

# **Wide-field Infrared Survey Explorer (WISE)**

## **COG Subsystem Design Document**

**Version 1.0**

**23 December 2008**

**Prepared by: Tracey L. Evans**



**Infrared Processing and Analysis Center  
California Institute of Technology**

**WSDC D-D020**

**Approved By:**

---

Roc Cutri, WISE Science Data Center Manager

---

Tim Conrow, WISE Science Data Center System Architect

---

Tracey Evans, WISE Science Data Center COG Cognizant Engineer

## **Revision History**

<b>Date</b>	<b>Version</b>	<b>Author</b>	<b>Description</b>
23 December 2008	1.0	Tracey L. Evans	Initial Draft

## Table of Contents

<b>1</b>	<b>INTRODUCTION .....</b>	<b>5</b>
<b>1.1</b>	<b>Subsystem Overview.....</b>	<b>5</b>
1.1.1	Requirements .....	5
1.1.2	Current Description .....	6
<b>1.2</b>	<b>Applicable Documents.....</b>	<b>6</b>
<b>1.3</b>	<b>Acronyms.....</b>	<b>6</b>
<b>2</b>	<b>INPUT .....</b>	<b>7</b>
<b>2.1</b>	<b>Control Input .....</b>	<b>7</b>
2.1.1	<i>WCOG</i> Command-Line Parameters .....	7
2.1.2	<i>COG</i> Command-Line Parameters.....	10
<b>2.2</b>	<b>Source Data Input.....</b>	<b>11</b>
<b>3</b>	<b>PROCESSING .....</b>	<b>12</b>
<b>3.1</b>	<b><i>WCOG</i> Processing .....</b>	<b>12</b>
3.1.1	<i>WCOG</i> Initialization .....	12
3.1.2	<i>WCOG</i> Post-processing .....	12
<b>3.2</b>	<b><i>COG</i> Processing.....</b>	<b>12</b>
3.2.1	<i>COG</i> Initialization .....	12
3.2.2	Source Data Input .....	12
3.2.3	Curve of Growth Calculation .....	13
3.2.4	Aperture Correction Calculation .....	13
<b>4</b>	<b>OUTPUT.....</b>	<b>14</b>
<b>4.1</b>	<b><i>COG</i> Output .....</b>	<b>14</b>
4.1.1	Output Curve of Growth Source Table.....	14
4.1.2	Output Median <i>COG</i> and Aperture Corrections Table .....	15
<b>4.2</b>	<b><i>WCOG</i> Output.....</b>	<b>16</b>
<b>5</b>	<b>TESTING AND PARAMETER TUNING .....</b>	<b>16</b>

# 1 INTRODUCTION

## 1.1 Subsystem Overview

This document presents the requirements, design, algorithms, and state of implementation of the *COG* (Curve-of-Growth) subsystem of the WSDC data processing system. *COG* consists of two programs, *COG*, which calculates a differential aperture curve of growth (COG) and aperture corrections for each PSF bin on the array(s), and *WCOG*, the wrapper for *COG* which also creates plots of its output data. These programs are run independent of the frame and scan pipelines, but can be part of the scan QA.

Herein, the initials “COG” in the regular typeface refer to a curve of growth; in the “smallcaps” typeface, *COG* and *WCOG* refer to the program names.

### 1.1.1 Requirements

The *COG* subsystem is required to determine the differential curve of growth of the aperture magnitudes for each PSF on the array(s), using many sources from multiple frames as input. It then must use the differential curve of growth to calculate the median aperture correction for each aperture to the “optimal” aperture in each PSF bin, and output those values to tables that can be read by other programs (notably WPHOT).

The Level 4 requirements supported by this subsystem are as follows:

L4WSDC-012: Flux measurements in the WISE Source Catalog shall have a SNR of five or more for point sources with fluxes of 0.12, 0.16, 0.65 and 2.6 mJy at 3.3, 4.7, 12 and 23 micrometers, respectively, assuming 8 independent exposures and where the noise flux errors due to zodiacal foreground emission, instrumental effects, source photon statistics, and neighboring sources.

