CUPID

"Customizable User Pipeline for IRS Data"

Abstract: The CUPID package will allow users to run the Spitzer IRS pipelines on their local machines. When run in the default mode, these pipelines will produce data files with pixel and table values identical to those produced by the operational pipelines. There will be, however, a number of options available for changing pipeline behavior. Users will have options for changing the value of any module processing parameter ("SET"). And users will also have options for controlling the flow of module processing ("START", "STOP", "SKIP", "DO.AFTER").   The package will include a new executive script (csh) and a few new pipeline "wrapper" scripts (Perl). It will also include existing operational executable modules and calibration files. Target platforms are Solaris, Mac, and Linux.
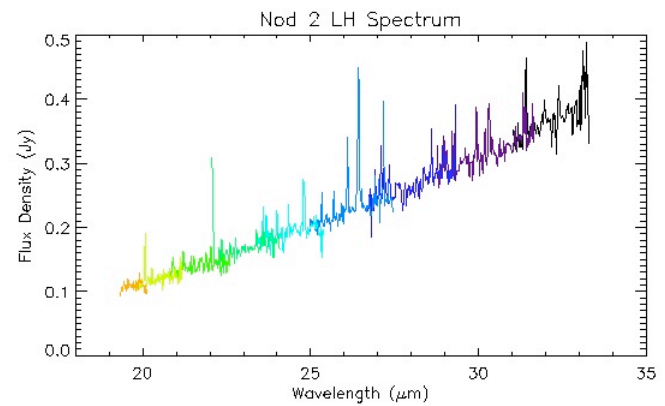
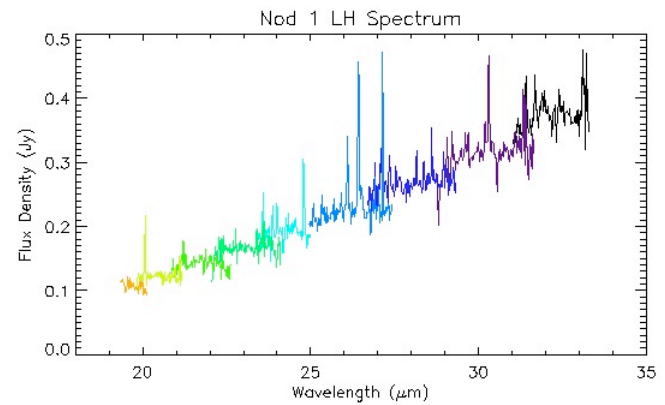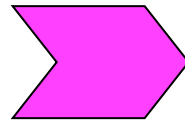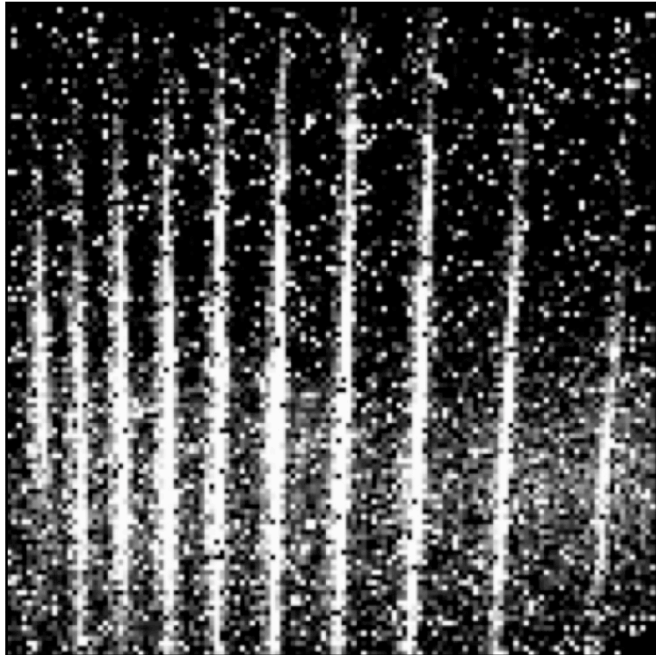# What it is CUPID ?

