



# **NHSC/PACS Web Tutorials**

## **HIPE Essentials**

### **PACS-102**

### *Reading and Executing Custom Scripts*



# Introduction

This tutorial shows how to load a custom script into HIPE and how to execute it.

## Pre-requisites

1. You should have completed the following tutorials:
  - ***PACS-101: How to use these tutorials.***

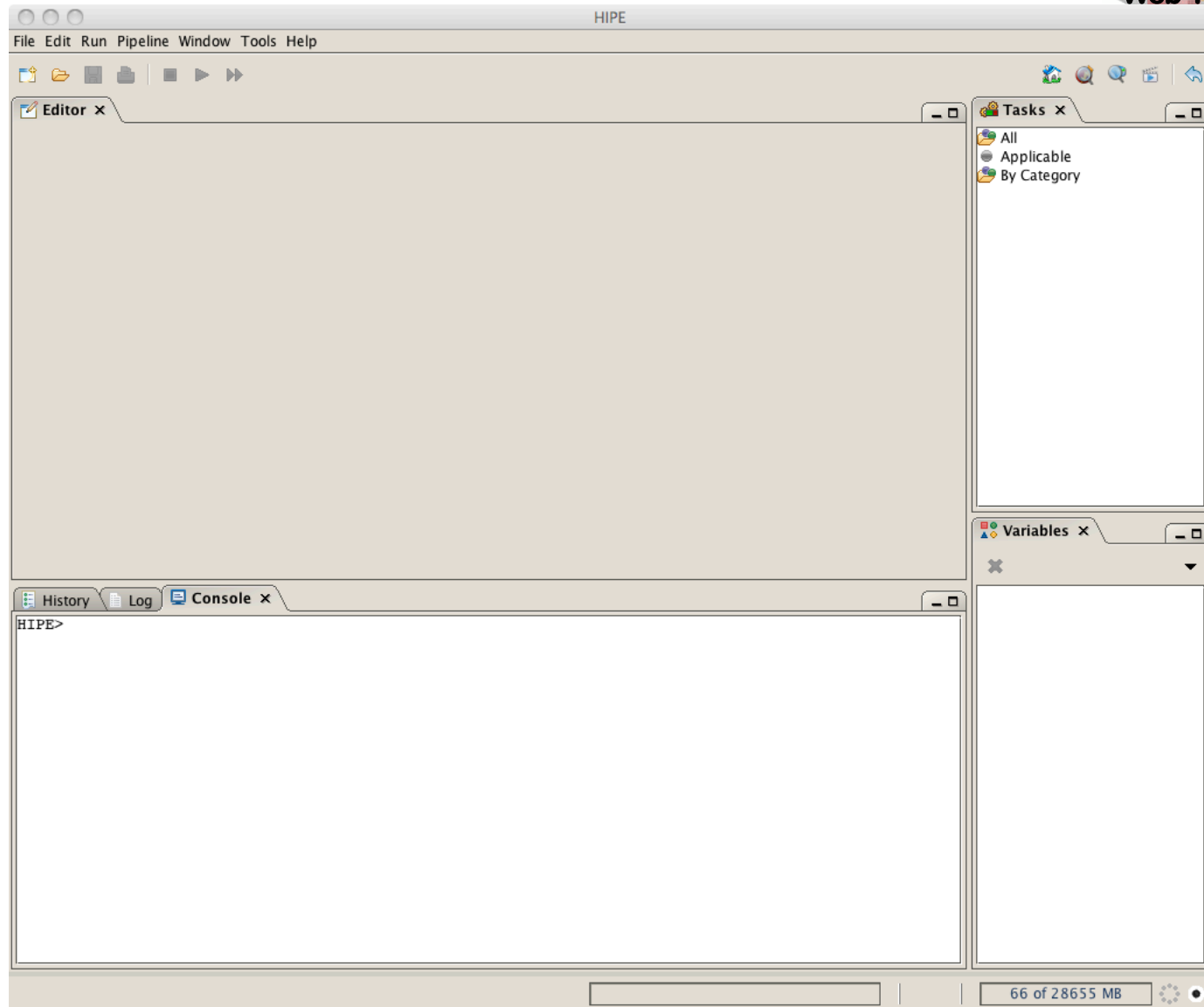


