Astronomical Image Reduction and Data Analysis Software

(AIRDAS)

Incorporating

- Image reduction and analysis
- Data reduction and analysis

Revision	Date	Comments	Released By
2013.07.00	2013-07-14	Initial manual release.	Gavin Blakeman

Table of Contents

1. Basics

This manual is designed to serve as a manual for beginner and advanced users. Additionally, the manual contains details for software developers wishing to extend or debug the code. The details of the database tables are also provided for developers that wish to extend the databases, or reuse the database in new applications.

1.1. Application Description

The application is intended to provide a means of analysing and reporting astronomical observations. The package has been written to take advantage of the ATID and ARID astronomical databases.

The package includes the following features:

- Astrometry Recording
- Photometry recording of results
- Light curve plotting
- Examination/Editing of FITS headers
- Writing Observation information to XML files
- ATID Database (Astronomical Target and Information Database)

  The ATID database contains over 3 million objects, including both deep sky and stellar objects. Each object in the database can be referenced by more than one name. Data is also included for variable stars.

  The use of this database is optional. The functionality of the database can be replaced by live internet lookups from the SIMBAD database.

  The ATID database is disabled by default.
- ARID Database (Astronomical Research Information Database)

  The ARID database supports observation planning and recording. Provision is also made for managing FITS image files and a library function is also included for FITS files. User comments are also allowed to be created against each object in database.

  The ARID database is disabled by default.
- Weather Database

  The weather database is used to store weather information. The intent is that the user can connect to an existing database of weather information, allowing the application to extract relevant weather information from the database to link to images. Relevant information is Temperature, Pressure and RH. These parameters are used for calculating refraction when performing precision measurements.

  The use of this database is optional.
- The software is multi-threaded to ensure complete use modern multi-core systems. This is most evident when CPU intensive functions are being used (image manipulation). During non-CPU intensive use, the effect of the multi-threading will be minimal. The software should also function adequately on older single core CPU's, albeit with correspondingly reduced performance.

1.2. Supported File Types

The application supports reading from a number of different file types. The following formats can be read:

- FITS – Fits file format
- ST7 – SBIG format files.

- DNG – Adobe Digital Negative
- RAW – May camera raw formats are supported. For a full list of supported formats, refer to the libRaw documentation.

When the application saves files for storage, by default it used the FITS file type. This is intentional and cannot be changed. Any other file type may lose relevant information and may not be able to store all the required information in a single file. The internal format used by the application is fully compatible with the FITS standard.

The application is able to export to a number of file types, this includes:

- JPEG
- TIFF
- PNG
- DNG

Support for exporting is mainly to allow files to be sent to others in an easily opened format. The export functions should not be used for archiving data.

1.3. Libraries and Licensing

1.3.1. AIRDAS License

AIRDAS is licensed under the GPL V2 software license. This allows the user to make and distribute copies of the software, provided that the source code for the software is included.

1.3.2. Third-party licenses

In order to improve the software and speed up development time, a number of third-party libraries are used. These libraries have different license agreements.

The list of third-party licenses and licensing terms and URL's for each of these libraries can be found in 7.2Standard Libraries. Efforts have been made to use the libraries without any changes. This is to ensure that the specified library revisions will work without modification.

1.3.3. Application Specific Libraries

There are also a number of libraries used that have been developed specifically for this application as well as other similar applications. These libraries are listed below and the relevant license restrictions are also listed.

- Astronomy Class Library (libAstroClass) - Licensed under the GPL V2
- Storage Class Library (SCL) - Licensed under the GPL V2
- Math Class Library (MCL) - Licensed under the GPL V2
- Physics Class Library (PCL) – Licensed under the GPL V2
- General Class Library (GCL) – Licensed under the GPL V2

1.4. Database

By using a standard SQL database to store application related information, the application gets the power and flexibility that is provided by a SQL database. While this approach may appear daunting to some users, it is relatively easy to set up a simple database server. There are a number of free (cost) applications that are applicable to both Linux and Windows. As home-servers become more common, this approach also allows the power of the home server to be tapped to serve the data to the application.

By using the Qt framework, a number of database servers are within easy reach. Any of database servers supported by the Qt framework may be used. This includes:

- Oracle (including Oracle XE)
- MS Access (ODBC)

- MySQL
- SQLite (V2 & V3) (V3 Recommended for ARID database.)
- IBM DB2
- Borland Interbase
- PostgreSQL (Version 7.3 and above)

When choosing a database server, the amount of support available on the internet should be a key consideration. The use of SQLite may also be simple to implement for the basic user.

1.5. Limitations

There are a number of limitations that are coded into the application. These are listed below:

1.5.1. Maximum Image Dimensions

The maximum image dimensions allowed are set by the size of the integer that is used for indexing. The size used by AIRDAS is a “size_t”. The size_t type is usually set dependant on whether the compiler is compiling for 32bits or for 64 bits.

1.5.2. Maximum Image Size

Due to the maths used to index arrays, the largest size of the image is 2,147,483,647 pixels. (2.1GP) Depending on the underlying type of the image, each image plane (colour) cannot exceed 2,147,483,647 bytes. When a “double” is used as the underlying type, this limits the image plane to 268,434,206 pixels (268MP)

When the application is compiled as a 32 bit application, the limits are smaller than if the application is compiled as a 64 bit application.

1.6. Known Bugs

The following are documented known limitations or bugs in the software. This list is not exhaustive and will be updated as further limitations are made known.

1.6.1. FITS Files

1.6.1.1. BITPIX=64

The application does not support reading FITS images (primary HDU or extension HDU type image) using BITPIX=64. Upon trying to open a file with BITPIX=64, the application will show an error. This error may not be specific to the actual problem.

This limitation is imposed by the CCfits library. Future versions of the library and software may correct this limitation.

1.6.1.2. Comments

There is a bug in the current release of CCfits that prevents the display of comments. All binaries released will have the relevant fix applied. The bug has been communicated to the development team of CCfits.

1.6.1.3. History

There is a bug in the current release of CCfits that prevents the display of history. All binaries released will have the relevant fix applied. The bug has been communicated to the development team of CCfits.

1.7. Revision Numbering

Each revision (version) of the software will be released with a unique revision number. The revision number will always be of the form YYYY.MM.BBBB where:

- YYYY – the year of the revision
- MM – The month of the revision.
- BBBB – Build number (Hexadecimal).

Major new releases will have a new revision number indicating the year and month of release.

It is intended that a single development path is used. This has the effect that between revisions, there may be builds of previous revisions that contain features intended for release in following revisions.

There is no intention to release beta versions of the software. As each version is released, it will be debugged as required.

The development trunk will always contain the most recent code. There is no intention to develop multiple releases simultaneously.

1.8. Notes

1.8.1. FITS Files

This application produces fully conforming FITS files. These files may have more than one HDU present. Some application are known to have issues when opening files from this application.

- AIP4Win – Will open file, but discards any additional HDU's. The additional HDU's are not available and are lost when the files are saved.
- CCDOps – Does not open files at all. (Suspect the issue is related to the use of BITPIX = -64.)

It is important to be aware that when additional HDU's are not saved, then astrometry and photometry information can be lost.

The WCS information is however stored in the primary HDU, and this should ensure that all applications open and save the data correctly.

1.8.2. Image Directories

While the user is free to choose any directory structure they wish to use to store astronomical images, some comments around the directories that are used by the application are applicable.

- Master Dark Directory – The application defines a directory in which to store master dark images. This is to make searching for master darks convenient.
- Master Flat Directory – The application defines this directory as being where the master flat files are stored. This is to make searching for master flats convenient.
- Master Bias Directory – The application expects that Master Bias files are stored in this directory.
- Scalable Dark Directory – The application expects that scalable dark files are stored in this directory.
- Image Directory – Images can be stored in this directory, or sub-directories off this directory.

2. Application Description

The tools provided include weather tools to allow weather on observational and non-observational nights to be captured and stored in a database.

2.1. Overview

2.1.1. Photometry

Photometry is the determination of the brightness of objects.