(not this one)

CUPID Is …

A software package that allows users to run all of the Spitzer IRS science pipelines on their own machine.

("IRS" is "InfraRed Spectrograph", an instrument that produces echelle spectra.)

The Data

Requirements

- Products have exactly the same data values as the Spitzer operational pipeline (on Solaris)
- Products will work with SPICE and CUBISM tools
- Products have same directory structure and file names as in the Spitzer Archive
- Software is easy to learn and use
- The package will run on MacOs, Linux, and Solaris
- Users can use their own calibration files
- Users can change values of processing constants
- Users can control flow of pipeline steps

How is CUPID Used?

Definition of "BCD"

- A BCD is a "Basic Calibrated Dataset"
  - Calibration files have been applied
  - Instrument characteristics have been removed
  - Pixels converted from 3D integer image to 2D floating point image
  - Header keywords have been updated and added

The Pipelines

1. BCD - Create BCD images
2. COADD - Create coadd images
3. BKSUB - Create background subtracted images
4. BCDX - Spectral extraction on BCD images
5. COADDX - Spectral extraction on coadd images
6. BKSUBX --- Spectral extraction on BKSUB images
7. CUTOUT --- Create blue and red cutout images
8. DARK   --- Create composite dark cube image

To Run CUPID

- To run CUPID, enter "cupid" followed by the following:

  - Processing Directories [at least one]
  - Pipeline selection [defaults to first five]
  - Override(s) of parameters or files [optional]
  - Flow control option(s) [optional]

Selecting Processing Directories

- Include directory(s) to be processed on the command line.

  E.g. **/data/r1234/ch0 /data/r1234/ch2**

  /data/r1234/ch*

  /data/r*/ch*

Selecting Pipelines to Run

- User may give explicit list
  e.g. **PIPE/BCD.COADD.COADDX**

- User may select first N pipelines
  e.g. **PIPE/6** to run the first 6 pipelines

- The default is to run the first 5 pipelines

Selecting Parameter Overrides

- Format: SET.module-flag=value

  e.g. **SET.flatap-f=/joe/myflat.fits**

  SET.snest-gain=5.6

Selecting Flow Control Options

- **START**.*STEPNAME*
- **STOP**.*STEPNAME*
- **SKIP**.*STEPNAME*
- **DO**.*STEPA*.**AFTER**.*STEPB*

Examples:

- **STOP.DARKBASE**
- **SKIP.STRAYCROSS**

Sample Command

- **cupid /data/r12345/* PIPE/BCD.BCDX
  SET.flatap-f=/joe/myflat.fits
  SET.snest-gain=5.6 SKIP.STRAYCROSS**

This command will run the BCD and BCDX
pipelines on all the directories in /data/
r12345. It will use the specified "flat" file and it
will use a gain value of 5.6 for the SNEST
step. Also, the STRAYCROSS step will be
skipped.

What's inside CUPID?

The CUPID Directory

(All of CUPID is in this directory.)

- **cupid** - main executable script (C-shell)
- **wrappers/** - "wrapper" scripts (Perl)
- **bin/ lib/ include/** - executable modules
- **cal/ cdf/** - calibration and control files
- **HELP** - help file (plain text)
- **VERSION** - version number and history

One Step in a "Wrapper" Script

```
#STEP SLOPE_FINDER compute slope estimates
my $integrationtime = $kwd_RAMPTIME + $kwd_GRPTIME;
run_module("SLOPE_FINDER slope_finder
IN  -i      <FLUX>                       # Input FLUX
IN  -iu     <UNC>                        # Input UNC
IN  -id     dmask.fits                   # Input DMASK
CDF -n      slope_finder.nl              # namelist file
CAL -ip     pmask.fits                   # PMASK CAL ile
V   -t      $integrationtime             # int time
OUT -o_flux slope_finder.fits <FLUX>     # Output FLUX
OUT -o_unc  slope_finder_unc.fits <UNC>  # Output UNC
OUT -ob     bmask.fits                   # Output BMASK
");
```

Files from the Operational Pipeline

	N files	MB
Executable Modules	32	10
Calibration Files	1500	2000
Control Files	125	1