# Step 1

## Start HIPE



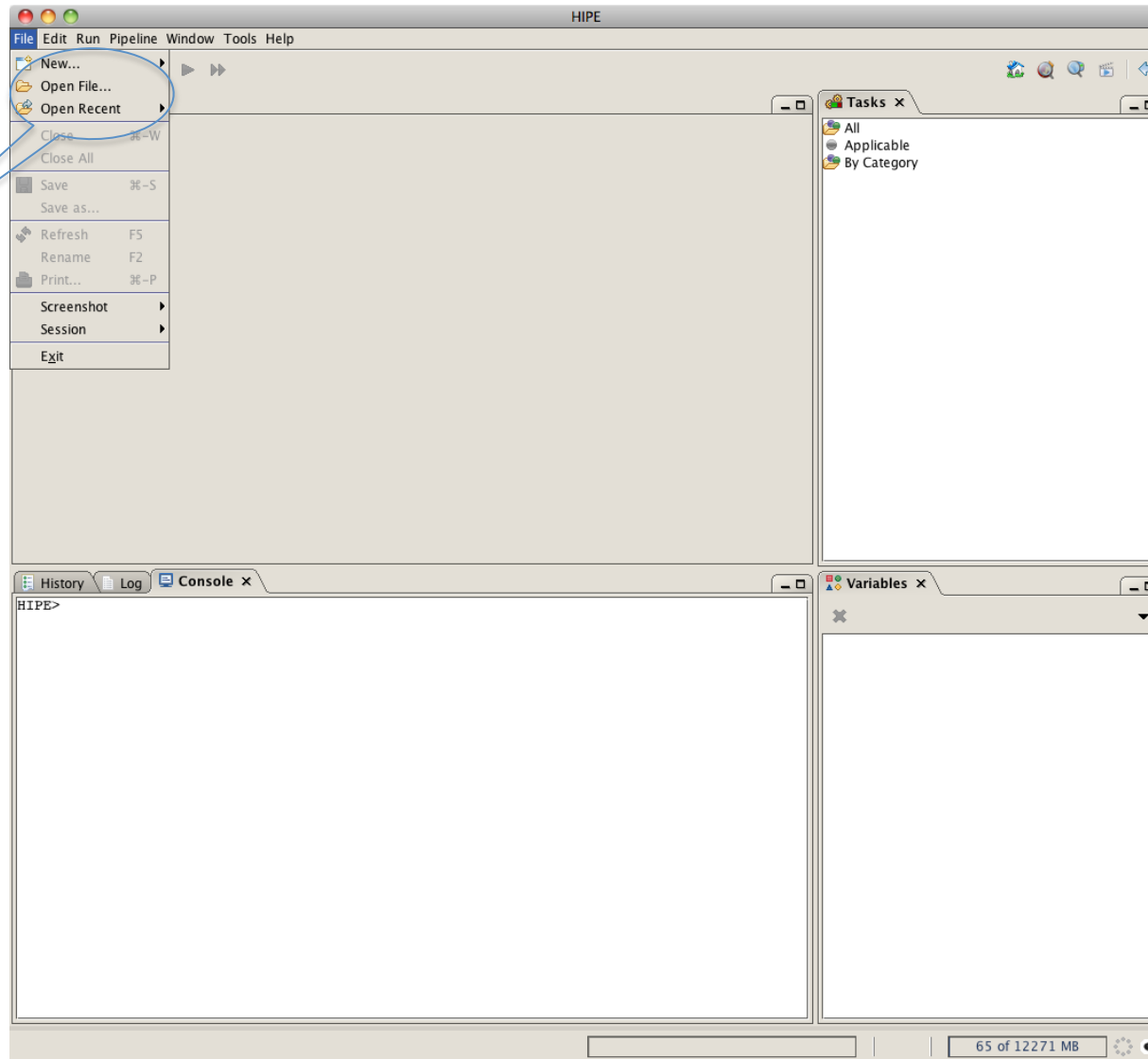
HIPE on startup



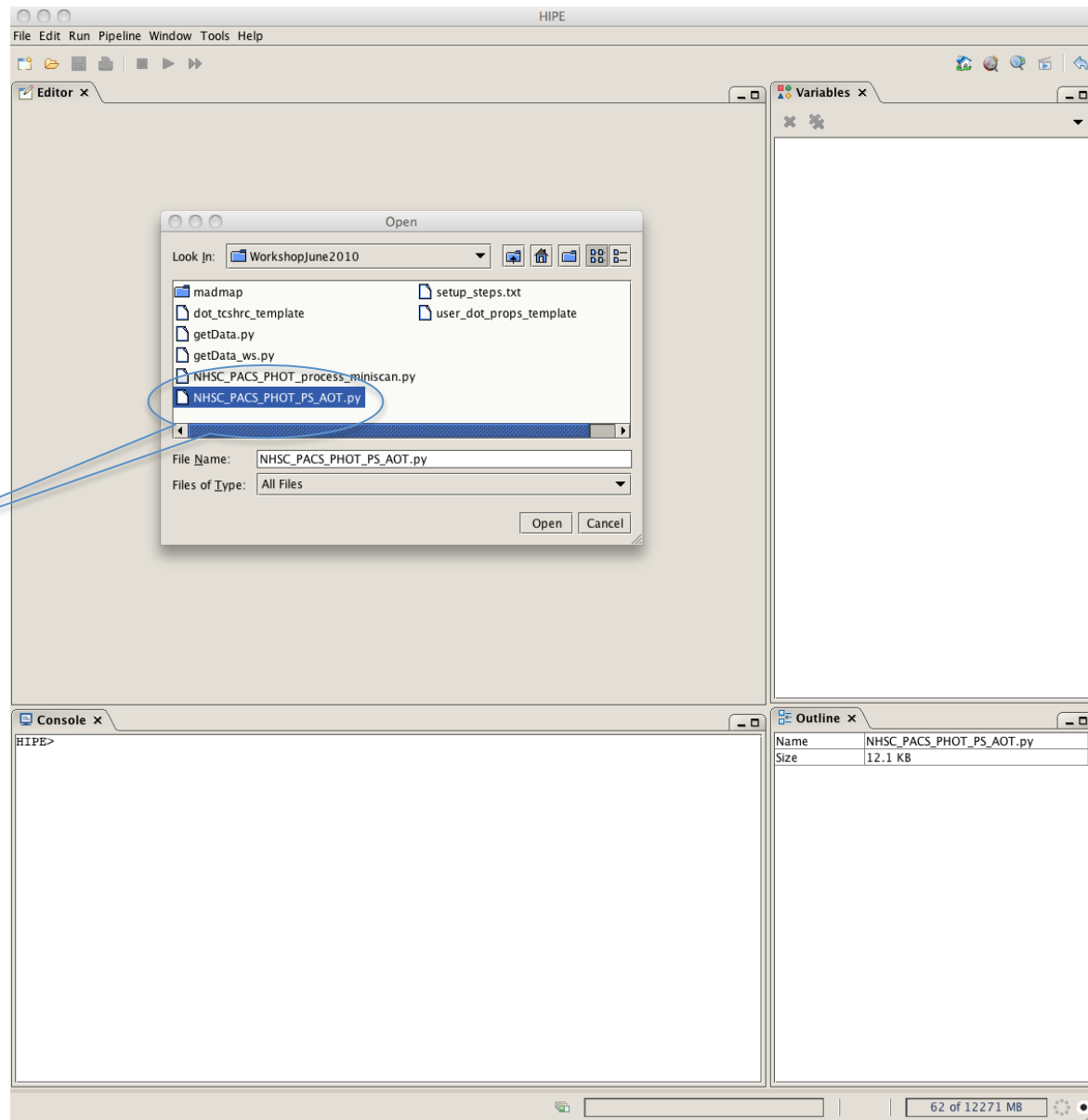


# Step 2

## Select script file



From the  
File Menu  
Select  
Open File...