L4WSDC-013: The root mean square error in relative photometric accuracy in the WISE Source Catalog shall be better than 7% in each band for unsaturated point sources with SNR>100, where the noise flux errors due to zodiacal foreground emission, instrumental effects, source photon statistics, and neighboring sources. This requirement shall not apply to sources that superimposed on an identified artifact.

L4WSDC-015: The WISE Source Catalog shall contain the measured in-band fluxes or flux upper-limits in the four WISE bands for objects detected in at least one band in the WISE Atlas Images.

### 1.1.2 Current Description

The *COG* program reads a set of frames' four-band source data output from WHPOT (files \* - mdex-1b.tbl), and calculates a differential aperture curve-of-growth (COG) for each PSF bin on the array(s). (Each array is broken up into a grid of  $n_x$  by  $n_y$  areas, each of which has a different PSF. Each of these areas is called a “PSF bin”.) The differential COG value  $dm$  for each source at aperture  $i$  is:

$$dm_i = m_i - m_{i+1}$$

where  $m$  is the aperture magnitude of the source.

For each PSF bin, *COG* then finds the median  $dm$  value from all sources in each aperture and determines curve of growth convergence or divergence, and calculates the aperture corrections for each aperture to the convergence/divergence aperture, or “optimal” aperture. The median differential curve of growth values and aperture corrections are calculated using an iterative median calculation with n-RMS rejection.

It outputs the COG data for each source into separate files for each PSF bin. The median curve of growth and aperture corrections are output into other files for each PSF bin.

## 1.2 **Applicable Documents**

This subsystem conforms to the specifications in the following project documents:

- WISE Science Data Center Functional Requirements Document, WSDC D-R001
- WISE Science Data System Functional Design, WSDC D-D001
- Software Management Plan, WSDC D-M003
- [WPHOT Output Photometry Table SIS](#), WSDC D-I107
- [COG Output Curve of Growth Source Table SIS](#), WSDC D-I127
- [COG Output Median COG and Aperture Corrections Table SIS](#), WSDC D-I128

## 1.3 **Acronyms**

COG	Curve of Growth
FRD	Functional Requirements Document
IPAC	Infrared Processing and Analysis Center
L4WSDC	Level 4 requirements for the WISE Science Data Center
PSF	Point-Spread Function used for source extraction
RMS	Root-Mean-Square value
SDS	Subsystem Design Specification

SIS	Software Interface Specification
SNR	Signal-to-Noise ratio
W1	WISE wavelength channel 1, 3.3 microns
W2	WISE wavelength channel 2, 4.7 microns
W3	WISE wavelength channel 3, 12 microns
W4	WISE wavelength channel 4, 24 microns
WISE	Wide-field Infrared Survey Explorer
WSDC	WISE Science Data Center
WSDS	WISE Science Data System

## 2 INPUT

### 2.1 Control Input

The Perl wrapper *WCOG* accepts the user's input parameters and uses them to construct a list of source files to read as input, then passes the input parameters along to the *COG* program. The control input parameters are thus analogous between *WCOG* and *COG*, but are given in slightly different formats.

#### 2.1.1 WCOG Command-Line Parameters

Typing “WCOG -help2” will provide program help, listing all of the possible input control parameters for *WCOG*. Most of this output is given below, with some more comments indicating how and where they are passed along to *COG*. The first large set of parameters in the help, up to (but not including) “bands,bnds”, is part of the “basic” parameter set for the WISE pipeline, and I've only included here the ones that are used by *WCOG*; for more information on this set of parameters see their documentation or source code. The *COG* program's use of the input parameters it receives is described in the next section.

Command => WCOG

Run COG (curve of growth) and aperture correction generator.

in_base,inbase,inb	: string	= cog
Input file(s) base name		
out_base,outbase,outb	: string	= %in_base%
Output file(s) base name		
out_dir,outdir,outd	: file	= .
Input directory		

`work_dir,workd` : file = %out\_dir%/work  
Directory for volatile output

*WCOG* uses the `work_dir` and `in_base` parameters to set the name of the list file, which it fills with the list of input data files created from the `frame_dirs` parameter below. The name of this list file is then passed to *COG*. *WCOG* uses the `out_base` and `out_dir` parameters to set up the `out_base` value passed to *COG*; this provides a full pathname.