2.1.2. Astrometry

Astrometry is the study of the position of objects in the sky. Measurements are normally made relative to other objects. Commonly stellar objects are used as reference points for targets.

Target objects can be solar objects, or other objects that need their position tracked over a period of time. Commonly, astrometry is used for orbit determination of asteroids and similar.

With an astrometry program, it is necessary to take repeated images of a portion of the sky to find objects of interest. These objects of interest will then have their position determined and possibly tracked over a period of time.

3. Menu Actions

The main menu consists of the following items:

- File menu – Actions concerned with files. Includes open, save, export, print.
- Edit menu –
- View menu – Actions to open dock widgets and controllers.
- Process menu – Actions concerned with processing images. This includes image calibration, creation of master frames, image stacking and image alignment.
- Analyse menu – Actions to analyse images. This includes source extraction, and WCS determination, astrometry and photometry.
- Transform menu – Geometric transformations of images.
- Enhance menu
- Colour menu
- Utilities menu – Useful tools and calculators.
- Window menu – Manage the open windows.
- Help menu – Provide help on different items.

3.1. File Menu

3.1.1. File | Open

This allows a file of type “.FITS” or “.ST7” to be opened and displayed for modification and editing. The file is opened in an image window.

The image window allows the various HDB's associated with the files to be selected and viewed or edited.

Typically a file would have at least a single HDB, the primary HDB. This will always contain an image, or data for an image. Additional HDB's may have information relating to the image, or additional images depending on the use.

3.1.2. File | Save

Saves an image. This action is used whenever an image needs to be saved after modification. If this action is not allowed, it will not be available.

Images are always saved in the FITS file format.

If the image has not been saved before and does not have an associated file name, then this function will automatically call the Save As option in preference.

3.1.3. File | Save As

Allows the file to be saved under a new file name. If the file has not been saved before, this action will automatically be called, even if the File | Save option is chosen.

3.1.4. File | Export

Allows an image to be exported into different formats. As a FITS file cannot be accurately represented in the possible formats, the HDB required for the FITS file is identified and the information from the single HDB is output to an appropriate format of file.

Generally images can be output to image formats, such as TIFF, JPEG and PNG. Tables will be output to .CSV files. Binary tables may not be able to be stored in a constructive fashion outside of the FITS file.

When a photometry table, or astrometry table is selected, the table can be exported as a .CSV file. The format of the .CSV file is fixed in the application.

3.1.5. File | Batch Convert

Allows a number of files to be converted to FITS format. The original file type can by any of the supported file types.

During the import process, additional data can be appended automatically to the file. This includes observer details, weather conditions, location information, telescope details and camera details.

A unique ID (UUID) will also be assigned to the FITS image, and the image registered and recorded in the ARID database.

When this option is selected, a dialogue box is displayed, allowing the selection of the files to be converted, and showing the parameters available for application to the file.

Once the parameters and file names are defined, the process will begin. The dialogue will be closed, and a separate thread will be opened to perform the conversion. The new thread will run in the background performing the conversion. The status of the new thread will be shown in the frame window, and it will also be possible to cancel the conversion process if required.

Note: Only one additional thread will be created per batch conversion. As these tasks will probably be IO/file read limited, there is no point creating additional threads.

3.1.6. File | Exit

This action exits the application. If any files need to be saved, the user will be prompted whether the file must be saved or not. If the user cancels during save actions, the exit will be abandoned.

3.2. Edit Menu

3.2.1. Edit | Configure | Sites

Allows observing site details to be added or updated to the database. The information required to create a site includes the location (latitude and longitude), the altitude and also some information on daylight saving time and time zones.

The don't display check allows the user to select sites that they do not wish to display in the observing site combo box at the bottom of the screen. This allows the list to be customised to a small number of frequently used sites.

3.2.2. Edit | Configure | Configure Observers

Allows observers to be updated or added to the database.

3.2.3. Edit | Configure | Instruments

Allows instruments (cameras) to be updated or added to the database. These are generic models for the cameras commonly available and can be customised to suit the actual parameters of specific observers cameras.

3.2.4. Edit | Configure | Telescopes

Allows the user to configure the telescopes that will be used as defaults for observations.

3.2.5. Edit | Configure | Set Catalogue Preferences

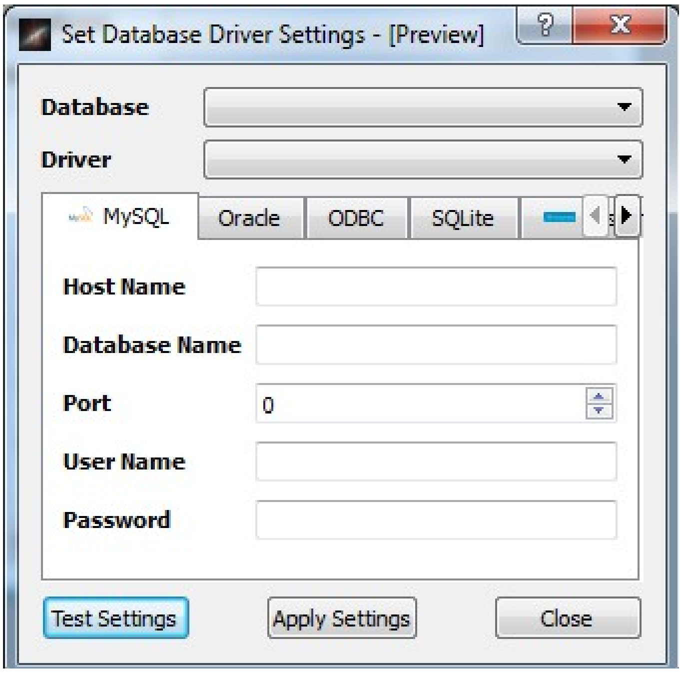
This allows the user to set-up catalogue sort-order preferences for names. As the catalogues themselves form part of the ATID database, they cannot be changed from this software. However the order of their use can be selected using this function.

3.2.6. Edit | Configure | Database

This allows the connection to the database to be configured.

The three database connections can be set up from this option.

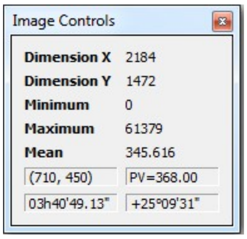
Note 1: During application start-up, connections to the databases are made. If the connections are not successful, the application will also show this dialogue to allow the user to update the connection settings.



3.3. View Menu

3.3.1. View | Image Information

This option brings up the Image control docking window. This allows the user to gain information about the state of the image. That is currently being displayed. If no image is displayed, this option will not be available. If the image control window is already

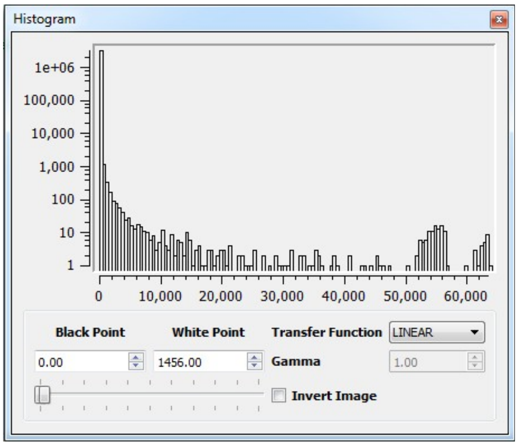


visible and the window is switched to a non-image window, then this information window will be disabled.

When the cursor is being moved over the selected image, the position on the image and the value of the pixels below the cursor are also shown.

If WCS information is available for the image, then the actual RA/DEC position will also be displayed in this dock widget.

3.3.2. View | Histogram



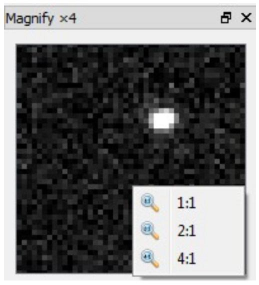
The histogram dock widget shows the histogram for the image.

The histogram dock widget allows a number of parameters associated with the display of the image to be modified.

Controls are also provided for changing the white point and black point of the image. This affects the display of the image, but does not change the image in any way. It is also possible to completely invert the image.