Select a file from the file dialog and click **Open** button.



```
HIPE - /Users/xliu/home/scripts/WorkshopJune2010/NHSC_PACS_PHOT_PS_AOT.py
File Edit Run Pipeline Window Tools Help
Editor x
NHSC_PACS...S_AOT.py x
1 #####
2 # PACS PHOT Point Source AOT Demo Script
3 # NASA Herschel Science Center
4 # Data Processing Workshop, 26-27 January 2010
5 # Prepared by Nicolas BILLOT (NHSC)
6 #####
7
8 from herschel.pacs.signal.context import PacsContext
9 from java.util import Date
10 import os
11 #
12 print ''
13 print 'Start Processing the Data at : ', Date()
14 print ''
15 #
16 # Define a couple of parameters to display verbose or graphs
17 # (recommended for the first execution of the script)
18 verbose = 'yes'
19 visual_check = 'yes'
20 #
21 # change directory path below to your local directory.
22 # enter it manually if necessary
23 #dir = os.getcwd() # get the current working directory
24 dir = '/home/nhscv[n]/work/' # where n is the number that was assigned to you
25 #
26 # Use the demo dataset OR edit the following parameters:
27 # Object HD 148387 # Demo Point Source Observation
28 #obsid=1342186141
29 #pooldir='/herscheldata/sc/workshop/PACS_PHOT/DATA/' # if using the demo data
30 # Your SDP data : write your OBSID and replace [n] with the number you have been given
31 obsid= Write your OBSID Here
32 myPoolName = obsid.toString()
33 pooldir='/home/nhscv[n]/pools/' # if using your own SDP data
34 #
35 # select blue or red camera
36 # Blue channel gives either 70 or 100 micron data
37 # Red channel gives 160 micron data independently of the blue channel filter
38 camera = 'blue'
```

The editor window shows that the file has been loaded into HIPE.





# Step 3

## Execute custom script



HIPE - /Users/xliu/home/scripts/WorkshopJune2010/NHSC\_PACS\_PHOT\_PS\_AOT.py

```
#####  
# PACS PHOT Point Source AOT Demo Script  
# NASA Herschel Science Center  
# Data Processing Workshop, 26-27 January 2010  
# Prepared by Nicolas BILLOT (NHSC)  
#####  
from herchel.pacs.signal.context import PacsContext  
from java.util import Date  
import os  
print ''  
print 'Start Processing the Data at : ', Date()  
print ''  
#  
# Define a couple of parameters to display verbose or graphs  
# (recommended for the first execution of the script)  
verbose = 'yes'  
visual_check = 'yes'  
#  
# change directory path below to your local directory.  
# enter it manually if necessary  
#dir = os.getcwd() # get the current working directory  
dir = '/home/nhscv[n]/work/' # where n is the number that was assigned to you  
#  
# Use the demo dataset OR edit the following parameters:  
# Object HD 148387 # Demo Point Source Observation  
#obsid=1342186141  
#pooldir='/hercheldata/sc/workshop/PACS_PHOT/DATA/' # if using the demo data  
# Your SDP data : write your OBSID and replace [n] with the number you have been given  
obsid= Write your OBSID Here  
myPoolName = obsid.toString()  
pooldir='/home/nhscv[n]/pools/' # if using your own SDP data  
#  
# select blue or red camera  
# Blue channel gives either 70 or 100 micron data  
# Red channel gives 160 micron data independently of the blue channel filter  
camera = 'blue'
```