`verbose,v` : switch  
Increase verbosity of informatory output

`log_file,logfile,log` : string  
Write (or inherit) standard log file  
'1' => do the usual thing.  
'+' prefix means append.  
'=' prefix means overwrite (the default).  
Otherwise assume a literal file name.

`test` : switch  
Do not actually run but display what would be run

`debug,dbg` : string private = 0  
Add targetted debug output

Setting the parameter `debug` to '1' or similar will turn on the debugging option for *WCOG* and *COG*. This sends a whole bunch of output to `STDOUT` and the file "`cog.dbug`" in the runtime directory. Most people will never want to use this parameter.

`bands,bnds` : list of 1 to 4 integer (1 to 4) = (1,2,3,4)  
Bands with `sdex` and image data to process

The `bands` parameter is the list of bands to process, with commas but no spaces between the numbers; for example, "1,3" or "1,2,3,4".

`frame_dirs,frdirs` : list of file (glob)  
Frame directories (or fileglobs) to process

The `frame_dirs` parameter is a list (comma-separated) of input frame dirs (containing `[basename]-mdex-lb.tbl` files) or fileglobs of frame dirs, *e. g.* '1d\*/fr/\*', to be used as input. *COG* can handle up to 1000 files per run. (The shell command line input buffer can only handle less than 100 in a list, though.)

`n_psf_xy,npsf` : list of integer = (5,5)  
Number of PSF bins in PSF grid in x and y

The list parameter `n_psf_xy` is separated into two parameters and passed to *COG* as `NpsfX` and `NpsfY`.

`min_srcs_psf,minsrcs,nmin` : integer = 20  
Minimum number of 'good' sources required in PSF bin to be able to calculate COG

The `min_srcs_psf` parameter is passed to *COG* as the `psfsrcs_min` parameter.

`n_rms_reject,nrej,nrms` : double = 3  
Multiple of RMS used for RMS-rejection when calculating iterative median

`max_loss_frac,loss_max,loss_frac` : double = 0.3  
Maximum fractional loss of sources allowed in RMS-rejection when calculating iterative median

`max_iter,iter_max,maxit` : integer = 30  
Maximum number of iterations allowed when calculating median

The `n_rms_reject` parameter is passed to *COG* as the `n_rej` parameter, the `max_loss_frac` parameter as the `loss_max` parameter, and the `max_iter` parameter as the `iter_max` parameter.

`delta_median,del_med,dmed` : double = 0.0005  
Delta median value (`med_i+1 - med_i`) used to determine convergence in iterative median calculation

The `delta_median` parameter is passed to *COG* as the `del_med` parameter.

`delta_converge,del_cnvg,cnvg` : double = 0.01  
Delta median value (`apm_i+1 - apm_i`) used to determine convergence in aperture correction calculation

`delta_diverge,del_div,div` : double = -0.0005  
Delta median value (`apm_i+1 - apm_i`) used to determine divergence in aperture correction calculation

The `delta_converge` and `delta_diverge` parameters are passed to *COG* as the `del_cnvg` and `del_div` parameters, respectively.

`snr_thresholds,snrthresh,snr` : list of double = (10,10000)  
SNR min and max thresholds to determine if source is a 'good' source

`max_latents_frac,max_frl,maxf` : double = 0.3  
Maximum latent fraction of pixels allowed if source is a 'good' source

`flag_ok_values,flag_ok,flg` : list of integer = (2,2,2,2)  
For each band, single flag value other than 0 allowed if source is a 'good' source

The `snr_thresholds` and `flag_ok_values` list parameters each separated and passed to *COG* as `snr_min` and `snr_max`, and the four `flg_ok_w[1234]` parameters. The `max_latents_frac` parameter is passed as the `frl_max` parameter.