The transfer function that is used to convert the data from the image into an image is also set from this option. It is important to note that changing the transfer function only changes the way the image is viewed and does not change the underlying information about the image.

3.3.3. View | Magnifier



The magnifier window provides a magnified view of the current cursor location.

There are three levels of magnification provided, 1×, 2× or 4×. The level of magnification is displayed on the viewer window and can be selected by bringing up a pop-up menu, by clicking the right mouse button with the mouse pointer over the magnifier window.

The magnifier window will always display the magnified view of the currently selected image window.

This magnifier will also work with the image stacking tool and the image alignment tool.

3.3.4. View | Navigator

The navigator image is a thumbnail image that displays the position of the current view on the larger image. The view is displayed by the red square on the Navigator window.

There are no interactions processed for the navigator window, it is simply an aid for determining location on a larger image.

The navigator window will also work with the stacking tool and the image alignment tool.

3.4. Process Menu

The “Process Menu” contains options relating to the processing of

3.4.1. Process | Calibrate Current Image

This option allows a single image to have the calibration frames applied. The user needs to identify the image to be calibrated, specify the extent of image calibration and provide information for the dark, flat and bias frames (where applicable). The option to keep the original data in the FITS file together with the calibration data and the calibrated image is also provided. (This would lead to a large sized file)

Checks are made during the calibration that the calibration frames match the extents of the frame to be calibrated, that the cameras are the same (if the camera information is present in the files), that the duration of the dark frame matches the duration of the image, that the temperature of the dark frame matches the temperature of the image.

If these checks are not met, the user is given the option to override the process and continue anyway.

See the image calibration topic for more information on image calibration.

3.4.2. Process | Calibrate Multiple Images

With this option, a number of images may have the same calibration frames applied. Checks are made to ensure that the flat and dark frames used for the calibration of the images are appropriate.

The images will all be output to the save directory. They are automatically saved as FITS images.

3.4.3. Process | Create Master Dark Frame

A master dark frame is the combination of a number of dark frames, that are stacked to form a single dark frame. As a dark frame is a capture of the noise present in the CCD, by combining a large number of dark frames together, a more accurate representation of the average noise present in the CCD can be obtained. The temperature of the master dark frame should be matched to the temperature of the image taken as the electron noise is proportional to the temperature of the CCD.

When creating a master dark frame, it is possible to create a scalable master dark by including the use of bias frame subtraction.

When using bias frame subtraction, either a master bias frame can be specified, or a number of bias frames can be selected. If a list of bias frames is supplied, the list of frames will be converted to a master bias frame through averaging of the bias frames. The user can also choose to save the master bias frame.

The same menu command is used for creating basic and advanced master dark frames.

After selecting the command, the first step is to identify the dark frames that will be combined for the dark frame. .

The last step is to identify the name for saving the new master dark frame. The “Create Dark” button can then be pressed.

The application will then load each dark frame and check that the integration times (“EXPTIME” or “EXPOSURE” keywords) and temperatures (if present) are the same for all images. If there are discrepancies, the application will give a warning, but will allow the user to choose to continue or to cancel the process.

If the file name to save as already exists, then the application will prompt to overwrite it. The user can then change the file name or allow it to be over-written.

If the exposure times are correct, then the application will continue and create the dark frame using the process for advanced or basic protocol. If bias frames or a master bias frame is used, then the master bias will be subtracted from each of the dark frames before the dark frames are added.

The master dark frame will always be saved to disk as a FITS file. It could be exported in another format, but the most accurate storage method for further use would be as a FITS file. (It will always have the extension .fits or .fts.)

The Master Dark can be created by using a mean combine or a median combine. The mean combine can suffer from the effects of cosmic ray strikes, while the median combine should not suffer the same effects. The median combine will however take a significantly longer amount of time to process, especially with large images.

3.4.4. Process | Create Master Flat Frame

The master flat frame is created in a similar fashion to the master dark frame and the master bias frame.

The master flat frame should be created from a number of flat frames that are combined to create a master flat. The number of frames required to create a representative master flat can be found in other publications.

When creating a master flat, there are two methods that can be used. The master flat can be created without using dark correction, or it can be created using dark correction. The more accurate method is to use flat darks. The flat darks should be the same exposure as the flats. This obviates the need to use bias correction as the flat-darks do not need to be scaled.

Two options are available for a the combining algorithm used for the creation of a master flat frame. This is the mean combine and the median combine. The mean combine will always be quicker especially when there are a large number of images.

3.4.5. Process | Create Bias Frame

Creating a master bias frame makes it possible to create advanced master darks. These advanced master darks can be scaled for time. The creation of a master bias frame is simple in that a number of bias frames are average combined to create the master bias.

The bias frames are checked before performing the combination of the frames. They need to have the same exposure times and dimensions of image. If this is not true, they cannot be combined into a master bias frame.

The master bias frames can be combined using a mean combine or a median combine. A median combine can take a large amount of time if large images or a large number of images are involved.

3.4.6. Process | Compare Images

Allows the comparison of a number of images. The images are compared by firstly aligning the images and then by blinking the images, allowing any differences between the images to be easily identified. The aligned images can also be saved, and any moving objects (astrometry) marked on the images and saved into the astrometry HDU. The moving objects can also be saved back into the original images without having to modify the original images. (Except by adding the astrometry HDU.)

The comparison process is a three step process.

1. Identify and load the images.
2. Identify the alignment points. (FUTURE: If WCS information is present in all the images, then the images will automatically align based on the included information.)
3. Blink the images to compare the information.
4. If there are moving objects observed, they can be marked on the image.
5. Either the aligned image can be saved, or the original image can be saved with the astrometry information embedded.

3.4.7. Process | Stack Images

The image alignment option allows the stacking of a number of images to form a composite image. The images may be calibrated concurrently as they are stacked.

There are several options associated with image stacking, these are:

- Sum Combine (FUTURE)
- Mean combine
- Median combine (FUTURE)
- Sigma Clip (FUTURE)

The combine options are mutually exclusive options. These control how the images are added together to get the output image. The sum combine simply adds each pixel value together. In the mean combine, all the images are added together and then averaged, in the median combine, each point in the final image is median of the points of the individual input images.

The images need to be registered before they can be combined. This is done by manually selecting two alignment points per image. These alignment points should correspond on each image. The first image is always the reference image. This image is not translated, rotated or scaled, but all the other images are altered to match to the reference image.

When performing the stacking function, the images may need to be rotated, scaled and translated to register all the images correctly. This rotation, translation and scaling operation does not change the size of the final image. Any areas that fall outside of the reference image are just clipped.

The output image is not saved automatically, and the “File | Save” or “File | Save As” menu items must be used to save the output image. If the image has not already been saved, then a prompt will show to allow the file name to be selected or input.

The file will always be saved in FITS format.

The output image can also be transferred to a normal imaging window. This will result in the image being removed from the stacked output display area and being re-opened in a new window.

For more details on the algorithms used for image stacking, please see the algorithms section.

3.5. Astrometry Menu

3.5.1. Astrometry | Measure

This function allows astrometry measurements to be made within an image. A number of reference stars must be defined. After the reference stars have been defined, targets can be identified that the position needs to be measured.

Identifying reference stars is a two step process. The first step is to mark a reference star. The reference star then needs to be identified. After two reference stars have been marked, the database can be queried for other stars in the vicinity. These can also be used as reference stars.

The more reference stars identified the better, as the least squares solution of the plate constants is more accurate if there are more knowns for the least squares solution.

3.5.2. Astrometry | Compare Images

This function allows a number of images to be compared. This will show up changes between the images. These changes can be due to phenomenon such as supernovae or moving objects in the field (asteroids, planets etc.)

Two methods of image comparison are provided, blinking and subtraction. Blinking works by switching between two images in succession, the human eye finds it easy to detect changes between two images that are blinked in succession.