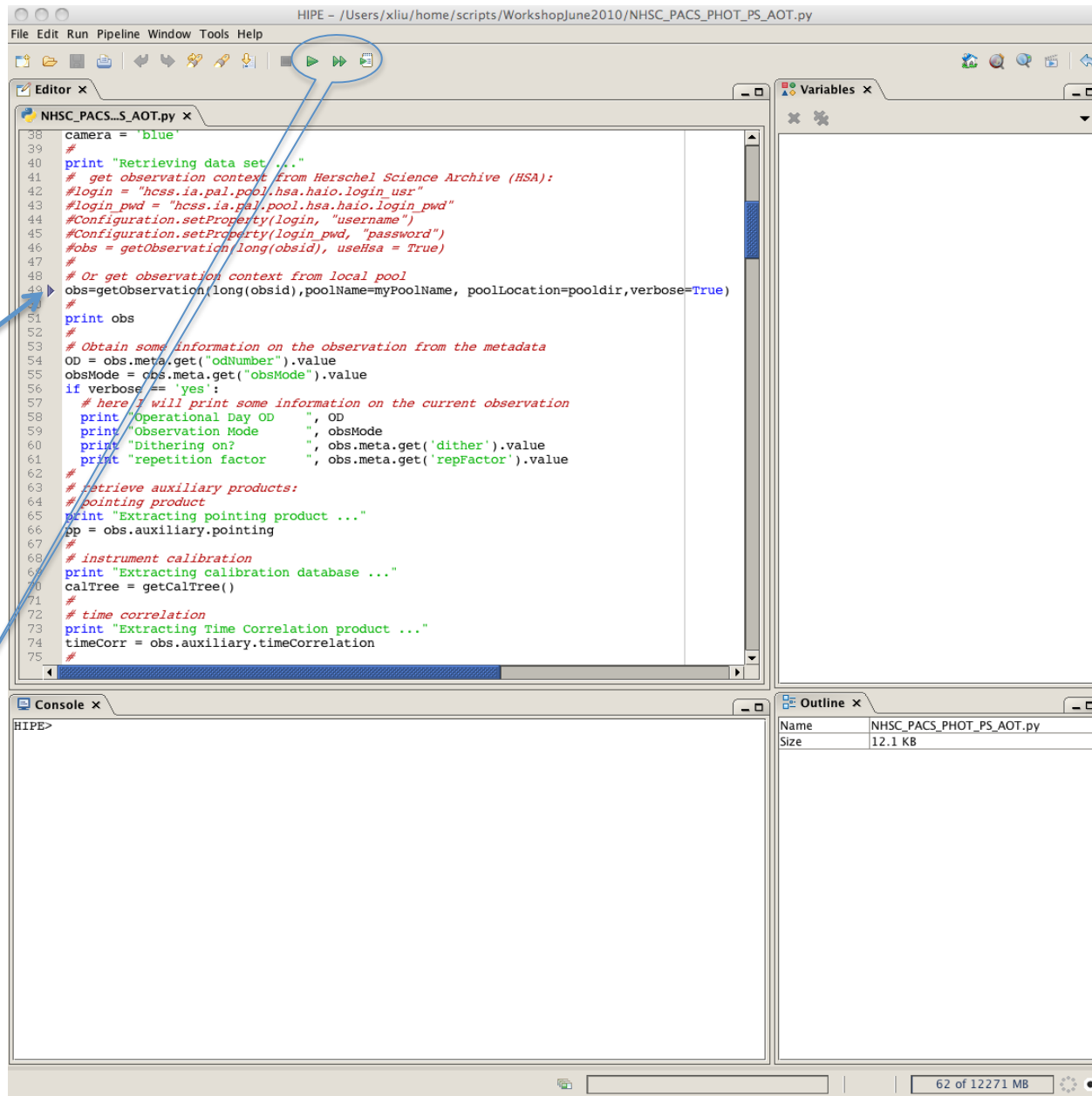
Name	Size
NHSC_PACS_PHOT_PS_AOT.py	12.1 KB

HIPE>

62 of 12271 MB

Click the double arrow to execute the script all at once.

These arrows turn green only if the cursor is in the Editor pane.