### 2.1.2 *COG* Command-Line Parameters

If the wrong number of input parameters is used on the command line, the program will print out usage information:

#### Usage:

```
cog listfile band#s outbase NpsfX NpsfY psfsrcs_min n_rej loss_max iter_max \  
del_med del_cnvg del_div snr_min snr_max frl_max \  
flg_ok_w1 flg_ok_w2 flg_ok_w3 flg_ok_w4 [debug]
```

`listfile`: file containing list of input file names  
`band#s`: string containing band numbers to use (ex: '123')  
`outbase`: output file base name  
`NpsfX`: number of PSFs in X (X x Y grid)  
`NpsfY`: number of PSFs in Y (X x Y grid)  
`psfsrcs_min`: min number of srcs required per PSF to calculate  
`n_rej`: multiple of RMS for RMS-rejection  
`loss_max`: max fractional loss of srcs per PSF allowed  
`iter_max`: max number of iterations allowed  
`del_med`: delta median value used to determine median calculation convergence  
`del_cnvg`: delta median value used to determine ap cor convergence  
`del_div`: delta median value used to determine ap cor divergence  
`snr_min`: min SNR for src to be used  
`snr_max`: max SNR for src to be used  
`frl_max`: max frL value allowed for src to be used  
`flg_ok_w[1234]`: flag value (besides 0) allowed in band 1,2,3,4 for src to be used  
`[debug]`: optional debugging file written

The `listfile` contains a list of frame `[basename]-mdex-1b.tbl` files to be used as input; it can handle up to 1000 files per run.

The `band#s` string contains the list of bands to process, without any spaces or punctuation between the numbers; for example, "124", "13", or "1234".

The `outbase` parameter is the basename for the output files.

`NpsfX` is the total number of PSF bins in the  $X$  direction on the array(s), and `NpsfY` is the total number of PSF bins in the  $Y$  direction. Note that the same values are used for all bands to be processed.

The `psfsrscs_min` parameter is the minimum number of sources required in a PSF bin to be able to calculate the median curve of growth values and aperture corrections.

The `n_rej` parameter gives the multiple  $n$  of the RMS used for the RMS-rejection in the median calculations. The `loss_max` parameter is the maximum fractional loss of sources per PSF bin allowed using the  $n$ -RMS rejection, and the `iter_max` parameter is the maximum number of iterations allowed.

The `del_med` parameter is the delta median value used to determine convergence in the iterative median calculation; if the median changes by less than `del_med` from one iteration to the next, it has converged and the iterations are stopped.

The `del_cnvg` parameter is the curve of growth's  $dm$  value used to determine the “infinite” or “optimal” aperture convergence, and the `del_div` parameter determines COG divergence.

Only “clean” sources with good magnitudes are used to determine curves of growth. The `snr_min` parameter gives the minimum SNR required for the source to be used in a COG, and the `snr_max` parameter gives the maximum SNR allowed. The `frl_max` parameter is the maximum fraction of pixels marked as latents allowed for a “clean” source. Certain aperture measurement flag values are also allowed for a “clean” source; the `flg_ok_w[1234]` parameters indicate what single flag value, besides 0, is allowed in that band. (Most of the time, it will be '0' or '2'.)

The final parameter `debug` is optional; if it appears at the end of the parameters the program runs in “debug” mode and sends a whole bunch of output to the file “`cog.debug`” in the runtime directory. Most people will never want to use this parameter.

## 2.2 Source Data Input

`COG` reads the source extraction data files listed in the file named by the `listfile` input parameter. Each of these files is generated by the upstream module `WPHOT` in the single-scan pipeline, and contains data columns for all four bands. They conform to the [WPHOT Output Photometry Table SIS](#), WSDC D-I107 and have names like `[basename]-mdex-1b.tbl`.

## 3 PROCESSING

### 3.1 *WCOG* Processing

#### 3.1.1 *WCOG* Initialization

The *WCOG* program initializes itself by reading and processing its control input parameters and prints out a program “banner” with version and date-time information. It uses the `work_dir` and `in_base` parameters to set the name of the list file. It then uses the `frame_dirs` parameter to get the directory paths, scan IDs and frame numbers for the source data files’ basenames, then adds “-mdex-1b.tbl” to create the list of input data files for the list file.