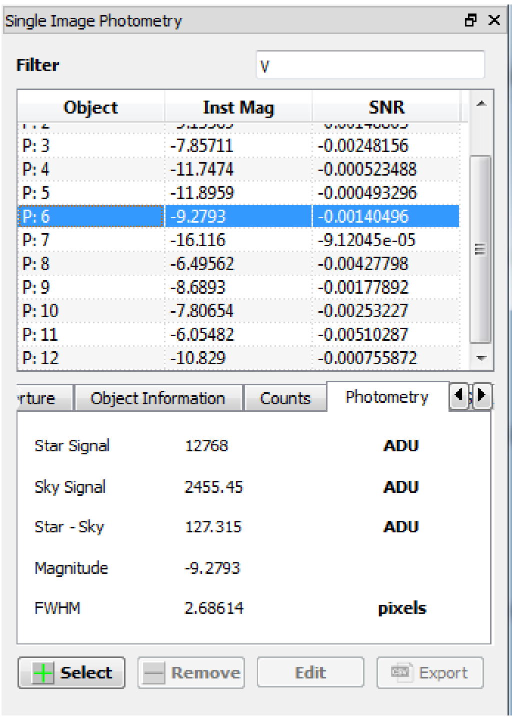
Any number of images can be selected for comparison. Once the images are selected, alignment points can be specified. This ensures that the images will display on top of one another and will not show differences due to misalignment.

3.6. Photometry Menu

3.6.1. Photometry | Single Image Photometry

This function is used for performing photometry on a single image. This allows the raw instrument data to be extracted for future analysis.

When the function is selected, the photometry control window opens up, and photometry information can then be controlled and entered.



When photometry is being performed, the information is written to the “Photometry HDU” in the FITS file. (Note the FITS file itself is only updated when the file is saved.)

If there is a “ZMAG” keyword in the FITS file, then the instrument magnitudes are updated to reflect this ZMAG value. This is not however a fully accurate correction.

A breakdown of the algorithm used for determining the FWHM and the magnitude can be found later in the manual. The equations used for the various calculations can also be found later in the manual.

Any number of sources can be selected and analysed in each image. Object names can also be associated with each object.

3.6.2. Transform | Mirror Horizontal

The image will be mirrored around the vertical axis. This allows images to be corrected for telescope reflections to allow the image to appear as the sky-view would appear.

3.6.3. Transform | Mirror Vertical

Mirrors the image around the horizontal axis. This allows images to be corrected for telescope reflections to allow the image to appear as the sky-view would appear.

3.6.4. Transform | Re-sample

This transform is used to re-sample the image. Re-sampling increases the number of pixels in the image, thus making the image appear smoother, although there is actually no additional information in the image. Re-sampling can also be used when the pixels from the camera are not square. By resampling, the pixels will be “squared-up” making the aspect ratio of the image correct.

When this action is selected, a dialogue is shown allowing the new dimensions for the image to be selected. There are two methods that can be used, a percentage increase, or choosing the new image size. Both have the same effect. There is also the option to maintain the aspect ratio of the original image, or allow any arbitrary aspect ratio to be used.

3.6.5. Transform | Float

Floating an image is the process of making the canvass larger, without increasing the size of the image itself.

When an image is floated, the extra space created is shared between the margins, so as to keep the original image details in the centre of the new image.

3.6.6. Transform | Rotation

This function can only act on images, and an image must be visible and selected as the current window for this function to be available.

When the function is selected, it allows an image to be rotated a fixed number of degrees to the left or the right. The image is always rotated around the (0, 0) point of the image.

The image will be automatically resized as required to ensure that the original image fits within the bounds after the rotation.

3.6.7. Colour | Convert->RGB

Converts a colour image into separate red, green, blue images. Each of the colour images will be a grayscale image and will open in a new window. The file name of the original image will be used, but the colour (red, green, blue) will be appended to the file name.

3.6.8. Colour | Convert ->LRGB

Converts a colour image into separate luminance, red, green, blue images. Each of the output images will be a grayscale image. The file name of the original image will be used, but the colour (luminance, red, green, blue) will be appended to the file name.

3.6.9. Colour | Convert->Grayscale

Converts a colour image into a grayscale image. The luminance channel is used for the grayscale image. The grayscale image will be opened in a new window, with the file name having “grayscale” appended to the file name.

3.7. Data Analysis

The data analysis tool allows the data generated from astronomical observations to be analysed. This is a number of graphical and other tools, that simplify the display of the data.

3.8. Utilities

4. Algorithms

4.1. Image Calibration

Image calibration is the process of correcting an image for known inaccuracies. In the case of CCD imaging, the inaccuracies are the noise generated during the exposure and digitising process and the noise generated by the optical system. These can be corrected for using a flat frame and a dark frame.

The software allows the Calibration to take place on a single image, or on a group of images.

Numerous descriptions of this process and the algorithms used can be found on the net, or in some of the reference books listed in the back of this document. Therefore, no detail will be supplied here.

4.2. Image Stacking

The image alignment option allows the stacking of a number of images to form a composite image. The images may be simultaneously calibrated as they are stacked. The user needs to select the images that are to be stacked. Alignment points in each image then need to be identified by the user. The images can then be stacked. If the user needs to have the images calibrated, the dark, bias and flat frames need to be identified. The work-flow followed for image stacking is:

1. Image calibration (If required)
2. Image registration
3. Image stacking

Algorithms are provided sum, mean and median stacking. In mean stacking, the stack is the mean of the images. With median stacking, the median value of each of the images is used to construct the output image. (mean and median have the usual statistical meanings in this sense.)

Standard image rotation and shifting algorithms are used.

5. FITS File Usage

This section should be read in conjunction with the FITS standard.

The FITS file format is used for saving images that have been edited, using the AIRDAS application. The FITS format is used as it is a standard format that is easy to interface and supports the required functionality. However, the software makes use of the FITS format in a software specific fashion, while still complying with the FITS standard.

The philosophy adopted by AIRDAS is to have each FITS file represent one image. (This may be a composite/stacked) image. Each additional HDU contained in the file then has additional information about the image.

A FITS file consists of a number of header and data units (HDU). The first HDU is called the primary HDU. The software treats each FITS file as a single image. This can be a monochrome or polychrome image. Accordingly, the primary HDU should also contain the image.

The Primary HDU always contains the primary image. This is the image that is to be operated upon. If there are additional images contained in the FITS file, they are considered to be support images to the primary image. Thus a FITS file could be created having a dark frame image, a flat frame image, the original CCD image and the primary image. In this case the primary image should be the CCD image with the dark frame and flat frame applied.

Any additional HDU's should refer to the primary image. Thus the astrometry HDU is the astrometry of the primary image. Similarly for the photometry HDU. When the primary image is manipulated (rotate, mirror etc), then the HDU's that are referring to the primary HDU are also manipulated. This ensures that the coordinate systems are kept in sync.

To enable file identification, each file is tagged with a UUID (UUID) when it is initially saved. This UUID is used to identify the file for future matching of the file.

5.1. HDU Details

The primary HDU is followed my a number of optional HDU's. If photometry or astrometry have been conducted on the image, then the software will create additional HDU's and append them to the image file. These additional HDU's will be to store astrometry and photometry information and will be ASCII Tables containing the stellar or solar object information for selected objects. This allows the data to be loaded automatically at runtime.

5.1.1. Primary HDU

The primary HDU shall contain the image. The primary HDU must contain location and weather information.

5.1.2. Astrometry HDU

The astrometry HDU contains data relating to astrometry of the image. This can include reference object data, target object data and plate constant data.

5.1.2.1. Keywords

The astrometry HDU may contain the following keywords:

5.1.2.2. Data Structure

The astrometry HDU uses an ASCII Table and has the following format:

Field Number	Start Column	Field Name	Format	Description
1	1	SEQ	I3	Sequential number of the object

2	5	OBJECTNAME	A26	Name of the object
3	32	R/T	A1	Target/Reference (T or R)
4	34	RA	F10.8	RA of the object (catalog place) (h.hhhh)
5	45	DEC	F10.7	Declination of the object (catalog place) (d.ddd)
6	56	EPOCH	F10.1	JD of the Epoch
7	67	PPX	F6.2	X coordinate of the object (Plate coordinate)
8	74	PPY	F6.2	Y coordinate of the object (Plate coordinate)

The observation time is already tied to the FITS file. Having the object name, and the relevant catalog, the observed place can be recalculated. Thus the observed place is stored, rather than the catalog information. The RA and declination are stored as decimal values.