The screenshot shows a Python IDE window titled "HIPE - /Users/xliu/home/scripts/WorkshopJune2010/NHSC\_PACS\_PHOT\_PS\_AOT.py". The main editor displays a Python script with various comments and code lines. A blue circle highlights the execution controls (play, stop, refresh) at the top. A blue arrow points from a text box to a single right-pointing arrow on line 49 of the script, which is the line being executed. The script content is as follows:

```
38 camera = 'blue'
39
40 # print "Retrieving data set..."
41 # get observation context from Herschel Science Archive (HSA):
42 #login = "hcass.ia.pal.pool.hsa.haio.login_usr"
43 #login_pwd = "hcass.ia.pal.pool.hsa.haio.login_pwd"
44 #Configuration.setProperty(login, "username")
45 #Configuration.setProperty(login_pwd, "password")
46 #obs = getObservation(long(obsid), useHsa = True)
47 #
48 # Or get observation context from local pool
49 obs=getObservation(long(obsid), poolName=myPoolName, poolLocation=pooldir, verbose=True)
50 #
51 print obs
52 #
53 # Obtain some information on the observation from the metadata
54 OD = obs.meta.get("odNumber").value
55 obsMode = obs.meta.get("obsMode").value
56 if verbose == 'yes':
57     # here I will print some information on the current observation
58     print "Operational Day OD", OD
59     print "Observation Mode", obsMode
60     print "Dithering on?", obs.meta.get('dither').value
61     print "repetition factor", obs.meta.get('repFactor').value
62 #
63 # retrieve auxiliary products:
64 # pointing product
65 print "Extracting pointing product ..."
66 pp = obs.auxiliary.pointing
67 #
68 # instrument calibration
69 print "Extracting calibration database ..."
70 calTree = getCalTree()
71 #
72 # time correlation
73 print "Extracting Time Correlation product ..."
74 timeCorr = obs.auxiliary.timeCorrelation
75
```

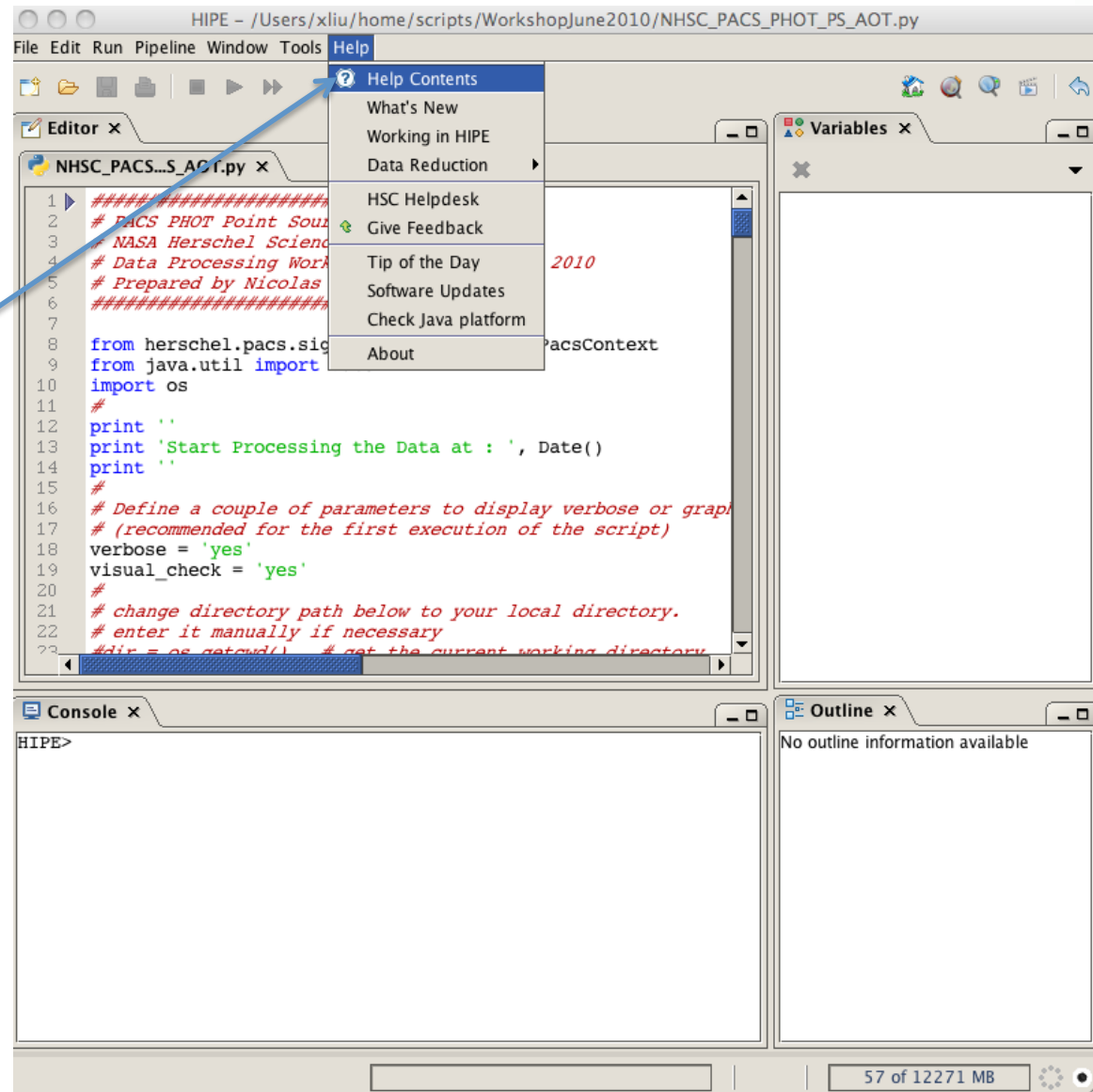
The single arrow executes the script one line at a time.

