  CPM  CPM
*****

Volume set: ROU:CACHE000  UCI: CPM000  Device: /dev/pts/1 (115.67.199.42)
ACCESS CODE:

vhainc000@vista.vainnovation.us X

VIEWing this message indicates that YOU agree with the terms of system
use.

MISUSE of this system constitutes a Federal Crime.

*****

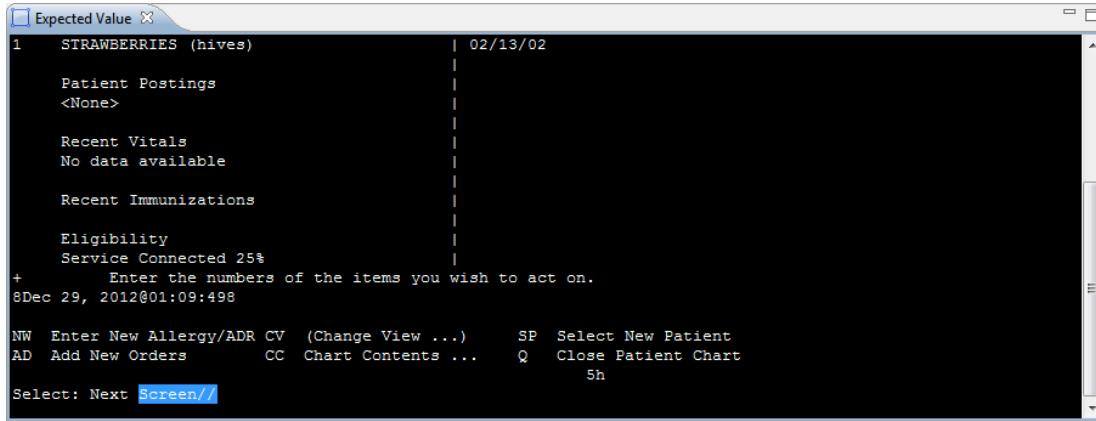
>>>> CPRS is installed in this account <<<<
>>  CPRS GUI IS AT VERSION 27.90 (OR*3*296)  <<
    Copy Date:
*****

MISUSE of this system constitutes a Federal Crime.
CPM  CPM  CPM  CPM  CPM  CPM  CPM  CPM  CPM  CPM  CPM
*****

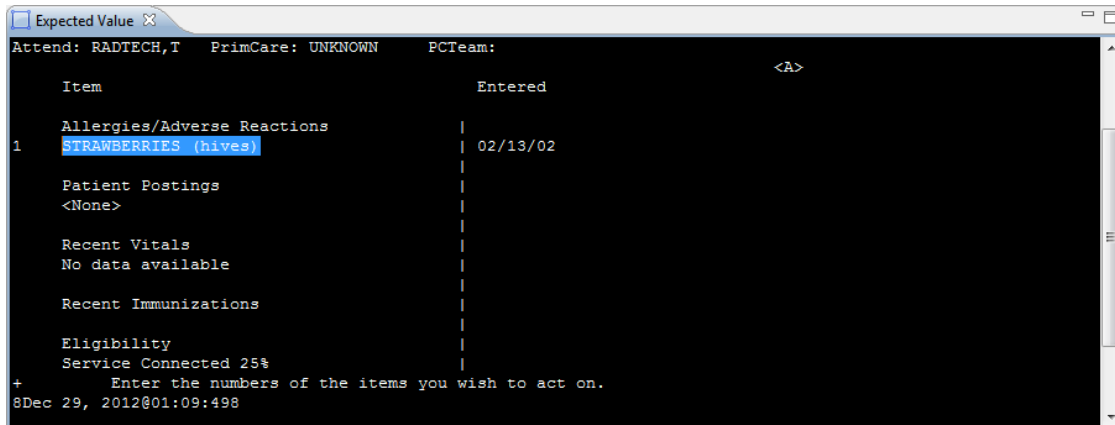
Volume set: ROU:CACHE000  UCI: CPM000  Device: /dev/pts/1 (115.67.199.42)
ACCESS CODE: 5)
```

3.2. Recording a Test

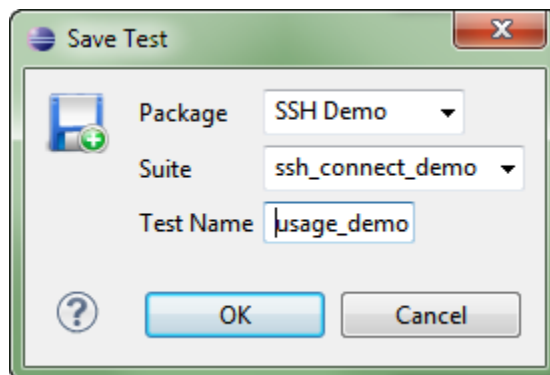
As you progress through the VistA screens, all entered input is recorded. However, for input coming back from the system, only the last word is recorded.



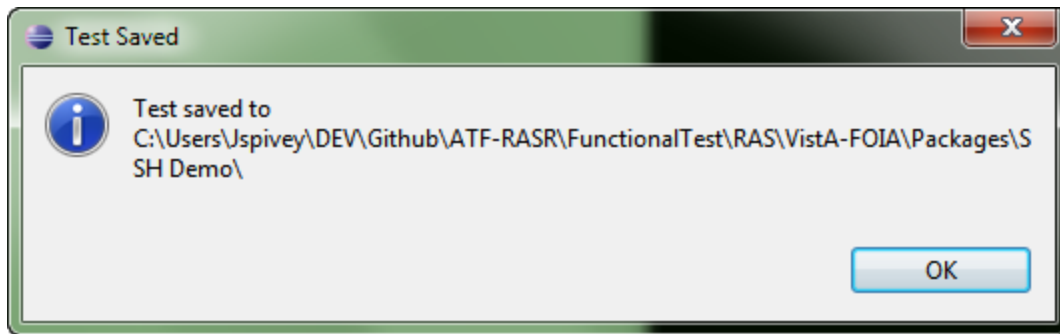
It is possible to override this default in the Expected Value view by selecting the text you want to record.



When you are ready to save your test into the OSEHRA ATF, click the save test button. Either select a new package and test suite or use the existing one, then give the test a name.



The test will be saved, and a notification displaying where it was saved to is displayed.



The test is now a part of the OSEHRA ATF and can be run via C Test. Invoking the Python file directly (and passing some required arguments) is also an option.