It next uses the `out_base` and `out_dir` input parameters to set up the full pathname `out_base` value passed to *COG*. It reformats other parameters as necessary (see Sec. 2.1.1), then creates the command line for *COG* and spawns the command to run it.

#### 3.1.2 *WCOG* Post-processing

After *COG* has completed, *WCOG* checks the output files and creates plots for each PSF bin in each band that was processed and which contained source data.

### 3.2 *COG* Processing

#### 3.2.1 *COG* Initialization

The *COG* program initializes itself by reading and processing its control input parameters. It also prints out a program “banner” with version and date-time information and opens a log and/or debugging file if requested.

#### 3.2.2 Source Data Input

The input routine is then called, which reads the list of data files to be processed from the list file created by *WCOG*. For each data file in turn, *COG* determines the corresponding metadata file, and reads the aperture size information from the metadata file. If this is the first file it has processed, it saves the aperture sizes; for all subsequent files, it checks the aperture sizes against those saved from the first file. If the sizes differ, it skips that file and moves to the next.

It then reads the source data file using the IPAC table file I/O library, and checks each source to see if it is a “clean” source. A source is “clean” in a certain band if it has an SNR in that band between the values of the `snr_min` and `snr_max` input parameters, the fraction of pixels marked as affected by latents is less than the value of the `frl_max` input parameter, all aperture magnitudes in that band are less than 99, and all aperture magnitude flag values in that band are 0 or the value given by the `flg_ok_w[1234]` input parameter for the band.

If the source is “clean”, *COG* calculates the PSF bin in which it is located, using the band, the source’s *X,Y* position on the array and the number of PSF bins in the *X* and *Y* directions on the array, given by the input parameters `NpsfX` and `NpsfY`. It then saves the necessary source data in a structure with the other sources found in that band and PSF bin, and continues to the next input source data file.

### 3.2.3 Curve of Growth Calculation

After reading in all of the source data, *COG* loops through each PSF bin in turn. It first checks to see if there are at least the minimum number of “clean” sources saved in that bin, given by the `psfsrscs_min` input parameter. If not, no calculations are done, a flag is set, and *COG* moves on to the next bin.

If there are enough “clean” sources saved in the PSF bin, it first calculates the set of “delta magnitudes” for the differential curve of growth. The value  $dm$  for each source at aperture  $i$  is:

$$dm_i = m_i - m_{i+1}$$

where  $m$  is the aperture magnitude of the source. Note that there is no  $dm$  value for the largest aperture.

The program then calculates the median  $dm$  value from all sources in each aperture, using an iterative median calculation with  $n$ -RMS rejection, where  $n$  is given by the input parameter `n_rej` and RMS is the root-mean-square of the set of  $dm_i$  minus the median value  $\rho_i$ . Any  $dm_i$  value that differs from the current median by more than  $n$  times the RMS is rejected from the data set, and the median is re-calculated in the next iteration. Iteration stops when the change in the median from the previous iteration to the next is less than the value of the `del_med` input parameter, the number of iterations reaches the value of the `iter_max` input parameter, or the fraction of rejected sources exceeds the value of the `loss_max` input parameter. In the last two cases, a warning message is printed to `STDOUT`.

### 3.2.4 Aperture Correction Calculation

Next, *COG* tests convergence to find the “optimal” or “infinite” aperture. This aperture, which contains all the light from a point source, is the smallest aperture  $i$  for which any of the following criteria are met:

- $\rho_i < del\_cnvg$ , where  $del\_cnvg$  is the value of the `del_cnvg` input parameter (**convergence**).
- $\rho_{i-1} - \rho_i < del\_div$ , where  $del\_div$  is the value of the `del_cnvg` input parameter (**divergence**).
- The aperture  $i$  is the largest aperture (**non-convergence**).

In the case of divergence or non-convergence, a warning message is written to STDOUT.

Once the “optimal” aperture is determined, *COG* calculates the difference between each aperture magnitude and the “optimal” aperture magnitude  $m_{opt}$ , or

$$dm_i = m_{opt} - m_i$$

for each source and aperture. The median aperture correction for each aperture  $i$  is then calculated iteratively with  $n$ -RMS rejection, as before.