Files Created for CUPID

	N files	N lines	KB
"cupid" Script	1	600	20
Wrappers	8	1600	60
HELP	1	2800	100
TOTAL	10	5000	180

EXAMPLE

Get Data From Spitzer Archive

"Processing Directory" Before Run

```
/data/r20136960/ch0/
  raw/
      SPITZER_S0_20136960_0002_0000_2_raw.fits
      SPITZER_S0_20136960_0002_0001_2_raw.fits
      SPITZER_S0_20136960_0003_0000_2_raw.fits
      SPITZER_S0_20136960_0003_0001_2_raw.fits
```

The 'cupid' Command

```
cupid /data/r20136960/ch0 SKIP.STRAYCROSS
```

Initial Display

```
******* S18.7-CUPID-beta-7b *******

- Pipelines:  BCD   COADD   BKSUB   BKSUBX COADDX

- processing directories: /data/r20136960/ch0

- BCD options:

            SKIP.STRAYCROSS

- X options:
```

Display for BCD Pipeline

```
2/4 /data/r20136960/ch0/bcd/SPITZER_S0_20136960_0002_0001_BCD.log
END_STEP  1. CVTI2R4       status=  0
END_STEP  2. ADDKWDS       status=  0
END_STEP  3. SNEST_IRS     status=  0
END_STEP  4. DNTOFLUX      status=  0
END_STEP  5. IMAGEST       status=  0
END_STEP  6. DROOPOP       status=  0
END_STEP  7. ROWDROOP      status=  0
END_STEP  8. DARKBASE      status=  0
END_STEP  9. CUBESUB       status=  0
END_STEP 10. LINEARIZ      status=  0
END_STEP 11. RADHIT        status=  0
END_STEP 12. DARKDRIFT     status=  0
END_STEP 13. SLOPE_FINDER  status=  0
END_STEP 14. DROOPRES      status=  0
END_STEP 15. FLATAP        status=  0
END_STEP 16. FPGEN         status=  0
0 OK  BCD SPITZER_S0_20136960_0002_0001_   2009/05/07 12:55:12
```

Display of Status Summary

```
BCD /data/r20136960/ch0/bcd
 0 OK BCD SPITZER_S0_20136960_0002_0000_  2009/05/07 12:55:00 (15 sec.)
 0 OK BCD SPITZER_S0_20136960_0002_0001_  2009/05/07 12:55:12 (11 sec.)
 0 OK BCD SPITZER_S0_20136960_0003_0000_  2009/05/07 12:55:24 (11 sec.)
 0 OK BCD SPITZER_S0_20136960_0003_0001_  2009/05/07 12:55:36 (11 sec.)
COADD /data/r20136960/ch0/pbcd
  0 OK  COADD SPITZER_S0_20136960_0002_  2009/05/07 12:55:38  (1 sec.)
  0 OK  COADD SPITZER_S0_20136960_0003_  2009/05/07 12:55:39  (1 sec.)
BKSUB /data/r20136960/ch0/pbcd
  0 OK  BKSUB SPITZER_S0_20136960_0002_  2009/05/07 12:55:39  (0 sec.)
  0 OK  BKSUB SPITZER_S0_20136960_0003_  2009/05/07 12:55:40  (0 sec.)
BKSUBX /data/r20136960/ch0/pbcd
  0 OK  BKSUBX SPITZER_S0_20136960_0002_  2009/05/07 12:55:44  (4 sec.)
  0 OK  BKSUBX SPITZER_S0_20136960_0003_  2009/05/07 12:55:46  (2 sec.)
COADDX /data/r20136960/ch0/pbcd
  0 OK  COADDX SPITZER_S0_20136960_0002_  2009/05/07 12:55:49  (2 sec.)
  0 OK  COADDX SPITZER_S0_20136960_0003_  2009/05/07 12:55:52  (2 sec.)
```