By storing all the information required to perform an image reduction future uses will be able to replicate any of the results obtained from the image.

5.1.3. Photometry HDU

This HDU contains information relating to photometry of the image. The HDU will only exist if photometry has been done to the image. The photometry HDU is stored as a binary table as specified by the FITS standard. The choice of a binary table is to allow the aperture and annulus information to be easily stored for different shapes of aperture/annulus.

Field Number	**Field Name**	**Format**	**Description**
1	Object Name	40A	Object name or sequence number.
2	Object RA	1D	
3	Object Dec	1D	
4	Object centroid X	1D	
5	Object centroid Y	1D	
6	Magnitude	1D	
7	Magnitude uncertainty	1D	
8	Source ADU	1D	
9	Source Area	1D	
10	Sky ADU	1D	
11	Sky Area	1D	
12	FWHM	1D	
12	Aperture	PB	Information about the aperture

The aperture information is variable length data that is used for serialising the aperture data. This allows the data for different types of apertures to be correctly written and read.

5.2. Special Keywords

While the FITS standard mandates a number of keywords, there are a number of keywords that the software reads as special keywords. These special keywords provide information that can be used programatically when calculating certain values or reducing information. The keywords that are used as special keywords come from the FITS standard as well as from other sources that are not standardised.

In choosing the keywords to use as special keywords, the following order of choice is used:

1. FITS standard.

2. Other large bodies that use the FITS format. (NOAO, UCOLICK etc)

3. Private company use of FITS keyword. (SBIG etc)

4. openAIRS specific keywords.

All the special keywords are listed in the following sections.

5.2.1. FITS Keywords

Each FITS file has a number of FITS keywords. Some of these keywords are mandated by the FITS standard, some are defined by the FITS standard. These keywords can be written into the FITS file at the time of the file being created, or at a later date when the file is being processed.

Non-standard FITS keywords can also be used.

The application uses a number of non-standard FITS keywords to store information about the images. Where possible, these non-standard FITS keywords are derived from other organisations that use FITS files. The application will display all included FITS keywords as well as allowing keywords to be written to file.

There are however a number of keywords that have special meaning/usage by the software and are read by the software for use by the application.

5.2.1.1. DATE-OBS

HDU: Primary

DATATYPE: string

COMMENT: date of the observation

DEFINITION: The date of the observation, in the format specified in the

FITS Standard. The old date format was 'yy/mm/dd' and may be used only for dates from 1900 through 1999. The new Y2K compliant date format is 'yyyy-mm-dd' or 'yyyy-mm-dd THH:MM:SS[.sss]'.

5.2.1.2. TIMESYS

HDU: Primary

DATATYPE: string

COMMENT: Time system to apply to all dates in the HDU/file

5.2.2. AIRDAS Keywords

These are keywords that are defined by the NOAO. The documentation can be found here: http://iraf.noao.edu/iraf/web/projects/ccdmosaic/imagedef/fitsdic.html.

5.2.2.1. CCD-TEMP

HDU	Image
DataType	Numeric
Unit	°C
Comment	CCD Temperature
Definition	The value field shall contain the temperature of the CCD at the time of the exposure.
Source	SBIG

5.2.2.2. ENVMJDn

HDU: primary

DATATYPE: real

UNIT: MJD

COMMENT: Environment measurement time

DEFINITION: Times for the site environment measurements given as modified Julian. For the wind measurements this is the start of the sampling period.

SOURCE: NOAO[1]

5.2.2.3. ENVTEM

HDU: primary

DATATYPE: real

UNIT °C

COMMENT: Temperature outside the dome.

DEFINITION: The temperature outside the dome at the time(s) given by the ENVMJD(n) keywords.

SOURCE: NOAO

5.2.2.4. EXPOSURE

HDU	Image
Data type	Numeric

1 These are keywords that are defined by the NOAO. The documentation can be found at http://iraf.noao.edu/iraf/web/projects/ccdmosaic/imagedef/fitsdic.html.

Unit	S (second)
Comment	Image Exposure Time
Definition	The value field shall contain a floating point number giving the exposure time of the observation in units of seconds. The exact definition of 'exposure time' is mission dependent and may, for example, include corrections for shutter open and close duration, detector dead time, vignetting, or other effects. This keyword is synonymous with the EXPTIME keyword.
Source	HEASARC

5.2.2.5. EXPTIME

HDU	Image
Data type	Numeric
Unit	S (second)
Comment	Image Exposure Time
Definition	The value field shall contain a floating point number giving the exposure time of the observation in units of seconds. The exact definition of 'exposure time' is mission dependent and may, for example, include corrections for shutter open and close duration, detector dead time, vignetting, or other effects. This keyword is synonymous with the EXPOSURE keyword.
Source	NOAO

5.2.2.6. OBSERVAT

HDU: Primary

DATATYPE: string

COMMENT: Observatory

DEFINITION: Observatory identification for the site of the observation.

SOURCE: NOAO

OpenAIRS: ~~This keyword may store an integer value, if so, the value is the key for the TBL_SITES.SITE_ID table.~~

If the value is a string, it is the site name. (TBL_SITES.SHORTTEXT)

If the value does not match any of the database table values, it is treated as a string value that has no reference to the TBL_SITES table.

If a value is specified for LATITUDE and LONGITUD, then this value is for information only. If no value is specified for LATITUDE and LONGITUD, then this value *may* be used to lookup the value of the lat/long for the observatory site.

The preferred value is a string value corresponding to the site name.

5.2.2.7. LATITUDE

HDU: Primary

DATATYPE: real

COMMENT: Geographic latitude where the observation was made

DEFINITION: The value field shall contain a floating point number giving the geographic latitude from which the observation was made in units of degrees.

SOURCE: UCOLICK

openAIRS: This keyword shall contain the latitude of the observatory. If the observatory is specified by name, after the first save this value will be correctly defined.

5.2.2.8. LONGITUDE

HDU	Primary
Data type	real
Unit	Degrees (d.ddd)
Comment	Geographic longitude where the observation was made
Definition	The value field shall contain a floating point number giving the geographic longitude from which the observation was made in units of degrees.
Source	AIRDAS

5.2.2.9. UUID

HDU	Primary
Data type	String
Unit	None
Comment	UUID of image
Definition	The value field shall contain a UUID that identifies the file.
Source	AIRDAS

The UUID is generated the first time that the file is saved, before the file is saved. The UUID is used for linking the image to a database primary key. The table also stores some image parameters. The primary key links the image to specific observations in the database.

6. Scripting Language

The application supports a scripting language that allows batch processing of files. The scripting language allows the user to set-up a process for calibrating images as well as extracting information from the images.

The scripting language is described in this chapter.

6.1. Scripting Module

The highest level of the scripting language is a module or program. The module can include other modules in it's code, but the module is essentially the program that determines how the images are processed.

The module is written in a scripting language that will be described shortly. The module will typically be written using a flat text editor (gedit, notepad) and opened and run by the application.

6.2. Script Commands

All script commands are single line commands. There can only be one command per line. The script interpreter is case sensitive.

#PROGRAM *program name – compulsary*

#AUTHOR *Authors Name – optional*

#DATE *yyyy-mm-dd – optional*

#SET

#VAR

#FUNCTION

#INCLUDE

7. Software Design Considerations

This chapter presents the design considerations and philosophy followed for the design and implementation of the software. This information is presented for persons wishing to contribute to the extension or improvement of the software. This section only needs to be read if you are contemplating development of the software.

7.1. Note to Developers

- Use “double” for floating point numbers.

- Write all code to use 32bit and 64 bit compilation.

7.2. Standard Libraries

When functionality is needed that is not a core requirement, then external libraries are sought that can perform the work. By using external libraries, the development time is reduced, and the robustness of the overall application is improved. There is however a trade-off of the development time vs finding and integrating the libraries.