## 4 OUTPUT

### 4.1 *COG* Output

The data output from the *COG* program consists of two sets of files, one file from each set for each PSF bin.

#### 4.1.1 Output Curve of Growth Sources Table

The output curve of growth sources table is written for each band and PSF bin that contains at least one “clean” source. The table name is determined by the input parameter `outbase`, the band, the total number of PSF bins in  $X$  and  $Y$ , and this PSF bin location; for example:

```
[outbase]-w[band]-cog-[NpsfX]x[NpsfY]-[thisNpsfX]x[thisNpsfY].tbl
```

The header of the file contains the following keywords:

- **Title:** The title of the file, containing the originating program name (*COG*) and the PSF bin identifier, which consists of the band and the PSF bin location ( $X \times Y$ ).
- **Version:** Version and runtime information.
- **PSFName:** The PSF bin identifier.
- **Nsracs:** The number of “clean” sources in this PSF bin.

There is one data line per aperture for each “clean” data source in the PSF bin. The data columns are:

- **nf**: The frame number, in the order of the input list file, containing the source.
- **id**: The ID number of the source within the frame.
- **r**: The aperture radius for this aperture, in arcseconds.
- **dm**: The *dm* value for this aperture and data source.

More information on these output files can be found in the *COG* Output Curve of Growth Source Table SIS, [WSDC D-I127](#).

#### 4.1.2 Output Median COG and Aperture Corrections Table

The output median COG and aperture corrections table is written for each band and PSF bin for which medians were calculated, *i. e.* the bin contains at least the minimum number of “clean” sources. The table name is determined by the input parameter `outbase`, the band, the total number of PSF bins in *X* and *Y*, and this PSF bin location; for example:

```
[outbase]-w[band]-apcor-[NpsfX]x[NpsfY]-[thisNpsfX]x[thisNpsfY].tbl
```

The header of the file contains the following keywords:

- **Title**: The title of the file, containing the originating program name (*COG*) and the PSF bin identifier, which consists of the band and the PSF bin location (*XxY*).
- **Version**: Version and runtime information.
- **PSFName**: The PSF bin identifier.
- **Nsrcs**: The number of “clean” sources in this PSF bin.
- **ApConverg**: The aperture number of the “optimal” aperture.
- **TypeConverg**: The type of convergence: convergence, divergence, or non-convergence.
- **Flag**: A flag indicating any problems in calculating the medians or convergence.

There is one data line per aperture, with the following columns:

- **ap**: The aperture number (with “1” being the smallest aperture).
- **r**: The aperture radius for this aperture, in arcseconds.
- **dm\_med**: The median value of *dm* for this aperture.
- **dm\_rms**: The RMS value of the median *dm* calculation for this aperture.
- **dm\_flg**: A flag indicating any problems in calculating *dm\_med*; a value of -99 indicates no calculation was done because it is the largest aperture.
- **apcor\_med**: The median value of the aperture correction for this aperture.
- **apcor\_rms**: The RMS value of the median aperture correction calculation for this aperture.
- **ap\_flg**: A flag indicating any problems in calculating *apcor\_med*.

More information on these output files can be found in the *COG* Output Median COG and Aperture Corrections Table SIS, [WSDC D-1128](#).

## 4.2 *WCOG* Output

The output from the *WCOG* program consists of two types of plot files, PostScript and SVG (Scalable Vector Graphics). For each PSF bin with at least one source, the differential curve of growth is plotted using the output curve of growth source table. If the output median COG and aperture corrections table also exists, the median  $dm$  values are overlaid on the plot, with the aperture of convergence/divergence/non-convergence identified by a different symbol (as specified in the plot key). The plots are then output to files of both types, with names similar to their input files:

```
[outbase]-w[band]-cog-[NpsfX]x[NpsfY]-[thisNpsfX]x[thisNpsfY].ps  
[outbase]-w[band]-cog-[NpsfX]x[NpsfY]-[thisNpsfX]x[thisNpsfY].svg
```

## 5 TESTING AND PARAMETER TUNING