"Processing Directory" After Run

```
/data/r20136960/ch0/
    raw/    (still contains original 4 raw files)
    bcd/
        SPITZER_S0_20136960_0002_0000_bcd.fits
        .... (168 files altogether)
    pbcd/
        SPITZER_S0_20136960_0002_bksub.fits
        .... (54 files altogether)
    cal/
        b0_flatfield.fits
        .... (17 files altogether)
    cdf/
        b0_basecal.nl
        .... (20 files altogether)
```
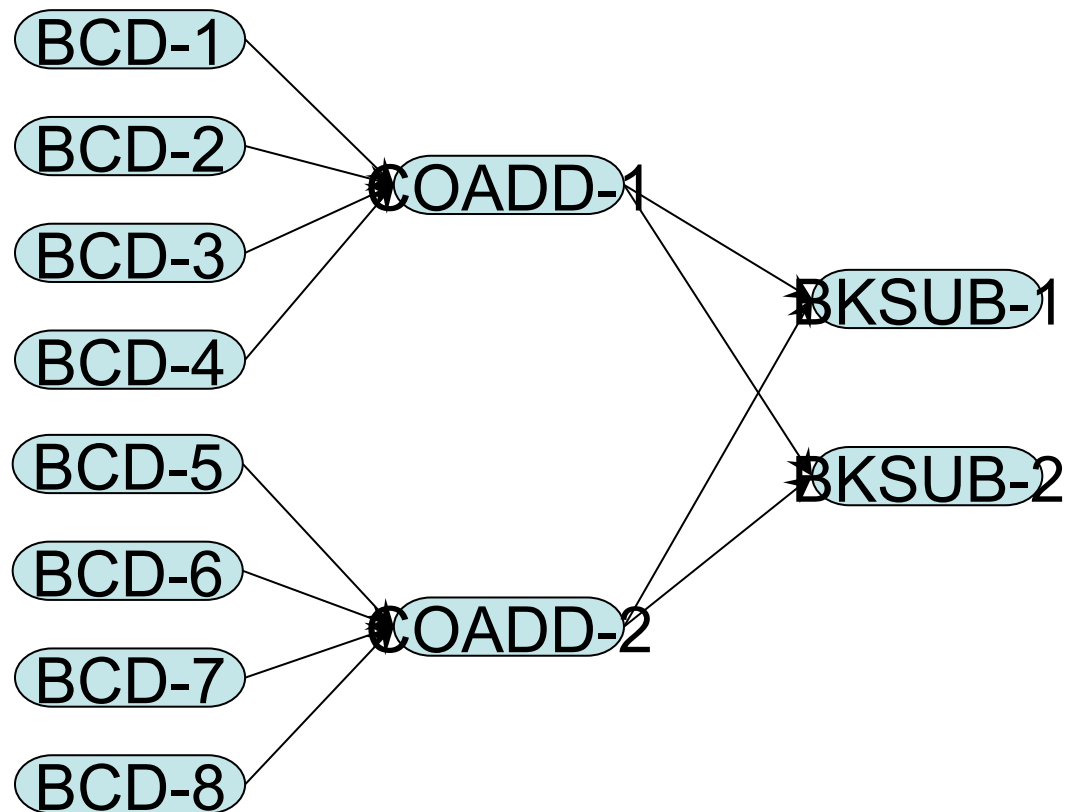
BACKUP SLIDES

How Images Are Combined

BCD-1
BCD-2
BCD-3
BCD-4
BCD-5
BCD-6
BCD-7
BCD-8

COADD-1
COADD-2

BKSUB-1
BKSUB-2

Finding the CUPID Directory

setenv CUPID (to directory containing CUPID package)

set COM = $0 # actual command used to call CUPID

if (-d ./wrappers)   setenv CUPID $cwd   # just name of script

if (-d $COM:h/wrappers)   setenv CUPID $COM:h # relative path of script

if (-d $cwd/$COM:h/wrappers) setenv CUPID $cwd/$COM:h # full path of script

Finding the Perl Directory

```
# set $perl to command to call Perl
if ($?PERL_PATH) then
  set perl = "$PERL_PATH/perl -w"
else
  foreach try_dir (/usr/bin /usr/local/bin /usr/sbin /usr/lib $path)
    if (-x $try_dir/perl) then
      set perl = "$try_dir/perl -w"
      break
    endif
  end
endif
```