The libraries already integrated into the application are:

- DLib
- LibRaw – Provides RAW file support.
- Qt Application Framework
- Qwt – Graphical Widgets for the Qt framework
- Boost Library – Smart pointers, file path management

- CCFits – C++ interface for the CfitsIO library
- CfitsIO – Library for reading/writing FITS files
- gsl
- SOFA
- Xerces
- Xalan

All of these libraries are C or C++ libraries and are supported by either an open source development community, or are produced by a scientific establishment and are for public use.

Depending on how individuals wish to build the libraries, they can be built as static or dynamically linked libraries.

With Windows binary distributions, all the libraries with the exception of the Qt framework are integrated as statically linked libraries. The aim of doing this is to reduce the number of distributable files to a minimum.

For each code release, the README.TXT file will contain the list of external libraries and revisions numbers of the external libraries that the code has been developed and compiled against.

7.2.1. LibRaw

License: LGPL

URL: www.libraw.org

The LibRaw library provides RAW file support for the AIRDAS application. This allows the opening of different types of RAW files, including Adobe DNG.

7.2.2. Boost Library

License:

URL: www.boost.org

Selected areas of the Boost library are used in the application. This include smart pointers, threading support and file path support.

7.2.3. GNU Scientific Library (GSL)

License: GPL V2

URL: http://www.gnu.org/software/gsl/

The GSL is used to provide maths support to the application. The GSL is written in C, and interfaces easily to to application.

The GSL is not used universally. Some of the statistics functions have been re-written into C++ as multi-threaded functions and included in the MCL library.

7.2.4. SOFA

License:

URL:

7.2.5. Cfitsio

License:

URL:

This library provides low-level C access to FITS files. While the library is required, the application should make no calls to the cfitsio routies. Rather all access to FITS files should be performed through using the CCfits library.

7.2.6. CCfits

License:

URL:

7.2.7. CCFits

7.3. Development Language

C++ is the preferred development language. However, the use of C is also allowed. Libraries used are typically C++ and C.

1. All new code should be written using C++.
2. The use of standard C++ and C libraries is encouraged, including the use of the STL.
3. The API used for development is the Qt API. This was specifically done to enable easy porting of the application to other platforms than Windows. The Qt framework should be used as far as possible for AIRDAS specific code. This includes all functionality provided by the Qt framework.
4. All code being developed for SCL, GCL, MCL, libAstroClass, should be independent of the Qt framework. The intent is to keep these libraries general and allow them to be used with any front-end code required. AIRDAS handles the user interface, while the libraries do all the heavy lifting.
5. Any database server supported by Qt can be used. Where the database allows embedded code (such as PL/SQL in Oracle), the developer can take a view on whether to make use of the embedded code functionality. However, if code is embedded into a database, the developer must make the embedded code available with the software, or provide the functionality in the application code to bypass the need for the embedded code. The database is viewed to be part of the front-end code.
6. The code should be written to compile with any of the popular compilers. Where language differences exist between compilers, then the differences should be resolved using #ifdef statements that refer to the compiler.
7. All counters and indexes of arrays should be based on size_t. This is required for 64bit compilation.
8. All pointers should be based on ptrdiff_t. This is required for 64bit compilation.

7.4. Application Libraries

The application libraries have been designed to be stand-alone libraries. However, due to the contingencies and amount of time available for coding, all the functionality required to make them generic libraries has not been implemented. The libraries have been implemented in such a way as to support the AIRDAS application, and therefore only appropriate elements may have been implemented. This is not to say that the libraries cannot be expanded for future use.

7.4.1. libAstroClass – Astronomy Class Library

This library contains a functions for manipulating astronomical images, calculating astronomical parameters etc. The library has been written in C++. There is no OS (Windows etc) associated with the library. IE, all the functionality is written to be generic and not tied to a specific OS. This has been done to ensure interoperability and portability of the code.

All future code for the libAstroClass must comply with the requirement that it does not rely on a specific OS.

The documentation for the libAstroClass class library has been developed using the DoxyGen tool. This allows all the documentation for the library to be contained within the source code of the library.

7.4.2. MCL – Maths Class Library

This library includes mathematical classes and functions where the functions are not provided by other libraries, such as the gsl.

7.4.3. SCL – Storage Class Library

Provides storage classes. Some of these classes replicate the STL classes, but others are used where less generic, more specific classes are required.

7.5. Application Files

The application uses a number of code files. These are detailed and explained below. Due to the way in which the application has changed over time, the file names do not always appear logical. This may be addressed in future versions.

7.6. Making Qt

To make the Qt libraries, the following process should be followed for windows.

7.6.1. Windows

Open the command line from Visual C++

CD to the Qt main directory

> CONFIGURE -PLATFORM WIN32-MSVC2010 -QT-SQL-MYSQL -QT-SQL-ODBC -QT-SQL-OCI

> NMAKE

7.7. Application Error Codes

Errors are problems with the software set-up, or with the software code itself. Errors are divided into three categories:

• Critical errors result in a shutdown of the application. These errors are non-recoverable.

• Recoverable errors - These are errors that will result in the application not performing as specified. However, the software can self correct for the error. These errors always relate to the operating environment.

• Informative errors - These errors provide information to the user. This could be around issues such as code problems, or inconsistencies in the environment. This type of error does not result in an application shutdown, but should be reported to the software developers to enable the development of more robust code.

• Code errors - These are errors where the code gets to a place that it should not be able to reach. These errors may or may not shut the application down, and should be reported to the software developers to enable the bugs to be removed from the code.

As far as possible errors should throw exceptions. Exceptions should be caught in the code and allow for application recovery as far as possible.

7.8. Application Settings

7.8.1. Settings

The settings are stored in different locations depending on the operating system in use. The settings contain important data to determine how the application operates. Following are the details of the settings. The section after the settings describes the various locations that the settings can be stored depending on the operating system in use.

Organisation Name = GPL2

Application Name = AIRDAS

7.8.2. Database

ATID Database	

Oracle XE	
DriverName	
HostName	
DatabaseName	
UserName	
Password	
Port	
ODBC	
DriverName	
DatabaseName	
MySQL	
DriverName	
HostAddress	
DatabaseName	
Port	
UserName	
Password	

Weather Database/Database
Weather Database/Disable
Weather Database/Oracle XE/DriverName
Weather Database/Oracle XE/HostName
Weather Database/Oracle XE/DatabaseName
Weather Database/Oracle XE/UserName
Weather Database/Oracle XE/Password
Weather Database/Oracle XE/Port
Weather Database/ODBC/DriverName
Weather Database/ODBC/DataSourceName
Weather Database/MySQL/DriverName
Weather Database/MySQL/HostAddress
Weather Database/MySQL/DatabaseName
Weather Database/MySQL/Port
Weather Database/MySQL/UserName
Weather Database/MySQL/Password

Weather Database/SQLite/DriverName

Weather Database/SQLite/DatabaseName

7.9. Coding recommendations

Strict coding standards will not be implemented, however, the following coding recommendations are suggested:

- All code should be written is C++ using C++ available language features..
- Rather than using macros for values, constants should be used. Constants have better scope rules than macros thus preventing clashes. (int const value = 1;)
- 'using namespace xxx' should not be used. Prefer fully qualified names are preferred. (std::vector...)
- Indents should be 2 spaces only and be represented in the source files as spaces, not tabs.
- All class names should begin with “C” (CAstroFile)
- All structure names should begin with “S” (SControlImage)

8. ATID Database

The ATID database provides information for over 3.3million astronomical objects. The ATID database is distinct from the ARID database, in that it provides only the information on the astronomical objects, and allows minimal user customisation. (This is to support easy updating of the database as it is optimised and corrected.) In contrast, the ARID database is designed to capture astronomical information for later analysis. This makes the ARID database more user customisable than the ATID database. (The ARID database consists of the table structure and minimal data, most of which is user customisable.) Although the ATID database can be concurrent with the ARID database, they are in fact separate database structures.

8.1. List of Tables

These tables are listed in the order in which they should be created. This is done as there are tables that depend on others for foreign keys.