The screenshot shows the HIPE software interface. The main window is titled "NHSC\_PACS\_S\_AOT.py" and contains Python code. A blue circle highlights the Run button in the toolbar. A blue arrow points from a text box to a highlighted block of code in the editor. The console shows the command "HIPE>" and the outline panel shows the file name "NHSC\_PACS\_PHOT\_PS\_AOT.py" and its size "12.1 KB".

```
53 # Obtain some information on the observation from the metadata
54 OD = obs.meta.get("odNumber").value
55 obsMode = obs.meta.get("obsMode").value
56 if verbose == 'yes':
57     # here I will print some information on the current observation
58     print "Operational Day OD      ", OD
59     print "Observation Mode      ", obsMode
60     print "Dithering on?          ", obs.meta.get('dither').value
61     print "repetition factor      ", obs.meta.get('repFactor').value
62
63 # retrieve auxiliary products:
64 # pointing product
65 print "Extracting pointing product ..."
66 pp = obs.auxiliary.pointing
67
68 # instrument calibration
69 print "Extracting calibration database ..."
70 calTree = getCalTree()
71
72 # time correlation
73 print "Extracting Time Correlation product ..."
74 timeCorr = obs.auxiliary.timeCorrelation
75
76 # instrument housekeeping
77 print "Extracting photometric housekeeping product ..."
78 photHK = obs.level0.refs["HPPHK"].product.refs[0].product["HPPHK"]
79
80 #####
81 # get the Level 0 data cube (frames) and access either the blue or red channel frames
82 #####
83 print "Starting processing from level 0 ..."
84 if camera == 'blue':
85     frames = obs.level0.refs["HPPAVGB"].product.refs[0].product
86 else:
87     frames = obs.level0.refs["HPPAVGR"].product.refs[0].product
88 # data can also be sliced per ABBA nod cycles (see HSC demo script from Markus Neilbock)
89 #
90 # what is the size of the cube I want to process ?
```

The single arrow can also execute a block of highlighted script.

From Help menu select **Help Contents** to bring up the help page.





Help window in a browser