The table and field names that are specified are also required by the AIRDAS software. If these table names and field names are not correctly named, then the application may not operate as expected.

8.1.1. TBL_CATALOG

This table stores the basic catalogue names. The catalogues are used for identifying objects by categories of names.

Field Name	Data Type	Index	Foreign Key	Comments
CATALOG_ID	UNSIGNED INT	Primary		Auto Increment
ABBREVIATION	VARCHAR(10)			UNIQUE
CATALOG_NAME	VARCHAR(100)			
COMMENTS	VARCHAR(4000)			
MASK	VARCHAR(50)			
DATEADDED	TIMESTAMP			

8.1.2. TBL_EPOCH

Epoch dates used for identifying the base position of objects. This table stores the standard epoch dates (J2000) for the various standard epochs to date.

Field Name	Data Type	Index	Foreign Key	Comments
EPOCH_ID	UNSIGNED INT	Primary		Auto Increment
EPOCH	VARCHAR(8)			UNIQUE
JULIANDAY	DOUBLE			

8.1.3. TBL_CONSTELLATIONS

The list of constellation names to apply to the TBL_OBJECTS

Field Name	Data Type	Index	Foreign Key	Comments
CONSTELLATION_ID	UNSIGNED INT	Primary		Auto Increment
EPOCH	VARCHAR(8)			
JULIANDAY	DOUBLE			

8.1.4. TBL_OBJECTTYPES

The objects type as applied to TBL_OBJECTS. This is typically planet, exo-planet, neutron star etc.

Field Name	Data Type	Index	Foreign Key	Comments
OBJECTTYPE_ID	UNSIGNED INT	Primary		Auto Increment
OBJECTTYPE	VARCHAR(22)	Unique		
COMMENTS	VARCHAR(158)			
CALCPOSN	INT			

8.1.5. TBL_NAMES

The names of the objects. More than one name per object is supported, duplicate names are also supported.

Field Name	Data Type	Index	Foreign Key	Comments
NAME_ID	UNSIGNED INT	Primary		Auto Increment
NAME	VARCHAR(26)			
OBJECT_ID	INT			
CATALOG_ID	INT			

8.1.6. TBL_SPECTRALCLASS

Field Name	Data Type	Index	Foreign Key	Comments
SPECTRALCLASS_ID	UNSIGNED INT	Primary		Auto Increment
SHORTTEXT	VARCHAR(1)			
LONGTEXT	VARCHAR(50)			

8.1.7. TBL_STELLAROBJECTS

The table of stellar objects. This list includes stellar objects as well as extended objects. There is limited support for exo-planets, in that star systems that contain known exo-planets are indicated as such.

There is no support for solar system (dynamic) objects in this table. All solar system objects are supported by the TBL_SOLAROBJECTS table.

Field Name	Data Type	Index	Foreign Key	Comments
OBJECT_ID	UNSIGNED INT	Primary		Auto Increment
NAME	VARCHAR(26)			
OBJECT_ID	INT			
CATALOG_ID	INT			
EPOCH_ID	INT			
FK4	INT(1)			Boolean
FK5	INT(1)			Boolean
ICRS	INT(1)			Boolean

8.1.8. TBL_VARSTAR_VARTYPE

The type information for variable stars. This is the M-M table for TBL_VARSTARS and TBL_VARTYPES.

Field Name	Data Type	Index	Foreign Key	Comments
OBJECT_ID	UNSIGNED INT	Primary		Auto Increment

5.1.7 TBL_VERSIONINFO

Information of the various versions of the ATID database.

8.1.9. TBL_VARTYPES

Field Name	Data Type	Index	Foreign Key	Comments
VARTYPE_ID	UNSIGNED INT	Primary		Auto Increment
SHORTTEXT	VARCHAR(10)			
LONGTEXT	VARCHAR(255)			
DESCRIPTION	TEXT(65535)			

8.1.10. TBL_SOLAROBJECTS

This table stores the information for solar system objects. This includes planets, minor planets, comets etc.

Field Name	Data Type	Index	Foreign Key	Comments
VARTYPE_ID	UNSIGNED INT	Primary		Auto Increment

8.1.11. TBL_VARSTARS The variable star information as applied to TBL_OBJECTS TBL_VARTYPES The types of variable stars.

8.1.12. TBL_VARSTARS2

Used when updating the TBL_VARSTARS table with new/modified information.

8.1.13. TBL_VERSIONINFO

This table contains information about the database itself. This information includes the current version as well as past versions.

Field Name	Data Type	Index	Foreign Key	Comments
REVISION	UNSIGNED INT	Primary		Auto Increment
STATUS	VARCHAR2(20)			
PUBLISH_DATE	DATE			
TARGETCOUNT	UNSIGNED INT			
IDENTIFIERCOUNT	UNSIGNED INT			

The information in the table needs to be updated manually for each subsequent issue of the database.

Allowed values for status are:

-

9. ARID Database

The ARID database is contains frequently changing information relating to observers, observations and other data related to observations. This contains a number of tables for the data. Tables in this database can be linked via foreign keys to tables in the ATID database, specifically where observation data is concerned.

This database is structured to allow observation information retrieval.

9.1. List of Tables

9.1.1. TBL_IMAGEINFORMATION

This table contains a list of images and the UUID corresponding to the images. Image information includes time/date information, file name and path information and provides a primary key that is used for linking observations of specific objects back to the images in which they appear.

Field Name	Data Type	Index	Foreign Key	Comments
IMAGE_ID	UNSIGNED INT	Primary		Auto Increment
IMAGE_UUID	VARCHAR(30)	Yes		
FILENAME	VARCHAR(255)			
FILEPATH	VARCHAR(255)			

IMAGEDATE	DATETIME	YES		
SITE_ID	UNSIGNED INT	YES		
INSTRUMENT_ID	UNSIGNED INT	YES		
TELESCOPE_ID	UNSIGNED INT	YES		

9.1.2. TBL_TELESCOPES

This is a list of all the telescopes available to an observer.

Field Name	Data Type	Index	Foreign Key	Comments
TELESCOPE_ID	UNSIGNED INT	Primary		Auto Increment
SHORTTEXT	VARCHAR(30)			Not NULL
MANUFACTURER	VARCHAR(30)			
MODELNUMBER	VARCHAR(30)			
APERTURE				
FRATIO				
OBSTRUCTION	DECIMAL(4,2)			
DONTDISPLAY	TINYINT(1)			

9.1.3. TBL_INSTRUMENTS

This is a list of all the instruments available to an observer.

Field Name	Data Type	Index	Foreign Key	Comments
INSTRUMENT_ID	UNSIGNED INT	Primary		Auto Increment
SHORTTEXT	VARCHAR(26)			Not NULL
MANUFACTURER	VARCHAR			
MODELNUMBER	VARCHAR			
SERIALNUMBER	VARCHAR			
NX	INT			unsigned
NY	INT			unsigned
DX	NUMBER(5,3)			
DY	NUMBER(5,3)			
ADU	NUMBER(4,2)			
DONTDISPLAY	TINYINT(1)			

9.1.4. TBL_OBSERVERS

This is a list of all the observer names.

Field Name	Data Type	Index	Foreign Key	Comments
OBSERVER_ID	UNSIGNED INT	Primary		Auto Increment
SHORTTEXT	VARCHAR(30)			Not NULL
AAVSOINITIALS	VARCHAR(3)			
DONTDISPLAY	TINYINT(1)			

9.1.5. TBL_SITES

This is a list of all observation sites available to the observer. There are a number of pre-defined sites that would be distributed with the application, however, the user is encouraged to customise the sites as required for themselves.

Field Name	Data Type	Index	Foreign Key	Comments
SITE_ID	UNSIGNED INT	Primary		Auto Increment
SHORTTEXT	VARCHAR(30)			Not NULL
LATITUDE	NUMBER			
LONGITUDE	NUMBER			
ALTITUDE	NUMBER			
TIMEZONE	NUMBER			
DAYLIGHTSAVING	NUMBER(1,0)			
IAUCODE	VARCHAR			
DONTDISPLAY	TINYINT(1)			

Latitude is given as a decimal value (ll.lllll)

9.1.6. TBL_OBSERVEDOBJECTS

This is a list of all observed objects.

Field Name	Data Type	Index	Foreign Key	Comments
OO_ID	UNSIGNED INT	Primary		Auto Increment

9.1.7. TBL_OBSPROGRAM

A list of all observation programs set-up by the user.

Field Name	Data Type	Index	Foreign Key	Comments
OBSPROG_ID	UNSIGNED INT	Primary		Auto Increment

9.1.8. TBL_PHOTOBJECTS

Objects that are linked to photometry programs. Includes information about how the observation should be handled and what should be observed.

Field Name	Data Type	Index	Foreign Key	Comments
PHOTOBJECT_ID	UNSIGNED INT	Primary		Auto Increment

RANK	UNSIGNED INT	Index		
OBSPROG_ID	UNSIGNED INT	Index	TBL_OBSPROGRAM:OBSPROG_ID	
OBSINTERVAL				
U				
B				
V				
R				
I				
COMP_ID			TBL_NAMES:NAME_ID	
CHECK_ID			TBL_NAMES:NAME_ID	
NAME_ID			TBL_NAMES:NAME_ID	

9.1.9. TBL_PHOTOBJFILTER

M-M table storing the filters to be applied to a photometry object.

Field Name	Data Type	Index	Foreign Key	Comments
PHOTOBJFILTER_ID	UNSIGNED INT	Primary		Auto Increment
PHOTOBJECT_ID	UNSIGNED INT		TBL_PHOTOBJECTS.PHOTOBJ_ID	
FILTER_ID	UNSIGNED INT		TBL_FILTERS.FILTER_ID	

9.1.10. TBL_FILTERS

This is a list of possible filters that may be used with Astronomical camera. All the most likely types are included in the initial list.

Field Name	Data Type	Index	Foreign Key	Comments
FILTER_ID	UNSIGNED INT	Primary		Auto Increment
SHORTTEXT	VARCHAR(6)			
LONGTEXT	VARCHAR2(20)			

Description This table stores a list of all available filters. This includes photometry filters and other filters. This is not a list of filters that the user has, but rather an exhaustive list of all possible filters.

10. Dependencies

These are the files required for the application to operate.

10.1. Windows

This is a list of .dll and similar files that are required for the application to function on Windows XP.

10.1.1. Qt

- QtCore4.dll
- QtGUI4.dll

- QtSQL4.dll

10.1.2. MySQL

These files are required if MySQL is being used as the database.

- Qsqlmysql4.dll
- Libmysql.dll

10.1.3. Oracle XE

- Qsqloci4.dll
- OCI.dll

11. Other

11.1. MySQL

Uploading csv files is best done on the local MySQL server. Copy the files to a directory on the server and then login to the MySQL command prompt

mysql -u root -p

Enter the MySQL password

Change to the correct database

```
mysql> USE ATID
mysql> LOAD DATA LOCAL INFILE '/var/www/site/site_users.csv'
INTO TABLE site_users FIELDS
TERMINATED BY ';'
ENCLOSED BY '"'
LINES TERMINATED BY '\r\n'
IGNORE 1 LINES;
```

12. Notes on Stellar Catalogues

12.1. Astrometric Catalogues

12.1.1. Hipparcos

12.1.2. Tycho 2

12.1.3. UCAC2

48 Million stars, in the range R=8.0 to R=16.0. Positional accuracies 20-70mas.

12.1.4. USNO B1.0

1,042,618,261 stars/galaxies. Positional error = 200mas.

12.2. Double Star Catalogues

12.2.1. Washington Double Star Catalogue

13. Algorithms

Some of the algorithms used are detailed in this section. As the software is open source, a complete description of the code can be found in the code itself. Where more detailed explanations are required, it may be found in this section.

13.1. Astrometry

13.1.1. Plate Constant Determination

To determine the plate constants, we need to begin with the following information:

1. The approximate centre of the plate;
2. At least three identified stars having the following:
    - Catalogue coordinates (α, δ)
    - Measured coordinates (x, y)

The following steps are then followed:

1. Determine the focal length

The angular distance (ω) between the stellar objects is given by:

$$\cos(\omega)=\sin(\delta_1)\sin(\delta_2)+\cos(\delta_1)\cos(\delta_2)\cos(\alpha_1-\alpha_2)$$

The focal length (F) is then:

$$F=\frac{s}{\omega}$$

where (s) is the linear distance between the two stars.

$$s=\sqrt{(x_1-x_2)^2+(y_1-y_2)^2}$$

2. Divide all the (x,y) by F

The coordinates on the tangent plane are taken to be (ζ, η), these are called the standard coordinates.

$$\zeta=\tan(\varepsilon)\cos(\gamma)=\frac{\sin(\alpha-A)}{\sin D\tan\delta+\cos D\cos(\alpha-A)}$$

$$\eta=\tan\varepsilon\cos\gamma=\frac{\tan\delta-\tan D\cos(\alpha-A)}{\tan D\tan\delta+\cos(\alpha-A)}$$

The plate constants are determined using the following equations:

$$\zeta-x=ax+by+c$$

$$\eta-y=dx+ey+f$$

13.2. Statistics

The statistics function used are multi-threaded as much as possible due to the large number of data points present in an image. The MCL library presents a number of multi-threaded statistics functions.

The variables below are used in the discussion of the statistical functions.

$$\sigma = standard\ deviation$$
$$s^2 = variance$$
$$\bar{x} = mean$$

13.2.1. Mean – MCL::mean(...)

The arithmetic mean is used for calculating the mean.

$$\bar{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i$$

The algorithm for calculating the mean can be found in the file “MCL/Include/Statistics/mean.hpp”.

13.2.2. Median

13.2.3. Min/Max

13.2.4. Standard Deviation – MCL::stdev(...)

The standard deviation is calculated from the square root of the variances.

$$\sigma = \sqrt{s^2}$$

The algorithm for standard deviation can be found in the file “MCL/Include/Statistics/stdev.hpp”.

13.2.5. Variance – MCL::variance(...)

$$s^2 = \frac{\sum_{i=1}^{n} x_i^2 - \frac{\left(\sum_{i=1}^{n} x_i\right)^2}{n}}{n-1}$$

The variance procedure uses two algorithms. One to calculate the variances of each subset, and the second to calculate the variance of the combined subsets.

The Knuth and Welford algorithm is used to calculate the data for each subset. The Chan et al. algorithm is used to calculate the variance of the subset data.

The code for the variance calculation can be found in “MCL/include/Statistics/variance.hpp”.

13.3. Determining the FWHM

The FWHM is determined by fitting a 2D Gaussian curve to the star image.

The equation to be fitted is:

$$I_{xy} = A \cdot e^{-\left(\frac{(x-x_o)^2 + (y-y_o)^2}{2w^2}\right)}$$

This is an equation in two dimensions with 4 parameters. There are thus 4 unknowns that need to be determined. Provided that we have more than 4 points, this can be solved using non-linear-least-squares methods.

Due to the effect of the long-tails that leads to poor accuracy, the curve is only fitted to the middle part of the data. This is done to improve accuracy.

Once the data has been non-linear-least-squares fitted, then the FWHM value can be calculated.

The Dlib function to solve non-linear least squares is used to calculate the four parameters. The FWHM is then calculated from:

$$FWHM = 2\sqrt{2\ln(2)} \cdot |w|$$
$$FWHM = 2.35482 \cdot |w|$$

The absolute value is used as there will be two valid solutions, one with w positive and one with w negative. By taking the absolute value, this is corrected.

14. Recommended Reading

The following books are recommended reading to understand how the various algorithms work. They can also give more details on the how-to of image analysis.

14.1. Books

1. Explanatory Supplement to the Astronomical Almanac,
2. Definition of the flexible image transport system (FITS). FITS Standard Version 3.0 2008 July 10
3. The Handbook of Astronomical Image Processing

15. GNU Free Documentation License

Version 1.3, 3 November 2008

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially.

Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover

Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may

be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or

for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification.

Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition.

Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material.

If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number.

Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.